Springer Series in Statistics

Advisors:
P. Bickel, P. Diggle, S. Fienberg, U. Gather,
I. Olkin, S. Zeger

Ingwer Borg
Patrick J.F. Groenen

Modern Multidimensional Scaling

Theory and Applications

Second Edition

With 176 Illustrations

Springer

Ingwer Borg
ZUMA
P.O. Box 122155
D-68072 Mannheim
Germany
borg@zuma-mannheim.de

Patrick J.F. Groenen
Erasmus University
Econometric Institute
P.O. Box 1738
3000 DR Rotterdam
The Netherlands
groenen@few.eur.nl

Library of Congress Control Number: 2005924955

ISBN-10: 0-387-25150-2 Printed on acid-free paper.
ISBN-13: 978-0387-25150-9

© 2005 Springer Science+Business Media, Inc.
All rights reserved. This work may not be translated or copied in whole or in part without the written permission of the publisher (Springer Science+Business Media, Inc., 233 Spring Street, New York, NY 10013, USA), except for brief excerpts in connection with reviews or scholarly analysis. Use in connection with any form of information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed is forbidden.
The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

Printed in the United States of America. (MVY)

9 8 7 6 5 4 3 2 1

springeronline.com

To Leslie,
Handan,
Sezen, and
Martti.

Preface

Multidimensional scaling (MDS) is a technique for the analysis of similarity or dissimilarity data on a set of objects. Such data may be intercorrelations of test items, ratings of similarity on political candidates, or trade indices for a set of countries. MDS attempts to model such data as distances among points in a geometric space. The main reason for doing this is that one wants a graphical display of the structure of the data, one that is much easier to understand than an array of numbers and, moreover, one that displays the essential information in the data, smoothing out noise.

There are numerous varieties of MDS. Some facets for distinguishing among them are the particular type of geometry into which one wants to map the data, the mapping function, the algorithms used to find an optimal data representation, the treatment of statistical error in the models, or the possibility to represent not just one but several similarity matrices at the same time. Other facets relate to the different purposes for which MDS has been used, to various ways of looking at or "interpreting" an MDS representation, or to differences in the data required for the particular models.

In this book, we give a fairly comprehensive presentation of MDS. For the reader with applied interests only, the first six chapters of Part I should be sufficient. They explain the basic notions of ordinary MDS, with an emphasis on how MDS can be helpful in answering substantive questions. Later parts deal with various special models in a more mathematical way and with particular issues that are important in particular applications of MDS. Finally, the appendix on major MDS computer programs helps the reader to choose a program and to run a job.

Contents of the Chapters

The book contains twenty-four chapters, divided into five parts. In Part I, we have six chapters:

- Chapter 1 is an introduction to MDS that explains the four purposes of MDS: MDS as a technique for data explorations, MDS as a method for testing structural hypotheses, MDS as a methodology for the discovery of psychological dimensions hidden in the data, and, finally, MDS as a model of mental arithmetic that explains how similarity judgments are generated. Depending on the particular field of interest, researchers have typically concentrated on just one of these purposes.

- Chapter 2 shows how MDS solutions can be constructed—in simple cases—by purely geometric means, that is, with ruler and compass. Although, in practice, one would almost always use a computer program for finding an MDS solution, this purely geometric approach makes some of the fundamental notions of MDS much clearer than to immediately look at everything in terms of algebraic formulas and computations. It shows, moreover, that the geometric model comes first, and coordinate systems, coordinates, and formulas come later.

- Chapter 3 introduces coordinates and distinguishes different MDS models by the particular functions one chooses for mapping data into distances. Relating data to distances in a particular way also leads to the question of measuring misfit. The Stress index is introduced. An extensive discussion follows on how to evaluate this index in practice.

- Chapter 4 discusses three real-life applications of MDS. The examples are fairly complex but do not require much substantive background. They serve to show the reader some of the trade-off decisions that have to be made when dealing with real data and also some of the most important ways of interpreting an MDS solution.

- Chapter 5 deals with a particular class of MDS applications where the emphasis lies on establishing or testing correspondences of regions in MDS space to classifications of the represented objects in terms of some content facets. It is asked whether objects classified as belonging to type $X, Y, Z, \ldots$ can be discriminated in MDS space such that they lie in different regions. A variety of regional patterns that often arise in practice is discussed and illustrated.

- Chapter 6 describes how to collect similarity or dissimilarity data. Four approaches are distinguished: direct similarity judgments and how to possibly reduce the labor to collect them; deriving similarity measures from the usual cases-by-variables data; converting non-

similarity measures into similarity measures; and some similarity measures defined for co-occurrence data.

Part II discusses technical aspects of MDS:

- Chapter 7 builds some matrix algebra background for later chapters. Eigendecompositions and singular value decompositions, in particular, are essential tools for solving many of the technical problems in MDS. These tools are put to work immediately for constructing a coordinate matrix from a distance matrix, and for principal axes rotations.

- Chapter 8 concentrates on algorithms for optimally solving MDS problems. To that end, basic notions of differentiation of functions and, in particular, of matrix traces are introduced. Then, the majorization method for minimizing a function is explained and applied to solve the MDS problem. This algorithm, known as the SMACOF algorithm, is presented in detail.

- Chapter 9 generalizes the approach of Chapter 8 by allowing for transformations of the dissimilarity data. First, ordinal transformations are discussed, both by monotone regression and rank-images. Then, monotone spline and power transformations are considered in some detail.

- Chapter 10 focuses on confirmatory MDS, where external constraints are enforced onto the MDS solution. These constraints typically are derived from a substantive theory about the data, and it is then tested to what extent this theory is compatible with the data. Two types of constraints are discussed: those imposed on the coordinates and those on the distances of the MDS solution.

- Chapter 11 considers some varieties of indices that assess the goodness of an MDS representation (such as different forms of Stress and the alienation coefficient) and shows some of their relations. Also, we discuss using weights on the dissimilarities and show their effects on MDS solutions.

- Chapter 12 is devoted to one of the first models used for MDS, Classical Scaling. This form of MDS attempts to transform given dissimilarity data into scalar products for which an optimal Euclidean distance representation can be found algebraically without an iterative algorithm.

- Chapter 13 discusses some technical problems that may occur in MDS applications. MDS solutions may degenerate, that is, they become almost perfect in terms of the fit criterion but, nevertheless, do not

represent the data in the desired sense. Another important problem is how to avoid local minimum solutions in iterative procedures. Various conditions and solutions for both problems are presented and discussed.

Part III is devoted to unfolding:

- Chapter 14 is concerned with unfolding, a special case of MDS. In unfolding, one usually has preference data from different individuals for a set of objects. Such data are represented by distances between two sets of points that represent individuals and objects, respectively. The model is psychologically interesting but poses a number of difficult technical problems when transformations are allowed on the data.

- Chapter 15 describes a variety of approaches designed to overcome the problem of degenerate solutions in unfolding. We discuss how to replace missing data with reasonable values, how to make the transformation that maps the data into the distances of the model more rigid, and how to properly adjust the loss function to avoid degeneracies.

- Chapter 16 introduces a number of special models for unfolding such as external unfolding, the vector model of unfolding, individual-differences unfolding with weighted dimensions and anti-ideal points, and a metric unfolding model that builds on scale values constructed within a particular (BTL) choice theory.

Part IV treats the geometry of MDS as a substantive model:

- Chapter 17 concentrates on one particular tradition of MDS where the MDS space is equated with the notion of a "psychological" space. Here, the formula by which we compute distances from point coordinates is taken as a model of the mental arithmetic that generates judgments of dissimilarity. Some varieties of such models (in particular, the Minkowski distance family) and their implications are investigated in some detail.

- Chapter 18 studies a particular function on pairs of multi-valued objects or vectors, scalar products. Scalar products have attractive properties. For example, one can easily find an MDS space that explains them. Hence, various attempts were made in the psychological literature to generate similarity judgments that can be directly interpreted as scalar products (rather than distance-like values).

- Chapter 19 concentrates on the most important distance function in practice, the Euclidean distance. It is asked what properties must

hold for dissimilarities so that they can be interpreted as distances or even as Euclidean distances. We also discuss what transformations map such dissimilarities into Euclidean distances. A further question is how to find a linear transformation that leads to approximate Euclidean distances in a small dimensionality.

Part V discusses some techniques and models that are closely associated with MDS:

- Chapter 20 treats Procrustean problems. Given one particular configuration or target, $\mathbf{X}$, it is asked how one can fit another configuration, $\mathbf{Y}$, to it without destroying meaningful properties of $\mathbf{Y}$. Procrustean solutions are important in practice because they serve to eliminate irrelevant—and often misleading—differences between different MDS solutions.

- Chapter 21 looks at generalized Procrustes analysis, where one wants to fit several configurations to a target or to each other. We also consider extensions where further fitting parameters are admitted that do not preserve the configurations' shapes but that have some meaning in terms of individual differences (e.g., different dimensional weights).

- Chapter 22 focuses on the question of how we can scale a set of K dissimilarity matrices into only one MDS solution and explain the differences among the K data sets by different weights on the dimensions of the "group space" of all K data sets. One algorithm for solving this problem, INDSCAL, is considered in some detail. Some algebraic properties of such models are also investigated.

- Chapter 23 concentrates on asymmetric proximities. They require special considerations or models. We show that asymmetric data can always be decomposed in a symmetric and a skew-symmetric part. Some models for visualizing asymmetry only study the skew-symmetric part and others try to represent both parts at the same time. We discuss several models such as Gower's decomposition for skew-symmetry, a model that represents the skew-symmetries as force vectors in an MDS solution of the the symmetries, unfolding, the slide-vector model, a hill-climbing model, and the radius-distance model.

- Chapter 24 focuses on two methods that are closely related to MDS: principal component analysis and correspondence analysis. We present their formal properties, show some applications to empirical data sets, and discuss how they are related to MDS.

In the Appendix, we cover two issues:

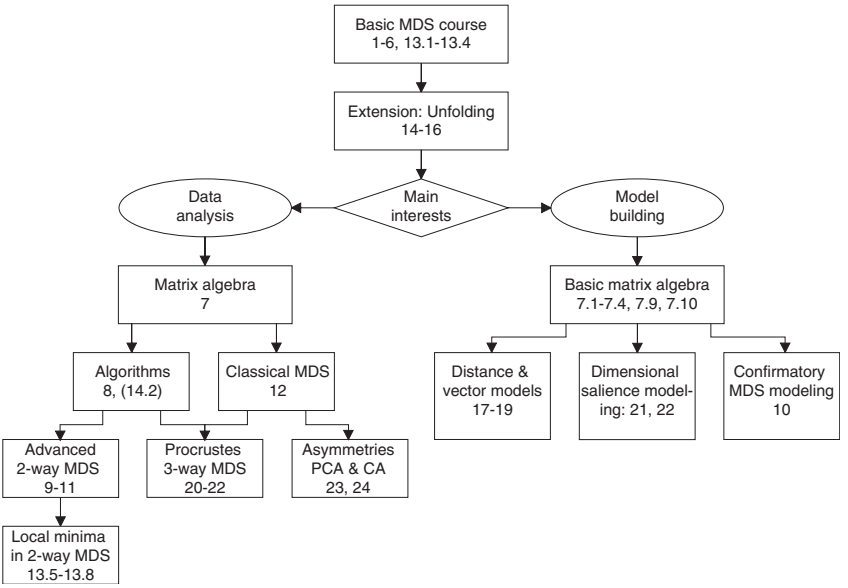


FIGURE 1. Some suggestions for reading this book.

- Appendix A describes in some detail the major computer programs available today for doing MDS. The programs selected are GGVIS, PERMAP, the MDS modules in SAS, SPSS (PROXSCAL and ALSCAL), STATISTICA, and SYSTAT, as well as the standalone programs NEWMDSX©, FSSA, and the classics KYST, MINISSA, and MULTISCALE.

- Appendix B contains a summary of the notation used throughout this book.

How to Read This Book

Beginners in MDS should first study Chapters 1 through 6. These chapters make up a complete introductory course into MDS that assumes only elementary knowledge of descriptive statistics. This course should be supplemented by reading Sections 13.1–13.4 because they cover, in the same nontechnical way, two technical problems (degenerate solutions, local minima) of which every MDS user should be aware.

The basic course can be extended by adding Chapters 14 to 16, if technical sections are skipped. These chapters add the idea of unfolding and discuss some variants of this model.

After mastering the fundamentals, the reader may either read on sequentially or first consider his or her primary interests. If these interests are primarily in the psychology of similarity and choice, then the reader

should move to the chapters on the right-hand side in Figure 1. That is, after reviewing some basic matrix algebra, the reader should move on to one of the topics of particular substantive interest. The most natural place to start is Chapter 17, which focuses directly on different attempts to model similarity by distance functions; to Chapter 18 which is concerned with how to assess scalar products empirically; and to Chapter 19 which studies some of the basic issues involved in modeling proximities in geometric models. Then, the essential ideas of of Chapters 21 and 22 are interesting candidates for further study. Also, the substantively relevant material in Chapter 10 should be of particular interest.

A student whose primary interest is data analysis should first study the matrix algebra in Chapter 7 in somewhat more detail to prepare for Chapter 12 (classical MDS). From Chapter 12, one can proceed to Chapter 23 (asymmetric models) and Chapter 24 (PCA and correspondence analysis) or to Chapters 20–22 (Procrustean methods, three-way models, individual-differences models). A different or additional route in Figure 1 is to turn to Chapter 8 (algorithms) after having studied Chapter 7. The discussion of how to find optimal transformations of the proximities (as in ordinal MDS) in Chapter 9 can be read, to a large extent, without knowledge of Chapter 8. Knowing how to solve MDS problems numerically is, however, a prerequisite for studying a number of advanced issues in Chapter 10 (confirmatory MDS and how to do it) and Chapter 11 (fit measures). From Chapter 11, one should proceed to the technical sections of Chapter 13, which discuss local minima problems.

History of the Book

One could say that the present book is the third edition of a book on multidimensional scaling. The book appeared in German in 1981 under the name *Anwendungsorientierte Multidimensionale Skalierung* by Ingwer Borg (Heidelberg, Germany: Springer). This book served as a basis for an English version. It was called, somewhat cryptically, *Multidimensional Similarity Structure Analysis*. Authored by Ingwer Borg and the late Jim Lingoes, it appeared in 1987 (New York: Springer). As the copies of this book sold out, a revised reprint was considered to bring the book up to date, but then this revision led to a complete overhaul and substantial additions, in particular on the algorithmic side. We have changed the order of presentation, excluded or shortened some material, and included recent developments in the area of MDS. To reflect these changes, we have added "Modern" to the book's title. We also replaced the term "Similarity Structure Analysis" by the better-known term "Multidimensional Scaling". Proponents of SSA may feel that this is an unfortunate regression in terminology, but the term MDS is simply much better known in general. In any case, the shift from SSA to MDS does not imply a change of perspective. We still consider all aspects of MDS representations as potentially interesting, not just

"dimensions." The present book is the second revised edition of *Modern Multidimensional Scaling*.

Preface to the Second edition

The second edition of *Modern Multidimensional Scaling* differs from the first edition on several aspects. The changes have increased the number of pages from 471 to 611 pages and the number of figures from 116 to 176. Two new chapters were added to the book. The first new chapter is devoted to the problem of how to avoid degeneracies in unfolding. New developments in this area are covered and several solutions are presented. One of these solutions, the PrefScal program, is scheduled to become available soon in SPSS.

The other new chapter is an expansion of a section on asymmetric models into a full chapter. There, we discuss several models for visualizing asymmetry and skew-symmetry in MDS. Some of these models are new and others are known in the literature.

In addition, we have updated, extended, and added several sections in existing chapters. Some of these additions reflect new insights from the literature; others are aimed at clarifying existing material. The appendix on MDS software contains the description of four new MDS programs.

Also, exercises have been added to each chapter. They should help the reader to better learn MDS by, first of all, actually doing MDS on empirical data sets, or by rethinking the various issues within a particular scientific context. The exercises differ, of course, with respect to their level. Some emphasize more practical skills such as actually using one or another MDS computer program; others are more demanding and have no simple right-or-wrong answers. These exercises make the book easier to use in a course on MDS. All data in the book are available on the Internet at

http://www.springeronline.com/0-387-25150-2.

Acknowledgment

There are several people we would like to thank for their comments and suggestions on this text. Their inputs certainly have been beneficial for the quality of this book. In particular, we would like to thank Frank Busing, Katrijn van Deun, Luc Delbeke, and Akinori Okada for their constructive feedback on parts of the manuscript. We are also grateful to Joost van Rosmalen for his careful reading and his remarks on the entire manuscript.

Ingwer Borg, Patrick J.F. Groenen, May, 2005, Mannheim and Rotterdam

Contents

Preface **vii**

I Fundamentals of MDS 1

1 The Four Purposes of Multidimensional Scaling 3
1.1 MDS as an Exploratory Technique 4
1.2 MDS for Testing Structural Hypotheses 6
1.3 MDS for Exploring Psychological Structures 9
1.4 MDS as a Model of Similarity Judgments 11
1.5 The Different Roots of MDS 13
1.6 Exercises 15

2 Constructing MDS Representations 19
2.1 Constructing Ratio MDS Solutions 19
2.2 Constructing Ordinal MDS Solutions 23
2.3 Comparing Ordinal and Ratio MDS Solutions 29
2.4 On Flat and Curved Geometries 30
2.5 General Properties of Distance Representations 33
2.6 Exercises 34

3 MDS Models and Measures of Fit 37
3.1 Basics of MDS Models 37
3.2 Errors, Loss Functions, and Stress 41

3.3 Stress Diagrams 42
3.4 Stress per Point 44
3.5 Evaluating Stress 47
3.6 Recovering True Distances by Metric MDS 55
3.7 Further Variants of MDS Models 57
3.8 Exercises 59

4 Three Applications of MDS 63
4.1 The Circular Structure of Color Similarities 63
4.2 The Regionality of Morse Codes Confusions 68
4.3 Dimensions of Facial Expressions 73
4.4 General Principles of Interpreting MDS Solutions 80
4.5 Exercises 82

5 MDS and Facet Theory 87
5.1 Facets and Regions in MDS Space 87
5.2 Regional Laws 91
5.3 Multiple Facetizations 93
5.4 Partitioning MDS Spaces Using Facet Diagrams 95
5.5 Prototypical Roles of Facets 99
5.6 Criteria for Choosing Regions 100
5.7 Regions and Theory Construction 102
5.8 Regions, Clusters, and Factors 104
5.9 Exercises 105

6 How to Obtain Proximities 111
6.1 Types of Proximities 111
6.2 Collecting Direct Proximities 112
6.3 Deriving Proximities by Aggregating over Other Measures . 119
6.4 Proximities from Converting Other Measures 125
6.5 Proximities from Co-Occurrence Data 126
6.6 Choosing a Particular Proximity 128
6.7 Exercises 130

II MDS Models and Solving MDS Problems 135

7 Matrix Algebra for MDS 137
7.1 Elementary Matrix Operations 137
7.2 Scalar Functions of Vectors and Matrices 142
7.3 Computing Distances Using Matrix Algebra 144
7.4 Eigendecompositions 146
7.5 Singular Value Decompositions 150
7.6 Some Further Remarks on SVD 152
7.7 Linear Equation Systems 154

7.8 Computing the Eigendecomposition . . . 157
7.9 Configurations that Represent Scalar Products . . . 160
7.10 Rotations . . . 160
7.11 Exercises . . . 163

8 A Majorization Algorithm for Solving MDS 169
8.1 The Stress Function for MDS . . . 169
8.2 Mathematical Excursus: Differentiation . . . 171
8.3 Partial Derivatives and Matrix Traces . . . 176
8.4 Minimizing a Function by Iterative Majorization . . . 178
8.5 Visualizing the Majorization Algorithm for MDS . . . 184
8.6 Majorizing Stress . . . 185
8.7 Exercises . . . 194

9 Metric and Nonmetric MDS 199
9.1 Allowing for Transformations of the Proximities . . . 199
9.2 Monotone Regression . . . 205
9.3 The Geometry of Monotone Regression . . . 209
9.4 Tied Data in Ordinal MDS . . . 211
9.5 Rank-Images . . . 213
9.6 Monotone Splines . . . 214
9.7 A Priori Transformations Versus Optimal Transformations . 221
9.8 Exercises . . . 224

10 Confirmatory MDS 227
10.1 Blind Loss Functions . . . 227
10.2 Theory-Compatible MDS: An Example . . . 228
10.3 Imposing External Constraints on MDS Representations . . 230
10.4 Weakly Constrained MDS . . . 237
10.5 General Comments on Confirmatory MDS . . . 242
10.6 Exercises . . . 244

11 MDS Fit Measures, Their Relations, and Some Algorithms 247
11.1 Normalized Stress and Raw Stress . . . 247
11.2 Other Fit Measures and Recent Algorithms . . . 250
11.3 Using Weights in MDS . . . 254
11.4 Exercises . . . 258

12 Classical Scaling 261
12.1 Finding Coordinates in Classical Scaling . . . 261
12.2 A Numerical Example for Classical Scaling . . . 263
12.3 Choosing a Different Origin . . . 264
12.4 Advanced Topics . . . 265
12.5 Exercises . . . 267

13 Special Solutions, Degeneracies, and Local Minima 269
13.1 A Degenerate Solution in Ordinal MDS 269
13.2 Avoiding Degenerate Solutions 272
13.3 Special Solutions: Almost Equal Dissimilarities 274
13.4 Local Minima 276
13.5 Unidimensional Scaling 278
13.6 Full-Dimensional Scaling 281
13.7 The Tunneling Method for Avoiding Local Minima 283
13.8 Distance Smoothing for Avoiding Local Minima 284
13.9 Exercises 288

III Unfolding 291

14 Unfolding 293
14.1 The Ideal-Point Model 293
14.2 A Majorizing Algorithm for Unfolding 297
14.3 Unconditional Versus Conditional Unfolding 299
14.4 Trivial Unfolding Solutions and σ_2 301
14.5 Isotonic Regions and Indeterminacies 305
14.6 Unfolding Degeneracies in Practice and Metric Unfolding . 308
14.7 Dimensions in Multidimensional Unfolding 312
14.8 Multiple Versus Multidimensional Unfolding 313
14.9 Concluding Remarks 314
14.10 Exercises 314

15 Avoiding Trivial Solutions in Unfolding 317
15.1 Adjusting the Unfolding Data 317
15.2 Adjusting the Transformation 322
15.3 Adjustments to the Loss Function 324
15.4 Summary 330
15.5 Exercises 331

16 Special Unfolding Models 335
16.1 External Unfolding 335
16.2 The Vector Model of Unfolding 336
16.3 Weighted Unfolding 342
16.4 Value Scales and Distances in Unfolding 345
16.5 Exercises 352

IV MDS Geometry as a Substantive Model 357

17 MDS as a Psychological Model 359
17.1 Physical and Psychological Space 359

17.2 Minkowski Distances 363
17.3 Identifying the True Minkowski Distance 367
17.4 The Psychology of Rectangles 372
17.5 Axiomatic Foundations of Minkowski Spaces 377
17.6 Subadditivity and the MBR Metric 381
17.7 Minkowski Spaces, Metric Spaces, and Psychological Models 385
17.8 Exercises 386

18 Scalar Products and Euclidean Distances 389
18.1 The Scalar Product Function 389
18.2 Collecting Scalar Products Empirically 392
18.3 Scalar Products and Euclidean Distances: Formal Relations 397
18.4 Scalar Products and Euclidean Distances: Empirical Relations 400
18.5 MDS of Scalar Products 403
18.6 Exercises 408

19 Euclidean Embeddings 411
19.1 Distances and Euclidean Distances 411
19.2 Mapping Dissimilarities into Distances 415
19.3 Maximal Dimensionality for Perfect Interval MDS 418
19.4 Mapping Fallible Dissimilarities into Euclidean Distances . 419
19.5 Fitting Dissimilarities into a Euclidean Space 424
19.6 Exercises 425

V MDS and Related Methods 427

20 Procrustes Procedures 429
20.1 The Problem 429
20.2 Solving the Orthogonal Procrustean Problem 430
20.3 Examples for Orthogonal Procrustean Transformations . . . 432
20.4 Procrustean Similarity Transformations 434
20.5 An Example of Procrustean Similarity Transformations . . 436
20.6 Configurational Similarity and Correlation Coefficients . . . 437
20.7 Configurational Similarity and Congruence Coefficients . . . 439
20.8 Artificial Target Matrices in Procrustean Analysis 441
20.9 Other Generalizations of Procrustean Analysis 444
20.10 Exercises 445

21 Three-Way Procrustean Models 449
21.1 Generalized Procrustean Analysis 449
21.2 Helm's Color Data 451
21.3 Generalized Procrustean Analysis 454
21.4 Individual Differences Models: Dimension Weights 457

21.5 An Application of the Dimension-Weighting Model 462
21.6 Vector Weightings 465
21.7 PINDIS, a Collection of Procrustean Models 469
21.8 Exercises 471

22 Three-Way MDS Models **473**
22.1 The Model: Individual Weights on Fixed Dimensions 473
22.2 The Generalized Euclidean Model 479
22.3 Overview of Three-Way Models in MDS 482
22.4 Some Algebra of Dimension-Weighting Models 485
22.5 Conditional and Unconditional Approaches 489
22.6 On the Dimension-Weighting Models 491
22.7 Exercises 492

23 Modeling Asymmetric Data **495**
23.1 Symmetry and Skew-Symmetry 495
23.2 A Simple Model for Skew-Symmetric Data 497
23.3 The Gower Model for Skew-Symmetries 498
23.4 Modeling Skew-Symmetry by Distances 500
23.5 Embedding Skew-Symmetries as Drift Vectors into MDS Plots 502
23.6 Analyzing Asymmetry by Unfolding 503
23.7 The Slide-Vector Model 506
23.8 The Hill-Climbing Model 509
23.9 The Radius-Distance Model 512
23.10 Using Asymmetry Models 514
23.11 Overview 515
23.12 Exercises 515

24 Methods Related to MDS **519**
24.1 Principal Component Analysis 519
24.2 Correspondence Analysis 526
24.3 Exercises 537

VI Appendices **541**

A Computer Programs for MDS **543**
A.1 Interactive MDS Programs 544
A.2 MDS Programs with High-Resolution Graphics 550
A.3 MDS Programs without High-Resolution Graphics 562

B Notation **569**

References **573**

Author Index **599**

Subject Index **605**

Part I

Fundamentals of MDS

1

The Four Purposes of Multidimensional Scaling

Multidimensional scaling (MDS) is a method that represents measurements of similarity (or dissimilarity) among pairs of objects as distances between points of a low-dimensional multidimensional space. The data, for example, may be correlations among intelligence tests, and the MDS representation is a plane that shows the tests as points that are closer together the more positively the tests are correlated. The graphical display of the correlations provided by MDS enables the data analyst to literally "look" at the data and to explore their structure visually. This often shows regularities that remain hidden when studying arrays of numbers. Another application of MDS is to use some of its mathematics as models for dissimilarity judgments. For example, given two objects of interest, one may explain their perceived dissimilarity as the result of a mental arithmetic that mimics the distance formula. According to this model, the mind generates an impression of dissimilarity by adding up the perceived differences of the two objects over their properties.

In the following, we describe four purposes of MDS: (a) MDS as a method that represents (dis)similarity data as distances in a low-dimensional space in order to make these data accessible to visual inspection and exploration; (b) MDS as a technique that allows one to test if and how certain criteria by which one can distinguish among different objects of interest are mirrored in corresponding empirical differences of these objects; (c) MDS as a data-analytic approach that allows one to discover the dimensions that underlie judgments of (dis)similarity; (d) MDS as a psychological model that explains judgments of dissimilarity in terms of a rule that mimics a particular type of distance function.

TABLE 1.1. Correlations of crime rates over 50 U.S. states.

Crime	No.	1	2	3	4	5	6	7
Murder	1	1.00	0.52	0.34	0.81	0.28	0.06	0.11
Rape	2	0.52	1.00	0.55	0.70	0.68	0.60	0.44
Robbery	3	0.34	0.55	1.00	0.56	0.62	0.44	0.62
Assault	4	0.81	0.70	0.56	1.00	0.52	0.32	0.33
Burglary	5	0.28	0.68	0.62	0.52	1.00	0.80	0.70
Larceny	6	0.06	0.60	0.44	0.32	0.80	1.00	0.55
Auto theft	7	0.11	0.44	0.62	0.33	0.70	0.55	1.00

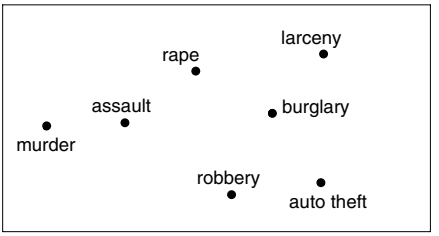


FIGURE 1.1. A two-dimensional MDS representation of the correlations in Table 1.1.

1.1 MDS as an Exploratory Technique

Exploratory data analysis is used for studying theoretically amorphous data, that is, data that are not linked to an explicit theory that predicts their magnitudes or patterns. The purpose of such explorations is to help the researcher to *see* structure in the data. MDS, too, can be used for such data explorations.

Consider an example. The U.S. Statistical Abstract 1970 issued by the Bureau of the Census provides statistics on the rate of different crimes in the 50 U.S. states (Wilkinson, 1990). One question that can be asked about these data is to what extent can one predict a high crime rate of murder, say, by knowing that the crime rate of burglary is high. A partial answer to this question is provided by computing the correlations of the crime rates over the 50 U.S. states (Table 1.1). But even in such a fairly small correlation matrix, it is not easy to understand the structure of these coefficients. This task is made much simpler by representing the correlations in the form of a "picture" (Figure 1.1). The picture is a two-dimensional MDS representation where each crime is shown as a point. The points are arranged in such a way that their distances correspond to the correlations. That is, two points are close together (such as murder and assault) if their corresponding crime rates are highly correlated. Conversely, two points are far apart if their crime rates are not correlated that highly (such as assault and larceny). The correspondence of data and distances is tight in this

example: the product-moment correlation between the coefficients in Table 1.1 and the distances in Figure 1.1 is $r = -.98$.

The reader need not be concerned, at this point, with the question of how such an MDS representation, $\mathbf{X}$, is found. We return to this issue in considerable detail in later chapters. For now, it suffices to assume that the data are fed to an MDS computer program and that this program provides a best-possible solution in a space with a dimensionality selected in advance by the user. The quality of this solution can be checked without knowing how it was found. All one has to do is measure the distances between the points of $\mathbf{X}$ and compare them with the data.[1] If distances and data are highly correlated in the sense of the usual product-moment correlation, say, then the distances represent the data well in a linear sense.[2] This is obviously true in the given case, and so the distances in Figure 1.1 represent the correlations in Table 1.1 very precisely.

What does the MDS picture in Figure 1.1 tell us? It shows that the crimes are primarily distributed along a horizontal dimension that could be interpreted as "violence vs. property" crimes. Moreover, the "property crimes" are less homogeneous, exhibiting some spread along the vertical axis, a dimension that could be interpreted as "hidden vs. street" crimes.

Although here we looked at dimensions, it is important to keep in mind that *any* property of the MDS representation that appears unlikely to result from chance can be interesting. The points may, for example, form certain groupings or clusters. Or, they may fall into different *regions* such as a center region surrounded with bands. The points may also lie on certain *manifolds* such as curved lines (a circle, for example) or on some surface in a higher-dimensional space. Looking for particular directions that would explain the points' distribution is just one possibility to search for structure. Later on in this book, we explore a variety of geometric regularities that have been found useful in practical research.

---

[1] Consider an analogy. Anyone can check the proposition that the number 1.414 approximates $\sqrt{2}$ simply by multiplying 1.414 by itself. The result shows that the proposition is nearly correct. For checking it, it is irrelevant how the number 1.414 was found. Indeed, few would know how to actually compute such a solution, except by trial and error, or by pushing a button on a calculator.

[2] With few points, one can even do (two-dimensional) MDS by hand. To find an MDS solution for the data in Table 1.1, first cut out seven small pieces of paper and write onto each of them one of the labels of the variables in Table 1.1, i.e., "murder", "rape", ..., "auto theft", respectively. Place these pieces of paper arbitrarily in a plane and then move them around in small steps so that higher correlations tend to correspond to smaller distances. Repeat these corrective point movements a few times until the match of distances and data is satisfactory or until it cannot be improved anymore. Such a manual approach is typically quite easy to perform as long as the number of variables is small. With many variables, computer algorithms are needed for doing the work. Good algorithms also make it more likely that one ends up with an optimal MDS solution, that is, a configuration whose distances represent the given data "best" (in some well-defined sense).

Such insights into the data structure are aided by the visual access made possible by the simple MDS picture. Of course, as it is true for exploratory data analysis in general, it is left to further studies to test whether the patterns thus detected are stable ones. Moreover, it is desirable to also develop a theory that provides a rationale for the findings and enables one to predict such structures.

1.2 MDS for Testing Structural Hypotheses

When more is known about a field of interest, exploratory methods become less important. The research items, then, are well designed and the general interest lies in studying effect hypotheses. That is, in particular, what one wants to know is if and how the facets (dimensions, factors, features, etc.) by which the items are conceptually distinguished are reflected in corresponding differences among observations on these items. MDS may be useful for studying such questions. Consider a case.

Levy (1983) reports a study on attitudes towards political protest behavior. She distinguished 18 types of attitudes towards political protest acts. These types correspond to the $3 \cdot 3 \cdot 2 = 18$ different ways of reading the following design scheme (mapping sentence):

$$\text{The} \left\{ \begin{array}{l} \underline{A\text{: modality of attitude}} \\ a_1 = \text{evaluation} \\ a_2 = \text{approval} \\ a_3 = \text{likelihood of own overt action} \end{array} \right\} \text{behavior of respondent } x$$

$$\text{with respect to} \left\{ \begin{array}{l} \underline{B\text{: strength of execution}} \\ b_1 = \text{demanding} \\ b_2 = \text{obstructive} \\ b_3 = \text{physically damaging} \end{array} \right\} \text{protest acts of}$$

$$\left\{ \begin{array}{l} \underline{C\text{: way to carry out}} \\ c_1 = \text{omission} \\ c_2 = \text{commission} \end{array} \right\} \rightarrow \left\{ \begin{array}{c} \underline{R\text{: direction}} \\ \text{very positive} \\ \text{to} \\ \text{very negative} \end{array} \right\} \text{behavior towards acts.}$$

Thirty items were selected from a study by Barnes et al. (1979), using this mapping sentence as a culling rule. Short verbal labels and the codings for the selected items with respect to the three facets of the mapping sentence are given in Table 1.2. For example, item no. 6 effectively asked: "To what extent is 'painting slogans on walls' effective when people use this act in pressing for change?" The respondent's answer was, for this item, recorded on a scale from "very effective" to "not effective". (This scale is the "range" R of the observational mapping.) According to Levy, this item asks about

TABLE 1.2. A classification of protest acts by three facets; numbers in table refer to item numbers.

Item	a_1	a_2	a_3		
Petitions	1	11	21	b_1	c_2
Boycotts	2	12	22	b_2	c_1
Lawful demonstrations	3	13	23	b_1	c_2
Refusing to pay rent	4	14	24	b_2	c_1
Wildcat strikes	5	15	25	b_2	c_1
Painting slogans on walls	6	16	26	b_3	c_2
Occupying buildings	7	17	27	b_2	c_2
Blocking traffic	8	18	28	b_2	c_2
Damaging property	9	19	29	b_3	c_2
Personal violence	10	20	30	b_3	c_2

an effectiveness evaluation ($= a_1$) of a physically damaging act ($= b_3$) of commission ($= c_2$).

How are these 18 different forms of attitudes towards protest behavior related to each other? Will the facets used by Levy for *conceptually* classifying the items show up in the survey data? The distinction "omission vs. commission", for example, is, after all, an organizing principle that comes from Levy. It may be clear enough and even useful to other researchers in the field of political behavior. However, that does not mean that the uninitiated respondent would use similar notions, especially not *implicitly* when making his or her ratings. In fact, it is not even guaranteed that evaluating protest acts in terms of "effectiveness", "approval", and "likelihood of own overt action" will lead to different ratings.

Levy (1983) approached these questions by MDS. The intercorrelations of the items from surveys taken in five different countries were first "scaled" by MDS. It turned out that three-dimensional spaces were needed in each case to adequately represent the correlations of the 30 items by corresponding distances. Figure 1.2 shows the MDS space for the German data.

One could inspect this space in an exploratory manner, as above. However, three-dimensional MDS configurations are hard to understand, in particular when projected onto paper or onto the computer screen. What we want here is, in any case, not exploration. Rather, we want to link the MDS configuration to the item design. For that purpose, it is easier not to look at the complete three-dimensional space at once, but only at certain projection planes. Such planes are, for example, the planes spanned by the three coordinate axes, that is, the plane spanned by axes X and Y, or by X and Z. Inspecting the X–Y plane or the "bottom" plane of Figure 1.2, one finds that Figure 1.3 can be split in two ways that clearly reflect the distinctions $a_1, \ldots, a_3$ and $b_1, \ldots, b_3$, respectively, made by the first two facets of the mapping sentence. The solid vertical lines show, for example, that all "demanding" items lie on the left-hand side, all "obstruction" items lie

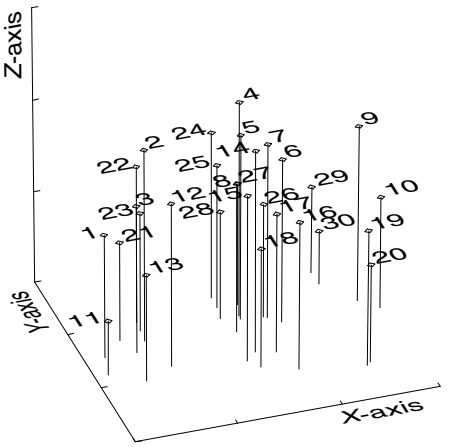


FIGURE 1.2. Three-dimensional MDS representation of protest acts.

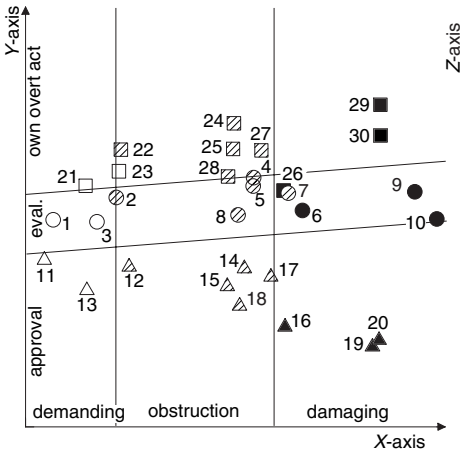


FIGURE 1.3. Plane spanned by X- and Y-axes of Fig. 1.2; drawn-in lines represent the first two facets (A and B) from mapping sentence. The filling of the markers reflects the levels of facet A, the shape the level of facet B.

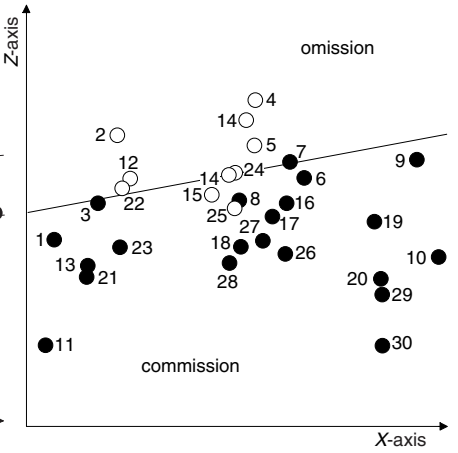


FIGURE 1.4. Plane spanned by X- and Z-axes of Fig. 1.2; drawn-in line represents third facet (C). The filling of the markers reflects the levels of facet C.

in the middle, and all "damaging" items lie on the right-hand side of the space. Figure 1.4 makes clear that the "omission" points are placed above the "commission" items along the Z-axis. Putting these findings together, one notes that the three-dimensional MDS space is thus cut into box-like regions that result from projecting the conceptual codings of the items onto the MDS configuration. Hence, Levy's distinctions on protest acts are not only conceptually possible, but they are also useful for explaining data variance.

1.3 MDS for Exploring Psychological Structures

MDS has been used primarily in psychology. Psychologists usually have psychological questions in mind. Even when used in an exploratory manner, MDS thus typically carried with it, as an implicit purpose, the search for "underlying dimensions" that would explain observed similarities or dissimilarities. In the exploratory MDS application on crime rates considered above, such notions were absent or had, at least, a much lower priority. The purpose of MDS, in the above crime context, was simply to enable the data analyst to look at the data structure in order to find rules that would help to *describe* the distribution of the points. One could thus say that in pure data-analytic MDS, one attempts to find rules of formation that allow one to describe the data structure in as simple terms as possible, whereas in the kind of exploratory MDS that is typical for psychologists the researcher is interested in discovering psychological dimensions that would meaningfully explain the data.

In psychology, the data used for MDS are often based on direct similarity judgments by the respondents. Wish (1971), for example, asked 18 students to rate the global similarity of different pairs of nations such as France and China on a 9-point rating scale ranging from 1 = very different to 9 = very similar. Table 1.3 shows the mean similarity ratings.

The similarity data of Table 1.3 are, roughly, represented by the distances of the two-dimensional MDS configuration in Figure 1.5. It thus holds that the higher the similarity measures, the smaller the corresponding distance. The dashed lines in this figure were not generated by MDS. Rather, they are an interpretation by Kruskal and Wish (1978) that can help to explain the distribution of the points. Interpreting an MDS representation means linking some of its geometric properties to substantive knowledge about the objects represented by the points. One such geometric property is the scatter of the points along a straight line or *dimension*. The lines are chosen by first identifying points that are far apart and about which one already knows something. Based on this prior knowledge, one attempts to formulate a substantive criterion that could have led *the subjects* to distinguish so

TABLE 1.3. Matrix of average similarity ratings for 12 nations (Wish, 1971).

Nation		1	2	3	4	5	6	7	8	9	10	11	12
Brazil	1	–											
Congo	2	4.83	–										
Cuba	3	5.28	4.56	–									
Egypt	4	3.44	5.00	5.17	–								
France	5	4.72	4.00	4.11	4.78	–							
India	6	4.50	4.83	4.00	5.83	3.44	–						
Israel	7	3.83	3.33	3.61	4.67	4.00	4.11	–					
Japan	8	3.50	3.39	2.94	3.83	4.22	4.50	4.83	–				
China	9	2.39	4.00	5.50	4.39	3.67	4.11	3.00	4.17	–			
USSR	10	3.06	3.39	5.44	4.39	5.06	4.50	4.17	4.61	5.72	–		
U.S.A.	11	5.39	2.39	3.17	3.33	5.94	4.28	5.94	6.06	2.56	5.00	–	
Yugoslavia	12	3.17	3.50	5.11	4.28	4.72	4.00	4.44	4.28	5.06	6.67	3.56	–

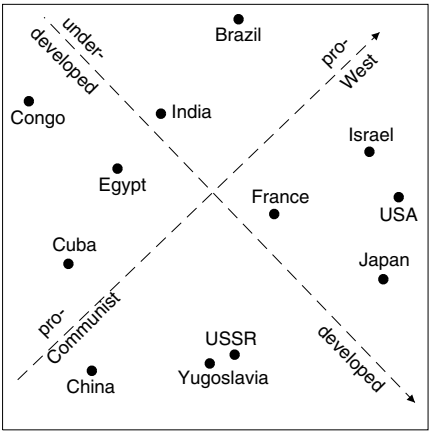


FIGURE 1.5. MDS for data in Table 1.3; dashed lines are an interpretation of the point scatter.

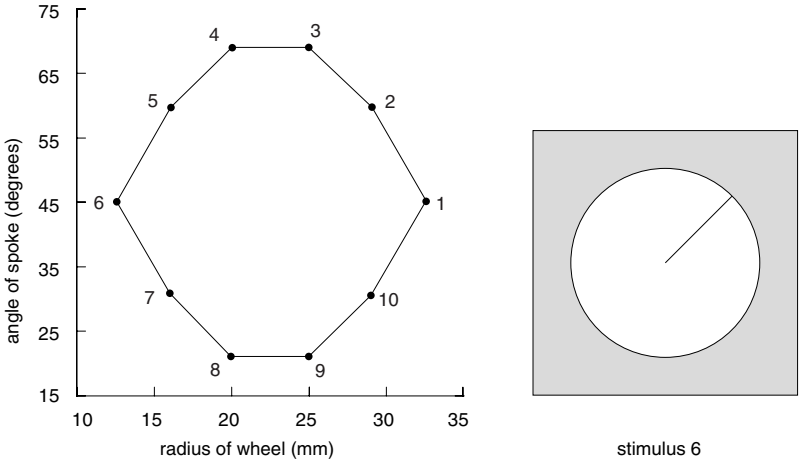


FIGURE 1.6. Design configuration for Broderson's one-spoked wheels; a specimen for such a stimulus is shown in the insert on the right-hand side.

clearly between these objects, placing them at opposite ends of a dimension. This is known as interpreting a dimension.

Interpreting an MDS space, therefore, involves data-guided speculations about the psychology of those who generated the similarity data. Testing the validity of the conclusions is left to further studies.

1.4 MDS as a Model of Similarity Judgments

Finally, the mathematics of MDS can serve as a model of similarity judgments. The most common approach is to hypothesize that a person, when asked about the dissimilarity of pairs of objects from a set of objects, acts *as if* he or she computes a distance in his or her "psychological space" of these objects.

Questions of this sort are studied mostly in the context of well-designed stimuli. One such example is the following. Broderson (1968) studied the dissimilarity of stimuli that looked like one-spoked wheels. That is, his stimuli were circles varying in diameter from 12.5 mm to 32.5 mm; they also had a drawn-in radius line at angles varying from 21° to 69°. Figure 1.6 shows an example of such a stimulus, together with a geometric description of the 10 stimuli selected for experimentation. (The line connecting the points in this figure has no particular meaning. It only helps to better understand the structure of the point configuration.)

Each of the 45 pairs of the one-spoked wheels $1, \ldots, 10$ from Figure 1.6 was drawn on a card and presented to subjects with the instruction to rate this pair's global similarity on a scale from 1 = minimal similarity to

TABLE 1.4. Mean similarity scores for one-spoked wheels described in Figure 1.6.

Item	1	2	3	4	5	6	7	8	9	10
1	–									
2	5.10	–								
3	3.86	5.42	–							
4	3.24	4.74	5.30	–						
5	3.52	4.98	4.56	5.06	–					
6	4.60	3.76	3.06	3.68	4.86	–				
7	4.02	3.08	2.88	3.26	4.82	5.06	–			
8	3.42	3.42	2.94	4.44	3.34	3.44	4.90	–		
9	3.98	3.36	4.30	3.26	2.92	3.06	4.64	5.48	–	
10	5.30	4.78	3.70	3.36	3.12	4.36	4.68	4.40	5.06	–

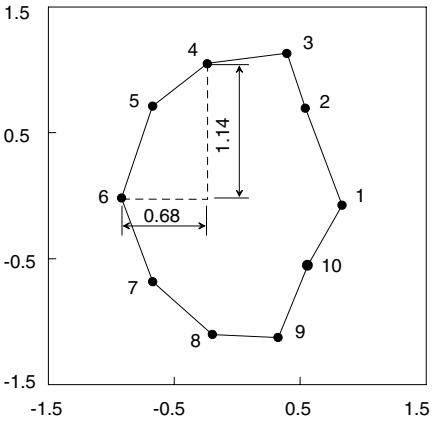


FIGURE 1.7. MDS representation of similarity data in Table 1.4; the combined lengths of the dashed line segments is the city-block distance of points 4 and 6.

7 = maximal similarity. This led to a 10×10 matrix of similarity scores for each subject. The mean scores for all 50 subjects are shown in Table 1.4.

It was hypothesized that a subject arrives at a similarity judgment by computing a particular distance in his or her psychological space. This space should essentially correspond to the physical design space in Figure 1.6. Given two points in this space, their *city-block distance* is the sum of their distances along the X- and Y-axes, respectively.

Figure 1.7 shows an MDS representation of the values in Table 1.4. One notes immediately that this spatial representation of the subjects' similarity scores is very similar to the design configuration in Figure 1.6.

The MDS representation has been computed so that its city-block distances correspond to the similarity scores in Table 1.4. In Figure 1.7, it is shown how such a city-block distance is computed. For points 4 and 6,

it is equal to the sum of the lengths of the dashed line segments connecting points 4 and 6: $0.68 + 1.14 = 1.82$. Broderson claims that his subjects arrived at their similarity ratings by comparing each pair of one-spoked wheels dimension by dimension, adding the perceived dimensional differences, and converting the resulting global dissimilarity impressions into the format of the response scale.

Do the similarity values in Table 1.4 support this theory? The answer is quite positive, because the (city-block) distances between any two points i and j in Figure 1.7 are highly correlated ($r = -.92$) with the similarity values in Table 1.4. Hence, this particular two-dimensional distance geometry is indeed a possible model of judgment of similarity for the given stimuli.

Such psychological model building goes considerably beyond a mere searching for structure in the data. It also differs from testing an abstract structural hypothesis. Rather, it involves a particular distance function that is defined on particular dimensions and is interpreted quite literally as a psychological *composition rule*.[3]

1.5 The Different Roots of MDS

The different purposes of MDS, and the existence of an enormous variety of related geometric models, have led to unnecessary confusion over the question of how MDS should be used. Social scientists such as sociologists, political scientists, or social psychologists, for example, are often interested in using MDS to test hypotheses on correlations in a way similar to what we saw above in Section 1.2. Consequently, they often do not even use the term multidimensional scaling but rather speak of *smallest space analysis* (Guttman, 1968) or of *multidimensional similarity structure analysis* (Borg & Lingoes, 1987).

Psychophysicists, on the other hand, are usually concerned not with correlations but with models that relate stimuli with well-known physical properties to their perceptual or cognitive representations. For them, the notion of multidimensional scaling has a very direct meaning in the sense that they study how *known* physical dimensions are represented psychologically. Because psychophysics is the domain where MDS came from [see De Leeuw

[3]There are theories closely related to MDS modeling that do not concentrate very much on the distance function, but instead concentrate on other properties of multidimensional geometry such as "incidence", "perpendicularity", or "inclusion". Often, geometries are chosen that appear very strange to the nonmathematician, such as curved spaces, bounded spaces, or finite geometries [see, for example, Drösler (1979) and Müller (1984)]. Such models are, however, typically highly specialized and thoroughly bound to a particular substantive field of interest (such as "monocular space perception" or "color vision"). There is usually no reason to use them for general data-analytic purposes, and so very little attention is given to them in this book.

and Heiser (1982) on the history of MDS], it is enlightening to read what Torgerson (1952) thought about MDS:

> *The traditional methods of psychophysical scaling presuppose knowledge of the dimensions of the area being investigated. The methods require judgments along a particular defined dimension, i.e., A is brighter, twice as loud, more conservative, or heavier than B. The observer, of course, must know what the experimenter means by brightness, loudness, etc. In many stimulus domains, however, the dimensions themselves, or even the number of relevant dimensions, are not known. What might appear intuitively to be a single dimension may in fact be a complex of several. Some of the intuitively given dimensions may not be necessary... Other dimensions of importance may be completely overlooked. In such areas the traditional approach in inadequate.*
>
> *Richardson, in 1938 (see also Gulliksen, 1946) proposed a model for multidimensional scaling that would appear to be applicable to a number of these more complex areas. This model differs from the traditional scaling methods in two important respects. First, it does not require judgments along a given dimension, but utilizes, instead, judgments of similarity between the stimuli. Second, the dimensionality, as well as the scale values, of the stimuli is determined from the data themselves.*

This clearly shows that early MDS was strongly dominated by notions of dimensional modeling of similarity judgments. Later consumers of MDS, even when they used MDS for purely exploratory purposes, were apparently so much influenced by this dimensional thinking that they often almost automatically looked for interpretable dimensions even though they set out to generally explore the data structure.

Data analysts, in contrast to psychophysicists, are generally not interested in building models for a particular substantive domain. Rather, they want to provide general-purpose tools for empirical scientists that will help the substantive researchers to better understand the structure of their data. For this purpose, of course, it would make no sense to employ a distance function such as the city-block distance used in Section 1.4 above, because the relations among the points of such geometries often are *not* what they appear to be. For example, the city-block distance between points 4 and 6 in Figure 1.7 is about the same as the city-block distance between points 1 and 6. The natural (Euclidean) distance between 4 and 6 is, in contrast, considerably shorter than the distance between 1 and 6. Hence, MDS representations that employ distance functions other than the Euclidean tend to be misleading when inspected intuitively. Therefore, they are useless for exploratory purposes.

1.6 Exercises

Exercise 1.1 Consider the following correlation matrix of eight intelligence test items (Guttman, 1965).

Item	1	2	3	4	5	6	7	8
1	1.00	.40	.25	.12	.67	.39	.26	.19
2	.40	1.00	.31	.39	.50	.24	.18	.52
3	.25	.31	1.00	.46	.28	.38	.42	.49
4	.12	.39	.46	1.00	.20	.14	.29	.55
5	.67	.50	.28	.20	1.00	.38	.26	.26
6	.39	.24	.38	.14	.38	1.00	.40	.22
7	.26	.18	.42	.29	.26	.40	1.00	.25
8	.19	.52	.49	.55	.26	.22	.25	1.00

(a) Use the procedure outlined in Footnote 2 on page 5 to find an MDS representation of these data in the plane by hand. That is, items should be represented as points, and the distances between any two points should be smaller the higher the corresponding items are correlated.

(b) The MDS representation will exhibit a particularly simple structure among the items. Use this structure to reorder the above correlation matrix. What pattern does this matrix exhibit?

(c) A typical beginner's mistake when using MDS is to incorrectly specify how the MDS distances should be related to the data. Correlations are indices of similarity, not of dissimilarity, and so correlations should be *inversely* related to MDS distances. Check what happens when you tell your MDS program that you want larger correlations represented by larger distances. (Hint: Depending on the MDS computer program, you may have to request something like "Regression=ascending" or you may have to specify that the correlations are "similarities." For a description of MDS programs, see Appendix A.)

Exercise 1.2 Consider the following correlation matrix of seven vocational interest scales (Beuhring & Cudeck, 1985).

Scale	Health	Science	Techn.	Trades	Bus.O.	Bus.C.	Social
Health	1.00						
Science	.65	1.00					
Technology	.45	.64	1.00				
Trades	.25	.44	.76	1.00			
Business Operations	.12	.16	.55	.49	1.00		
Business Contact	.22	.21	.57	.46	.75	1.00	
Social	.50	.26	.37	.20	.47	.65	1.00

(a) Use the procedure outlined in Footnote 2 on page 5 to find an MDS representation of these data in the plane by hand.

(b) Interpret the resulting MDS representation: What does it tell you about interests?

Exercise 1.3 Consider the data in Table 1.4 on page 12. They were scaled in Figure 1.7 by using the city-block distance, not the "usual" (that is, Euclidean) distance. What happens to city-block distances if the coordinate system is rotated by, say, 30 degrees? What happens to Euclidean distances in the same case? Based on your answers to these two questions above, what can you say about the coordinate system when dealing with city-block distances?

Exercise 1.4 Representing proximity data such as correlations in an MDS plane is often useful for an exploratory investigation of the data structure. Yet, the MDS configuration can also be misleading. When?

Exercise 1.5 Replicate the experiment of Section 1.3 with 10 U.S. States or countries of your choice.

(a) Prepare a list of all possible pairs of states. Rate the similarity of the states in each pair on a scale from 0=not different to 10=very different. (You may want to begin by first picking the two states that appear most different and by setting their similarity equal to 10. This establishes a frame of reference for your judgments.)

(b) Scale the resulting similarity ratings by hand or by an MDS computer program.

(c) Study the MDS solution and search for a dimensional interpretation.

Exercise 1.6 Consider the matrix below (Lawler, 1967). It shows the correlations among nine items. The items assess three performance criteria (T_1 = quality of job performance, T_2 = ability to perform the job, T_3 = effort put forth on the job) by three different methods (M_1 = superior ratings, M_2 = peer ratings, M_3 = self ratings). Such a matrix is called a multitrait-multimethod matrix.

Item	No.	1	2	3	4	5	6	7	8	9
T_1M_1	1	1.00								
T_2M_1	2	.53	1.00							
T_3M_1	3	.56	.44	1.00						
T_1M_2	4	.65	.38	.40	1.00					
T_2M_2	5	.42	.52	.30	.56	1.00				
T_3M_2	6	.40	.31	.53	.56	.40	1.00			
T_1M_3	7	.01	.01	.09	.01	.17	.10	1.00		
T_2M_3	8	.03	.13	.03	.04	.09	.02	.43	1.00	
T_3M_3	9	.06	.01	.30	.02	.01	.30	.40	.40	1.00

(a) Check whether the facets trait and method are reflected as regions in an MDS representation of the correlations.

(b) What substantive conclusions can you derive with respect to the facets trait and method? Is there, for example, reason to conclude that the facets may be ordered rather than just categorical?

(c) What other insights can you derive from the MDS solution concerning performance appraisals? How do the different kinds of appraisals differ?

Exercise 1.7 Consider Table 1.5 on page 18. It shows data from an experiment where 10 experienced psychiatrists each fabricated archetypal psychiatric patients by characterizing them on the 17 variables of the Brief Psychiatric Rating Scale (Mezzich, 1978). The variables are A = somatic concern, B = anxiety, C = emotional withdrawal, D = conceptual disorganization, E = guilt feelings, F = tension, G = mannerism and posturing, H = grandiosity, I = depressive mood, J = hostility, K = suspiciousness, L = hallucinatory behavior, M = motor retardation, N = uncooperativeness, O = unusual thought content, P = blunted affect, Q = excitement.

(a) Correlate the rows of this data matrix to get similarity coefficients for the 40 patients. Then use MDS to explore the structure of the correlations.

(b) Does a 2D MDS representation allow you to distinguish the four psychiatric types?

(c) The MDS representation indicates that the four types are ordered in certain ways. Describe and explain that order.

Exercise 1.8 Consider the data in Table 1.5 on page 18.

(a) Compute the Euclidean distance of any two rows. Use these distances as proximities and do a two-dimensional MDS with them. Compare the resulting solution to an MDS solution that uses correlations as proximity measures.

(b) Repeat the above for city-block distances as proximity measures.

(c) Are the MDS solutions very different? Discuss why this is so.

TABLE 1.5. Severity ratings (on 0..6 scale) of four prototypical psychiatric patients on 17 symptoms by 10 psychiatrists (Mezzich, 1978).

Type	No.	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
	1	4	3	3	0	4	3	0	0	6	3	2	0	5	2	2	2	1
	2	5	5	6	2	6	1	0	0	6	1	0	1	6	4	1	4	0
	3	6	5	6	5	6	3	2	0	6	0	5	3	6	5	5	0	0
	4	5	5	1	0	6	1	0	0	6	0	1	2	6	0	3	0	2
De-	5	6	6	5	0	6	0	0	0	6	0	4	3	5	3	2	0	0
pressive	6	3	3	5	1	4	2	1	0	6	2	1	1	5	2	2	1	1
	7	5	5	5	2	5	4	1	1	6	2	3	0	6	3	5	2	3
	8	4	5	5	1	6	1	1	0	6	1	1	0	5	2	1	1	0
	9	5	3	5	1	6	3	1	0	6	2	1	1	6	2	5	5	0
	10	3	5	5	3	2	4	2	0	6	3	2	0	6	1	4	5	1
	11	2	2	1	2	0	3	1	6	2	3	3	2	1	4	4	0	6
	12	0	0	0	4	1	5	0	6	0	5	4	4	0	5	5	0	6
	13	0	3	0	5	0	6	0	6	0	3	2	0	0	3	4	0	6
	14	0	0	0	3	0	6	0	6	1	3	1	1	0	2	3	0	6
Manic	15	3	4	0	0	0	5	0	6	0	6	0	0	0	5	0	0	6
	16	2	4	0	3	1	5	1	6	2	5	3	0	0	5	3	0	6
	17	1	2	0	2	1	4	1	5	1	5	1	1	0	4	1	0	6
	18	0	2	0	2	1	5	1	5	0	2	1	1	0	3	1	0	6
	19	0	0	0	6	0	5	1	6	0	5	5	4	0	5	6	0	6
	20	5	5	1	4	0	5	5	6	0	4	4	3	0	5	5	0	6
	21	3	2	5	2	0	2	2	1	2	1	2	0	1	2	2	4	0
	22	4	4	5	4	3	3	1	0	4	2	3	0	3	2	4	5	0
	23	2	0	6	3	0	0	5	0	0	3	3	2	3	5	3	6	0
	24	1	1	6	2	0	0	1	0	0	3	0	1	0	1	1	6	0
Schizo-	25	3	3	5	6	3	2	5	0	3	0	2	5	3	3	5	6	2
phrenic	26	3	0	5	4	0	0	3	0	2	1	1	1	2	3	3	6	0
	27	3	3	5	4	2	4	2	1	3	1	1	1	4	2	2	5	2
	28	3	2	5	2	2	2	2	1	2	2	3	1	2	2	3	5	0
	29	3	3	6	6	1	3	5	1	3	2	2	5	3	3	6	6	1
	30	1	1	5	3	1	1	3	0	1	1	1	0	5	1	2	6	0
	31	2	4	3	5	0	3	1	4	2	5	6	5	0	5	6	3	3
	32	2	4	1	1	0	3	1	6	0	6	6	4	0	6	5	0	4
	33	5	5	5	6	0	5	5	6	2	5	6	6	0	5	6	0	2
	34	1	4	2	1	1	1	0	5	1	5	6	5	0	6	6	0	1
Paranoid	35	4	5	6	3	1	6	3	5	2	6	6	4	0	5	6	0	5
	36	4	5	4	6	2	4	2	4	1	5	6	5	1	5	6	2	4
	37	3	4	3	4	1	5	2	5	2	5	5	3	1	5	5	1	5
	38	2	5	4	3	1	4	3	4	2	5	5	4	0	5	4	1	4
	39	3	3	4	4	1	5	5	5	0	5	6	5	1	5	5	3	4
	40	4	4	2	6	1	4	1	5	3	5	6	5	1	5	6	2	4

2
Constructing MDS Representations

An MDS representation is found by using an appropriate computer program. The program, of course, proceeds by computation. But one- or two-dimensional MDS representations can also be constructed by hand, using nothing but a ruler and compass. In the following, we discuss such constructions in some detail for both ratio MDS and for ordinal MDS. This leads to a better understanding of the geometry of MDS. In this context, it is also important to see that MDS is almost always done in a particular family of geometries, that is, in flat geometries.

2.1 Constructing Ratio MDS Solutions

An MDS representation is in practice always found by using an appropriate computer program (see Appendix A for a review of such programs). A computer program is, however, like a black box. It yields a result, hopefully a good one, but does not reveal how it finds this solution.

A good way to build an intuitive understanding for what an MDS program does is to proceed by hand. Consider an example. Table 2.1 shows the distances between 10 cities measured on a map of Europe. We now try to reverse the measurement process. That is, based only on the values in Table 2.1, we want to find a configuration of 10 points such that the distances between these points correspond to the distances between the 10 cities on the original map. The reconstructed map should be proportional in size to the original map, which means that the ratios of its distances

TABLE 2.1. Distances between ten cities.

	1	2	3	4	5	6	7	8	9	10
1	0	569	667	530	141	140	357	396	570	190
2	569	0	1212	1043	617	446	325	423	787	648
3	667	1212	0	201	596	768	923	882	714	714
4	530	1043	201	0	431	608	740	690	516	622
5	141	617	596	431	0	177	340	337	436	320
6	140	446	768	608	177	0	218	272	519	302
7	357	325	923	740	340	218	0	114	472	514
8	396	423	882	690	337	272	114	0	364	573
9	569	787	714	516	436	519	472	364	0	755
10	190	648	714	622	320	302	514	573	755	0

should correspond to the ratios of the values in Table 2.1. This defines the task of *ratio MDS*. We find the *solution* of this task as follows.

A Ruler-and-Compass Approach to Ratio MDS

For convenience in laying out the map, we first identify those cities that are farthest from each other. Table 2.1 shows that these are the cities 2 and 3, whose distance is $d_{23} = 1212$ units. We then want to place two points on a piece of paper such that their distance is proportional to $d_{23} = 1212$ units. To do this, we choose a *scale factor*, s, so that the reconstructed map has a convenient overall size. If, for example, we want the largest distance in the map to be equal to 5 cm, then $s = 0.004125$ so that $s \cdot 1212 = 5$. All values in Table 2.1 are then multiplied by s. The scale factor s leaves invariant the proportions or ratios of the data in Table 2.1.

Having fixed the scale factor, we draw a line segment with a length of $s \cdot 1212$ cm on a piece of paper. Its endpoints are called 2 and 3 (Figure 2.1).

We now elaborate our two-point configuration by picking one of the remaining cities for the next point. Assume that we pick city 9. Where must point 9 lie relative to points 2 and 3? In Table 2.1 we see that the distance between cities 2 and 9 on the original map is 787 units. Thus, point 9 must lie anywhere on the circle with radius $s \cdot 787$ cm around point 2. At the same time, point 9 must have a distance of $s \cdot 714$ cm to point 3. Consequently, point 9 also must lie on the circle with radius $s \cdot 714$ cm around point 3 (Figure 2.2). Hence, for point 9, there are exactly two *solutions*—labeled as 9 and $9'$, respectively, in Figure 2.1—that satisfy the conditions $d_{29} = s \cdot 787$ cm and $d_{39} = s \cdot 714$ cm. We arbitrarily choose point 9.

We continue by adding further points to our MDS configuration. It does not matter which city we pick next. Assume that it is city 5. Where, relative to points 2, 3, and 9, should point 5 lie? It should lie on (a) the circle around point 2 with radius $s \cdot d_{25}$, (b) on the circle around point 3 with

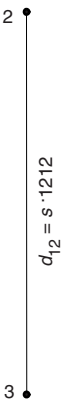


FIGURE 2.1. First construction step for MDS representation of distances in Table 2.1.

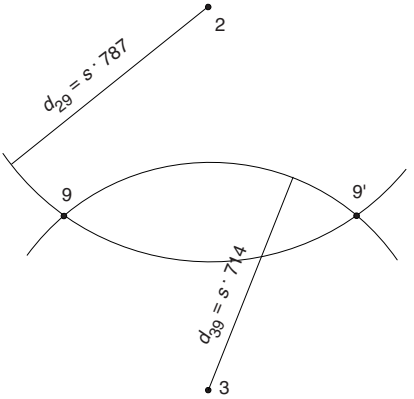


FIGURE 2.2. Positioning point 9 on the map.

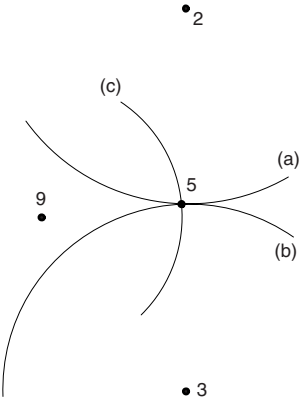


FIGURE 2.3. Positioning point 5 on the map.

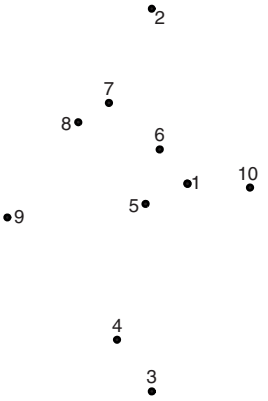


FIGURE 2.4. Final MDS representation for data in Table 2.1.

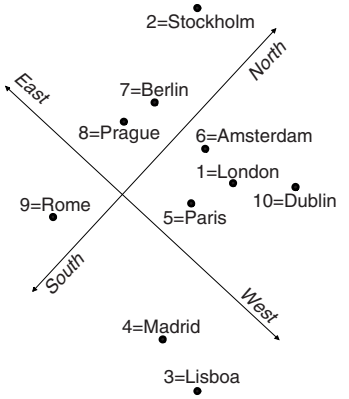


FIGURE 2.5. Identification of points and geographical compass.

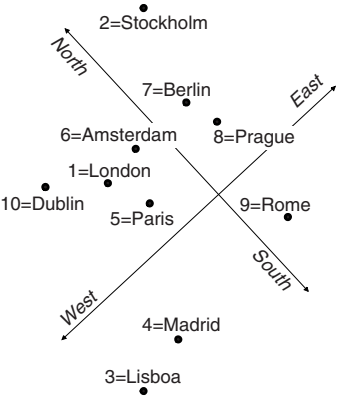


FIGURE 2.6. Horizontal reflection of configuration in Fig. 2.5 so that East is to the right.

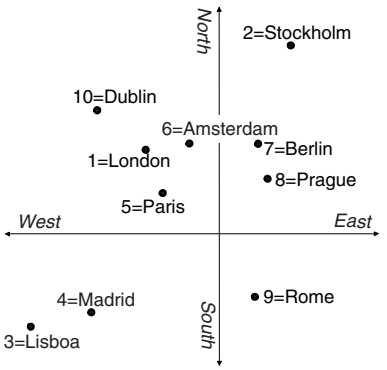


FIGURE 2.7. Rotation of configuration in Fig. 2.5 so that North is up.

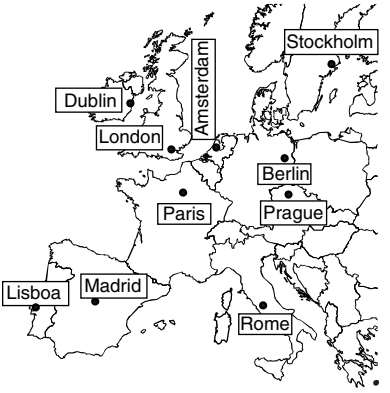


FIGURE 2.8. Configuration of Fig. 2.7 over a map of Europe.

radius $s \cdot d_{35}$, and (c) on the circle around point 9 with radius $s \cdot d_{95}$, as in Figure 2.3. Point 5 satisfies all three conditions and, in contrast to the above construction for point 9, there is only one solution point.

Once all of the cities have been considered, the configuration in Figure 2.4 is obtained. The configuration *solves* the representation problem, because the distances between its points correspond to the distances in Table 2.1, except for an overall scale factor s.

If we replace the numbers with city names, then Figure 2.5 shows that the reconstructed map has an unconventional orientation. But this can be easily adjusted. We first *reflect* the map along the horizontal direction so that West is on the left-hand side, and East is on the right-hand side (Figure

2.6). Second, we *rotate* the map somewhat in a clockwise direction so that the North–South arrow runs in the vertical direction, as usual (Figure 2.7).

Admissible Transformations of Ratio MDS Configuration

The final "cosmetic" *transformations* of the MDS configuration—rotation and reflection—are obviously without consequence for the reconstruction problem, because they leave the distances unchanged (*invariant*). Rotations and reflections are thus said to be *rigid motions*. Another form of a rigid motion is a *translation*, that is, a displacement of the entire configuration relative to a fixed point. A translation of the configuration in Figure 2.7 would, for example, move all points the same distance to the left and leave the compass where it is.

There are two ways to think of rigid motions, the *alibi* and the *alias*. The former conceives of the transformation as a motion of the points relative to a fixed frame of reference (e.g., the pages of this book) and the latter as a motion of the frame of reference relative to points that stay put in their positions in space.

Transformations often make MDS representations easier to look at. It is important, though, to restrict such transformations to *admissible* ones, that is, to those that do not change the relations among the MDS distances that we want to represent in the MDS configuration. *Inadmissible* transformations are, on the other hand, those that destroy the relationship between MDS distances and data. For the problem above, rigid motions are certainly admissible. Also admissible are *dilations*, that is, enlargements or reductions of the entire configuration. Dilations do not affect the ratios of the distances.

Rigid motions and dilations together are termed *similarity transformations*, because they leave the shape (but not necessarily the size) of a figure unchanged. For a better overview, a summary of these transformations is given in Table 2.2. The term *invariance* denotes those properties of geometrical objects or configurations that remain unaltered by the transformation. Instead of rigid motions, one also speaks of *isometries* or, equivalently, of *isometric transformations*. This terminology characterizes more directly what is being preserved under the transformation: the metric properties of the configuration, that is, the distances between its points.

2.2 Constructing Ordinal MDS Solutions

The ruler-and-compass construction in the above attempted to represent the data such that their ratios would correspond to the ratios of the distances in the MDS space. This is called ratio MDS. In *ordinal MDS*, in contrast, one only requires that the order of the data is properly reflected

TABLE 2.2. Two important transformation groups and their invariances.

Transformation Group	Transformations	Invariance
Rigid motion (isometry)	Rotation Reflection Translation	Distances
Similarity transformation	Rotation Reflection Translation Dilation	Ratio of distances

TABLE 2.3. Ranks for data in Table 2.1; the smallest distance has rank 1.

	1	2	3	4	5	6	7	8	9	10
1	–	26	34	25	3	2	14	16	27	5
2	26	–	45	44	31	20	11	17	41	33
3	34	45	–	6	29	40	43	42	36	36
4	25	44	6	–	18	30	38	35	23	32
5	3	31	29	18	–	4	13	12	19	10
6	2	20	40	30	4	–	7	8	24	9
7	14	11	43	38	13	7	–	1	21	22
8	16	17	42	35	12	8	1	–	15	28
9	27	41	36	23	19	24	21	15	–	39
10	5	33	36	32	10	9	22	28	39	–

by the order of the representing distances. The reason for such a weaker requirement is usually that the scale level of the data is taken as merely ordinal. If only greater than and equal relations are considered informative, we could simplify Table 2.1 and replace its values by ranking numbers, because the original data are (order-)equivalent to their ranking numbers. This replacement renders Table 2.3.

Ordinal MDS is a special case of MDS, and possibly the most important one in practice. Thus, we may ask how we can proceed with our geometrical tools, ruler and compass, in constructing such an ordinal MDS solution.

A Ruler-and-Compass Approach to Ordinal MDS

The first step in ordinal MDS remains the same as above. That is, we begin by picking a pair of cities that define the first two points of the configuration. If the cities 2 and 3 are picked as before, we can use Figure 2.1 as our starting configuration. Assume now that we want to add point 9 to this configuration. What can be derived from the data to find its position relative to points 2 and 3?

Clearly, the following holds: point 9 must be closer to 3 than to 2, because the distance d_{39} must be smaller than d_{29}. This follows from Table 2.3,

because the ranking number for the distance of 3 and 9 is 36, whereas the ranking number for the distance of 2 and 9 is 41. (Note that the ranking numbers here are *dissimilarities* or *distance-like* measures; hence, a greater ranking number should lead to a greater distance.) The distances in the MDS configuration are ordered as the data are only if $d_{39} < d_{29}$. Thus, the plane in Figure 2.9 is divided into two *regions* by the perpendicular line through the middle of the line segment that connects points 2 and 3. The shaded area indicates that point 9 must lie in the region below the horizontal line if the condition $d_{39} < d_{29}$ is to be met. We call the set of points below this line the *solution set* or the *solution space* for the problem of placing point 9. Each point of this region, for example, 9, 9′, or 9″, could be chosen as point 9.

But Table 2.3 also requires that point 9 must be closer to 2 than the distance between point 2 and 3, because the rank of pair 2 and 9 is 41 and that of pair 2 and 3 is 45. Hence, $d_{29} < d_{23}$, which means that point 9 must be placed within a circle around point 2 whose radius is somewhat smaller than d_{23}. This condition is graphically illustrated in Figure 2.10 by the circle with radius $\max(d_{29})$, where $\max(d_{29})$ is "somewhat" smaller than d_{23}. Moreover, point 9 must also be placed such that $d_{39} < d_{23}$. This leads to the second circle in Figure 2.10, a circle whose radius is somewhat smaller than d_{23}.

Of course, point 9 must satisfy all three conditions at the same time. Therefore, the desired solution space in Figure 2.11 results from superimposing Figures 2.9 and 2.10.

Comparing Figure 2.2 with Figure 2.11, we see that the second solution is much more *indeterminate*, offering infinitely many possible candidates for point 9, not just two. The reason for this increased indeterminacy lies in the weaker constraints that ordinal MDS puts onto the MDS configuration: only the order of the data, not their ratios, determines the distances in MDS space. In spite of that, point 9 cannot lie just anywhere. Rather, the inequalities have led to "some" reduction of freedom in placing point 9 in the given plane.

We now arbitrarily select one point from the solution set to represent object 9: let this be point 9 in Figure 2.11. We then add a fourth point representing object 5 to the present configuration consisting of points 2, 3, and 9. Table 2.3 says that the resulting configuration must satisfy (a) $d_{25} < d_{29}$, because the corresponding ranking numbers in Table 2.3 are 31 and 41, and because the distances in the MDS representation should be ordered as the data are; (b) $d_{35} < d_{39}$, because for the corresponding ranking we find $29 < 36$; (c) $d_{59} < d_{35}$, because $19 < 29$; (d) $d_{59} < d_{25}$, because $19 < 31$; and (e) $d_{35} < d_{25}$, because $29 < 31$. These conditions each induce a boundary line bisecting the plane in Figure 2.12 into a region whose points all satisfy one of the inequalities, and a complementary region whose points violate it. Point 5 must then be so placed that it satisfies all inequality conditions, (a) through (e).

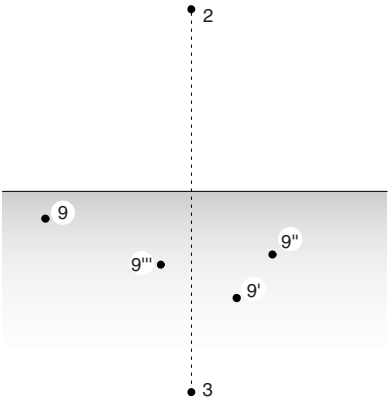


FIGURE 2.9. Solution space (shaded) for all points 9 so that $d_{39} < d_{29}$.

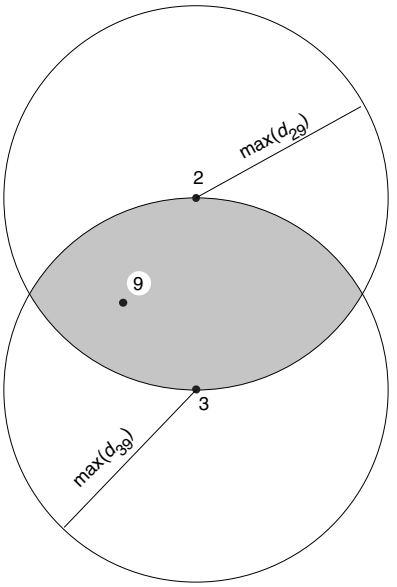


FIGURE 2.10. Solution space (shaded) for all points 9 so that $d_{29} < d_{23}$ and $d_{39} < d_{23}$.

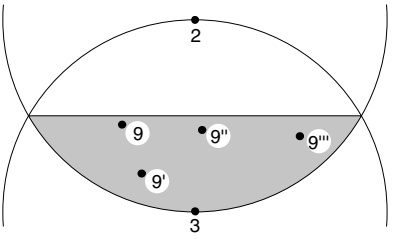


FIGURE 2.11. Solution space (shaded) for all points 9 simultaneously satisfying conditions of Figs. 2.9 and 2.10.

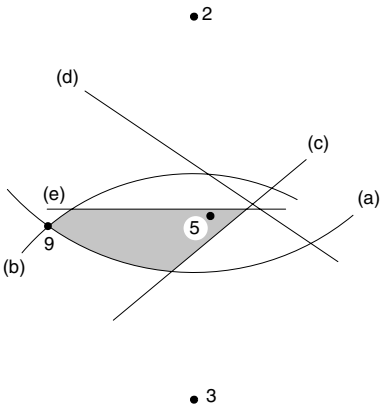


FIGURE 2.12. Solution space (shaded) for point 5.

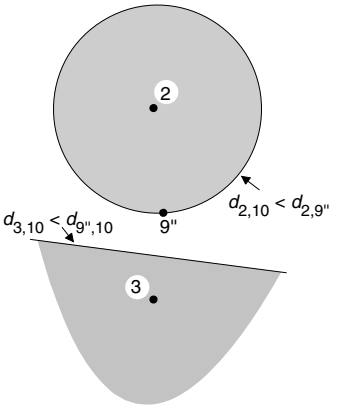


FIGURE 2.13. No point 10 can be placed into the configuration $\{2, 3, 9''\}$ so that it satisfies the shown inequalities.

Figure 2.12 shows the solution space for point 5 as a shaded area. We note that this area is smaller than the solution space for point 9 in Figure 2.11. Hence, the freedom with which we can choose a point for object 5 is less than it was for point 9 in Figure 2.11.

Proceeding in this way, we end up with an MDS representation whose distances are ordered as the ranking numbers in Table 2.3. However, finding this solution turns out not to be as straightforward as it may seem, because our construction method, in practice, would run into dead-end alleys over and over again. We show this in the next section.

Solution Spaces in Ordinal MDS

It may happen that the solution space is *empty*. In the example above, this occurs, for example, if we pick a "wrong" point for 9 in the sense that the chosen point will make it impossible to add *further* points in the desired sense. Consider an example. Assume that we had picked point $9''$ in Figure 2.11. We then would try to add a point for object 10 to the configuration $\{2, 3, 9''\}$. From Table 2.3 we note that point 10 must be closer to 2 than to $9''$, and so it must lie within the shaded circle in Figure 2.13. At the same time, point 10 must also lie below the line that is perpendicular through the midpoint of the line connecting points 3 and $9''$, because point 10 must satisfy the condition $d_{3,10} < d_{9'',10}$. But no point can simultaneously lie below this line and within the shaded circle, and so we see that the solution space for point 10 is empty. Thus, had we decided on point $9''$, we later would have had to reject this point as unacceptable for the enlarged representation problem and start all over again with a new point 9.

We also note that the solution space for each newly added point shrinks in size at a rapidly accelerating rate. Therefore, the chances for picking

wrong points for later construction steps also go up tremendously as each new point is added. Indeed, new points also have, in a way, a backwards effect: they reduce the size of the solution spaces for the old points. Every new point that cannot be properly fitted into a given configuration (as in Figure 2.13) forces one to go back and modify the given configuration until all points fit together.

The shrinkage of the solution spaces as a consequence of adding further points occurs essentially because the number of inequalities that determine the solution spaces grows much faster than the number of points in the configuration. We see this easily from our example: the solution space in Figure 2.11 is defined by three inequalities, namely, $d_{29} > d_{39}, d_{23} > d_{29}$, and $d_{23} > d_{39}$. When point 5 is added, we have four points and six distances. Because every distance can be compared to any other one, the MDS configuration must pay attention to 15 order relations.

More generally, with n points, we obtain $n \cdot n = n^2$ distances d_{ij}. Of these n^2 distances, n are irrelevant for MDS, namely, all $d_{ii} = 0, i = 1, \ldots, n$. This leaves $n^2 - n$ distances. But $d_{ij} = d_{ji}$, that is, the distance from i to j is always equal to the distance from j to i, for all points i, j. Thus, we obtain $(n^2 - n)/2 = (n)(n-1)/2$ relevant distances. This is equal to the number of pairs out of n objects, which is denoted by $\binom{n}{2}$ [read: n-take-2]. But all of these $\binom{n}{2}$ distances can be compared among each other. Consequently, we have (n-take-2)-take-2 or $\binom{\binom{n}{2}}{2}$ order relations (assuming that all values of the data matrix are different). Hence, the ranking numbers for $n = 4$ objects imply 15 inequalities; for $n = 50$, we obtain 749,700 inequalities, and for $n = 100$ there are 12,248,775 inequalities. We can understand intuitively from the sheer number of independent constraints why the ordinal MDS solution is so strongly determined, even for a fairly small n.

Isotonic Transformations

Isotonic transformations play the same role in ordinal MDS as similarity transformations in ratio MDS. Isotonic transformations comprise all transformations of a point configuration that leave the order relations of the distances unchanged (*invariant*). They include the isometric transformations discussed above as special cases.

An ordinal MDS solution is determined *up to* [1] isotonic transformations—just as the ratio MDS configurations are fixed *up to* similarity transformations—because as long as the order of the distances is not changed, any configuration is as good an ordinal MDS representation as any other. However, unless only a very small number of points is being considered, isotonic transformations allow practically no more freedom for changing the point

[1] "Up to" means that *weaker* transformations which leave even more properties invariant are also admissible.

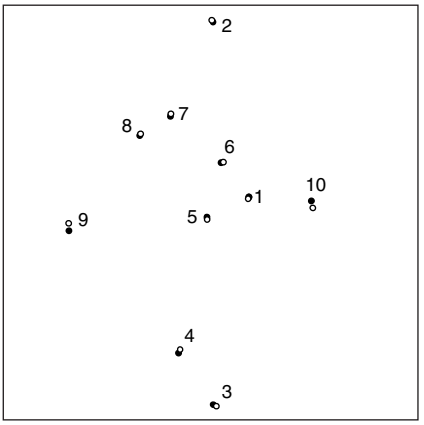


FIGURE 2.14. Comparing ratio MDS (solid points) and ordinal MDS (open circles) after fitting the latter to the former.

locations than isometric transformations. This is a consequence of the rapid shrinkage of the solution sets for the points.[2]

2.3 Comparing Ordinal and Ratio MDS Solutions

The solutions of both the ratio MDS and the ordinal MDS are shown together in Figure 2.14. The solid black points are the ratio MDS solution, and the open circles are the ordinal MDS configuration. We notice that the two configurations are very similar. This similarity has been brought out by admissibly transforming the ordinal MDS configuration so that it matches the ratio MDS configuration as much as possible. That is, leaving the former configuration fixed, we shifted, rotated, reflected, and dilated the ordinal MDS configuration so that its points $1, \ldots, 10$ would lie as close as possible to their respective target points $1, \ldots, 10$ in the ratio MDS configuration. (How this fitting is done is shown in Chapter 20.)

The fact that we obtain such highly similar structures demonstrates that treating the data as ordinal information only may be sufficient for reconstructing the original map. This seems to suggest that one gets something for free, but it really is a consequence of the fact that the order relations in a data matrix like Table 2.3 are on *pairs of pairs* of objects, not just on *pairs* of objects. In the second case, we would have weak information, indeed, and in the first, obviously not.

[2]The solution sets in ordinal MDS are also called *isotonic regions*, because the distances of each point in this set to a set of particular points outside of this set are ordered in the same way.

Ratio and ordinal MDS solutions are almost always very similar in practice. However, there are some instances when an ordinal MDS will yield a degenerate solution (see Chapter 13). Also, the positions of the points in an ordinal MDS are practically just as unique as they are in ratio MDS, unless one has only very few points. With few points, the solution spaces remain relatively large, allowing for much freedom to position the points (see, e.g., Figure 2.11).

But why do ordinal MDS at all? The answer typically relates to scale level considerations on the data. Consider the following experiment: a subject is given a 9-point rating scale; its categories range from 1 = very poor to 9 = very good; the subject judges three pictures (A, B, and C) on this scale and arrives at the judgments $A = 5, B = 7$, and $C = 1$. Undoubtedly, it is correct to say that the subject has assigned the pictures A and B more similar ratings than A and C, because $|A-B| = 2$ and $|A-C| = 4$. But it is not so clear whether the subject really felt that pictures A and B were more alike in their quality than pictures A and C. The categories of the rating scale, as used by the subject, need not correspond in meaning to the arithmetical properties of the numbers $1, 2, \ldots, 9$. For example, it is conceivable that the subject really only makes a poor-average-good distinction, or that she understands the category "very good" as "truly extraordinary", which might mean that 8 is much farther from 9 than 5 is from 6. In this case, the assigned scores 5, 7, and 1 would have a much weaker interpretability, and we could really only assert that the subject regarded B as best, A as next best, and C as worst.

2.4 On Flat and Curved Geometries

Taking a closer look at the European map in Figure 2.8, one notes that Stockholm has about the same Y-coordinate as points in Scotland. Geographically, however, Stockholm lies farther to the north than Scotland. Hence, the map is incorrect in the sense suggested by the compass in Figure 2.8, because points with the same Y-coordinates generally do not have the same geographical latitude. The distances in Table 2.1 are, on the other hand, correctly represented in Figure 2.8. But these distances were measured on a map printed on the pages of an atlas, and not measured over the *curved* surface of the globe.

Any geographical map that is *flat* is wrong in one way or another. Consider the globe in Figure 2.15, and assume that we want to produce a flat map of a relatively small region of its surface such as, for example, one of the shown spherical rectangles. This can only be done by *projecting* this region onto a flat plane, and any such projection will distort some feature of the original geometry. The usual method, for example, projects the globe's surface (except for the areas close to the poles) by rays emanating

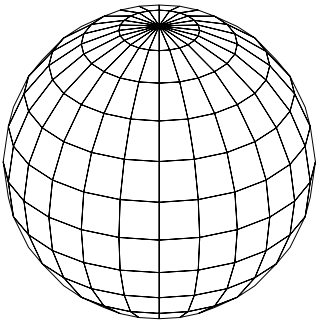


FIGURE 2.15. Globe with meridians (North–South lines) and parallels (East–West lines).

from the globe's center onto the surface of a cylinder that encompasses the globe and touches it on the Equator. The converging meridians—the lines running from the North Pole to the South Pole in Figure 2.15—thus are mapped onto parallel lines on the flat map. This projection properly represents the points' North–South coordinates on the Y-axis of the flat map. It also preserves the points' meridians as lines with the same X-coordinates. However, although the map is quite accurate for small areas, the size of the polar regions is greatly exaggerated so that, for example, Alaska looks much larger than it is. There are many other projections, which are used for different purposes. The map in Figure 2.8 is a projection that preserves area, but it is misleading when one naively reads its X–Y-coordinates as geographical longitude and latitude, respectively.

Anyone who approaches a point configuration first looks at it in the Euclidean sense. Euclidean geometry is *flat* geometry, with the flat plane as its most prominent example. Euclidean geometry is the *natural* geometry, because its properties are what they appear to be: circles look like circles, perpendicular lines look perpendicular, and the distance between two points can be measured by a straight ruler, for example. Euclidean geometry is a formalization of man's experience in a spatially limited environment. Other geometries besides the Euclidean one were discovered only by a tremendous effort of abstraction that took some 2000 years.[3] The surface of the globe

[3]Euclid, in his *Elements*, had systematically explained and proved well-known theorems of geometry such as the theorem of Pythagoras. The proofs rely on propositions that are not proved (*axioms*). One of these axioms is the parallel postulate. It says that through a point outside a straight line there passes precisely one straight line parallel to the first one. This seems to be a very special axiom. Many attempts were made to show that it is superfluous, because it can be deduced from the other axioms. "The mystery of why Euclid's parallel postulate could not be proved remained unsolved for over two thousand years, until the discovery of non-Euclidean geometry and its Euclidean models revealed the impossibility of any such proof. This discovery shattered the traditional conception of geometry as the true description of physical space ... a new conception

in Figure 2.15 is an example for a *curved* geometry. Distance is measured on the globe not with a straight ruler but with a thread stretched over its surface. This yields the shortest path between any two points and thus defines their distance. Extending the path of the thread into both directions defines a straight line, just as in Euclidean geometry. (From the outside, this path appears curved, but for the earthbound, it is "straight".) On the globe, any two straight lines will meet and, hence, there are no parallels in this kind of plane. Moreover, they will intersect in two points. For example, any two meridians meet both at the North and the South pole. Following any straight line brings you back to the point you started from, and so the globe's surface is a finite but unbounded plane. Another one of its "odd" properties is that the sum of the angles in a triangle on this plane is not a fixed quantity, but depends on the size of the triangle, whereas in Euclidean geometry these angles always add up to 180°.

Thus, the globe's surface is a geometry with many properties that differ from Euclidean geometry. Indeed, most people would probably argue that this surface is not a plane at all, because it does not correspond to our intuitive notion of a plane as a flat surface. Mathematically, however, the surface of the sphere is a consistent geometry, that is, a system with two sets of objects (called points and lines) that are linked by geometrical relations such as: for every point P and for every point Q not equal to P there exists a unique line L that passes through P and Q.

Some curved geometries are even stranger than the sphere surface geometry (e.g., the locally curved four-dimensional space used in modern physics) but none ever became important in MDS. MDS almost always is carried out in Euclidean geometry. If MDS is used as a technique for data analysis, then it is supposed to make the data accessible to the eye, and this is, of course, only possible if the geometric properties of the representation space are what they seem to be. Conversely, if MDS is non-Euclidean, then it is never used as a tool for data explorations. Rather, in this case, the properties of the representing geometry are interpreted as a substantive theory. Curved geometries play a minor role in this context. Drösler (1981), for example, used the properties of a particular two-dimensional constant-curvature geometry to model monocular depth perception. Most non-Euclidean modeling efforts remained, however, restricted to flat geometries such as the city-block plane discussed in Chapter 1, Section 1.4. In this book, we only utilize flat geometries and, indeed, mostly Euclidean geometry, unless stated otherwise.

---

emerged in which the existence of many equally consistent geometries was acknowledged, each being a purely formal logical discipline that may or may not be useful for modeling physical reality" (Greenberg, 1980, p. xi).

2.5 General Properties of Distance Representations

A geometry—whether flat or curved—that allows one to measure the distances between its points is called a metric geometry. There are usually many ways to define distances. In the flat plane, the natural way to think of a distance is the Euclidean distance that measures the length of the ruler-drawn line between two points. Another example is the city-block distance as shown in Figure 1.7. These two variants of a distance, as well as all other distances in any geometry, have a number of properties in common. These properties are important for MDS because they imply that proximities can be mapped into distances only if they too satisfy certain properties.

Consider a plane filled with points. For any two points i and j, it holds that

$$d_{ii} = d_{jj} = 0 \leq d_{ij}; \tag{2.1}$$

that is, the distance between any two points i and j is greater than 0 or equal to 0 (if $i = j$). This property is called *nonnegativity* of the distance function. Furthermore, for any two points i and j, it is true that

$$d_{ij} = d_{ji}; \tag{2.2}$$

that is, the distance between i and j is the same as the distance between j and i (*symmetry*). Finally, for all points i, j, k, it holds that

$$d_{ij} \leq d_{ik} + d_{kj}. \tag{2.3}$$

This *triangle inequality* says that going directly from i to j will never be farther than going from i to j via an intermediate point k. If k happens to be on the way, then (2.3) is an equality.

These properties, which are obviously true for distances in the familiar Euclidean geometry, are taken as the definitional characteristics (*axioms*) of the notion of distance. One can check whether any given function that assigns a numerical value to pairs of points (or to any pair of objects) possesses these three properties.

Consider, for example, the *trivial distance* defined by $d_{ij} = 1$ (if $i \neq j$) and $d_{ij} = 0$ (if $i = j$). To prove that this function is a distance, we have to show that it satisfies the three distance axioms. Starting with nonnegativity, we find that we have $d_{ii} = 0$ for all i by the second part of the definition, and that $d_{ij} > 0$ for all $i \neq j$ by the first part of the definition. Symmetry also holds because the function is equal to 1 for all $i \neq j$. Finally, for the triangle inequality, we obtain $1 < 1 + 1$ if i, j, and k are all different; $1 = 1 + 0$ if $k = j$ and so on. Hence, the left-hand side of the inequality can never be greater than the right-hand side.

Naturally, the trivial distance is not a particularly interesting function. Even so, it can still serve as a nontrivial psychological model. If it is used

as a primitive model for liking, it may turn out empirically wrong for a given set of persons if there are persons who like somebody else more than themselves.

We may also have proximities where $p_{ij} = p_{ji}$ does not hold for all i and j. The proximities, in other words, are not symmetric. Such proximities are rather typical for the social relation "liking" between persons. If such nonsymmetry is observed and if it cannot be interpreted as due to error, then the given data cannot be represented directly in *any* metric geometry. Symmetry, thus, is always a precondition for MDS.

The other properties of distances may or may not be necessary conditions for MDS. If one has observed "self"-proximities for at least two p_{ii}s and if they are not all equal or if any p_{ii} is greater than any p_{ij} (for $i \neq j$) proximity then, strictly speaking, one cannot represent these proximities by any distance. If the proximities violate the triangle inequality, it may or may not be relevant for MDS. In ordinal MDS, it is no problem because adding a sufficiently large constant to all p_{ij}s eliminates all violations (see Section 18.2). In ratio MDS, however, the proximities are assumed to have a fixed origin and no such arbitrary additive constants are admissible. Hence, violations of the triangle inequality are serious problems. If they are considered large enough, they exclude any distance representation for the data.

2.6 Exercises

Exercise 2.1 If you square the correlations in Exercises 1.1, 1.2, or 1.4, and then do ordinal MDS, you obtain exactly the same solutions as for the original values.

(a) Explain why.

(b) Specify three other transformations that change the data values substantially but lead to the same ordinal MDS solutions as the raw data.

(c) Specify a case where such a transformation of the data values changes the ordinal MDS solution.

Exercise 2.2 Specify the admissible transformations for the city-block ordinal MDS solution in Figure 1.7.

Exercise 2.3 Consider the table of distances between five objects below.

Object	1	2	3	4	5
1	0	1.41	3.16	4.00	8.06
2	1.41	0	2.00	3.16	8.54
3	3.16	2.00	0	1.41	8.06
4	4.00	3.16	1.41	0	7.00
5	8.06	8.54	8.06	7.00	0

(a) Use the ruler-and-compass method described in Section 2.1 to construct a ratio MDS solution. Choose the scale factor s equal to 1, so that the distance between points 1 and 4 should be equal to 4 cm in your solution.

(b) Connect points 1 to 2 by a line, points 2 and 3, etc. What pattern emerges?

(c) Verify your solution by using an MDS program. Explain possible differences between the two solutions obtained by hand and by using the computer program.

Exercise 2.4 A psychologist investigates the dissimilarity of the colors red, orange, green, and blue. In a small experiment, she asks a subject to rank the six pairs of colors on their dissimilarity (1 = most similar, 6 = most dissimilar). The resulting table of ranks is given below.

Item	R	O	G	B
Red	–			
Orange	1	–		
Green	3	2	–	
Blue	5	6	4	–

The psychologist wants to do an ordinal MDS in two dimensions on these data but does not have an MDS program for doing so. So far, she has found the coordinates for Red (0, 3), Orange (0, 0), and Green (4, 0).

(a) Use the ruler-and-compass method described in Section 2.2 to find a location for point Blue that satisfies the rank-order of the data. Specify the region where Blue may be located.

(b) Interpret your solution substantively.

(c) Suppose that none of the coordinates were known. Try to find an ordinal MDS solution for all four points. Does this solution differ from the one obtained in (a)? If so, explain why.

3
MDS Models and Measures of Fit

MDS models are defined by specifying how given similarity or dissimilarity data, the proximities p_{ij}, are mapped into distances of an m-dimensional MDS configuration $\mathbf{X}$. The mapping is given by a *representation function* $f(p_{ij})$ that specifies how the proximities should be related to the distances $d_{ij}(\mathbf{X})$. In practice, one usually does not attempt to strictly satisfy f. Rather, what is sought is a configuration (in a given dimensionality) whose distances satisfy f as closely as possible. The condition "as closely as" is quantified by a badness-of-fit measure or *loss function*. The loss function is a mathematical expression that aggregates the representation errors, $e_{ij} = f(p_{ij}) - d_{ij}(\mathbf{X})$, over all pairs (i, j). A normed sum-of-squares of these errors defines *Stress*, the most common loss function in MDS. How Stress should be evaluated is a major issue in MDS. It is discussed at length in this chapter, and various criteria are presented.

3.1 Basics of MDS Models

In this section, MDS models are defined and discussed on a level sufficient for most practical applications. In later chapters, we revisit some of the relevant issues in greater detail.

Assume that measures of similarity or dissimilarity, for which we use the general term *proximity*, p_{ij}, are given for the pairs (i, j) of n objects. Some examples for such proximities were discussed in Chapter 1: similarities of crimes, assessed by the correlations of their frequencies over different U.S.

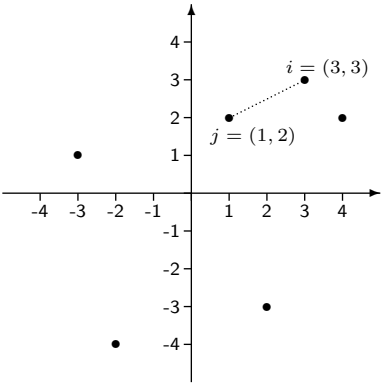


FIGURE 3.1. A Cartesian plane with some points; the length of the line segment connecting points i and j is the (Euclidean) distance of points i and j.

states; correlations among attitudes towards political protest behaviors; direct ratings of the overall similarity of pairs of different countries; and similarity judgments on one-spoked wheels. All of these cases are examples of measures of similarity, because the higher a correlation (or a rating of similarity), the more similar the objects i and j. However, instead of asking for judgments of similarity, it is just as easy—or even easier—to ask for judgments of dissimilarity, for example, by presenting a rating scale ranging from 0 = no difference to 10 = very dissimilar.

Coordinates in the MDS Space

MDS attempts to represent proximities by distances among the points of an m-dimensional configuration $\mathbf{X}$, the MDS space. The distances can be measured by a ruler, up to a certain level of precision, and if the MDS space is at most three-dimensional. But distances can also be *computed* with arbitrary precision, and this can be done in a space of arbitrarily high dimensionality. Computation is made possible by *coordinating* the MDS space. The most common such coordination is first to define a set of m directed axes that are perpendicular to each other and intersect in one point, the *origin* O. These axes—in the applied context often called *dimensions*—are then divided up into intervals of equal length so that they represent, in effect, a set of perpendicular "rulers".

Each point i, then, is uniquely described by an m-tuple $(x_{i1}, x_{i2}, \ldots, x_{im})$, where x_{ia} is i's projection onto dimension a. This m-tuple is point i's *coordinate vector*. The origin O is given the coordinates $(0, 0, \ldots, 0)$. Figure 3.1 shows some points and their coordinate vectors in a *Cartesian plane*, that is, in a plane coordinated by a set of perpendicular dimensions.

Computing Distances

Given a Cartesian space, one can compute the distance between any two of its points, i and j. The most frequently used and the most natural distance function is the Euclidean distance. It corresponds to the length of the straight line[1] segment that connects the points i and j. Figure 3.1 shows an example.

The Euclidean distance of points i and j in a two-dimensional configuration $\mathbf{X}$ is computed by the following formula:

$$d_{ij}(\mathbf{X}) = \sqrt{(x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2}. \tag{3.1}$$

Thus, $d_{ij}(\mathbf{X})$ is equal to the square root of the sum of the intradimensional differences $x_{ia} - x_{ja}$, which is simply the Pythagorean theorem for the length of the hypotenuse of a right triangle. For Figure 3.1, thus, formula (3.1) yields $d_{ij} = \sqrt{(3-1)^2 + (3-2)^2} = \sqrt{5}$. Formula (3.1) can also be written as

$$d_{ij}(\mathbf{X}) = \left[\sum_{a=1}^{2}(x_{ia} - x_{ja})^2\right]^{1/2}, \tag{3.2}$$

which can easily be generalized to the m-dimensional case as

$$d_{ij}(\mathbf{X}) = \left[\sum_{a=1}^{m}(x_{ia} - x_{ja})^2\right]^{1/2}. \tag{3.3}$$

MDS Models and Their Representation Functions

MDS maps proximities p_{ij} into corresponding distances $d_{ij}(\mathbf{X})$ of an MDS space $\mathbf{X}$. That is, we have a *representation function*

$$f : p_{ij} \to d_{ij}(\mathbf{X}), \tag{3.4}$$

where the particular choice of f specifies the *MDS model*. Thus, an MDS model is a proposition that given proximities, after some transformation f, are equal to distances among points of a configuration $\mathbf{X}$:

$$f(p_{ij}) = d_{ij}(\mathbf{X}). \tag{3.5}$$

[1]The term "straight" corresponds to what we mean by straight in everyday language. In Euclidean geometry, a straight line can be drawn by tracing with a pen along a ruler. More generally, a straight line is the *shortest* path (geodesic) between two points. The notion of straightness, therefore, presupposes a distance measure. With different distance measures, straight lines often do not look straight at all. An example is the straight line between points 4 and 6 in Figure 1.7, which consists of the two dashed line segments that *look* like a "corner" line.

The distances $d_{ij}(\mathbf{X})$ in (3.4) and (3.5) are always unknowns. That is, MDS must find a configuration $\mathbf{X}$ of predetermined dimensionality m on which the distances are computed. The function f, on the other hand, can either be completely specified or it can be restricted to come from a particular class of functions. Shepard (1957), for example, collected similarities p_{ij} for which he predicted, on theoretical grounds, that they should be related to distances in an unknown two-dimensional space $\mathbf{X}$ by the exponential function. That is, it was hypothesized that $p_{ij} = \exp[-d_{ij}(\mathbf{X})]$. Similarly, Thurstone (1927) predicted that choice probabilities p_{ij} should be equal to unknown distances between points i and j on a line ("scale values") after transforming the p_{ij}s by the inverse normal distribution function. This choice of f, again, was theoretically justified.

In most applications of MDS, there is some looseness in specifying f. That is, for example, f is only restricted to be "some" exponential function or "some" linear function. The exact parameters of these functions are not specified. An important case is *interval MDS*. It is defined by

$$p_{ij} \to a + b \cdot p_{ij} = d_{ij}(\mathbf{X}), \tag{3.6}$$

for all pairs (i, j). The parameters a and b are free and can be chosen such that the equation holds. Another case is *ordinal MDS*, where f is restricted to be a monotone function that preserves the order of the proximities. That means—assuming, for simplicity, that the proximities are dissimilarity scores—that

$$\text{if } p_{ij} < p_{kl}, \text{ then } d_{ij}(\mathbf{X}) \leq d_{kl}(\mathbf{X}). \tag{3.7}$$

If $p_{ij} = p_{kl}$, (3.7) requires no particular relation of the corresponding distances. This is known as the *primary approach* to tied proximities, where ties can be "broken" in the corresponding distances. The *secondary approach* to ties requires that if $p_{ij} = p_{kl}$, then also $d_{ij} = d_{kl}$. The primary approach is the default in most ordinal MDS programs. A slight modification of (3.7) is to replace the relation $\leq$ by $<$. The first relation specifies a *weak* monotone function f, the second one a *strong* monotone function. Most often, ordinal MDS is used with a weak monotone function.

How should one choose a particular representation function? If no particular f can be derived by theoretical reasoning, one often restricts f to a particular class of functions on the basis of the scale level of the proximities. For example, if the proximities are direct similarity ratings on, say, pairs of nations, one might feel that only their rank-order yields reliable information about the respondent's true cognitions. Differences ("intervals") between any two ratings, in contrast, would not represent any corresponding psychological quantities. Under these assumptions, there is no reason to insist that these intervals be faithfully represented by distances in the MDS space. Moreover, a weak scale level makes it easier to approximately represent the essential information in an MDS space of low dimensionality.

Conversely, starting from an MDS model, one can choose a representation function g in the regression hypothesis $g : d_{ij}(\mathbf{X}) \rightarrow p_{ij}$. This hypothesis needs to be tested against the data. One can pick any g: if it leads to a model that is empirically satisfied—and provided that the model does not hold for formal reasons only—one has shown a nontrivial empirical regularity. No further justification is needed for picking a particular function g.

3.2 Errors, Loss Functions, and Stress

MDS models require that each proximity value be mapped *exactly* into its corresponding distance. This leaves out any notion of error. But empirical proximities always contain noise due to measurement imprecision, unreliability, sampling effects, and so on. Even the distances used in Table 2.1 are not completely error-free, because reading off values from a ruler only yields measures of limited precision. Hence, one should not insist, in practice, that $f(p_{ij}) = d_{ij}(\mathbf{X})$, but rather that $f(p_{ij}) \approx d_{ij}(\mathbf{X})$, where $\approx$ can be read as "as equal as possible". Given that the proximities contain some error, such approximate representations make even *better* representations—more robust, reliable, replicable, and substantively meaningful ones—than those that are formally perfect, because they may smooth out noise.

If one has a theory about the proximities, one would be interested to see how well this theory is able to explain the data, and so a best-possible MDS representation (of some sort) is sought. If the error of representation is "too large," one may reject or modify the theory, but obviously one first needs to know how well the theory accounts for the data. Any representation that is precise enough to check the validity of this theory is sufficiently exact. A perfect representation is not required.

Further arguments can be made for abandoning the equality requirement in $f(p_{ij}) = d_{ij}(\mathbf{X})$. Computerized procedures for finding an MDS representation usually start with some initial configuration and improve this configuration by moving around its points in small steps ("iteratively") to approximate the ideal model relation $f(p_{ij}) = d_{ij}(\mathbf{X})$ more and more closely. As long as the representation is not perfect, one only has $f(p_{ij}) \approx d_{ij}(\mathbf{X})$, where $\approx$ means "equal except for some small discrepancy".

The Stress Function

To make such notions as "almost", "nearly", and so on, more precise, we employ the often used statistical concept of error. A (squared) *error of representation* is defined by

$$e_{ij}^2 = [f(p_{ij}) - d_{ij}(\mathbf{X})]^2. \tag{3.8}$$

Summing e_{ij}^2 over all pairs (i, j) yields a badness-of-fit measure for the entire MDS representation, *raw Stress*,

$$\sigma_r = \sigma_r(\mathbf{X}) = \sum_{(i,j)} [f(p_{ij}) - d_{ij}(\mathbf{X})]^2. \tag{3.9}$$

The raw Stress value itself is not very informative. A large value does not necessarily indicate bad fit. For example, suppose that the dissimilarities are road distances between cities in kilometers. Suppose that an MDS analysis on these data yields $\sigma_r(\mathbf{X}_1) = .043$. Redoing the analysis with dissimilarities expressed in meters yields the same solution, but on a scale that is 1000 times as large, and so one gets $\sigma_r(\mathbf{X}_2) = 43,000$. This does not mean that $\mathbf{X}_2$ fits the data worse than $\mathbf{X}_1$; it merely reflects the different calibration of the dissimilarities. To avoid this scale dependency, σ_r can, for example, be normed as follows,

$$\sigma_1^2 = \sigma_1^2(\mathbf{X}) = \frac{\sigma_r(\mathbf{X})}{\sum d_{ij}^2(\mathbf{X})} = \frac{\sum [f(p_{ij}) - d_{ij}(\mathbf{X})]^2}{\sum d_{ij}^2(\mathbf{X})}. \tag{3.10}$$

Taking the square root of σ_1^2 yields a value known as *Stress-1* (Kruskal, 1964a). The reason for using σ_1 rather than σ_1^2 is that σ_1^2 is almost always very small in practice, so σ_1 values are easier to discriminate. Thus, more explicitly,

$$\text{Stress-1} \ = \sigma_1 = \sqrt{\frac{\sum [f(p_{ij}) - d_{ij}(\mathbf{X})]^2}{\sum d_{ij}^2(\mathbf{X})}}. \tag{3.11}$$

The summations extend over all p_{ij} for which there are observations. Missing data are skipped. In the typical case of symmetric proximities, where $p_{ij} = p_{ji}$ (for all i, j), it suffices to sum over one half of the data-distance pairs only. Obviously, $\sigma_1 = 0$ only if $d_{ij}(\mathbf{X}) = f(p_{ij})$.

Minimizing Stress-1 always requires finding an optimal $\mathbf{X}$ in a given dimensionality m. Moreover, if f is only specified up to certain free parameters, then optimal values for these parameters must also be found. This problem typically is solved by regressing the proximities onto the distances computed on $\mathbf{X}$. In interval MDS, one uses linear regression, in ordinal MDS monotone regression (see Section 9.2). The regression yields transformed proximities, $f(p_{ij})$s, that are "approximated distances" or "d-hats" ($\widehat{d}_{ij}$s) also referred to as *disparities* in the MDS-literature.

3.3 Stress Diagrams

Loss functions such as Stress are indices that assess the mismatch of (admissibly transformed) proximities and corresponding distances. Stress is, in a way, similar to a correlation coefficient, except that it measures the badness-of-fit rather than the goodness-of-fit. Experienced researchers know that

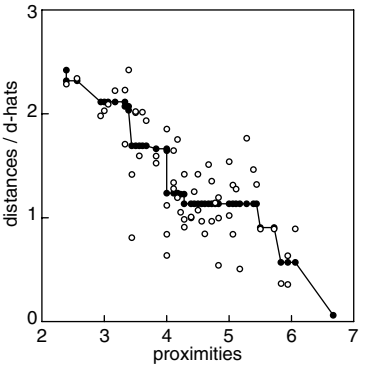


FIGURE 3.2. Shepard diagram for MDS solution shown in Fig. 1.5.

correlations can be high or low for various reasons. For example, a correlation can be artificially high because of outliers. It can also be misleadingly low because the regression trend is not linear. What one usually does to study such questions is to take a look at the scatter diagram.

Exactly the same approach is also customary in MDS. The most informative scatter diagram plots proximities on the X-axis against the corresponding MDS distances on the Y-axis. Typically, a regression line that shows how proximities and approximated distances ($\widehat{d}_{ij}$s) are related is also shown. This plot is known as a *Shepard diagram*.

Figure 3.2 gives an example. The Shepard diagram exhibits, as open circles, the similarities of Table 1.3 plotted against the corresponding distances of Figure 1.5. The filled circles represent the $(p_{ij}, \widehat{d}_{ij})$ pairs. They all lie on a monotonically descending line, as requested by the ordinal MDS model used to scale these data. The vertical distance of each (p_{ij}, d_{ij}) point (open circle) from the $(p_{ij}, \widehat{d}_{ij})$ point (filled circle) represents the error of representation for this particular proximity, e_{ij}. The Y-axis of the Shepard diagram has two labels: distances (d_{ij}s) and approximated distances ($\widehat{d}_{ij}$s).

What can be learned from this Shepard diagram? First, it gives an overall impression of the scatter around the representation function. In Figure 3.2, one notes that there is quite a bit of scatter around the monotone regression curve. The vertical distances of the points from the step function (e_{ij}s) are generally quite large, and thus $\sigma_1 = .186$. Then, one notes that there are no real outliers, although some points contribute relatively much to Stress. The most prominent case is the point with coordinates $(3.44, 0.82)$. Its error or "residual," which enters the Stress function quadratically, is -0.877, and the second greatest residual is only 0.636. One finds in Table 1.3 that there are two dissimilarity estimates of 3.44, one for India vs. France and one for Brazil vs. Egypt. The MDS program keeps track of each and every proximity and informs us that the large residual is related to the pair India–France. Hence, this observation is explained worst by

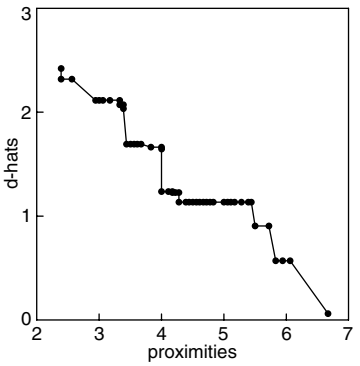


FIGURE 3.3. A transformation plot (scatter diagram of proximities vs. d-hats) for the MDS solution shown in Fig. 1.5.

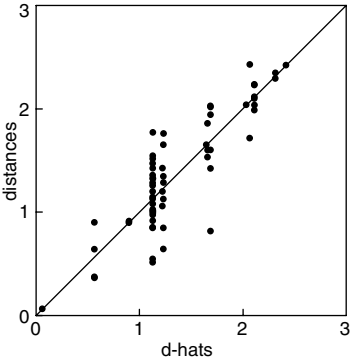


FIGURE 3.4. A residual plot (scatter diagram of d-hats vs. distances) for the MDS solution shown in Fig. 1.5.

the MDS space in Figure 1.5, possibly because it brings in an additional dimension.

A Shepard diagram is particularly informative in the case of ordinal MDS. This model requires a monotone representation function f, but its particular shape is left open. It is often interesting to see which shape it acquires in scaling real data. [Indeed, this question motivated the invention of ordinal MDS (see Chapter 17).] In Figures 3.2 and 3.3, we note that the regression curve is roughly linear, although it shows a number of marked steps.

Some MDS programs also provide scatter plots of the $\widehat{d}_{ij}$s vs. the corresponding d_{ij}s. Figure 3.4 gives an example for the data in Table 1.3. The points in such a plot scatter around the bisector from the lower left-hand corner to the upper right-hand corner. If Stress is zero, they all lie on this bisector; otherwise, they do not. The vertical distance of the points from the bisector corresponds to the error of approximation, but the horizontal distances have the same magnitude, $|e_{ij}|$. The outlier discussed above, the proximity for France vs. India, has coordinates 0.815 on the vertical axis and 1.69 on the horizontal axis. It lies farthest from the bisector. Generally, what one studies in such plots is the distribution of the points around this bisector for possible outliers, anomalies, gaps, and so on.

3.4 Stress per Point

In the previous section, we have looked at how well each proximity p_{ij} or its transformation $\widehat{d}_{ij}$ is fitted by the corresponding distance d_{ij}. The error for one particular proximity is the vertical distance between $\widehat{d}_{ij}$ and the d_{ij}

TABLE 3.1. Squared error for the solution in Figure 1.5 of the similarity ratings for 12 nations (Wish, 1971). The last row (and column) contains the average per row (or column) and is called Stress per point.

Nation		1	2	3	4	5	6	7	8	9	10	11	12	SPP
Brazil	1	–	.02	.24	.09	.00	.08	.08	.02	.00	.00	.07	.00	.05
Congo	2	.02	–	.01	.07	.00	.03	.00	.04	.01	.00	.00	.05	.02
Cuba	3	.24	.01	–	.09	.01	.05	.05	.02	.01	.00	.00	.00	.04
Egypt	4	.09	.07	.09	–	.01	.02	.07	.01	.01	.00	.08	.00	.04
France	5	.00	.00	.01	.01	–	.23	.21	.17	.01	.02	.01	.01	.06
India	6	.08	.03	.05	.02	.23	–	.00	.04	.03	.01	.01	.00	.04
Israel	7	.08	.00	.05	.07	.21	.00	–	.04	.00	.00	.00	.02	.04
Japan	8	.02	.04	.02	.01	.17	.04	.04	–	.10	.01	.00	.02	.04
China	9	.00	.01	.01	.01	.01	.03	.00	.10	–	.00	.00	.06	.02
USSR	10	.00	.00	.00	.00	.02	.01	.00	.01	.00	–	.04	.00	.01
U.S.A	11	.07	.00	.00	.08	.01	.01	.00	.00	.00	.04	–	.00	.02
Yugoslavia	12	.00	.05	.00	.00	.01	.00	.02	.02	.06	.00	.00	–	.01
Stress per point		.05	.02	.04	.04	.06	.04	.04	.04	.02	.01	.02	.01	.03

in the Shepard diagram. Instead of looking at a single error only, it may be more interesting to consider all errors of one object to all others. We examine the definition of raw Stress in (3.9) more closely. Clearly, raw Stress is a sum of the squared errors over all pairs of objects. Table 3.1 contains the squared error for the solution in Figure 1.5 of the similarity ratings for twelve nations (Wish, 1971). Note that for convenience, this table shows the squared errors below and above the diagonal, although because of the symmetry the errors below (or above) the diagonal would suffice. Now, a simple measure to indicate how badly each individual point is fitted can be obtained by averaging the squared errors between the current object and all other objects. We call this measure *Stress per point* and it is shown in the last column (and the last row) of Table 3.1. For example, the Stress per point for France can be obtained by averaging all the squared errors in the row of France in Table 3.1. Equivalently, the same value is obtained by averaging column 5 for France in this table. An additional feature of Stress per point is that their average equals the total Stress. Because Stress per point is defined on the squared errors, we must square σ_1 to compare it with the average Stress per point. In the previous section, we found that $\sigma_1 = .186$, so that $\sigma_1^2 = .0346 \approx .03$ is the same value indeed as the element in the lower right-hand corner of Table 3.1.

Several conclusions can be drawn from this table. First, most points are fitted rather well by this solution, because their Stress per point is reasonably low. Second, the best fitting points are Yugoslavia and the USSR, followed by U.S.A., China, and Congo. Third, the worst fitting points are France and Brazil. When interpreting the solution, this information should be kept in mind. Apparently, the MDS solution in Figure 1.5 is not very well able to represent the points for France and Brazil. Their Stress per

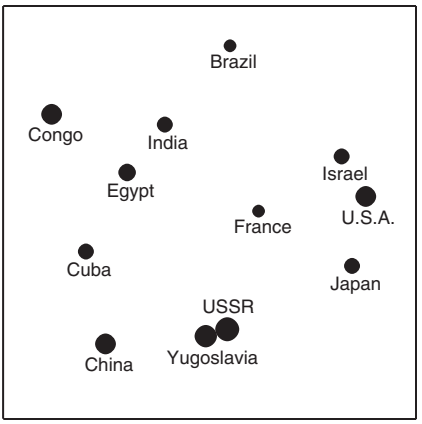


FIGURE 3.5. Bubble plot of fit per point derived from Stress per point of the similarity nations data of Wish (1971). Big bubbles indicate points with good fit, small bubbles indicate points with poor fit.

point is relatively high because there is quite some difference between the distances and the transformed data with all other countries. It can be verified in Table 3.1 that the high Stress per point for France is caused, in particular, by high errors of France with India, Israel, and Japan. A high Stress per point indicates that we cannot be certain about the exact location of this point. It may be an indication that an additional dimension is needed for these points to reduce the error.

To inspect the Stress per point graphically, it is simpler to switch to the *fit per point* that is defined as one minus the Stress per point. Generally, the fit per point is a value between zero and one. Usually, the fit per point is close to one. In our example, the fit per point varies between .99 for Yugoslavia and USSR and .94 for France. In Figure 3.5, the fit per point is expressed by the radius of the bubble representing the point. The centers of the bubbles are the locations of the points, just as in Figure 1.5. To avoid too little discrimination in the size of the bubbles, we linearly transformed the radii such that the worst fitting point (France) has a radius twice as small as the best fitting point (Yugoslavia). It can be seen in Figure 3.5 that the best fitting points (with the largest bubble) are mostly located around the edges (with the exception of Brazil) and that the worst fitting points are located towards the center (such as, for example, France). To interpret the solution, Figure 3.5 shows immediately which points should be emphasized in the interpretation because of their good fit per point, and which points should not be emphasized because of their bad fit.

3.5 Evaluating Stress

How should one evaluate the Stress of a given MDS solution? One approach is to study the Shepard diagram. It shows the number of points that have to be fitted, the optimal regression line, the size of the deviations, possible outliers, and systematic deviations from the requested regression line. Thus, Shepard diagrams are highly informative. Nevertheless, it is customary to condense all of this information into a single number, Stress.

In ordinal MDS, *any* matrix of proximities p_{ij} $(i < j)$ can be represented, with zero Stress, in $m = n - 2$ dimensions (see Chapter 19). However, such perfect solutions are not desired, as we saw above. Therefore, one seeks an MDS representation with considerably fewer dimensions. The problem is how to choose the "proper" dimensionality. Scaling with too few dimensions may distort the true (reliable) MDS structure due to *over-compression* or may lead to technical problems (see Chapter 13). Being too generous on dimensions may, on the other hand, blur the MDS structure due to *overfitting* noise components. If information is available about the reliability of the data, one should choose a dimensionality whose Stress corresponds to the random component of the data. Inasmuch as this information is rarely given, one has to resort to other criteria.

Simple Norms for Stress

Beginners in multivariate data analysis typically ask for simple (often overly so) norms. In MDS, a number is requested so that whenever Stress is less than that benchmark value, the MDS solution should be considered acceptable. Guttman (in Porrat, 1974) proposes such a norm for a coefficient closely related to Stress: he required that the coefficient of alienation K should be less than 0.15 for an acceptably precise MDS solution. He later added that what he had in mind when he made this proposal were "the usual circumstances" (Guttman, personal communication). [Note that here and in the following, we are considering ordinal MDS only.]

It is easy to see that such circumstances are important. Any global fit measure will be low, for example, when the number of points n is small relative to the dimensionality of the space, m. Guttman thus assumed for the $K < 0.15$ rule that n "clearly" exceeds m (as another rule of thumb, at least fourfold: Rabinowitz, 1975; Kruskal & Wish, 1978). Conversely, if n is much larger than m (more than 10 times as large, say), higher badness-of-fit values might also be acceptable.

Another rough criterion is to pick that solution "for which further increase in $[m]$ does not significantly reduce Stress" (Kruskal, 1964a, p. 16). To find that m, one should first compute MDS solutions for different dimensionalities (e.g., for $m = 1, 2, \ldots, 5$) and then plot the resulting Stress values (on the Y-axis) against the m-values (on the X-axis). If the points in this diagram are connected by a line, starting at $m = 1$ and ending at

$m = \max$, one obtains a *scree plot*. (An example of a scree plot is given in Figure 4.5.)

The curve in a scree plot is generally monotonically decreasing, but at an increasingly slower rate with more and more dimensions (convex curve).[2] What one looks for is an *elbow* in this curve, a point where the decrements in Stress begin to be less pronounced. That point corresponds to the dimensionality that should be chosen. The rationale of this choice is that the elbow marks the point where MDS uses additional dimensions to essentially only scale the noise in the data, after having succeeded in representing the systematic structure in the given dimensionality m.

For the Stress-1 coefficient σ_1 using ordinal MDS, Kruskal (1964a), on the basis of his "experience with experimental and synthetic data" (p. 16), suggests the following benchmarks: .20 = poor, .10 = fair, .05 = good, .025 = excellent, and .00 = perfect.[3] Unfortunately, such criteria almost inevitably lead to misuse by suggesting that only solutions whose Stress is less than .20 are acceptable, or that all solutions with a Stress of less than .05 are good in more than just a formal sense. Neither conclusion is correct. An MDS solution may have high Stress simply as a consequence of high error in the data, and finding a precise representation for the data does not imply anything about its scientific value.

Obviously, one needs more systematic insights into how Stress depends on the number of points, the dimensionality of the MDS solution, the kind and amount of error in the proximities, the type of the underlying true configuration, and so on. Computer simulation studies can help to answer such questions. In the following, we consider some such studies.

Stress for Random Data

The most extreme case that can be studied is concerned with the "nullest of all null hypotheses" (Cliff, 1973), that is, with the question of whether the Stress for some given data is significantly lower than for random data. Stenson and Knoll (1969) and Klahr (1969) compute the distribution of Stress values for ordinal MDS under H_0 as follows: (a) pick some values for n, the number of the points, and m, the dimensionality of the MDS space; (b) randomly insert the numbers $1, 2, 3, \ldots, \binom{n}{2}$ into the cells of a lower-half proximity matrix; (c) use ordinal MDS on these proximities and

[2] An exception to that rule can result, for example, when the MDS computer program does not succeed in finding the optimal solution for some dimensionality. The scree test can, therefore, occasionally be useful to identify such suboptimal solutions.

[3] A Stress value of .20, say, is often written as 20%. Why this language became popular is not entirely clear. However, if one replaces Stress by squared Stress, then one can show that, for example, 20% (squared) Stress means that 80% of the variance of the d-hats is explained by the distances (see Section 11.1).

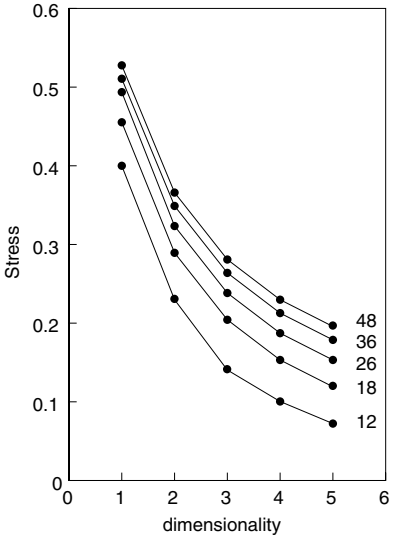


FIGURE 3.6. Average Stress for random proximities among n objects, represented via ordinal MDS in different dimensionalities (Spence & Ogilvie, 1973).

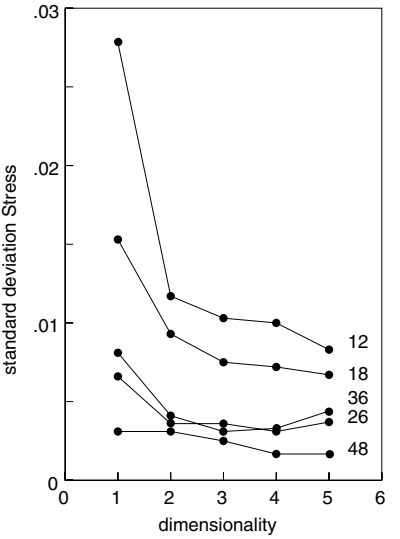


FIGURE 3.7. Standard deviations for curves in Fig. 3.6.

compute Stress; and (d) repeat the above for many permutations of the data, so that a distribution of Stress values results.

These simulations show that if n grows, then expected Stress also grows and its variance becomes smaller; if m grows, then expected Stress becomes smaller. If the data contain ties, the primary approach leads to lower Stress (because ties are optimally broken) than the secondary (where ties in the data must be preserved in the distances); the more ties there are, the larger the difference.

Spence and Ogilvie (1973) conduct a similar investigation for $n = 12, 13, \ldots, 48$ points and $m = 1, 2, \ldots, 5$ dimensions, a useful range for many practical purposes. Figure 3.6 shows the average Stress curves for various n values, using ordinal MDS. The curves indicate again that Stress depends on n and m. One also notes that each additional dimension reduces Stress increasingly less. The confidence intervals of the expected Stress values are quite narrow, as the standard deviations of the Stress distributions in Figure 3.7 show. The standard deviations are so small that lowering the curves by about 0.03 should result in reliable cutoff values for testing this H_0.

Spence (1979) has shown that one can closely approximate the curves in Figure 3.6 and curves interpolated therein for $n = 12, 13, \ldots, 48$ by the

formula

$$\sigma_1 = .001(a_0 + a_1 m + a_2 n + a_3 \ln(m) + a_4 \sqrt{\ln(n)}), \tag{3.12}$$

where $a_0 = -524.25, a_1 = 33.8, a_2 = -2.54, a_3 = -307.26$, and $a_4 = 588.35$. A comparison of the results with those from Stenson and Knoll (1969) shows very good agreement, so that formula (3.12) can be used to estimate expected "random" Stress for the range $n = 10, \ldots, 60$ and $m = 1, \ldots, 5$.

We show that the Stress values in all real-data MDS applications discussed in this book lie definitely under the values expected for H_0. This also shows that this kind of null hypothesis represents a very small hurdle indeed. On the other hand, if one does not even succeed in rejecting this H_0, then it seems unreasonable to study the MDS representation further.

The Hefner Model

Simulations that study the distribution of Stress for random data (of some sort) are useful from a data-analytic point of view. They do not attempt to simulate an MDS model in the sense of a psychological theory about similarity judgments. If MDS is used in this way, then one also needs a more explicit model for what is meant by the "random" component of the data.

Consider the similarity-of-nations example in Section 1.3. We may want to assume that a respondent arrives at his or her overall similarity judgment by first computing the distance of two nations in his or her system of dimensions or *perceptual space*, and then mapping this distance into the response format provided by the researcher. Moreover, we could postulate that the perceptual space is not static, but that its points "oscillate" about their characteristic position over time. The oscillations could be due to unsystematic variations in attention, fluctuating discrimination thresholds, activation and decay processes on the memory traces, and so on. Under these conditions, the respondent would compute a distance at each point in time, but these distances would not fit together in a plane, because each distance depends on the particular positions of the points at time t, and these positions are not constant over time. An observed proximity, after transformation by f, is thus conceived as $f(p_{ij}) = d_{ij}^{(e)} = \sum_{a=1}^{m} [x_{ia}^{(e)} - x_{ja}^{(e)})^2]^{1/2}$, where $x_{ia}^{(e)} = x_{ia} + e_{ia}$ and e_{ia} is a value from the random distribution of point i.

For the "error" terms e_{ia}, one can postulate a particular distribution over time. A commonly used assumption is that the points oscillate symmetrically in all directions of the MDS space around their characteristic (true) locations. It is usually assumed that these distributions are normal, of equal size, and uncorrelated among each other, so that e_{ia} is modeled as

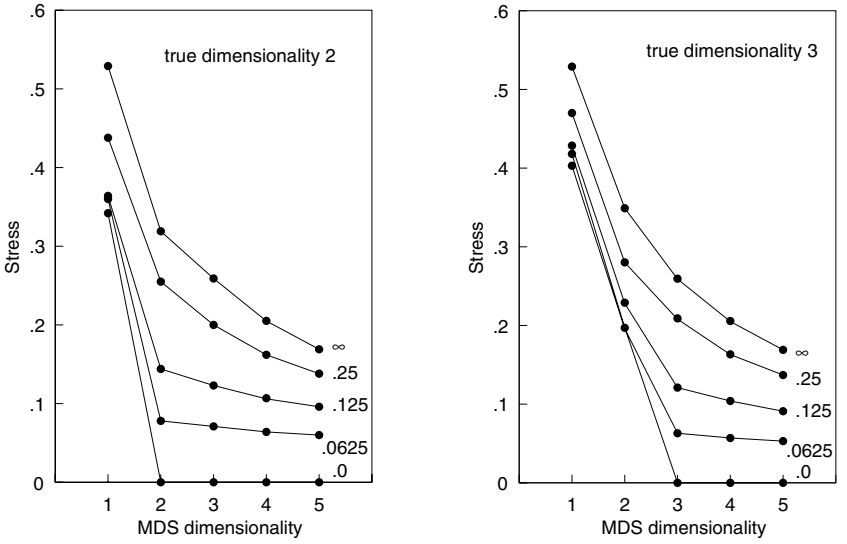


FIGURE 3.8. Expected Stress for distances in evenly scattered 2D (left panel) and 3D (right panel) configurations of 36 points with random error ranging from .0 to ∞, when represented in 1D through 5D (Spence & Graef, 1974).

a value sampled randomly from $N(0, \sigma^2)$. These definitions constitute the Hefner (1958) generalization of the Thurstone (1927) Case-V model.

Wagenaar and Padmos (1971) and Spence and Graef (1974) report simulation studies based on the Hefner model. They randomly pick n points from within a unit (hyper-)square or (hyper-)disk, and add error components sampled from $N(0, \sigma^2)$ to each of its coordinates. This leads to error-affected distances that are subsequently taken as proximities.

In contrast to the study by Spence and Ogilvie (1973) described above, this simulation allows one to specify the true (underlying) MDS space as the point configuration used in computing the proximities. Wagenaar and Padmos (1971) simulate this case for $n = 12, 18, 26, 36$; in $m = 1, \ldots, 4$ dimensions; and with error variances of $\sigma = 0.0, 0.0625, 0.125, 0.25$, and ∞ (i.e., pure random data).

Figure 3.8 shows the Stress curves obtained for proximities computed from 36 points in 2D and 3D MDS spaces, respectively, and represented in MDS spaces of one to five dimensions. One notes that all Stress curves are convex downwards. The upper curves in both diagrams almost have the same shape: they result from the condition of pure error. For the other conditions, we note elbows in the Stress curves for MDS dimensionalities of 2 and 3, respectively, that is, for the true dimensionalities of the underlying MDS spaces. These elbows are most pronounced in the error-free case, but are washed out with more and more error.

How large is the error component in these studies? One can check, by computer simulation, that the absolute difference of an error-affected distance (computed in an evenly scattered configuration of points within a unit disk) and the corresponding true distance, $|d_{ij}^{(e)} - d_{ij}|$, can be expected to be somewhat larger than the σs utilized by Spence and Graef (1974). That is, for example, for $m = 2$ and $\sigma = 0.25$, one finds that the expected absolute difference is 0.27, whereas for $\sigma = 0.0625$ and $m = 3$ it is 0.07. An error of judgment of about 25% does not seem excessive for many data in the social sciences. This may explain why elbows in scree plots are virtually never observed in practice, because, for $\sigma = 0.25$ or smaller σs, they are not obvious in Figure 3.8 either.

For real data, Spence and Graef (1974) propose comparing the Stress values for MDS solutions in different dimensionalities with their simulation curves in order to determine both the portion of error as well as the true dimensionality of the observations.

If one has an independent estimate of the error component in the data, the true dimensionality may be found by identifying that simulation curve among all those for the given error level that most closely matches the Stress curve for the given data. If the true dimensionality is known, one can proceed analogously for the error level. The conclusion depends, however, on the validity of the simulated error model.

Taking a closer look at the Hefner model, one notes that the normal error distribution is only a convenient approximation, because it puts no restrictions on the range of the point oscillations. Apart from that, however, the Hefner model has some interesting properties. It implies that error-affected distances tend to over-estimate true distances, because, by expanding the definition of $d_{ij}^{(e)}$, $E[(d_{ij}^{(e)})^2] = d_{ij}^2 + 2m\sigma^2$. Indeed, the error-affected distances are distributed as the noncentral χ^2 distribution (Suppes & Zinnes, 1963; Ramsay, 1969). Thus, a true distance of zero will only be over-estimated; small true distances can be expected to be more often over- than under-estimated; and the larger the true distance, the more balanced over- and under-estimation. This is a plausible model that prevents distance estimates from becoming negative.

Empirically, however, one often finds that dissimilarity judgments for very similar objects are more reliable than those for very dissimilar objects. Ramsay (1977), therefore, suggests making the error on the distances proportional to their size. In one particular model, the true distances are multiplied by a random factor whose logarithm has a normal distribution with mean 0 and standard deviation σ. This leads to a log-normal distribution for the error-affected distances where: (a) $d_{ij}^{(e)} \geq 0$; (b) the larger the true distance, the larger the noise; and (c) error-affected distances are

more likely to be over-estimated than under-estimated.[4] These properties seem to hold for many empirical contexts. However, what remains less clear is how this error model could be conceived in terms of what is going on in the psychological space.

Recovering Distances Under Noise

Simulation studies with error-perturbed distances do not assess how precisely the true distances are recovered by ordinal MDS. This question is investigated in the early days of MDS by Young (1970), Sherman (1972), Isaac and Poor (1974), and Cohen and Jones (1973), among others. Young (1970) proceeds as follows. (a) A true configuration with dimensionality t is defined by randomly sampling point coordinates. This yields true distances and, after adding error to the point coordinates, error-perturbed distances, as above. (b) The error-perturbed distances are monotonically transformed. (c) The resulting values are taken as data for an ordinal MDS procedure. (d) An MDS representation is then assessed with respect to the degree to which it recovers the true distances.

Young's simulations for different numbers of points, error levels, and monotone transformations—always setting $m = t$, so that the MDS analysis is in the true dimensionality—show that the precision of recovered distances grows with the number of points, and decreases with a higher error level in the data and with larger dimensionality of the solution space. This is intuitively plausible, because the isotonic regions in ordinal MDS shrink dramatically as a function of the number of points. Indeed, in the distances-among-cities example of Chapter 2, we found that the distances of the original map were almost perfectly reconstructed in a metric sense by the ordinal 2D MDS solution.

The effect of error on recovery precision is also easy to understand. More error simply reduces the correspondence of true distances and proximities. However, the harmful effect of error on recovery decreases with more points, because, with many points, the error-affected distances randomly over-estimate and under-estimate the true distances in so many ways that the effect of error on the configuration is balanced out and the solution essentially reconstructs the true distances. Stress, on the other hand, *in-*

[4]This error model, and related ones, is incorporated into the program MULTISCALE (see Appendix A). MULTISCALE does not minimize a loss function such as Stress. Rather, it tries to find that configuration $\mathbf{X}$ which, given a particular error model, maximizes the likelihood to yield $d_{ij}^{(e)}$s that correspond to the observed dissimilarities (maximum likelihood estimation). Given that the assumed error model holds, this allows one to determine confidence regions for the points and to make a number of inferential decisions, such as one on the proper dimensionality. Maximum likelihood MDS methods also exist for the Hefner error model (Zinnes & MacKay, 1983) and for ordinal MDS (Takane & Carroll, 1981).

creases when the number of points goes up, other conditions being equal! Cox and Cox (1990) even showed, by simulation, that Stress is an almost perfectly linear function of noise, given some special circumstances such as $m = t = 2$, but independently of the spatial pattern of points (ranging from extremely regular patterns through complete spatial randomness to cluster-like aggregations of points) and also independently of n. Cox and Cox (1992) report similar results for $m > 2$, but without such a strong linear relation between Stress and noise.

These findings have important practical implications. Global fit indices such as Stress are closely related to the proportion of error in the data. They are largely useless as measures of how well an MDS solution represents the "true" structure of the data. Therefore, it is quite possible that one obtains an MDS representation that has high Stress but that, nevertheless, is highly reliable over replications of the data. This means that a given Stress value should always be evaluated against some rough estimate of how much error is contained in the data.

An interesting further investigation on recovering true MDS spaces by means of ordinal MDS is presented by Sherman (1972), who studied, in particular, the effects of *over-* and *under-compression*. These notions refer to the question of whether the MDS dimensionality (m) is smaller or greater than the dimensionality of the space from which the proximities were derived (t). Sherman finds that picking the wrong dimensionality ($m \neq t$) has a pronounced effect: although Stress goes down monotonically when m goes up, the metric determinacy is best when $m = t$ and decreases with the extent of both over- and under-compression. There are slight differences though: under-compression, especially when there are many points in a relatively low-dimensional space, is somewhat less serious. This again shows that lower Stress (as a consequence of higher dimensionality) does not imply better metric recovery.

Summary on Stress

Stress is a badness-of-fit measure that depends, as we saw, on many factors. Some of them are:

- n, the number of points: the higher n, the higher Stress in general;
- m, the dimensionality of the MDS space: the higher m, the lower Stress;
- the error in the data: more error means higher Stress;
- the number of ties in the data (for ordinal MDS with weak monotonicity): more ties allow for lower Stress in general;
- the number of missing data: more missing data lead to lower Stress, in general;

TABLE 3.2. Average recovery coefficients, r^2s, for proximities related to true distances by $p_{ij} = d_{ij}^k$, under choice of different MDS models (Green, 1974).

Power k	Ratio MDS	Interval MDS
1.2	.99	.99
2.2	.94	.99
3.2	.85	.97
4.2	.78	.96
5.2	.72	.94

- the MDS model: interval MDS generally leads to higher Stress than ordinal MDS, particularly if f is markedly nonlinear and/or has major steps.

All of these criteria are mechanical ones. They are not sufficient for evaluating Stress, nor are they always important. Kruskal (1964a) writes: "A second criterion lies in the interpretability of the coordinates. If the m-dimensional solution provides a satisfying interpretation, but the $(m+1)$-dimensional solution *reveals no further structure* [our emphasis], it may be well to use only the m-dimensional solution" (p. 16). It is in this sense that Guttman (personal communication) called Stress a mere "technical measure." A measure of scientific significance, in contrast, would take into account the degree to which an MDS solution can be brought into a meaningful and replicable correspondence with prior knowledge or with theory about the scaled objects.

3.6 Recovering True Distances by Metric MDS

So far, we have investigated the performance of ordinal MDS only. In metric MDS, many of the above questions can be answered rather directly. For example, for interval MDS and error-free proximities, increasing the number of points has no effect on the goodness of recovery. If we scale under $t = m$, we can expect that Stress is zero for any n. Moreover, the correlation of the true and the recovered distances should be one. In ordinal MDS, in contrast, we cannot easily infer from the obtained Stress value how high the metric recovery is. This depends, among other things, on n, because the number of points is related to the size of the isotonic regions. If the data are not error-free, then interval MDS succeeds in representing somewhat more error variance in general when n is small, so that the metric recovery is likely to be less than perfect. If n grows, then both Stress and metric recovery go up, just as in ordinal MDS. Thus, it can be seen that the behavior of metric MDS is quite predictable without simulation studies.

The situation is not as easily diagnosed if we ask how well interval MDS does if the true relation between proximities and distances is not linear.

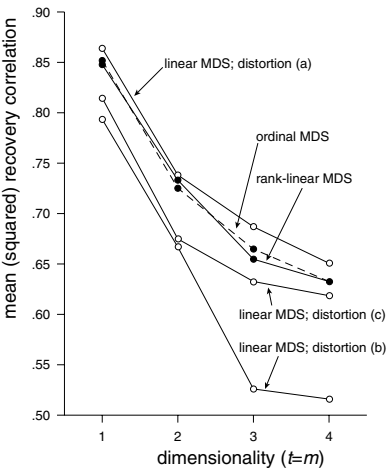


FIGURE 3.9. Recovery performance of MDS under choice of different models, number of dimensions, and distortions on proximities (after Weeks & Bentler, 1979).

Some answers are given by Green (1974). He selects $n = 10, 20$, and 30 points in $t = 2$ and $t = 3$ dimensions. Distances were computed and transformed into proximities by the function $p_{ij} = d_{ij}^k$, with $k = 1.2$, 2.2, 3.2, 4.2, and 5.2. Interval and ratio MDS were used to recover the underlying configurations from these proximities. The recovery coefficients in Table 3.2 show that ratio MDS is quite robust against such monotonic distortions of the function relating distances and proximities, as long as they are not extremely nonlinear. Interval MDS is almost unaffected by these (appreciable) nonlinear transformations.

Green (1974) demonstrates further that if we first substitute ranking numbers for the p_{ij} values, and then use ratio or interval MDS on these numbers, recovery is even better. The idea of *rank-interval MDS* was studied in more detail by Weeks and Bentler (1979). They used the following parameters for their simulations: $n = 10$, 20, 30; $t = 1$, 2, 3, 4; and $e = 0.25$, 0.75, 2.0, defined as the proportion of the error variance to the variance of the true distances. The proximities were derived from the error-perturbed distances by (a) $p_{ij} = d_{ij}^{(e)}$, (b) $p_{ij} = [d_{ij}^{(e)}]^4$, (c) $p_{ij} = [d_{ij}^{(e)}]^{1/4}$, or (d) $p_{ij} = \text{rank}[d_{ij}^{(e)}]$. Condition (d) is Green's ranking number substitution, and condition (a) simply means that the error-perturbed distances were taken directly as data, without any further distortions. These data were represented by both ordinal and interval MDS. The dimensionality of the solution space, m, varied from 1 to 6.

Figure 3.9 shows the main result of the study. The various curves are defined by the average values of the (squared) metric determinacy coefficient under the different conditions. As expected, all curves drop as $t = m$

goes up, because the higher the dimensionality, the more error variance can be represented by MDS, which negatively affects the metric determinacy of the solution. Ordinal MDS leads to the same recovery curve under all conditions. For (a), interval MDS does slightly better than ordinal MDS, but its recovery performance is definitely worse than that of ordinal MDS under the nonlinear distortions (b) and (c), as expected. However, with the ranking number substitutions, interval MDS leads to virtually the same recovery curve as ordinal MDS, as one can see from comparing the two lines with the solid black points in Figure 3.9. (Note that ranking number substitutions make all of the data sets used by Weeks and Bentler (1979) equivalent.) This replicates the finding of Green (1974). The "linearizing" effect of ranking number substitutions was also known to Lingoes (1965), who used this method in constructing initial configurations for ordinal MDS procedures.

Two conclusions can be derived from these results. (1) If proximities and distances are related in a linear way, then the metric information contained in the data is only marginally more powerful than the ordinal information contained in the data for recovering the true distances. (2) If proximities and data are related in a monotonic way, then ordinal and rank-interval MDS can be expected to lead to essentially the same solutions. This is important insofar as metric MDS methods are more robust in a numerical sense; that is, they generally are more likely to yield globally optimal solutions and are less likely to produce degenerate solutions (see Chapter 13).

3.7 Further Variants of MDS Models

The generic model relation (3.4) leaves room for many variants of MDS models not discussed so far. The most obvious way to generate such models is to specify the representation function f in different ways. There are many possibilities, and some of them are considered in Chapter 9. Further possibilities arise out of considering particular patterns of missing data. A whole model class, called unfolding, is discussed at length in Chapters 14 to 16. Then, one could partition the proximities into subsets, and specify independent fs or even different fs for each such subset rather than just one single f for all proximities as in (3.4).

At this point, we need not go into such models. We introduce, however, one generalization of (3.4) that allows us to introduce some notions useful for further classifying MDS models. Assume that we have more than one proximity for each pair (i, j). Such a case can arise, for example, if the data collection is replicated K times or if there are K persons, each giving rise to one set of proximities. In such a case, the proximities can be given three indices, p_{ijk} $(i, j = 1, \ldots, n; k = 1, \ldots, K)$. This means that they can be

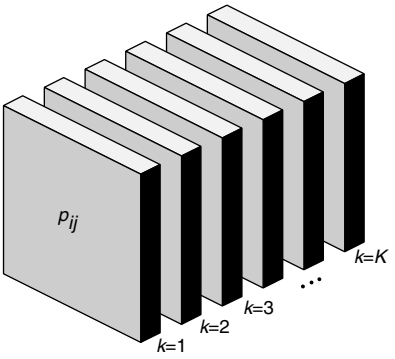


FIGURE 3.10. Symbolic representation of a three-way proximity array; $r = 1$ indicates replication 1.

collected in a *three-way* data array, as illustrated in Figure 3.10. This array can be conceived as a "deck" of K proximity matrices, where each "card" comes from one replication or person.

One could analyze such data by scaling each replication separately and then comparing or aggregating the different MDS solutions (see Chapter 20), or by first averaging the proximities over the replications and then scaling the aggregated scores. Another possibility is the following model,

$$f : p_{ijk} \rightarrow d_{ij}(\mathbf{X}), \tag{3.13}$$

for all pairs (i, j) and all ks, given that p_{ijk} is nonmissing. Note that this model relation differs from (3.4) only with respect to the proximities: it maps K proximities p_{ijk} (rather than just a single p_{ij}) into just one distance d_{ij}.

The three-way proximity block in Figure 3.10 suggests further possibilities for MDS models. Carroll and Arabie (1980) developed a taxonomy for MDS models according to which the three-way data in Figure 3.10 would also be characterized as *two-mode* data: "A mode is defined as a particular class of entities. ... Entities could be, for example, subjects, stimuli, test items, occasions, experimental conditions, geographical areas, or components of a 'multiattribute stimulus'. ... A K-way array is defined as the Cartesian product of a number of modes, some of which may be repeated. For example, an array associated with three-way multidimensional scaling might be of the form $A \times B \times B$, where A denotes subjects, and B stimuli" (p. 610). Hence, the "ways" of a proximity array refer, in a sense, to the number of subscripts of its proximities, whereas the "modes" distinguish whether these ways are qualitatively different ones.

There exist particular MDS models for three-way two-mode proximities, especially those where the "third" way denotes different individuals (see Chapters 21 and 22). There are also special models for two-way two-mode proximities, where one mode represents individuals and the other denotes

choice objects (see Chapter 17). Typical MDS data are, however, two-way one-mode proximities such as item intercorrelations or direct similarity ratings.

3.8 Exercises

Exercise 3.1 Consider the configuration in Figure 3.1. Compute the Euclidean distances among its $n = 6$ points.

(a) From these distances, generate dissimilarities by adding random error to each value. That is, $\delta_{ij} = d_{ij} + e_{ij}$, where e_{ij} is a value taken from a normal distribution $N(0, \sigma)$. (Alternatively, add random error to the point coordinates and then compute the distances. This may be easier to do within your statistics package.) Use different σs to simulate data with small, medium, and large error components. Run ordinal MDS with these dissimilarities. Compare the MDS solutions to Figure 3.1 and check the ability of ordinal MDS to recover the d_{ij}s from the δ_{ij}s.

(b) Repeat (a) using interval MDS.

(c) Repeat with $n = 20$ and $n = 40$ points that you choose at random in the plane shown in Figure 3.1, that is, with points (x, y), where $x, y \in [-4, +4]$.

Exercise 3.2 Suppose that the solution of Exercise 2.4 is given by the coordinates

	Dim 1	Dim 2
Red	0	3
Orange	0	0
Green	4	0
Blue	6	6

(a) Make a scatter plot of these points. Compute the distances between the points.

(b) Summarize the results in a table that has as its rows the six pairs of colors. Then, add a column that contains the proximity data for these pairs (see Exercise 2.4). Add a second column with the corresponding distances, computed from the table above. Finally, order the rows so that the row with the smallest proximity value is on top, and the row with the largest proximity at the bottom. Does the rank-order of the proximities match the rank-order of the distances? What do you conclude about the quality of the MDS solution?

Exercise 3.3 Consider data from 13 stock market indices of 784 daily measures from January 1, 1995, to December 31, 1997 (Groenen & Franses, 2000). From these data, the so-called return values are derived by taking the difference of the log of two subsequent index values. A correlation matrix of these stock market indices is given below.

Stock market		1	2	3	4	5	6	7	8	9	10	11	12	13
1	Brus	1.00												
2	CBS	.62	1.00											
3	DAX	.64	.69	1.00										
4	DJ	.29	.36	.21	1.00									
5	FTSE	.52	.69	.54	.38	1.00								
6	HS	.43	.40	.50	.11	.35	1.00							
7	Madrid	.51	.61	.57	.31	.59	.33	1.00						
8	Milan	.49	.50	.60	.15	.41	.37	.47	1.00					
9	Nikkei	.25	.28	.29	.04	.24	.33	.24	.23	1.00				
10	Sing	.34	.26	.36	.05	.25	.67	.26	.29	.29	1.00			
11	SP	.28	.35	.20	.96	.37	.09	.29	.14	.05	.04	1.00		
12	Taiwan	.04	.05	.07	-.03	.03	.15	.05	.07	.10	.19	-.03	1.00	
13	VEC	.52	.71	.62	.33	.63	.37	.61	.45	.25	.27	.32	.04	1.00

Now, the question is how different (or similar) the fluctuations are among the indices of the 13 stock markets.

(a) Use a computer program to do an interval MDS in 1 to 6 dimensions. Make a scree plot of the Stress values. Motivate your choice for the dimensionality of the solution.

(b) Can the Stress values be compared to the ones obtained for random data (see Figure 3.6) and the Hefner model? Explain why.

(c) Inspect Stress diagrams of your solution. What can you say about the fit? Do all points fit equally well?

(d) Interpret the solution. Can you distinguish groups of stock markets that have similar fluctuations?

(e) In what stock markets should you invest your money, if you want to spread the risks of your investment? Motivate.

(f) Redo the analysis with an ordinal transformation. Is the resulting configuration different? Compare the Shepard plots or the transformation plots. Is the difference in Stress small or large? Explain why this is so.

Exercise 3.4 Use the solution you like best from the previous exercise and compute the Stress per point and the fit per point. Produce a bubble plot that shows the fit per point either by hand or by a graphics program. Which are the worst fitting points? Which are the best fitting points? Interpret the solution again. Is it different from your first interpretation?

Exercise 3.5 Run an ordinal MDS analysis with your MDS program on the data from Table 2.3. The Stress of the resulting MDS solution is most likely not equal to zero even though we know that the distances were measured on a flat map.

(a) Explain why Stress is not zero.

(b) Try to get your MDS program to come up with a smaller Stress value.

(c) Compare the solution generated under the program's default settings with any one that has an even lower Stress. What do you conclude?

4
Three Applications of MDS

Three applications of MDS are discussed in some depth. Emphasis is given to the questions of how to choose a particular MDS solution and how to interpret it. First, data on the perceived similarity of colors are studied. The predicted MDS configuration is a color circle, which is indeed found to be the best representation for the data. Second, confusion data on Morse codes are investigated. The MDS space shows two regional patterns, which reflect two physical properties of the signals. Third, global similarity judgments on different facial expressions are studied. A dimensional system can be found that relates to three empirical scales for the faces.

4.1 The Circular Structure of Color Similarities

We now look at some applications of MDS in somewhat more depth and not just in an illustrative way. We start with a classic case where the MDS solution is particularly revealing.

Some Data on the Perceived Similarity of Colors

A person asked to somehow orderly arrange chips of different colors will almost certainly come up with an order from orange over yellow, green, blue, to blue-violet, corresponding to the order of the electromagnetic wavelengths of these colors. For the color red-violet, the respondent would probably not be sure whether it should lie on the red end or the violet end of

the scale (or on both). This problem is solved by arranging the colors in a horseshoe or circle. It may be supposed that most persons confronted with this ordering task would sooner or later arrive at such a solution.

Both perceptual and intellectual components seem to be involved in solving this task, but to what relative extent? It could be argued that the color circle is already implied by the way we perceive the similarity of colors. Let us look at some data by Ekman (1954). Ekman used 14 colors differing only in their wavelengths, but not in their brightness or saturation. Each of all possible 91 pairs of different colors was projected onto a screen, and 31 subjects were asked to rate the "qualitative similarity" of each such pair on a scale from 0 (no similarity) to 4 (identical). The ratings for each pair were averaged over all subjects. Finally, the resulting scores were divided by 4, that is, scaled down to the interval from 0 to 1. This led to the similarity matrix in Table 4.1 (lower half). Note that only one-half of the matrix was collected empirically, and so it suffices to show this half: the complete matrix, if needed, can be constructed by setting $p_{ii} = 1.00$ and $p_{ij} = p_{ji}$, for all i, j. (Most MDS programs need only a half-matrix as input.)

The proximities in Table 4.1 could be interpreted as correlations, so that a principal component analysis (PCA; see also Chapter 24) is possible. A PCA yields five different factors. These factors correspond to five different groups of points on the electromagnetic spectrum. The factors comprise the colors 434–445, 465–490, 504–555, 584–600, and 610–674, which roughly correspond to the subjective color qualities blueish-purple, blue, green, yellow, and red, respectively. Chopping up the colors into qualitative categories, however, does not throw much light on the question we are asking.

An inspection of the coefficients in Table 4.1 shows that the data do not support the notion of discrete color categories. Rather, they possess a simple pattern of interrelatedness, a peculiar gradient of similarities, with larger coefficients towards the main diagonal and the lower left-hand corner, respectively. So, using MDS, which establishes a direct relationship between dissimilarity measures and geometric distance (unlike PCA), we would possibly get a simple geometric expression for this data gradient.

MDS Representations of the Color Similarities

For the MDS analysis, we use ordinal MDS, the usual choice for a first approximation. Thus, a configuration of 14 points is sought such that the rank-order of the distances between these points corresponds (inversely) to the rank-order of the data.

Any MDS program requires the user to specify the dimensionality (m) of the desired representation. What value m should be chosen in the given case? Surely, setting $m \geq 13$ would be an uninteresting choice, because dissimilarities among n objects can always be perfectly represented in a space with dimensionality $m \geq n - 1$. For example, in a plane with points

TABLE 4.1. Similarities of colors with wavelengths from 434 to 674 nm (lower half) of Ekman (1954); residuals of 1D MDS representation (upper half).

nm	434	445	465	472	490	504	537	555	584	600	610	628	651	674
434	–	.14	.17	.38	.22	-.73	-1.07	-1.21	-.62	-.06	.42	.38	.28	.26
445	.86	–	.25	.11	-.05	-.75	-1.09	-.68	-.35	-.04	.44	.65	.55	.53
465	.42	.50	–	.08	-.32	-.57	-.47	-.06	.00	-.32	.17	.12	.91	.82
472	.42	.44	.81	–	.12	-.36	-.26	.15	.00	-.11	.00	.33	.23	1.03
490	.18	.22	.47	.54	–	-.07	.08	.48	.40	.00	.22	.17	.07	.00
504	.06	.09	.17	.25	.61	–	.31	.28	.45	.68	.01	.00	.00	-.15
537	.07	.07	.10	.10	.31	.62	–	.13	.35	.09	.31	.00	.00	-.75
555	.04	.07	.08	.09	.26	.45	.73	–	-.05	.17	-.09	-.22	-.32	-.34
584	.02	.02	.02	.02	.07	.14	.22	.33	–	-.05	-.01	-.06	-.16	-.18
600	.07	.04	.01	.01	.02	.08	.14	.19	.58	–	.21	.07	-.39	-.40
610	.09	.07	.02	.00	.02	.02	.05	.04	.37	.74	–	-.08	-.13	-.11
628	.12	.11	.01	.01	.01	.02	.02	.03	.27	.50	.76	–	-.03	-.16
651	.13	.13	.05	.02	.02	.02	.02	.02	.20	.41	.62	.85	–	-.11
674	.16	.14	.03	.04	.00	.01	.00	.02	.23	.28	.55	.68	.76	–

A, B, and C, it is possible, by moving the points around, to eventually arrive at a configuration whose distances perfectly represent any given proximities $p(A, B)$, $p(A, C)$, $p(B, C)$, no matter what values they have. Analogously, for four points, a perfect representation always exists in three dimensions, and so on.

The minimal dimensionality for a perfect MDS representation is only of formal interest. In practice, we always try to represent the data in an MDS space of considerably lower dimensionality. The rationale for choosing a low dimensionality is the expectation that this will cancel out over- and underestimation errors in the proximities, thus smoothing the representation (see Chapter 3). Moreover, a low-dimensional and preferably two-dimensional solution is often precise enough for a first interpretation. For the data in Table 4.1, we first try solutions in 1D, 2D, and 3D space (using the MDS module of SYSTAT 5.0).

With three solutions, we have to decide which one we should consider most appropriate. We first look at the 2D solution in Figure 4.1. It shows a circular arrangement of the points representing the colors. Moreover, the points are perfectly ordered along the drawn-in line in terms of their wavelengths. This circular structure corresponds to the color circle.

How well does this configuration represent the data? The best answer to this question is provided by looking at the Shepard diagram of the 2D MDS solution (Figure 4.2). The plot shows a tight correspondence of proximities and distances. The points lie very close to the monotone regression line. The regression line is almost straight, and so the dissimilarities of Table 4.1 are almost linearly and almost perfectly related to the distances in Figure 4.1. In contrast, in the Shepard diagram for the 1D solution (Figure

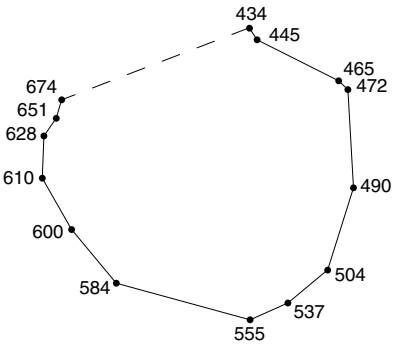


FIGURE 4.1. Ordinal MDS representation for color proximities in Table 4.1.

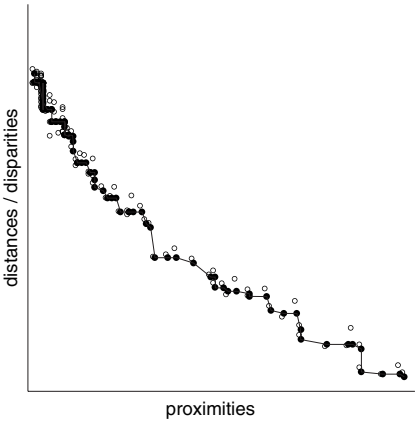


FIGURE 4.2. Shepard diagram for Fig. 4.1.

4.4), the deviations of the points from the shown best-possible monotonic decreasing line are excessive.

Measured in terms of Stress, the badness-of-fit of the 1D, 2D, and 3D solutions is 0.272, 0.023, and 0.018, respectively. These values are a rare example for a definite elbow in the scree test. The 1D solution has high Stress, and adding one additional dimension leads to a major Stress reduction. Adding yet another dimension has very little further effect and, indeed, cannot have much of an effect because the 0.023 for the 2D solution is so close to zero already.

Thus, the 2D solution appears to be a reasonably precise representation of the data. Adding a third dimension is not sensible, because of several reasons: (a) the point configuration in the X–Y-plane of the 3D solution (Figure 4.3) corresponds closely to the 2D configuration (Figure 4.1); (b) the decrement in Stress by allowing for a third dimension is negligible, satisfying the elbow criterion; (c) the scattering of the points in 3D space along the third dimension appears to be uninterpretable in substantive terms; and (d) no a priori theory exists for a 3D solution. Analogous arguments hold for comparing the 1D and 2D solutions. Hence, we have formal and substantive reasons to consider the 2D representation in Figure 4.1 as the best MDS representation of the given data.

A Closer Look at Model Fit

The Shepard diagram in Figure 4.4 shows that the 1D solution is a relatively poor representation of the data. Why there cannot exist a really good 1D solution can be seen from Figure 4.1. If we had to locate a straight line in this plane so that the distances between the projections of the points onto this line mirror most closely the order of the data, then this line would be

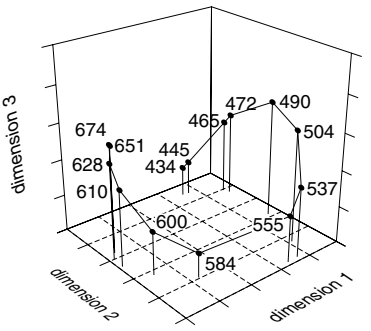


FIGURE 4.3. 3D MDS space of color data.

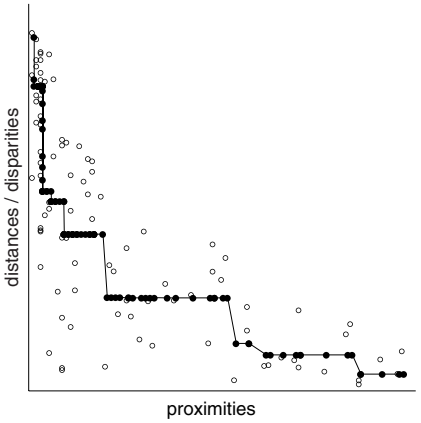


FIGURE 4.4. Shepard diagram for 1D MDS representation of color data.

oriented roughly horizontally. Such a line is the best 1D approximation to the given 2D distance structure, because most of the straight lines connecting any two points in Figure 4.1 run more or less in this direction. For point 610, for example, we see in Figure 4.1 that the projections of the rays from this point to all other points onto a horizontal line are ordered (in length) almost as the rays themselves. However, there would also be misrepresentations on this line. For example, the points 434 and 555, if projected onto a horizontal line, would be very close to each other, whereas the similarity between 434 and 555 is among the lowest ones observed. Hence, this datum is not represented well by the points' distance on this 1D subspace.

We should expect, then, that a 1D MDS solution for the color data represents the proximities of such colors as 610 and 472 with respect to all other colors quite well, but that it runs into problems with pairs such as 434 and 555. One could assess such effects quantitatively by computing, for each color C in turn, the correlation between the similarities of C to all other colors and the distances of point C to all other points. To be consistent with the ordinal approach, an ordinal correlation (e.g., Spearman's ρ) would be appropriate. Each such coefficient is a *conditional* fit measure, because it hinges on one fixed point or variable (C, here).

Using Spearman correlations, one finds that they are, for each point C, close to -1.00 for the 2D and 3D solutions. For the 1D case, in contrast, there is much more variance. The coefficients are particularly low for points 434 ($r = -.075$) and 445 ($r = -.360$) at the upper end of the horseshoe in Figure 4.1. Low conditional fit measures imply that the overall precision of the 1D MDS representation (as measured by Stress, e.g.) cannot be very good, because conditional agreements between distances and data are a necessary condition for a globally good solution.

Spearman's correlation is, however, not very robust, because it is based on the ranks of the data, and major changes of the rank-order sometimes result from minor changes of the data. A correlation coefficient that assesses the degree of monotone correspondence directly on the data is μ_2 (Guttman, 1968; see also Chapter 14). For the given data, however, one arrives at the same conclusion using μ_2: points 434 and 445 are the major sources of Stress.

An even more fine-grained analysis of the sources of Stress is possible by studying the residuals, e_{ij}, for all i, j. Table 4.1 (upper half) shows these residuals for the 1D MDS representation of the color data. One notes, for example, that the similarity measures for the pairs $(434, 555)$ and $(434, 537)$ are relatively poorly represented by their corresponding distances, as expected. For the pair $(610, 472)$, in contrast, the residual is zero.

Most MDS computer programs provide these residuals upon request. Some also compute some kind of average residual value — such as the root mean squared residual — for each point in turn. Such coefficients are conditional fit measures closely related to the Stress formula (Borg, 1978b).

4.2 The Regionality of Morse Codes Confusions

The next example we consider is also from perception on stimuli that are physically well structured. This most complex data matrix requires special considerations and some simplifications. The MDS configuration, then, clearly reflects the structural properties of the stimuli.

Morse Code Confusions and Their Representability by Distances

Consider now the data matrix in Table 4.2 (Rothkopf, 1957). The scores are confusion rates on 36 Morse code signals (26 for the alphabet; 10 for the numbers $0, \ldots, 9$). Each Morse code signal is a sequence of up to five "beeps." The beeps can be short (0.05 sec) or long (0.15 sec), and, when there are two or more beeps in a signal, they are separated by periods of silence (0.05 sec). For example, the signal for A is "short-silence-long," with a total temporal length of 0.25 seconds. We code such a signal as 12 (1 = short and 2 = long, or "di-da").

Rothkopf (1957) asked 598 subjects to judge whether two signals, presented acoustically one after another, were the same. The values given in Table 4.2 are the percentages with which the answer "Same!" was given in each combination of row stimulus i and column stimulus j, where i was the first and j the second signal presented. Each stimulus pair was presented in two orders, for example, B following A (confusion rate is 4%) and also A following B (5%). Moreover, the rate of confusion of each signal with itself

was assessed. For example, the relative frequency of confusing A with itself is 92%, and for B, 84%.

If we attempt an MDS representation, we notice several problems. First, we observe that the nonnegativity axiom does not hold, because the values in the main diagonal of the data matrix are not all the same. Because the distance from any point to itself is always 0, we will therefore necessarily incur a misrepresentation of the empirical data in the MDS space. On the other hand, the second part of the nonnegativity axiom poses no problem, because all data values in the main diagonal are greater than any off-diagonal value, and this can be properly expressed by distances in an MDS space.

Then, we see that the symmetry condition [axiom (2.2)] also does not hold for the data. For example, the signal I is more frequently confused with a subsequent A (64%) than A is with a subsequent I (46%). But if we represent I and A by one point each, then we will necessarily have the relation $d_{IA} = d_{AI}$, so that the asymmetry of the observed relation is lost, that is, not represented.

Finally, the triangle inequality can be checked only if the data are on a ratio scale. For all weaker MDS models, it is always possible to find a constant k so that every $p_{ij}+k$ satisfies the triangle inequality. The minimal constant k is found by first identifying the triangle inequality violated most and then computing the value that, when added to each proximity in this inequality, turns the inequality into an equality. Thus, unless we consider the Rothkopf data as ratio-scaled distances (apart from error), axiom (2.3) is immaterial.

Distance axioms (2.1) and (2.2), on the other hand, remain violated even if one allows for ordinal transformations of the data. Yet, we should take into account that none of Rothkopf's subjects knew Morse codes. It is a very demanding task for an untrained subject to distinguish consistently between different signals, and we might, therefore, argue that the violations of these axioms are unsystematic and due to error. (We test this in Chapter 24.) Under this assumption, we can think of the data for each (i, j) and (j, i) pair as replicated observations of a basically symmetric relation, and then obtain a better estimate of the true relation by averaging the two observed values. In other words, from Table 4.2 we form a new proximity matrix, where, say, $p_{AB} = p_{BA} = (.05 + .04)/2 = .045$. The main diagonal could then be filled with the value 1.00, say, although this is immaterial, because MDS programs ignore these values anyway.

TABLE 4.2. Confusion percentages between Morse code signals (Rothkopf, 1957).

Morse Code		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	1	2	3	4	5	6	7	8	9	0	
·—	A	92	4	6	13	3	14	10	13	46	5	22	3	25	34	6	6	9	35	23	6	37	13	17	12	7	3	2	7	5	5	8	6	5	6	2	3	A
—···	B	5	84	37	31	5	28	17	21	5	19	34	40	6	10	12	22	25	16	18	2	18	34	8	84	30	42	12	17	14	40	32	74	43	17	4	4	B
—·—·	C	4	38	87	17	4	29	13	7	11	19	24	35	14	3	9	51	34	24	14	6	6	11	14	32	82	38	13	15	31	14	10	30	28	24	18	12	C
—··	D	8	62	17	88	7	23	40	36	9	13	81	56	8	7	9	27	9	45	29	6	17	20	27	40	15	33	3	9	6	11	9	19	8	10	5	6	D
·	E	6	13	14	6	97	2	4	4	17	1	5	6	4	4	5	1	5	10	7	67	3	3	2	5	6	5	4	3	5	3	5	2	4	2	3	3	E
··—·	F	4	51	33	19	2	90	10	29	5	33	16	50	7	6	10	42	12	35	14	2	21	27	25	19	27	13	8	16	47	25	26	24	21	5	5	5	F
——·	G	9	18	27	38	1	14	90	6	5	22	33	16	14	13	62	52	23	21	5	3	15	14	32	21	23	39	15	14	5	10	4	10	17	23	20	11	G
····	H	3	45	23	25	9	32	8	87	10	10	9	29	5	8	8	14	8	17	37	4	36	59	9	33	14	11	3	9	15	43	70	35	17	4	3	3	H
··	I	64	7	7	13	10	8	6	12	93	3	5	16	13	30	7	3	5	19	35	16	10	5	8	2	5	7	2	5	8	9	6	8	5	2	4	5	I
·———	J	7	9	38	9	2	24	18	5	4	85	22	31	8	3	21	63	47	11	2	7	9	9	9	22	32	28	67	66	33	15	7	11	28	29	26	23	J
—·—	K	5	24	38	73	1	17	25	11	5	27	91	33	10	12	31	14	31	22	2	2	23	17	33	63	16	18	5	9	17	8	8	18	14	13	5	6	K
·—··	L	2	69	43	45	10	24	12	26	9	30	27	86	6	2	9	37	36	28	12	5	16	19	20	31	25	59	12	13	17	15	26	29	36	16	7	3	L
——	M	24	12	5	14	7	17	29	8	8	11	23	8	96	62	11	10	15	20	7	9	13	4	21	9	18	8	5	7	6	6	5	7	11	7	10	4	M
—·	N	31	4	13	30	8	12	10	16	13	3	16	8	59	93	5	9	5	28	12	10	16	4	12	4	16	11	5	2	3	4	4	6	2	2	10	2	N
———	O	7	7	20	6	5	9	76	7	2	39	26	10	4	8	86	37	35	10	3	4	11	14	25	35	27	27	19	17	7	7	6	18	14	11	20	12	O
—···	P	5	22	33	12	5	36	22	12	3	78	14	46	5	6	21	83	43	23	9	4	12	19	19	41	30	34	44	24	11	15	17	24	23	25	13	P
——·—	Q	8	20	38	11	4	15	10	5	2	27	23	26	7	6	22	51	91	11	2	3	6	14	12	37	50	63	34	32	17	12	9	27	40	58	37	24	Q
·—·	R	13	14	16	23	5	34	26	15	7	12	21	33	14	12	12	29	8	87	16	2	23	23	62	14	12	13	7	10	13	4	7	12	7	9	1	2	R
···	S	17	24	5	30	11	26	5	59	16	3	13	10	5	17	6	6	3	18	96	9	56	24	12	10	6	7	8	2	2	15	28	9	5	5	5	2	S
—	T	13	10	1	5	46	3	6	6	14	6	14	7	6	5	6	11	4	4	7	96	8	5	4	2	2	6	5	5	3	3	3	8	7	6	14	6	T
··—	U	14	29	12	32	4	32	11	34	21	7	44	32	11	13	6	20	12	40	51	6	93	57	34	17	9	11	6	6	16	34	10	9	9	7	4	3	U
···—	V	5	17	24	16	9	29	6	39	5	11	26	43	4	1	9	17	10	17	11	6	32	92	17	57	35	10	10	14	28	79	44	36	25	10	1	5	V
·——	W	9	21	30	22	9	36	25	15	4	25	29	18	15	6	26	20	25	61	12	4	19	20	86	22	25	22	10	22	19	16	5	9	11	6	3	7	W
—··—	X	7	64	45	19	3	28	11	6	1	35	50	42	10	8	24	32	61	10	12	3	12	17	21	91	48	26	12	20	24	27	16	57	29	16	17	6	X
—·——	Y	9	23	62	15	4	26	22	9	1	30	12	14	5	6	14	30	52	5	7	4	6	13	21	44	86	23	26	44	40	15	11	26	22	33	23	16	Y
——··	Z	3	46	45	18	2	22	17	10	7	23	21	51	11	2	15	59	72	14	4	3	9	11	12	36	42	87	16	21	27	9	10	25	66	47	15	15	Z
·————	1	2	5	10	3	3	5	13	4	2	29	5	14	9	7	14	30	28	9	4	2	3	12	14	17	19	22	84	63	13	8	10	8	19	32	57	55	1
··———	2	7	14	22	5	4	20	13	3	25	26	9	14	2	3	17	37	28	6	5	3	6	10	11	17	30	13	62	89	54	20	5	14	20	21	16	11	2
···——	3	3	8	21	5	4	32	6	12	2	23	6	13	5	2	5	37	19	9	7	6	4	16	6	22	25	12	18	64	86	31	23	41	16	17	8	10	3
····—	4	6	19	19	12	8	25	14	16	7	21	13	19	3	3	2	17	29	11	9	3	17	55	8	37	24	3	5	26	44	89	42	44	32	10	3	3	4
·····	5	8	45	15	14	2	45	4	67	7	14	4	41	2	0	4	13	7	9	27	2	14	45	7	45	10	10	14	10	30	69	90	42	24	10	6	5	5
—····	6	7	80	30	17	4	23	4	14	2	11	11	27	6	2	7	16	30	11	14	3	12	30	9	58	38	39	15	14	26	24	17	88	69	14	5	14	6
——···	7	6	33	22	14	5	25	6	4	6	24	13	32	7	6	7	36	39	12	6	2	3	13	9	30	30	50	22	29	18	15	12	61	85	70	20	13	7
———··	8	3	23	40	6	3	15	15	6	2	33	10	14	3	6	14	12	45	2	6	4	6	7	5	24	35	50	42	29	16	16	9	30	60	89	61	26	8
————·	9	3	14	23	3	1	6	14	5	2	30	6	7	16	11	10	31	32	5	6	7	6	3	8	11	21	24	57	39	9	12	4	11	42	56	91	78	9
—————	0	9	3	11	2	5	7	14	4	5	30	8	3	2	3	25	21	29	2	3	4	5	3	2	12	15	20	50	26	9	11	5	22	17	52	81	94	

An MDS of the Symmetrized Morse Code Data

Let us now study the symmetrized data by ordinal MDS.[1] In comparison with the color data examined above, we start here from a weaker position. Previously, we had a clear expectation about the MDS configuration and its dimensionality; here we make no attempt to predict anything. Hence, we must proceed in a purely descriptive way at the beginning.

Proceeding as Kruskal (1964a) did, we compute solutions in 1D through 5D, for which we obtain the Stress values shown graphically in Figure 4.5. The figure shows that the 1D solution has about .28 Stress. These values lie way below the expected Stress for random data reported in Figure 3.6, but that is always true for structured proximities. Adding one more dimension reduces Stress considerably to .18. By Kruskal's criteria (see Section 3.5), this would still be evaluated as a "poor" goodness-of-fit value. However, this simple norm does not take n, the number of points, into account, and what we have here is a relatively big data set compared to, say, the color data in Table 4.1. Fitting proximities for more objects to distances in an MDS space always requires a higher dimensionality if the data contain a certain amount of experimental error.

But how large is this error? We could take up the proposal of Spence and Graef (1974) and compare the observed Stress values to those obtained from simulating the Hefner model. This should allow us to determine both the true dimensionality and the error level. The observed Stress values are 0.35, 0.20, 0.14, 0.10, and 0.08 for $m = 1, \ldots, 5$, respectively. Their scree plot (Figure 4.5) shows no elbow. Turning to Figure 3.8, we note that the curves that most closely approximate the observed Stress values are the ones for an error level of 0.13. However, the Spence and Graef (1974) simulations do not clearly indicate what the true dimensionality of the MDS configuration is for these data.

Turning to interpretability, we first consider the 2D MDS configuration in Figure 4.6. Interpretation means to link some of the configuration's geometrical properties to known or assumed features of the represented objects. In the given case, we find that the points arrange themselves in a pattern that reflects the composition of the represented Morse signals, as shown in Figure 4.7. Following a suggestion by Wish (1967), we note that the 2D MDS space can be cut by the solid lines such that each region of the space contains signals of the same total duration. For example, this puts M (coded as 22), R (=121), D (=211), U (=112), and H (=1111) into the same equivalence class, because their signals all last 35/100 sec.

---

[1]The first MDS analysis of these data was done by Shepard (1963) and then by Kruskal (1964a) with the program M-D-SCAL (Kruskal & Carmone, 1969). M-D-SCAL has been replaced, in the meantime, by KYST (Kruskal, Young, & Seery, 1978). Most modern MDS programs (see Appendix A for an overview) usually lead to very similar solutions (Spence, 1972).

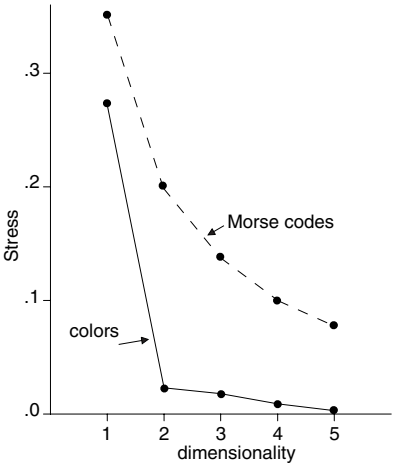


FIGURE 4.5. Scree plot (Stress vs. dimensionality) for MDS of color and Morse code data, respectively.

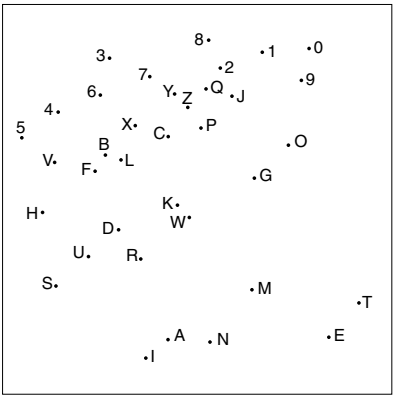


FIGURE 4.6. Ordinal MDS representation of Morse code data in Table 4.2.

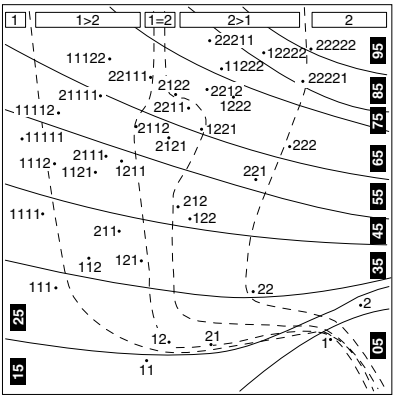


FIGURE 4.7. Morse code MDS configuration with two sets of partitioning lines: dashed lines split space into different signal types; solid lines differentiate signal lengths.

Technically, cutting a space into regions is called *partitioning* the space. Generally, partitioning a set means splitting it into subsets such that each element belongs to exactly one such subset. The resulting subsets are *exhaustive* and *disjoint*.

The configuration can also be partitioned in other ways by using other criteria. The dashed lines partition the space into regions that contain signals with only short (coded as 1) beeps, more short than long (coded as 2) beeps, a balanced number of short and long beeps, more long than short beeps, and long beeps only, respectively. The structure of this partitioning could be simplified—provided we are admitting some minor and one major misclassification of points—to a North–South slicing of the MDS plane into parallel stripes. The one major misclassification would result from point E. E, the Morse code that consists of one short beep only, seems to play a particular role. It is close to T, the other one-beep Morse code.

Without E, a typical *dimensional interpretation* of the MDS space would suggest itself: after a little rotation, the Y-axis could be interpreted as "duration", the X-axis as "kind of composition", ranging from signals consisting of short beeps only over signals with both short and long beeps to signals with long beeps only. Hence, at this stage, further research should first clarify the reliability of E's position. If E turns out to be reliable, we could possibly design a theory that explains the subjective similarity of Morse codes not by two independent dimensions but by two dimensions where the points' variance with respect to one dimension depends on the scale values on the other dimension, giving rise to a fan-like partitioning.

In any case, we see that the 2D MDS configuration can be interpreted in a simple but nontrivial way. Known properties of the signals, not just plausible posthoc insights, are used to explain the point scatter. The simplicity of the resulting geometric structure suggests, moreover, that we have found something real, not just an apparent structure in random data.

If we go on to higher-dimensional solutions, the points do not appear to reflect further systematic structure. Because no substantive hypothesis could be derived on the dimensionality of the MDS configuration, we may decide to give considerable weight to this simple interpretability of the solution over a formal precision-of-representation criterion such as Stress. This turns out to be a fruitful strategy in general. In any case, the data could be replicated, and then we would hope to find the same organizational patterns again. Without several such replications, we should be wary of making fine-grained interpretations.

4.3 Dimensions of Facial Expressions

There are many principles that can be used for interpreting an MDS configuration. What one always looks for is some way to organize the point

scatter, to account for it or to "explain" it by a parsimonious but substantively meaningful generating function. The typical, often almost mechanical approach to this question in the literature has been the interpretation by dimensions. Dimensional interpretations assign substantive meaning to coordinate axes. We now examine a relatively refined example where a dimensional theory is given a priori.

Rating Facial Expressions on Simple Scales

Some of the early research on the psychology of facial expressions was occupied with the question of whether subjects could correctly identify the intended emotional message from a person's facial expression. It was found that misinterpretations were not random; the perceived emotion usually seemed "psychologically similar" (Woodworth, 1938) to the one actually expressed by the sender. Schlosberg and others then attempted to develop a theory of the differentiability of facial expressions, concluding that three perceptual "dimensions" were needed for a meaningful classification of facial expressions: pleasant–unpleasant (PU); attention–rejection (AR); and tension–sleep (TS). In different studies, it could be shown that subjects were able to classify facial expressions on these dimensions.

Engen, Levy, and Schlosberg (1958) published scale values, empirically arrived at, for the 48 photographs of the Lightfoot Series. This series shows the face of a woman acting out a series of different situations. Some of the situations and their coordinate values are given in Table 4.3. If these values are taken as Cartesian coordinates, distances between the different expressions can be computed and used to predict confusion rates. However, "... the particular three dimensions used by Schlosberg are not necessarily the only dimensions or the best dimensions for explaining confusion data .... There is the possibility that one or more of Schlosberg's scales, while understandable when made explicit to judges, are unimportant in uninstructed perception of facial expression; or conversely, that one or more important scales have been omitted .... [The experimenter] imposes particular dimensions of his own choosing and is arbitrarily forced to give them equal weight" (Abelson & Sermat, 1962, p. 546).

MDS of Facial Expressions and Internal Scales

MDS offers another way of testing the theory of three dimensions. We can ask the subjects to globally judge, without external criteria provided by the experimenter, the overall similarities of different facial expressions. The proximities are then mapped into MDS distances. The resulting configuration should be three-dimensional, with dimensions that correspond to the Schlosberg scales.

Abelson and Sermat (1962) asked 30 students to rate each pair of the 13 pictures described in Table 4.3 on a 9-point scale with respect to overall

TABLE 4.3. Scale values on three scales for faces of a woman acting different scenes (Engen et al., 1958); values are medians on 9-point scales.

	Scene	PU	AR	TS
1	Grief at death of mother	3.8	4.2	4.1
2	Savoring a Coke	5.9	5.4	4.8
3	Very pleasant surprise	8.8	7.8	7.1
4	Maternal love–baby in arms	7.0	5.9	4.0
5	Physical exhaustion	3.3	2.5	3.1
6	Something wrong with plane	3.5	6.1	6.8
7	Anger at seeing dog beaten	2.1	8.0	8.2
8	Pulling hard on seat of chair	6.7	4.2	6.6
9	Unexpectedly meets old boyfriend	7.4	6.8	5.9
10	Revulsion	2.9	3.0	5.1
11	Extreme pain	2.2	2.2	6.4
12	Knows plane will crash	1.1	8.6	8.9
13	Light sleep	4.1	1.3	1.0

dissimilarity. Dissimilarity was defined as "a difference in emotional expression or content." For each subject, 78 proximities resulted, which were then rescaled over individuals by the method of successive intervals (Diederich, Messick, & Tucker, 1957). The means of these intervals were taken as the proximity data (Table 4.4).

We now analyze the data in Table 4.4 by ordinal MDS. The resulting Stress values for 1D up to 5D solutions are .24, .11, .06, .04, and .02, respectively. On purely formal grounds, we would probably decide that the 2D solution is reasonably accurate. However, because we are particularly interested in testing Schlosberg's theory of three dimensions, we should also consider the 3D solution. To make things simpler, we first start with the 2D solution.

The point coordinates of the 2D solution (Figure 4.9) are shown in Table 4.5. One can check that the values in each column add up to zero. Geometrically, this means that the MDS configuration is *centered*; that is, its center of gravity lies at the origin of the coordinate axes. The coordinate vectors are also uncorrelated. This is so because the MDS configuration has been rotated to its principal axes orientation or, expressed differently, because the dimensions X and Y are the principal axes (see also Section 7.10) of this plane. Principal axes (PAs) are always uncorrelated.[2] The PAs can be found by locating an axis so that it accounts for as much of the points' scattering as possible. That is, an axis is located such that it lies as close as possible to all points in the sense that the sum of squared distances of

[2]One can formulate the problem of finding PAs as finding that rotation of a given Cartesian dimension system that makes the point coordinates uncorrelated [see, for example, Strang (1976)].

TABLE 4.4. Proximities for faces from Table 4.3.

	1	2	3	4	5	6	7	8	9	10	11	12	13
1	–												
2	4.05	–											
3	8.25	2.54	–										
4	5.57	2.69	2.11	–									
5	1.15	2.67	8.98	3.78	–								
6	2.97	3.88	9.27	6.05	2.34	–							
7	4.34	8.53	11.87	9.78	7.12	1.36	–						
8	4.90	1.31	2.56	4.21	5.90	5.18	8.47	–					
9	6.25	1.88	0.74	0.45	4.77	5.45	10.20	2.63	–				
10	1.55	4.84	9.25	4.92	2.22	4.17	5.44	5.45	7.10	–			
11	1.68	5.81	7.92	5.42	4.34	4.72	4.31	3.79	6.58	1.98	–		
12	6.57	7.43	8.30	8.93	8.16	4.66	1.57	6.49	9.77	4.93	4.83	–	
13	3.93	4.51	8.47	3.48	1.60	4.89	9.18	6.05	6.55	4.12	3.51	12.65	–

TABLE 4.5. Coordinates for points in 2D MDS space.

Point/Picture	Dim 1 (X)	Dim 2 (Y)
1	−0.41	−0.46
2	0.54	0.14
3	1.22	0.75
4	0.97	−0.21
5	0.06	−0.72
6	−0.67	0.24
7	−1.34	0.45
8	0.48	0.62
9	1.05	0.27
10	−0.59	−0.69
11	−0.62	−0.31
12	−1.02	0.98
13	0.32	−1.04

the points from it is minimal. The second PA then is fixed automatically, because it must be perpendicular to the first axis.

Internal and External Scales

We now test whether the *external* scales of Table 4.3 account for the relative locations of the points. A crude first test is to correlate each of the columns of Table 4.5 (*internal scale*) with the columns in Table 4.3. Table 4.6, left panel, shows that there is a considerable correlation, $r = .94$, between the coordinates of the points on the X-axis and the values of the corresponding facial expressions on the PU scale. Similarly, the point coordinates on the Y-axis correlate highly with both the AR ($r = .86$) and the TS ($r = .87$) scales.

TABLE 4.6. Correlations between principal axes of 2D and 3D MDS solutions and Schlosberg scales in Table 4.3.

	2D MDS			3D MDS			
Scale	Dim 1	Dim 2	R^2	Dim 1	Dim 2	Dim 3	R^2
PU	.94	.21	.92	.93	.20	-.09	.91
AR	-.02	.86	.74	-.05	.83	-.34	.81
TS	-.38	.87	.90	-.37	.89	.06	.96

Yet, Schlosberg's theory does not claim that the principal axes should be of particular substantive importance. Maybe there are other dimensions that better satisfy the theory and, in particular, correlate higher with the scales of Table 4.3. This question can be answered as follows. Using multiple correlation, we can assess how well an optimal linear combination of the principal axes explains the scales. Because principal axes are uncorrelated, the squared multiple correlations are simply the sum of the squared bivariate correlations in Table 4.6. For example, for the PU scale on the one hand and the principal axes of the 2D solution, we find $R(PU.12) = .92^{1/2}$ from $R^2 = (0.94)^2 + (0.21)^2 = 0.92$. Thus, because the multiple correlation of the PU scale with the principal axes is higher than any bivariate correlation of PU with a given principal axis, there must exist an axis (i.e., another internal scale) in the MDS space that correlates even higher with PU than the X-axis. This is now investigated.

Optimally Fitting External Scales

In addition to correlating the points' coordinates on some internal scale with an external scale, we can also express their relationship geometrically. This is done by representing an external scale S by a directed line[3] Q located such that the point projections on it (Q-values or Q-coordinates) mirror as closely as possible the corresponding scale values of S. This can mean, for example, that the point projections on Q are spaced such that the ordinal Stress between the Q- and the S-values is minimal. Or, because we have treated the S scales above as interval scales, we could require that the intervals of the Q- and the S-values correspond most closely in their proportions. Thus, Q should be located such that, over all points i, $[s_i - (a + b \cdot q_i)]^2 = \min$, where q_i is the coordinate value of point i's projection on line Q. This looks like a linear regression problem, except that not only the weights a and b, but also the q_i values are unknowns. But any line Q is simply a linear combination of the coordinate vectors in

[3]A *directed line* is a line on which the direction from one end to the other has been indicated as positive, and the reverse direction as negative. The points on this line are ordered.

TABLE 4.7. Multiple regression problem to account for external PU scale by MDS coordinate vectors. Weights w_1, w_2 and additive constant a are to be chosen such that $\approx$ means "as nearly equal as possible." Optimal values are $g_1 = 2.679, g_2 = 0.816, a = 4.523$.

$$\begin{bmatrix} 3.8 \\ 5.9 \\ 8.9 \\ 7.0 \\ 3.3 \\ 3.5 \\ 2.1 \\ 6.7 \\ 7.4 \\ 2.9 \\ 2.2 \\ 1.1 \\ 4.1 \end{bmatrix} \approx g_1 \begin{bmatrix} -0.41 \\ 0.54 \\ 1.22 \\ 0.97 \\ 0.06 \\ -0.67 \\ -1.34 \\ 0.48 \\ 1.05 \\ -0.59 \\ -0.62 \\ -1.02 \\ 0.32 \end{bmatrix} + g_2 \begin{bmatrix} -0.46 \\ 0.14 \\ 0.75 \\ -0.21 \\ -0.72 \\ 0.24 \\ 0.45 \\ 0.62 \\ 0.27 \\ -0.69 \\ -0.31 \\ 0.98 \\ -1.04 \end{bmatrix} + a \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \\ q_5 \\ q_6 \\ q_7 \\ q_8 \\ q_9 \\ q_{10} \\ q_{11} \\ q_{12} \\ q_{13} \end{bmatrix} = \begin{bmatrix} 3.05 \\ 6.08 \\ 8.40 \\ 6.95 \\ 4.10 \\ 2.92 \\ 1.30 \\ 6.32 \\ 7.56 \\ 2.38 \\ 2.61 \\ 2.59 \\ 4.53 \end{bmatrix}$$

Table 4.7. Hence, for each point i, it holds that $q_i = w_1 \cdot x_i + w_2 \cdot y_i$, where x_i and y_i are the coordinate values of point i on the given X- and Y-axes, respectively, and w_k is a weight.

Inserting this expression for q_i into the above loss function, we note that b can be pulled into the w_is so that the multiple regression problem in Table 4.7 emerges, where $g_i = b \cdot w_i$. Because X and Y are uncorrelated, the weights in Table 4.7 are simple regression weights. The additive constant a is simply the mean of the external scale. (One can eliminate a entirely by transforming the s_i-values into deviation scores.) The regression equation thus says that, given some point P such as point 1 with coordinates $(-0.41, -0.46)$, its corresponding q_1-value is $2.679 \cdot (-0.41) + 0.816 \cdot (-0.46) + 4.523 = 3.05$.

Overall, the resulting point coordinates on Q correlate with the external scale PU with .96, which checks with the $R^2 = .92$ from Table 4.6.

For the origin $O = (0.00, 0.00)$, we get $q_O = 0.00$, and so it is convenient to run Q through the origin O. For actually drawing Q in an MDS space, we have to find a second point on Q besides the origin. It can be shown that the regression weights g_i are the coordinates of such a point, provided Q runs through the origin O. Hence, we have two points that lie on Q, and this determines the line. In the given case, these points are $O = (0.00, 0.00)$ and $(2.679, 0.816)$.

A second possibility is locating the line Q on the basis of its angles to the coordinate axes. The direction cosine[4] of line Q with the ath coordinate

[4]The *direction cosine* of Q with the coordinate axis A_i is the cosine of the angle that rotates the positive end of Q onto the positive end of A_i.

TABLE 4.8. Coordinates of points $1, \ldots, 13$ of Fig. 4.8 projected onto the axes A_1, A_2, and A_3 of Fig. 4.8; r is the correlation of axis A_i with the PU values in Table 4.3.

	A_1	A_2	A_3
1	-0.564	0.136	0.526
2	0.802	-0.421	-0.368
3	1.748	-0.719	-1.133
4	1.479	-0.980	-0.338
5	0.165	-0.417	0.475
6	-1.029	0.722	0.170
7	-2.049	1.426	0.356
8	0.655	-0.121	-0.672
9	1.544	-0.810	-0.709
10	-0.811	0.185	0.774
11	-0.895	0.399	0.528
12	-1.635	1.412	-0.173
13	0.590	-0.811	0.564
$r =$	0.920	-0.780	-0.800

axis can be computed directly by the formula $\alpha_a = \cos^{-1}(g_a / \sum_{a=1}^{m} g_a^2)$, where g_a is the regression weight of the ath coordinate axis.

Because of the close relationship between regression weights and direction angles, we can conceive of the problem of representing an external scale by a line as a rotation problem: the task is to turn a line running through the origin such that the projections of the points on it correspond best to a given external scale. Figure 4.8 demonstrates this notion. A line or, rather, a directed axis is spun around the origin until it reaches an orientation where the points of the MDS configurations project on it so that these projections correlate maximally with the external scale. Three axes (A_1, A_2, and A_3) are shown graphically. The corresponding point projections are exhibited in Table 4.8. The table also shows the correlations of the projections with the scale values for the external PU scale from Table 4.3. One notes that A_1 has a high positive correlation with the PU scale, which indicates that the X-axis of the MDS solution can be interpreted as a continuum ranging from unpleasant to pleasant (see also Figure 4.9).

3D MDS of the Faces Data with Embedded External Scales

Because Schlosberg's theory is a theory of three dimensions, we also take a look at the 3D MDS solution. Figure 4.10 exhibits this configuration, together with the embedded external scales. Before going into further interpretations, we note that such a 3D configuration is not easy to look at, because what we see here is only a projection of this configuration onto a plane. The reader always has to mentally reconstruct the original configuration from this projection, which is often a difficult task. We note,

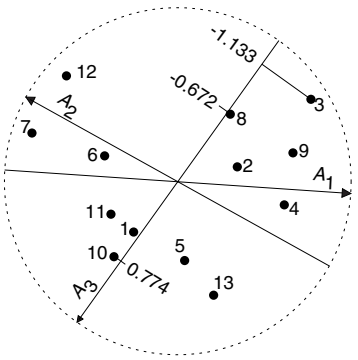


FIGURE 4.8. Embedding of an external scale into an MDS configuration (faces data) as a rotation problem.

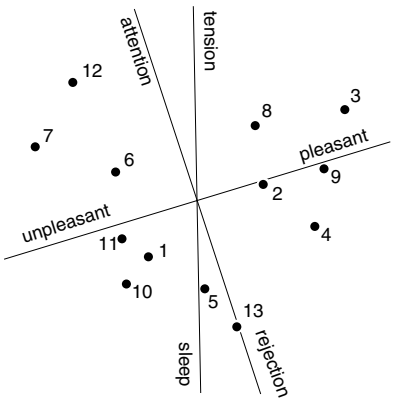


FIGURE 4.9. 2D MDS of faces data, with optimally fitted external scales.

for example, that it is almost impossible to see from Figure 4.10 how the embedded scales are oriented in the space.

One gets a clearer picture from the correlations of the embedded scales with the coordinate axes (Table 4.6). In addition, it is sometimes worthwhile to make use of features offered by the graphical environment of some MDS programs, in particular the possibility of rotating 3D configurations online in space. This allows one to inspect the configuration from different perspectives on the computer screen, which may suffice to understand the spatial relationships.

Figure 4.10, in any case, seems to suggest that the external scales PU and TS essentially correspond to Cartesian dimensions, whereas AR does not explain much additional variance. This is not surprising because r(TS, AR) = .75 in Table 4.3. Yet, there is quite a bit of scatter of the points in the third dimension. That this can only be partially explained by the external scales may be a consequence of the different psychology involved in generating the global similarity judgments and the ratings on the external scales. The given evidence is at least not contradictory to Schlosberg's theory of three dimensions.

4.4 General Principles of Interpreting MDS Solutions

The above MDS applications are chosen to show the reader some real-data examples, with substantive questions linked to them. The question of interpretation asked for connections of geometric properties of the MDS

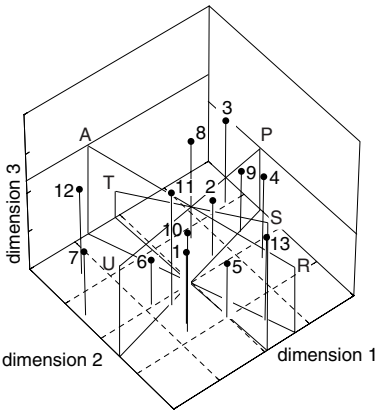


FIGURE 4.10. 3D MDS of faces data, with fitted external scales.

representation and substantive aspects of the represented objects. In the case of the color data, we found that the electromagnetic wavelengths of the colors were reflected in a corresponding array of the points in MDS space along a (curved) line. For the Morse code signals, we found that certain physical properties of the signals had a systematic relationship to various regions of the MDS space. The facial expression data led to an MDS configuration whose dimensions were strongly related to external scales for the same stimuli.

These examples illustrate the three most common principles used in interpreting MDS solutions. The color circle is an instance of a particular *manifold*, which is any set of points that form objects in space that are nearly "flat" in the neighborhood of any of their points ("locally" Euclidean). Most often, manifolds refer to points that form smooth curves or surfaces in space.

The regional interpretation of the Morse code data resulted from partitioning the space in multiple ways. The criteria used were different physical properties of the Morse code stimuli. In each case, the goal was to split the space such that each region would contain only points representing stimuli with equivalent properties on the partitioning criterion. Nothing else is required by this interpretational approach and, therefore, many different regional patterns may arise. Special cases are clusters—that is, very dense regions separated from each other by "empty space"—and dimensions. The latter partition the space into intervals, checkerboard patterns, box-like cells, and so on, depending on the dimensionality m. A regional interpretation is also possible for the color data: if we use wavelength as the physical property of the stimuli, each region contains but a single point, but coarser partitionings result from lumping together the stimuli into such classes as red, blue, yellow, and green.

Finally, the facial expression example illustrated the dimensional approach, the most common interpretation in practice. Note, however, that interpreting dimensions means that one is trying to link a very particular geometric feature to substantive features of the represented objects. One should not expect that this will always be successful.

These applications were, in a sense, confirmatory ones, because, in each case, there was at least an implicit expectation about certain properties of the MDS configuration. But even in a more exploratory context, interpreting MDS configurations complies with the same logic, except that some of the features of the stimuli one links to the MDS geometry are hypothesized or assumed. That is, looking at an MDS configuration and trying to make sense out of it simply means that one projects various forms of prior knowledge onto this space in order to explain the configuration. If this prior knowledge is solid, then exploratory MDS is also solid. Otherwise, one has to test the stability of such interpretations over replications.

In principle, *any* geometric property of an MDS solution that can be linked to substance is an interesting one. However, in the literature, certain standard approaches for interpretation are suggested, that is, particular geometric properties that one should consider. By far, the most popular approach is to look for meaningful directions or dimensions in the MDS space. Naturally, dimensions may not be related in any interesting way to the objects' substance, nor is any other feature of an MDS configuration.

4.5 Exercises

Exercise 4.1 In this exercise, we have a closer look at the choice of dimensionality for the color data of Ekman (1954) from Section 4.1.

(a) Compute MDS solutions for the data in Table 4.1 in 1, 2, 3, 4, 5, and 6 dimensions. Make a scree plot. What do you conclude with respect to the proper dimensionality of the MDS solution?

(b) Discuss a few criteria from Section 3.5 for choosing the proper dimensionality of the MDS solution.

Exercise 4.2 Figure 4.1 gives an MDS representation for the subjective similarity assessments of different colors. These colors are characterized by their electromagnetic wavelengths. Yellow corresponds to about 570 nm, green to 520 nm, blue to 480 nm, and violet to about 380–450 nm. Orange starts at about 600 nm and turns into red at the end of the visible spectrum (above 650 nm). For answering (b) and (c), you may want to consult an introductory psychology textbook.

(a) With this background, interpret the MDS configuration in terms of two meaningful color dimensions.

(b) What kind of MDS configuration could be expected if the color stimuli would vary not only in hue, but also in saturation?

(c) What color would you expect to lie at the center of the circle?

Exercise 4.3 Consider the facial expression data of Section 4.3.

(a) Compute the angle between the X-axis and the lines that best represent the scales PU, AR, and TS, respectively, of Table 4.3 in the MDS configuration of Table 4.5.

(b) The angles for AR and TS are similar. What does that mean in terms of the data?

(c) What substantive conclusions do you draw from (b)?

Exercise 4.4 Rosenberg and Kim (1975) studied the similarity of 15 kinship terms. College students sorted the terms on the basis of their similarity into groups. Each student generated a dissimilarity matrix where a pair of objects was coded as 1 if the objects were sorted in different groups and as 0 if the objects were sorted in the same group. The table below gives the percentage of how often terms were *not* grouped together over all students.

Kinship Term	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1 Aunt	–														
2 Brother	79	–													
3 Cousin	53	67	–												
4 Daughter	59	62	74	–											
5 Father	73	38	77	57	–										
6 Granddaughter	57	75	74	46	79	–									
7 Grandfather	77	57	76	77	51	57	–								
8 Grandmother	55	80	78	54	70	32	29	–							
9 Grandson	79	51	72	72	54	29	31	57	–						
10 Mother	51	63	79	31	29	56	75	50	79	–					
11 Nephew	56	53	51	74	59	74	58	79	51	81	–				
12 Niece	32	76	53	52	81	51	79	58	74	60	27	–			
13 Sister	58	28	70	37	63	50	79	57	75	39	76	53	–		
14 Son	80	38	73	29	32	72	55	78	47	57	52	74	62	–	
15 Uncle	27	57	51	80	51	80	55	77	58	73	33	56	79	59	–

In addition, for each of the kinship terms, external scales can be set up for gender (1 = male, 2 = female, 9 = missing), generation (-2 = two back, -1 = one back, 0 = same generation, 1 = one ahead, 2 = two ahead), and degree (1 = first, 2 = second, etc.) of the kinship term. The table below presents these external scales.

	Kinship Term	Gender	Generation	Degree
1	Aunt	2	-1	3
2	Brother	1	0	2
3	Cousin	9	0	4
4	Daughter	2	1	1
5	Father	1	-1	1
6	Granddaughter	2	2	2
7	Grandfather	1	-2	2
8	Grandmother	2	-2	2
9	Grandson	1	2	2
10	Mother	2	-1	1
11	Nephew	1	1	3
12	Niece	2	1	3
13	Sister	2	0	2
14	Son	1	1	1
15	Uncle	1	-1	3

(a) Do an ordinal multidimensional scaling analysis in two dimensions. Interpret the solution.

(b) Inspect the Shepard diagram or the transformation and residual diagrams. Are all proximities properly fitted?

(c) Compute the correlations between the dimensions and the external scales generation and degree, respectively. Use a multiple regression program to find optimal weights g_1 and g_2 to predict each external scale out of the two dimensions. Plot the two external scales in the solution. How can you interpret the solution in terms of generation and degree?

(d) Suppose that we would also like to represent gender in the MDS solution. Explain how this could be done. Elaborate your solution in the plot.

Exercise 4.5 Wolford and Hollingsworth (1974) were interested in the confusions made when a person attempts to identify letters of the alphabet viewed for some milliseconds only. A confusion matrix was constructed that shows the frequency with which each stimulus letter was mistakenly called something else. A section of this matrix is shown in the table below.

Letter	C	D	G	H	M	N	Q	W
C	–							
D	5	–						
G	12	2	–					
H	2	4	3	–				
M	2	3	2	19	–			
N	2	4	1	18	16	–		
Q	9	20	9	1	2	8	–	
W	1	5	2	5	18	13	4	–

(a) Are these data similarity or dissimilarity measures?

(b) Use MDS to show their structure.

(c) Interpret the MDS solution in terms of regions. What do you conclude with respect to letter confusion? (Hint: Letter confusion may be based, e.g., on visual features or on the similarity of sounds.)

Exercise 4.6 Consider the data on the subjective similarity of different countries in Table 1.3. The table below supplements these data by two external scales. The first scale consists of rankings on "economic development" that one particular student could have assigned to these countries in the 1960s. The second scale shows the population of these countries in about 1965.

Country	No.	Economic Development	Population (ca. 1965)
Brazil	1	3	87
Congo	2	1	17
Cuba	3	3	8
Egypt	4	3	30
France	5	8	51
India	6	3	500
Israel	7	7	3
Japan	8	9	100
China	9	4	750
USSR	10	7	235
U.S.A.	11	10	201
Yugoslavia	12	6	20

(a) Find the coordinates of a two-dimensional ordinal MDS representation of the data in in Table 1.3.

(b) Fit the external scales into this MDS space by linear regression. Plot the embedded scales as directed lines.

(c) Interpret the MDS solution in terms of the external scales, if possible. Discuss how successful these two scales are in explaining the MDS configuration.

5
MDS and Facet Theory

Regional interpretations of MDS solutions are very general and particularly successful approaches for linking MDS configurations and substantive knowledge about the represented objects. Facet theory (FT) provides a systematic framework for regional interpretations. FT structures a domain of interest by partitioning it into types. The typology is generated by coding the objects of interest on some facets of their content. The logic is similar to stratifying a sample of persons or constructing stimuli in a factorial design. What is then tested by MDS is whether the distinctions made on the conceptual (design) side are mirrored in the MDS representation of the objects' similarity coefficients such that different types of objects fall into different regions of the MDS space.

5.1 Facets and Regions in MDS Space

Interpreting an MDS solution means linking geometric properties of the configuration to substantive features of the represented objects. A very general approach is to interpret regions of an MDS space. Regional interpretations are put into a systematic framework in facet theory (Guttman, 1959, 1991; Borg & Shye, 1995).

Elements of Facet Theory

The central notion of facet theory (FT) is that of a *facet*. A facet is a scheme used to classify the elements of a domain of interest into types. The facet "gender", for example, classifies persons into males and females. Similarly, the facet "behavior modality" classifies attitudinal behavior into emotional, cognitive, and actional behavior. Using several facets at the same time partitions a domain of interest into multifaceted types. Consider the tasks contained in an intelligence test, for example. In FT, such tasks are *intelligence items*, defined as questions that ask about an individual's behavior and assess it on a scale from "very right" to "very wrong" according to an objective rule (Guttman, 1965). A particular case of intelligence items are the tests in paper-and-pencil intelligence test batteries. Such tests require the testee to find verbal analogies, solve arithmetic problems, and identify patterns that complete series of figures, for example. Hence, they can be classified by the facet "language of presentation" into numerical, verbal, and geometrical ones. At the same time, such tests relate to different abilities, which gives rise to a second facet, "required mental operation". It classifies tests into those where the testee has to infer, apply, or learn a rule, respectively (Guttman & Levy, 1991). In combination, these two facets distinguish nine types of intelligence: numerical tests requiring the testee to infer a rule, numerical tests requiring the testee to apply a rule, ..., geometrical tests requiring the testee to learn a rule.

In FT, facets are typically not just listed but rather expressed in the framework of a *mapping sentence*. It shows the roles the facets play relative to each other and relative to what is being observed, that is, the *range* of the items. An example is the following.

$$\text{Person } \{p\} \text{ performs on a task presented in}$$

$$\left\{ \begin{array}{c} \underline{\text{language}} \\ \text{verbal} \\ \text{numerical} \\ \text{geometrical} \end{array} \right\} \text{language and requiring} \left\{ \begin{array}{c} \underline{\text{requirement}} \\ \text{learning} \\ \text{applying} \\ \text{inferring} \end{array} \right\} \text{an}$$

$$\text{objective rule} \rightarrow \left\{ \begin{array}{c} \underline{\text{range}} \\ \text{very right} \\ \text{to} \\ \text{very wrong} \end{array} \right\} \text{according to that rule.}$$

The terms enclosed in braces denote the facets.[1] The set of persons, p, is not stratified further in this example, whereas the questions are structured

[1]Instead of braces, one often uses vertical arrays of parentheses. Braces, however, correspond to the usual mathematical notation for listing the elements of a set. Formally, a facet is a set or, more precisely, a component set of a Cartesian product.

by the two facets from above, "requirement" and "language". The *range* of the mapping sentence is the scale on the right-hand side of the arrow. The arrow symbolizes an observational mapping of every person in p crossed with every (doubly coded) test into the range (data). Each such mapping specifies the response of a given person to a particular *type* of question. For each question type, there are generally thousands of concrete items.

Facets are invented for a particular *purpose*, that is, for systematically breaking up a domain of interest into subcategories or types in order to conceptually structure this domain. Take plants, for example. Botanists, painters, children, perfume makers, and the like, all invented category systems that allow them to order plants in some way that is meaningful for them. Good classification systems allow the user to unambiguously place each and every object into one and only one category. But good classification systems also serve a particular purpose beyond providing conceptual control: the different types distinguished by the classification system should, in one way or another, "behave" differently in real life. Whether this is true can be tested empirically and, hence, implies a hypothesis.

Facet Theory and Regions in MDS Spaces

A traditional specification of the hypothesis of empirical usefulness of a facet is that it should explain the data in some way. One way of testing this is to check whether the *distinctions* made by the facets are mirrored, facet by facet, in corresponding *differences* of the data. For example, tests that require the testee to infer, apply, or learn a rule, should lead to different responses of the testee. One particular specification of what is meant by "different" is that inferential tests are most difficult, in general, and learning tests are least difficult, with application tests in between. Another form of hypothesis is that different item types fall into different regions of an MDS representation of the item intercorrelations.

A regional hypothesis thus links content facets to regions of the empirical MDS space. The hypothesis is that the MDS space can be partitioned such that each region represents a different facet element.[2] That is, all points within a particular region should be associated with the same facet element, and points in different regions should be associated with different facet elements.

Consider an example. Table 5.1 shows the intercorrelations of eight intelligence test items, together with *structuples*, that is, codings of the items on the facets "language" and "requirement" discussed above. Item 1 in Ta-

[2]In a plane, a region is defined as a connected set of points such as the inside of a rectangle or a circle. More generally, a set of points is connected if each pair of its points can be joined by a curve all of whose points are in the set. Partitioning a set of points into regions means to split the set into classes such that each point belongs to exactly one class.

TABLE 5.1. Intercorrelations of eight intelligence tests, together with content codings on the facets "language" = {N = numerical, G = geometrical} and "requirement" = {A = application, I = inference} (Guttman, 1965).

Language	Requirement	Test	1	2	3	4	5	6	7	8
N	A	1	1.00	.67	.40	.19	.12	.25	.26	.39
N	A	2	.67	1.00	.50	.26	.20	.28	.26	.38
N	I	3	.40	.50	1.00	.52	.39	.31	.18	.24
G	I	4	.19	.26	.52	1.00	.55	.49	.25	.22
G	I	5	.12	.20	.39	.55	1.00	.46	.29	.14
G	A	6	.25	.28	.31	.49	.46	1.00	.42	.38
G	A	7	.26	.26	.18	.25	.29	.42	1.00	.40
G	A	8	.39	.38	.24	.22	.14	.38	.40	1.00

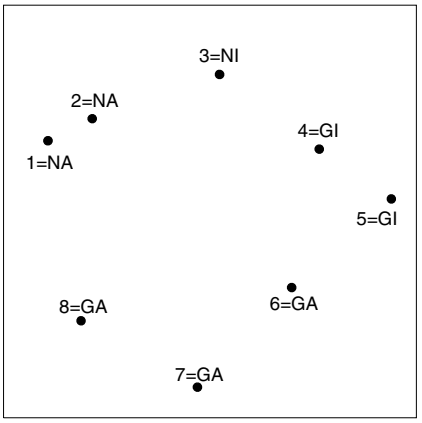


FIGURE 5.1. 2D MDS of correlations in Table 5.1.

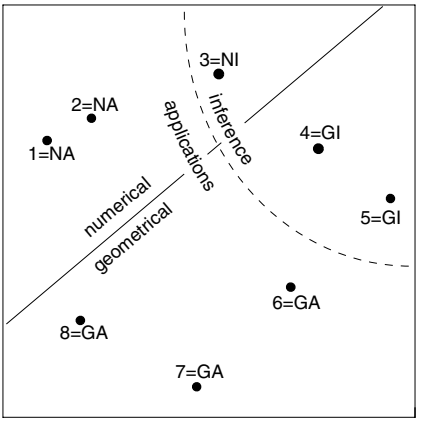


FIGURE 5.2. MDS space with four regions resulting from G- vs. N-, and A- vs. I-distinctions, respectively.

ble 5.1 is coded as numeric (on the facet "language") and as application (on the facet "requirement"), whereas item 5 is geometrical and inference. Rather than looking at these correlations directly, we represent them in a 2D MDS space (Figure 5.1). This can be done with the low Stress of .015.

Figure 5.2 demonstrates that the MDS configuration can indeed be cut such that each partitioning line splits it into two regions containing only points of one type: points of the N-type lie above the solid line, and points of the G-type below that line. The dashed line separates I-type points from A-type points. One notes in Figure 5.2 that there is considerable leeway in choosing the partitioning lines. Why, then, was a curved line chosen for separating I-type points from A-type points? The reason is that this line yields a structure that looks like a slice from the *universe* of all possible item types discriminated by the given two facets. If items of all nine types

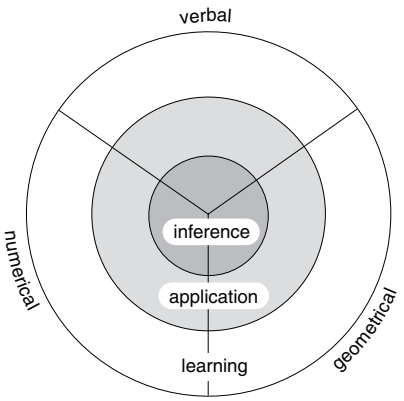


FIGURE 5.3. Schematic radex of intelligence items.

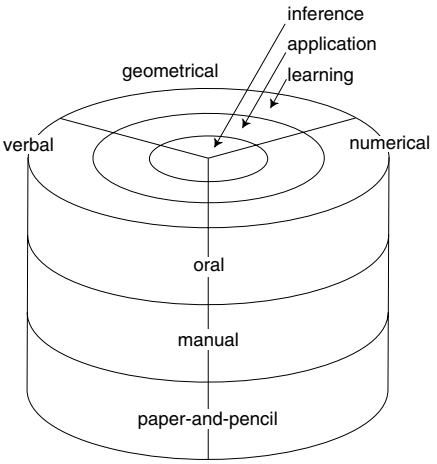


FIGURE 5.4. Cylindrex of intelligence items (after Guttman & Levy, 1991).

had been observed, one can predict that the MDS configuration would form a pattern similar to a dart board, or *radex*, shown schematically in Figure 5.3. If, in addition, one adds another facet, "communication", which distinguishes among oral, manual, and paper-and-pencil items, one obtains a 3D *cylindrex*, shown in Figure 5.4. In the cylindrex, "communication" plays the role of an axis along which the radexes for items using a fixed form of communication are stacked on top of each other.

Summarizing, we see that every facet contains additional information on the items in MDS. In a way, a facet can be seen as a design variable of the items: every item belongs to one of the categories of each and every facet. The facets are combined into a mapping sentence so that every item corresponds to one particular way of reading this mapping sentence. Some combinations of the categories may not be expressed by items, whereas other combinations may have more than one item. The facets and their categories (elements) are chosen on substantive grounds. Given a set of items classified by such facets, MDS tests whether the classification is reflected in a corresponding regionality of the representation space.

5.2 Regional Laws

The cylindrex structure has been confirmed so often for intelligence test items that now it is considered a *regional law* (Guttman & Levy, 1991). What Figure 5.2 shows, therefore, is a partial replication of the cylindrex law.

What does such a regional law mean? First of all, it reflects regularities in the data. For example, restricting oneself to items formulated in a particular language (such as paper-and-pencil tests) and, thus, to a radex as in Figure 5.3, one notes that inference items generally correlate higher among each other than application items, and learning items are least correlated. Thus, knowing that some person performs well on a given inference item allows one to predict that he or she will most likely also perform well on other inference items, whereas good performance on a given learning item says little about the performance on other learning items. One can improve the predictions, however, if one constrains them to learning tasks that use a particular language of presentation such as numerical tasks.

One notes, moreover, that the MDS regions for inference, application, and learning are ordered. This order cannot be predicted or explained from the properties of the qualitative facet "requirement", but it reliably shows up in hundreds of replications (Guttman & Levy, 1991). Thus, it seems unavoidable to ask for an explanation for this lawfulness. Ideally, what one wants is a definitional system that allows one to *formally derive* such ordered regions from its facets.

Snow, Kyllonen, and Marshalek (1984) proposed an explanation in this direction. They report a factor analysis that suggests that items which relate to points in the center of the radex (i.e., inference tasks) are "complex" items and those represented at the periphery (such as learning tasks) are "specific" items. This repeats, to some extent, what the radex says: items whose points are closer to the origin of the radex tend to be more highly correlated with other items. Snow et al. (1984) add, however, that more complex tasks show "increased involvement of one or more centrally important components." Hence, their explanation for the inference-application-learning order seems to be that these facet elements are but discrete semantic simplifications of a smooth gradient of complexity.

One can ask the complexity question in a different way and define a task t_1 as more complex than t_2 if "it requires everything t_1 does, and more" (Guttman, 1954, p. 269). Formally, this implies an interlocking of content structuples, which is analogous to the perfect Guttman scale. Specifying such structuples requires one to identify basic content facets with a common range, where the concepts "inference", "application", and "learning" then become only global labels for comparable (hence ordered) content structuples of these underlying facets. For a fixed element of the "language" facet, such a system would allow one to predict a particular order of regions (*simplex*).

But this leads to the question of what pulls the different simplexes—one for each type of required mental operation, that is, one for items that require application, learning, or inference of an objective rule, respectively—to a common origin? To explain this empirical structure requires an additional pattern in the structuples. Formally, for the three directions of the intelligence radex, it would suffice to have an additional coding of the items in

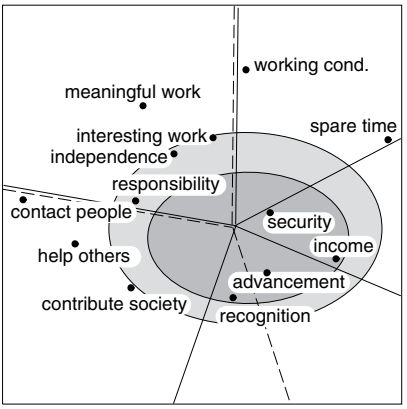


FIGURE 5.5. Radex partitionings of 13 work value items.

terms of the extent to which they require each of the three mental operations. In any case, with many points and/or differentiated facets, a simple correspondence between regions and structuples is a remarkable finding. Arbitrary assignments of structuples to the points do not, in general, lead to such lawfulness. Partitionings with relatively smooth cutting lines are generally also more reliable. Moreover, they help clarify the roles the various facets play with respect to the data. Such roles are reflected in the particular ways in which they cut the space.

5.3 Multiple Facetizations

A given object of interest can always be facetized in more than one way. Every new facet offers a new alternative. But then one may ask whether each such facetization is reflected in different statistical effects on the data side. Consider work values, for example. Work value items ask the respondent to assess the importance of different outcomes of his or her work. An example is the questionnaire item: "How important is it to you personally to make a lot of money?" with the range "very important . . . not important at all." Conceptually, two different kind of facets have been proposed for organizing such items: one facet distinguishes the work outcomes in terms of the need they satisfy, and the other facet is concerned with the allocation criterion for rewarding such outcomes. Consider Table 5.2, in which Borg and Staufenbiel (1993) coded 13 work value items in terms of seven facets. The facets and the structuples were taken from the literature on organizational behavior. Moorhead and Griffin (1989) argue, for example, that security in Maslow's sense interlocks with both Alderfer's relatedness and existence, but an item that is both Maslow-type security and Alderfer-type relatedness (item 10 in Table 5.2) is missing in the given sample of items.

TABLE 5.2. Work value items with various facet codings: H(erzberg) = {h = hygiene, m = motivators}; M(aslow) = {p = physiological, s = security, b = belongingness, r = recognition, a = self-actualization }; A(lderfer) = {e = existence, r = relations, g = growth}; E(lizur) = {i = instrumental-material, k = cognitive, a = affective-social}; R(osenberg) = {e = extrinsic, i = intrinsic, s = social}; L(evy-Guttman) = {i = independent of individual performance, g = depends on group performance, n = not performance dependent}; B(org-Elizur) = {1 = depends much on individual performance, 2 = depends more on individual performance than on system, 3 = depends both on individual performance and on system, 4 = depends on system only}.

Item	H	M	A	E	R	L	B	Work Value
1	m	a	g	k	i	g	3	Interesting work
2	m	a	g	k	i	g	3	Independence in work
3	m	a	g	k	i	g	3	Work that requires much responsibility
4	m	a	g	k	i	n	4	Job that is meaningful and sensible
5	m	r	g	k	e	i	1	Good chances for advancement
6	m	r	r	a	s	i	1	Job that is recognized and respected
7	h	b	r	a	s	n	4	Job where one can help others
8	h	b	r	a	s	n	4	Job useful for society
9	h	b	r	a	s	n	4	Job with much contact with other people
10	-	s	r	-	-	-	-	(No item of this type asked in study)
11	h	s	e	i	e	i	2	Secure position
12	h	s	e	i	e	i	1	High income
13	h	p	e	i	e	n	4	Job that leaves much spare time
14	h	p	e	i	e	n	4	Safe and healthy working conditions

Figure 5.5 shows a 2D MDS representation for the correlations of the 13 work value items assessed in a representative German sample. The radex partitioning is based on the facets "M(aslow)" (solid radial lines), "R(osenberg)" (dashed radial lines), and "L(evy-Guttman)" (concentric ellipses). It is easy to verify that the other facets also induce perfect and simple partitionings of this configuration. These partitionings are, moreover, quite similar: the respective regions turn out to be essentially congruent, with more or fewer subdivisions. Differences of the various wedge-like partitionings are primarily related to the outcome advancement, which is most ambiguous in terms of the need that it satisfies. Hence, one can conclude that all of these theories are structurally quite similar in terms of item intercorrelations. This suggests, for example, that Herzberg's motivation and hygiene factors correspond empirically to Elizur's cognitive and affective/instrumental values, respectively.

We note, moreover, that such similar partitionings of the MDS space into wedge-like regions—induced by different facets that are formally not equivalent—give rise to a partial order of the induced sectors. The interlocking of the Herzberg and the Maslow facets implies, for example, that the hygiene region contains the subregions "physiological", "security", and "belongingness", and the motivators' region contains the subregions "esteem" and "self-actualization". Hence, the subregions are forced into a certain neighborhood relation that would not be required without the hierarchical nesting. Similarly, the conceptual interlocking of the Maslow and the Alderfer facet requires "esteem" to fall between "self-actualization" and 'belongingness".

Elizur, Borg, Hunt, and Magyari-Beck (1991) report further studies on work values, conducted in different countries, which show essentially the same radex lawfulness. Note that this does not imply similarity of MDS configurations in the sense that these configurations can be brought, by admissible transformations, to a complete match, point by point (for such matchings; see Chapter 20). Rather, what is meant here is that several configurations (which do not even have to have the same number of points) exhibit the same law of formation: they can all be partitioned in essentially the same way (i.e., in the sense of a radex) by just one fixed coding of the items, thus showing similar *contiguity patterns* (Shye, 1981).

5.4 Partitioning MDS Spaces Using Facet Diagrams

Partitioning an MDS space is facilitated by using *facet diagrams*. Facet diagrams are simply subspaces—usually 2D projection planes—of the MDS space where the points are labeled by their structuples or, better, by their codings on just one facet (*structs*). This usually enables one to see the

distribution of the points in terms of the particular typology articulated by each facet.

Consider an example that also explicates further aspects of facet theory (Galinat & Borg, 1987). In experimental investigations a number of properties of a situation have been shown, one by one, to have an effect on judgments of duration of time. The following mapping sentence shows four of these properties within a design meant to measure symbolic duration judgments, that is, duration judgments on hypothetical situations.

$$\text{Person } \{p\} \text{ believes that the } \left\{ \begin{array}{c} \underline{\text{positivity}} \\ p_1 = \text{pleasant} \\ p_2 = \text{neutral} \\ p_3 = \text{unpleasant} \end{array} \right\} \text{ situation with}$$

$$\left\{ \begin{array}{c} \underline{\text{number}} \\ m_1 = \text{many} \\ m_2 = \text{few} \end{array} \right\} \left\{ \begin{array}{c} \underline{\text{variability}} \\ v_2 = \text{monotonous} \\ v_1 = \text{variable} \end{array} \right\} \text{ events that are}$$

$$\left\{ \begin{array}{c} \underline{\text{difficulty}} \\ s_1 = \text{difficult} \\ s_2 = \text{easy} \end{array} \right\} \text{ to handle is felt as } \rightarrow \left\{ \begin{array}{c} \underline{\text{reaction}} \\ \text{very short in duration} \\ \text{to} \\ \text{very long in duration} \end{array} \right\}.$$

The mapping sentence first shows a placeholder for the population of respondents $\{p\}$. In each particular way of reading the mapping sentence, one element of $\{p\}$ is picked and crossed with one particular combination of the elements of the *content facets*. The content facets distinguish among different situations by considering four properties of its events: "positivity of events", "number of events", "variability of events", and "difficulty to handle events". With the number of facet elements we have specified here—3 on "positivity", 2 on "number", 2 on "variability", and 2 on "difficulty"—we have $3 \cdot 2 \cdot 2 \cdot 2 = 24$ different situation types. For example, a situation with structuple (p_3, m_2, v_1, s_2) or, for short, 3212 is defined to be an unpleasant one, where few things are happening, with much variability, and no problems to cope with what is going on.

What we are interested in is how persons judge the duration of these 24 situation types. The mapping sentence identifies the characteristics of these situations in a relatively abstract way. For each type of situation, concrete examples must be constructed in order to have items that can be presented to respondents for assessment. The following story illustrates a concrete item for a situation of type $p_1 m_1 v_1 s_2$. "You are playing a simple card game with your children. It is quite easy for you to win this game because your kids are no serious opponents. The game requires you to exchange many different cards. The game is fun throughout the three minutes that it lasts." This description is supplemented by the question, "What do you think; how long would this card game seem to last? Would it seem longer or shorter than three minutes?"

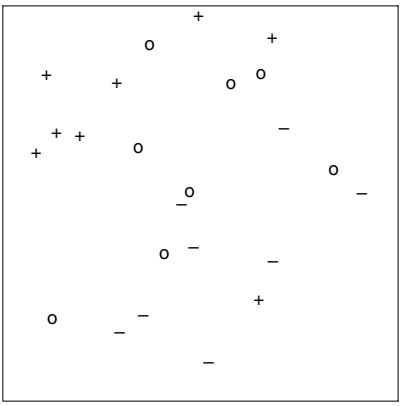


FIGURE 5.6. Facet diagram for duration judgments and facet "positivity".

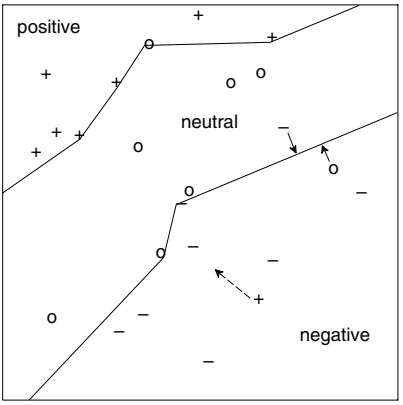


FIGURE 5.7. Facet diagram with axial partitioning.

A sample of persons rated this and 23 other hypothetical situations on a 7-point scale from "a lot shorter" (coded as 1) to "a lot longer." This bipolar scale, together with the question, "What do you think: how long ...?", is a concrete specification for the generic response range "very long ... very short in duration" in the above mapping sentence.

The intercorrelations of the 24 items are mapped into a 4D MDS space (with Stress = .13). Four dimensions are chosen because we assume that each facet can be completely crossed with any other. We now look at this space in terms of two projection planes. Figure 5.6 shows the plane spanned by the first two principal axes of the MDS configuration. Its points are labeled by the structs of each point on the facet "positivity". That is, points labeled as – in this facet diagram represent situations defined as p_3 = unpleasant. (Instead of –, one could also have chosen p_3, "unpleasant", "neg", "3", or any other symbolism, of course.) The facet diagram shows immediately that the points marked as +, o, and – are not distributed randomly. Rather, the plane can be partitioned into regions so that each region contains only or almost only points of one particular type. Figure 5.7 shows such a partitioning. It contains two minor errors: the two solid arrows indicate where these points "should" lie to be in the appropriate regions. Obviously, they are not far from the boundaries of these regions. There is also one further, and gross, error: a "positive" point located in the "negative" region. The dashed arrow attached to this point indicates the direction of required shifting.

Figure 5.8 represents an alternative partitioning that is error-free. This partitioning depends, however, very much on the position of the one point marked by an arrow. Thus, it may be less reliable in further replications. Moreover, the two partitionings imply different things. The concentric regions of Figure 5.8 predict that the duration ratings on unpleasant situa-

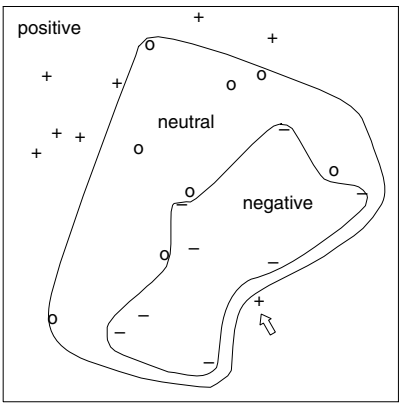


FIGURE 5.8. Facet diagram with modular partitioning.

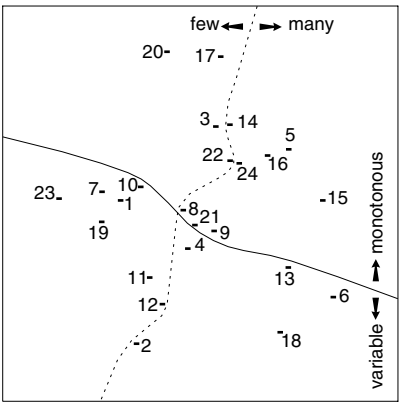


FIGURE 5.9. MDS projection plane of 24 duration situations, spanned by third and fourth principal components, partitioned by facets "variability" and "number".

tions should correlate higher among each other, on the average, than those for pleasant situations. The parallel regions of Figure 5.7 do not thus restrict the correlations. Nevertheless, both partitions are similar in splitting the plane into *ordered* regions, where the neutral region lies in between the positive and the negative regions. Hence, the regions are ordered as the facet "positivity" itself. Neither the spatial organization induced by the straight lines nor that induced by concentric circular lines would therefore have problems in accommodating a "positivity" facet, which distinguishes many more than just three levels. This is important because what we want, eventually, is not a theory about some particular sample of stimuli but one about the *universe* of such situation types. We thus see that the facet "positivity" is reflected in the structure of the duration ratings. The decision on which of the two partitionings is ultimately correct requires further data.

Figure 5.9 shows another plane of the 4D MDS space. It is spanned by principal axes 3 and 4 of the space and is therefore *orthogonal* to the plane in Figures 5.6–5.8. That is, each of its axes is perpendicular to both axes used in Figures 5.6–5.8. One recognizes from the respective facet diagrams (not shown here) that the configuration in this plane can be partitioned by the facet "number"—without error—and also by "variability"—with two errors.

The facet "difficulty" does not appear to show up in the MDS configuration; that is, the points representing easy and difficult situations, respectively, seem to be so scrambled that they can be discriminated only by very "irregular" partitionings. Such partitionings are, however, rarely useful. Note, though, that just looking at various orthogonal planes does not guarantee that one will detect existing regional patterns because such

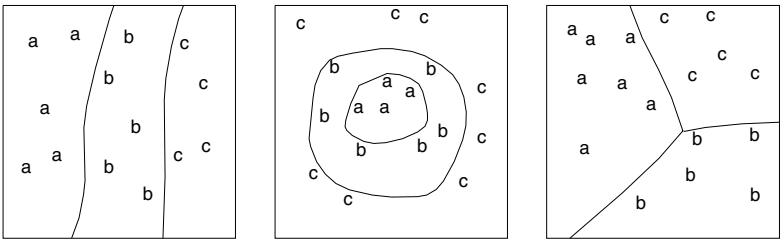


FIGURE 5.10. Three prototypical roles of facets in partitioning a facet diagram: axial (left panel), modular (center), and polar (right).

patterns may be positioned obliquely in space. This remains an unsolved problem that is particularly relevant in higher-dimensional spaces. In any case, using various spatial rotations and projections, we at least could not identify any simple regions related to the facet "difficulty" (Galinat & Borg, 1987).

MDS thus shows that the structure of the duration ratings can, in a way, be explained by three of the four facets of the design. This explanation is, moreover, compatible with considerations that extend beyond the sample of the given 24 concrete situations and that relate to their universe.

5.5 Prototypical Roles of Facets

With the partitionings shown in Figures 5.7 and 5.9, one arrives at an embryonic Cartesian coordinate system spanned by the three facets "positivity", "number", and "variability". Another coordinate system is suggested if we accept the circular partitioning shown in Figure 5.8. In this case, we have some evidence for a *polar* coordinate system of these facets.

The coordination of the MDS configuration in these examples is not chosen arbitrarily. Rather, it relates naturally to content. We stress this point here because the data determine only the distances among the points, not any dimensions. Dimensions are either superimposed onto the distance geometry in order to be able to replace ruler-and-compass construction methods by computation, or they may result from projecting content onto the geometry, as we saw earlier.

The content facets often play one of three prototypical roles in this context. This is shown in the three panels of Figure 5.10. The panels exhibit schematic facet diagrams, whose points are labeled as a, b, and c. In the panel on the left-hand side, the space is partitioned in an *axial* way. The panel in the center shows a *modular* partitioning. The panel on the right-hand side shows a *polar* facet. An axial facet is one that corresponds to a dimension; that is, the partitioning lines cut the space into subspaces

that look like parallel stripes of the plane (*axial simplex* of regions; see also Figure 5.7). A modular facet leads to a pattern that looks like a set of concentric bands (*radial simplex* of regions; see also Figure 5.8). Finally, a polar facet cuts the space, by rays emanating from a common origin, into sectors, similar to cutting a pie into pieces (*circumplex* of regions; see also Figure 5.3).

A number of particular combinations of facets that play such roles lead to structures that were given special names because they are encountered frequently in practice. For example, the combination of a polar facet and a modular facet in a plane, having a common center, constitutes a *radex* (see Figure 5.3). Adding an axial facet in the third dimension renders a *cylindrex*. Another interesting structure is a *multiplex*, a conjunction of at least two axial partitionings (see Figure 5.9). Special cases of the multiplex are called *duplex* (two axial facets), *triplex* (three axial facets), and so on. The multiplex corresponds to the usual (Cartesian) coordinate system ("dimensions") as a special case if the facets are (densely) ordered and the partitioning lines are straight, parallel, and orthogonal to each other.

There is also a variety of structures that are found less frequently in practice, for example, the *spherex* (polar facets in three-dimensional space) or the *conex* (similar to the cylindrex, but with radexes that shrink as one moves along its axial facet).

5.6 Criteria for Choosing Regions

Partitionings of geometric configurations that consist of only a few points are relatively easy to find. However, there is often so much leeway for choosing the partitioning lines that their exact shape remains quite indeterminate. More determinacy and greater falsifiability are brought in by increasing the number of items. Another principle for restricting the choice of partitioning lines is to think beyond the sample. In Figure 5.2, the partitioning lines were chosen, in part, by considering the universe of all intelligence items, a cylindrex.

Thinking beyond what was observed is always desirable, although it is, of course, impossible to say in general how this could be done. Most researchers typically are interested in generalizing their findings to the entire content universe, to additional populations, and over replications. The system of partitioning lines therefore should be *robust* in this respect, and not attend too much to the particular sample. Simple partitionings with relatively smooth cutting lines are typically more robust. But what is simple? Surely, a regionalization consisting of simply connected regions as in an axial or an angular system is simple, but so are the concentric bands of a circumplex. Hence, simple means, above all, that the partitioning is simple to characterize in terms of the roles of the facets that induce the

regions. Naturally, if one admits greater irregularities (i.e., not requiring the lines to be so stiff locally), then the number of errors of classification can generally be reduced or even eliminated. However, such error reduction typically makes it more difficult to describe the structure and, as a consequence, makes it harder to express how the facets act on the MDS space. Moreover, irregular ad hoc partitionings also reduce the likelihood of finding similar structures in replications and in the universe of items. One thus faces a trade-off decision of the following kind. Should one use relatively simple partitionings at the expense of more errors? Or should one choose more irregular lines to avoid classification errors, and then leave it to the reader to simplify these patterns? Obviously, one has to decide what seems most appropriate in the given context.

Irregular lines cast doubts on the falsifiability of regional hypotheses. Partitionings become less likely to result from chance the more points they classify correctly, the more differentiated the system of facets is, the simpler the partitioning lines are, and the greater the stability of the pattern is over replications. For arbitrary structuples, one should not expect to find regional correspondences in the data. To see this, we simulate this case by randomly permuting the structuples in Table 5.1. Assume that this has led to the assignments 1 = GA, 2 = NI, 3 = GA, 4 = NA, 5 = GI, 6 = NA, 7 = GI, and 8 = GA. If we label the points in Figure 5.2 by these structuples, we find that the plane can be partitioned in a modular way by the facet {A, I}, but that the A-points are now in the center in between the I-points. That does not correspond to the structure of the content universe, the cylindrex, which was replicated in hundreds of data sets (Guttman & Levy, 1991). The second facet, {G, N}, leads to a partitioning line that winds itself snake-like through the circular MDS configuration. It thus shows that separating the G- from the N-points with a reasonably regular line is only possible because we have so few points. It can hardly be expected that such an artificial partitioning can be replicated in other and richer data sets.

In addition to these formal criteria, one must request that the pattern of regions also ultimately makes sense. Yet, irregular lines are already difficult to describe as such and, as a consequence, complicate the search for explaining the way in which the regions are related to the facets. Moreover, in the given case, the radial order of inference, application, and learning is not only replicable, but also seems to point to an ordered facet "complexity", where inference is the most complex task (see above). If application items, then, come to lie in the radex center, such further search for substantive meaning is thwarted.

To avoid seemingly arbitrary partitionings or to aid in partitioning MDS spaces, Shye (1991) proposed a computerized method for partitioning facet diagrams in three ways: (1) in an axial way, by parallel and straight lines; (2) in a modular way, by concentric circles; and (3) in a polar way, by rays emanating from a common origin. The program yields graphical dis-

plays of three optimal partitionings, and measurements of the goodness of these partitionings by providing a *facet separation index* based on the sum of distances of the "deviant" points from their respective regions and normalized by the separability that can be expected for random data (Borg & Shye, 1995). Using this procedure suggests, for example, that a concentric-circles partitioning is best in terms of separability for the facet E = {i = instrumental-material, k = cognitive, a = affective-social} for the configuration in Figure 5.5. This finding conflicts with our previous decision to use polar partitioning for the very similar facet suggested by Rosenberg. On closer inspection, one notes, however, that it hinges on the location of one point, that is "good chances for advancement." This work value was categorized by Elizur as cognitive, but for a representative sample it may be better categorized as instrumental-material, because higher pay, more job security, and better working conditions may be more what most people have in mind when they assess the importance of advancement. Another criterion that speaks against the concentric-circles partitioning is that it induces ordered regions. The concentric circles that lead to the best separability index for facet E with respect to the given MDS configuration place the affective region in between the instrumental region and the cognitive region. Hence, the regions are ordered in this partitioning, while the facet only makes nominal distinctions, and no rationale for this order seems obvious a posteriori, except that affective values may be more highly intercorrelated than cognitive or instrumental values, in general. Naturally, such content considerations, as well as generalizability and replicability, must be considered in addition to formal separability measures for a given sample representation.

5.7 Regions and Theory Construction

Definitions and data are intimately linked through correspondence hypotheses not only at a particular point in time, but they are also related to each other over time in a "partnership" (Guttman, 1991) of mutual feedback. The definitions serve to select and structure the observations. The data then lead to modifications, refinements, extensions, and generalizations in the definitional framework. There is no natural beginning of this partnership between data and definitions. Hence, a correspondence between data and definitions can also be established a posteriori. That is, one may recognize certain groupings or clusters of the points, and then think about a rationale afterwards to formulate new hypotheses. When the definitional framework is complex, one typically does not predict a full-fledged regional system (such as a cylindrex) unless past experience leads one to expect such a system. Rather, one uses a more modest strategy with exploratory characteristics, and simply tries to partition the space, facet by facet, with mini-

mum error and simple partitioning lines. Even more liberal and exploratory is the attempt to identify space partitions according to new content facets that were not conceived in advance. The stability of such partitions is then tested in replications.

Replicating a regional correspondence, and thereby establishing an empirical law, is not sufficient for science. Researchers typically also want to understand the law. Why, for example, are work values organized in a radex? An answer to this question can be derived, in part, from reasoning in Schwarz and Bilsky (1987). These authors studied general values. One of the facets they used was "motivational domain" = {achievement, self-direction, security, enjoyment, ...}. These distinctions were considered nominal ones, but there was an additional notion of substantive *opposition*. Four such oppositions were discussed, for example, achievement vs. security: "To strive for success by using one's skills usually entails both causing some change in the social or physical environment and taking some risks that may be personally or socially unsettling. This contradicts the concern for preserving the status quo and for remaining psychologically and physically secure that is inherent in placing high priority on security values" (p. 554). Hence, the region of achievement values was predicted to lie opposite the security region. If we use this kind of reasoning post hoc on the work value radex of Figure 5.5, we could explain the opposite position of the sectors v and a (in Maslow's sense) by a certain notion of "contrast" of striving for self-actualization and for recognition, respectively. This notion of contrast is derived from a basic facet analysis of action systems (Shye, 1985). The same facet analysis also explains the neighborhood of regions like recognition and security, for example.

To predict regional patterns requires one to clarify the expected roles of the facets in the definitional framework. This involves, first of all, classifying the scale level of each facet. For ordered facets, one predicts a regional structure whose regions are also ordered so that the statement that some region R comes "before" another region R′ has meaning. The order of the regions should correspond to the order specified for the elements of the corresponding facet. For qualitative facets, any kind of simple partitionability of the point configuration into regions is interesting. The distinction of facets into *qualitative* and *ordinal* ones represents a "role assignment" (Velleman & Wilkinson, 1994) that is "not governed by something inherent in the data, but by interrelations between the data and some substantive problem" (Guttman, 1971, p. 339), that is, by certain correspondence hypotheses linking the observations and the definitional system. Hence, if one can see a conceptual order among the facet's elements and hypothesize that this order is mirrored in the observations collected on corresponding items, then the facet "is" ordered—for testing the hypothesis. Scale level thus remains context-related.

Consider as an example the facet "color" = {red, yellow, green, blue, purple}. One would be tempted to say, at first, that this "is" a nominal

facet. Yet, with respect to similarity judgments on colors, "color" has been shown to be ordered empirically in a circular way (see Chapter 4). Furthermore, with respect to the physical wavelength of colors, "color" is linearly ordered.

5.8 Regions, Clusters, and Factors

As is often true with concepts used in FT relative to similar ones in data analysis, the FT notion is more general. An important example is that regions include clusters as a special case. Lingoes (1981) proposes a faceted way to distinguish among different types of regions. He suggests that a cluster is a particular region whose points are all closer to each other than to any point in some other region. This makes the points in a cluster look relatively densely packed, with "empty" space around the cluster. For regions, such a requirement generally is not relevant. All they require is a rule that allows one to decide whether a point lies within or outside the region. The points 5 and 6 in Figure 5.2 are in different regions, but complete linkage clustering (a common type of cluster analysis), for example, puts them into one cluster together with point 4, and assigns points 7 and 8 to another cluster. For regions, the distance of two points—on which clustering is based—does not matter. Indeed, two points can be very close and still be in different regions. Conversely, two points may be far apart and still belong to the same region. As an analogy, consider Detroit (Michigan) and Windsor (Ontario). These cities are much closer than Detroit and Los Angeles, for example, but Detroit and Los Angeles are both in the same country, whereas Detroit and Windsor are not. In regions, all that counts is discriminability. Moreover, clusters are usually identified on purely formal criteria, whereas regions are always based on substantive codings of the represented objects. Guttman (1977) commented therefore as follows: "... theories about non-physical spaces ... generally call for continuity, with no 'vacuum' or no clear separation between regions... The varied data analysis techniques going under the name of 'cluster analysis' generally have no rationale as to why systematic 'clusters' should be expected at all... The term 'cluster' is often used when 'region' is more appropriate, requiring an outside criterion for delineation of boundaries" (p. 105).

Factors from factor analyses are not directly related to regions or to clusters. However, it is often asked in practice what one would have found if one had analyzed a correlation matrix by factor analysis rather than by MDS. Factor analysis, like cluster analysis, is a procedure that is substantively "blind" (Guttman, 1977) or that, if used in a confirmatory way, forces a preconceived formal structure onto the data representation, namely "factors". The factors are (rectilinear) dimensions that are run through point clusters, usually under the additional constraint of mutual orthogonality.

For Table 5.1, a factor analysis yields three factors with eigenvalues greater than 1. After varimax rotation, one finds that these factors correspond to three clusters in Figure 5.1, {1,2,3}, {4,5,6}, and {6,7,8}. Hence, in a way, the factors correspond to a polar partitioning of the MDS configuration in the given case, with three factors or "regions" in a 2D MDS space. With positive correlation matrices, this finding is rather typical; that is, one can expect $m + 1$ factor-induced regions in an m-dimensional MDS space. The reason for this is that positive correlations are conceived of in factor analysis as a vector bundle that lies in the positive hyperoctant of the Cartesian representation space, whereas MDS—which does not fix the origin of the space—looks only at the surface that contains the vector endpoints. Thus, Figure 5.1 roughly shows the surface of a section of the sphere whose origin lies somewhere in the center of the points but behind (or above) the plane (Guttman, 1982). The factors then correspond to a tripod fixed to the origin and rotated such that its axes lie as close as possible to the points. Hence, one notes that the location of this dimension system is highly dependent on the distribution of the points in space, whereas this is irrelevant for regions, although, of course, a very uneven distribution of the points in space will influence the MDS solution through the Stress criterion.

5.9 Exercises

Exercise 5.1 Consider the multitrait-multimethod matrix below (Bagozzi, 1993). It shows the correlations among nine items. The items assess the traits global self-esteem, social self-esteem, and need for order. Each trait is measured by three methods: true–false, multipoint, and simple self-rating scales.

Item	No.	1	2	3	4	5	6	7	8	9
T_1M_1	1	(.83)								
T_2M_1	2	.58	(.85)							
T_3M_1	3	.17	.14	(.74)						
T_1M_2	4	.75	.45	.23	(.93)					
T_2M_2	5	.72	.74	.16	.65	(.91)				
T_2M_2	6	.09	.06	.68	.25	.08	(.85)			
T_1M_3	7	.58	.53	.14	.62	.68	.09	(.63)		
T_2M_3	8	.47	.74	.10	.40	.69	.07	.58	(.74)	
T_3M_3	9	.22	.18	.63	.34	.22	.56	.30	.23	(.82)

(a) Do an MDS of this data matrix and check the configuration for possible correspondences to the trait and the method facet, respectively. Try both 2D and 3D solutions.

(b) What can you conclude about the relative weight of trait and method in these data?

(c) Characterize the roles of facets T and M in this MDS configuration.

(d) Compare the roles of facets T and M to the roles that T and M play in Exercise 1.6.

Exercise 5.2 Consider the data matrix below based on a representative survey in the U.S.A. It shows the intercorrelations of items asking about satisfaction with different aspects of one's life. According to Levy (1976), one can classify these items by the following mapping sentence. The extent of satisfaction of respondent x with the $\{a_1$ = state of, a_2 = resources for$\}$ his or her activities in area of life $\{b_1$ = education, b_2 = economy, b_3 = residence, b_4 = spare time, b_5 = family, b_6 = health, b_7 = work, b_8 = general$\}$ $\rightarrow$ {very positive ... very negative} satisfaction with life.

Item	A	B	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1 City as place to live	2	3														
2 Neighborhood	2	3	54													
3 Housing	2	3	44	49												
4 Life in the U.S.	2	3	33	28	29											
5 Amount of educat.	2	1	19	18	23	12										
6 Useful education	2	1	14	14	19	15	54									
7 Job	1	7	22	21	26	23	25	24								
8 Spare time	1	4	22	19	27	23	26	23	33							
9 Health	2	9	05	00	06	06	18	17	13	21						
10 Standard of living	1	2	33	32	45	24	32	24	35	37	17					
11 Savings, investmt.	2	2	25	23	29	19	28	20	27	32	17	59				
12 Friendships	2	4	24	19	23	21	16	17	25	40	09	25	24			
13 Marriage	2	5	14	13	21	13	09	12	25	30	12	25	23	21		
14 Family life	1	5	24	19	23	21	18	18	27	40	14	32	25	31	48	
15 Life in general	1	8	28	23	30	24	28	24	34	50	26	45	36	32	38	50

(a) According to Levy facets A and B establish a radex in a 2D MDS representation of these data. Verify.

(b) Characterize the roles of facets A and B in the MDS space.

(c) What item lies at the origin of the radex? Can you give a substantive explanation of why this makes sense?

(d) Items that lie more at the center of the radex are more similar to each other. What does that mean in this particular context?

Exercise 5.3 Consider the data matrix below. It shows the correlations for 12 intelligence tasks from the Wechsler test. The coefficients below the main diagonal are based on 2200 U.S. children; the coefficients above the main diagonal come from 1097 Israeli children. Following Guttman and Levy (1991), the tasks can be described by the following mapping sentence. The correctness of the response of testee x to a task that requires {I = inference, A = application, L = learning} of an objective *rule* through {o = oral, m =

manual manipulation, p = paper and pencil} *expression* → {high ... low} correctness.

Item	Rule	Exp	1	2	3	4	5	6	7	8	9	10	11	12
1 Information	A	o		51	52	58	46	36	40	38	42	34	31	30
2 Similarities	I	o	62		42	58	49	31	36	41	41	35	29	25
3 Arithmetic	A	o	54	47		44	36	43	34	33	44	33	33	32
4 Vocabulary	A	o	69	67	52		60	35	41	44	41	37	31	27
5 Comprehension	I	o	55	59	44	66		24	38	40	38	36	30	30
6 Digit span	L	o	36	34	45	38	26		28	28	32	23	29	26
7 Picture completion	A	o	40	46	34	43	41	21		45	47	45	25	31
8 Picture arrangement	A	m	42	41	30	44	40	22	40		45	48	28	35
9 Block design	A	m	48	50	46	48	44	31	52	46		57	32	39
10 Object assembly	A	m	40	41	29	39	37	21	48	42	60		27	40
11 Coding	L	p	28	28	32	32	26	29	19	25	33	24		23
12 Mazes	L	p	27	28	27	27	29	22	34	32	44	37	21	

(a) Do an MDS analysis of both the U.S. and the Israeli correlation matrices.

(b) Check whether the facets *rule* and *expression* allow you to structure ("explain") the MDS configurations.

(c) Characterize the roles these facets play in the MDS spaces.

(d) Which tasks are more central ones in terms of the spatial regions? Discuss in substantive terms what it means that "the closer an intelligence item is to being a 'rule inference', the weaker its affinity is to a single kind of material" (Shye, Elizur, & Hoffman, 1994)[p. 112]. ("Material" here corresponds to what Guttman calls "expression".)

Exercise 5.4 Consider the MDS configuration in Figure 5.5. Its interpretation is based on regions induced by some of the facets exhibited in Table 5.2. A special case of a region is a cluster. Clusters may emerge "out of substance" when one partitions an MDS space by facets defined for the entities represented by the points. However, clusters sometimes are also used in the MDS context in a purely exploratory way to help interpret MDS solutions. For that purpose, the proximities are subjected to a hierarchical cluster analysis, and the emerging cluster hierarchy is superimposed onto the MDS plane by expressing each cluster as a convex hull around the points that belong to the cluster. With hierarchical clusters, this often leads to *families* of such hulls that look like altitude or contour lines on a geographic map. We now use this approach on the data on which Figure 5.5 is based. These data are shown in the table below.

No.	Work Value	1	2	3	4	5	6	7	8	9	10	11	12	13
1	Interesting													
2	Independence	.44												
3	Responsibility	.61	.58											
4	Meaningful work	.49	.48	.53										
5	Advancement	.32	.44	.33	.39									
6	Recognition	.39	.34	.41	.47	.38								
7	Help others	.38	.35	.41	.45	.27	.65							
8	Contribute society	.36	.29	.44	.43	.16	.49	.64						
9	Contact people	.21	.10	.22	.21	.16	.29	.35	.45					
10	Security	.28	.18	.30	.39	.15	.36	.37	.49	.61				
11	Income	.37	.32	.36	.46	.21	.33	.45	.45	.43	.68			
12	Spare time	.32	.29	.35	.34	.23	.56	.49	.44	.40	.47	.49		
13	Working cond.	.50	.37	.39	.40	.30	.45	.44	.35	.26	.37	.37	.60	

(a) Do a hierarchical cluster analysis on the work values correlations. Plot the resulting clusters as nested "altitude" lines onto the MDS plane for the same data.

(b) Compare the cluster structure to the regions in Figure 5.5. Discuss where they agree and where they differ.

(c) Cluster analysis is sometimes used to check whether the clusters that one sees in an MDS solution are but scaling artifacts. Green & Rao write: "As a supplementary step, the ... data ... were submitted to ... [a] clustering program ... the program was employed to determine how well the low-dimensional scaling solutions preserved the original relationships in the input data" (Green & Rao, 1972, p. 33). Discuss what they mean by that statement.

(d) Superimpose hierarchical clusters onto the similarity of nations data in Table 1.3.

(e) Test out different clustering criteria (in particular, single linkage and average linkage) and check how they differ in clustering the points of Figure 1.5. Discuss why they differ.

Exercise 5.5 Facets are often superimposed by the substantive researcher on a theoretical basis. The facets, then, are typically not obtrusive ones, and many alternative facetizations are possible using different theories. Yet, facets can also be obtrusive features of the entities. That is true, for example, for the items in factorial surveys ("vignettes") or for stimuli within a factorial design. In these cases, the objects possess a certain facet profile by construction. It is also true for the following matrix which shows rank-order correlations of favorite leisure activities for groups defined by gender, race, and self-defined social class (Shinew, Floyd, McGuire, & Noe, 1995).

No.	Group	1	2	3	4	5	6	7	8
1	Lower-class black women	–							
2	Middle-class black women	.71	–						
3	Lower-class black men	.54	.54	–					
4	Middle-class black men	.35	.45	.61	–				
5	Lower-class white women	.23	.52	.17	.55	–			
6	Middle-class white women	.29	.66	.20	.52	.77	–		
7	Lower-class white men	.20	.33	.51	.87	.54	.41	–	
8	Middle-class white men	.11	.07	.25	.81	.51	.26	.26	–

(a) Represent these data in an MDS plane.

(b) Partition the space by the facets gender, race, and class, respectively.

(c) Discuss the resulting regions. Which facets show up in simple regions; which facets do not? What do you conclude about the leisure activities of these groups?

6
How to Obtain Proximities

Proximities are either collected by directly judging the (dis-)similarity of pairs of objects, or they are derived from score or attribute vectors associated with each of these objects. Direct proximities typically result from similarity ratings on object pairs, from rankings, or from card-sorting tasks. Another method, called the anchor stimulus method, leads to conditional proximities that have a restricted comparability and require special MDS procedures. Derived proximities are, in practice, most often correlations of item scores over individuals. Because there is so much work involved in building a complete proximity matrix, it is important to know about the performance of incomplete proximity matrices (with missing data) in MDS. It turns out that MDS is quite robust against randomly distributed missing data. MDS is also robust when used with coarse proximities, for example, dichotomous proximities.

6.1 Types of Proximities

MDS procedures assume that proximities are given. How one collects these proximities is a problem that is largely external to the MDS procedures discussed in this book.[1] However, because proximities are obviously needed,

[1]Some authors (e.g., Müller, 1984) approach MDS axiomatically. They formulate relational systems that, if satisfied, guarantee the existence of certain forms of MDS representations. Ideally, these axioms can be directly assessed empirically by asking the

and because the way these proximities are generated has implications for the choice of an MDS model, we devote some space to this topic.

In the previous chapters, we encountered different forms of proximities. For example, the proximities in Table 2.1 were distances generated by direct measurement on an atlas. In all other cases, the proximities were but distance *estimates* related to distances by some MDS model. The color similarity data in Table 4.1 were collected by averaging similarity ratings (0 = no similarity, ..., 4 = identical) for all different pairs of colors over all subjects. The Morse code proximities in Table 4.2 were obtained by computing the relative frequencies of "same" and "different" judgments for all pairs of Morse codes over different subjects. The data in Table 4.4 that indicate the similarity of facial expressions are based on scaling dissimilarity assessments for all pairs of faces over all subjects by the method of successive intervals.

These examples all involve some form of *direct* (dis-)similarity assessment for its object pairs, be it ratings on a scale from "no similarity" to "identical", judgments of "same" or "different", or orderings of object pairs on a similarity scale.

In practice, such direct approaches are rather atypical. Proximities usually are not based on direct similarity judgments, but rather are indices *derived* from other information. The most prominent ones are correlation coefficients, such as the product-moment correlations in Table 5.1 that assess the similarity of intelligence test items.

6.2 Collecting Direct Proximities

Direct proximities arise from directly assessing a binary relation of similarity or dissimilarity among the objects.[2] There are many possible ways to collect such data. The most obvious method is to ask respondents for a similarity judgment.

Some Varieties of Collecting Direct Proximities

The most popular method for collecting direct proximities is to *rate* the object pairs with respect to their overall similarity or dissimilarity. Krantz

---

subjects for simple judgments, such as partitioning every subset of at least three stimuli into two groups of relatively similar stimuli. In such an approach, the data collection is intimately related to the axiomatization of the MDS model.

[2]In order to keep the discussion uncluttered, we skip the case of dominance data in this section. Dominance data assess which object in a pair of objects dominates the other one in some sense, such as, for example, preference. They are treated later when discussing unfolding models in Part III.

and Tversky (1975), for example, wanted proximities for pairs of rectangles. They read the following instruction to their subjects (p. 14).

> *In this experiment we will show you pairs of rectangles and we'll ask you to mark an X in the appropriate cell on the scale from 1 to 20 [answer booklet was before subject] according to the degree of dissimilarity between rectangles.*
>
> *For example: if the rectangles are almost identical, that is, the dissimilarity between them is very small, mark X in a low-numbered cell. In the same fashion, for all intermediate levels of dissimilarity between the rectangles, mark X in an intermediate-numbered cell.*
>
> *We are interested in your subjective impression of degree of dissimilarity. Different people are likely to have different impressions. Hence, there are no correct or incorrect answers. Simply look at the rectangles for a short time, and mark X in the cell whose number appears to correspond to the degree of dissimilarity between the rectangles.*

This method of gathering proximities is called *pairwise comparison*. The subject rates every pair of objects on a dissimilarity scale.

Instead of ratings, market researchers often use some method of *ranking* the object pairs in terms of their overall similarity. For that purpose, each object pair is typically presented on a card. The subject is then asked to sort these cards so that the most similar object pair is on top of the card stack and the most dissimilar one at the bottom.

A complete ranking often may be too demanding a task or too time-consuming. Indeed, respondents often have difficulty ranking nonextreme objects. Thus, the "intermediate" ranks may be unreliable. It therefore makes sense to soften the ranking procedure as follows. The respondent is asked first to sort the cards into two stacks (not necessarily of equal size) one containing "similar" pairs and the other containing "dissimilar" pairs. For each stack, this sorting can be repeated until the respondent feels that it becomes too difficult to further partition a given stack into similar and dissimilar objects. The stack with the most similar objects is then scored as 1, the stack containing the next most similar objects as 2, and so on. The object pairs are given as proximities the score of the stack to which they belong. This usually leads to a weak rank-order (i.e., one containing ties), but that is no problem for MDS.

In *Q-sort* techniques (Stephenson, 1953), the respondents are asked to sort the cards with the object pairs into the categories of a scale that ranges, for example, from "very similar" to "not similar at all". The sorting must be done so that the stack on each scale category contains a preassigned number of cards. Typically, these numbers are chosen such that the card stacks are approximately normally distributed over the scale, with few cards

at the extremes and many cards in the middle. Computer programs exist that support this type of data collection.

Free sorting, in contrast, imposes a minimum number of constraints onto the respondents. They are simply asked to sort the cards onto different stacks so that cards showing object pairs that appear similar in some sense are in the same stack. The number of stacks is not specified. It can range from just one stack for all cards to the case where each stack contains only one card. To pairs of objects that are on the same stack, we assign a dissimilarity of 0, and for pairs of objects on different stacks, a 1 (see below, Section 6.5 on co-occurrence data). The advantage of this method is that the subject's task is not demanding, even for a large number of objects, and subjects report to enjoy the task.

Another technique for collecting direct proximities is the *anchor stimulus method*. Given n objects, one object is picked as a fixed comparison A, and the subject is asked to judge the similarity of all other $n-1$ objects to A. Each and every object serves, in turn, as an anchor. This leads to n sets with $n-1$ proximities each. The proximities resulting from the anchor stimulus method are *conditional* ones. Two proximities resulting from the anchor stimulus method have a meaningful relation only if they have the anchor stimulus as a common element. Thus, for example, the proximity for A and X and the proximity for A and Y can be compared because they share the anchor stimulus A. However, comparing the proximity for A and X with the proximity for B and Y (with A and B anchor stimuli) does not make sense, because the anchor stimuli are different. Hence, such data require particular MDS methods, with weaker loss functions that only assess, point after point, how well the distances of each anchor point to all other points represent the respective proximities. The relations of distance pairs that involve four different points are irrelevant.

Conditional data have the advantage that less data have to be ranked at the same time. Instead of ranking $\binom{n}{2}$ different pairs of objects, the anchor method only needs to rank $n-1$ pairs of objects at one time. The task of conditional ranking relative to fixed anchors is easier and yields more reliable data. These data, however, require more judgments altogether and are less comparable.

A systematic comparison among several methods for collecting direct proximities was done by Bijmolt and Wedel (1995). They found that free sorting and pairwise comparisons rate positively with respondents whereas collecting conditional data was considered to be boring and fatiguing. In terms of the data quality and the quality of the MDS solution, pairwise comparisons ranked best followed by free sorting.

On Ordering Object Pairs for Collecting Direct Proximities

The perceived similarity of two objects may depend on the order in which they are presented. For example, we note in Table 4.2 that the Morse code

signal for I is more frequently confused with a subsequent A (64%) than A is with a subsequent I (46%). Tversky (1977) gives another example: it seems likely that North Korea is assessed as similar to Red China, but unlikely that someone feels that Red China is similar to North Korea. Other order effects may arise if certain objects are presented relatively often in a given section of the data collection. For example, if the Morse code for A appears in the first 20 comparisons, it is most likely to have some anchoring effect.

Position effects can be reduced by randomly picking which of the objects of a pair will be in first position. This method avoids that a given object is always first or second in those pairs where it appears. *Timing effects* can be balanced by picking a random order for the object pairs.

An alternative approach is to balance position and timing effects by explicit planning. Ross (1934) developed a method for that purpose. It generally should be superior to the random method if the number of objects is relatively small. A computer program for Ross ordering was written by Cohen and Davison (1973).

Planned Incomplete Data Designs

One of the more obvious obstacles for doing an MDS analysis is that one needs many proximities, which are expensive to collect. The cheapest way to reduce the labor involved in data collection is to replace data by assumptions. Two assumptions are typical in MDS applications. First, it is taken for granted that the proximities are essentially symmetric. This obviates the need to collect both p_{ij} and p_{ji}. Second, the proximity of an object to itself, p_{ii}, is also not assessed empirically, because it seems even more justified to consider this information trivial: the dissimilarity of an object to itself is assumed to be essentially zero. For an MDS program, it is sufficient to have the proximities for one half-matrix.

However, even with a half-matrix, one needs to assess $\binom{n}{2} = n(n-1)/2$ proximities. The quantity $\binom{n}{2}$ grows rapidly with n. For example, for $n = 10$ one needs to collect 45 proximities, whereas for $n = 20$ one needs 190 proximities. Few subjects would be willing or able to rank 190 pairs of objects with respect to their global similarity. Hence, the need for incomplete data collection becomes obvious. Some structured incomplete designs are displayed in Table 6.1 (after Spence, 1983).

How should one plan an incomplete data design? A good solution is to *randomly* eliminate a certain proportion of cells in the proximity matrix and define them as missing data. Spence and Domoney (1974) studied this question in detail. They computed the distances in a given MDS space with dimensionality t, and then took these distances as input to MDS in order to see how well they would be reconstructed by MDS in t dimensions under a variety of conditions. One of these conditions was to add random error to the distances. Another one was to define some of the proximities as

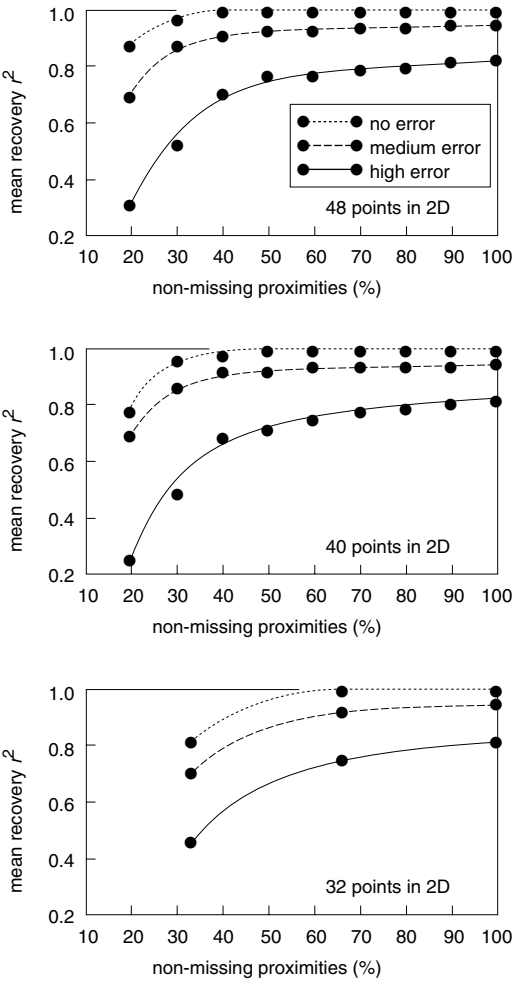


FIGURE 6.1. Recovery of MDS distances (Y-axis) among 48, 40, and 32 points, respectively, under different error levels (upper curves = no error, lower curves = high error) and percentages of nonmissing data (X-axis) (after Spence & Domoney, 1974).

TABLE 6.1. Examples of some incomplete designs (after Spence, 1983). A 0 indicates absence of a proximity, a 1 presence of the proximity.

(a) Cyclic design

	1	2	3	4	5	6	7	8
1	-							
2	1	-						
3	0	1	-					
4	0	0	1	-				
5	1	0	0	1	-			
6	0	1	0	0	1	-		
7	1	0	1	0	0	1	-	
8	1	1	0	1	0	0	1	-

(b) Random design

	1	2	3	4	5	6	7	8
1	-							
2	1	-						
3	0	1	-					
4	0	1	0	-				
5	1	0	1	0	-			
6	1	1	0	0	1	-		
7	0	0	1	0	1	0	-	
8	1	0	1	1	0	0	1	-

(c) Block design

	1	2	3	4	5	6	7	8
1	-							
2	1	-						
3	1	1	-					
4	1	1	1	-				
5	0	0	1	1	-			
6	0	0	1	1	1	-		
7	0	0	0	1	1	1	-	
8	0	0	0	1	1	1	1	-

missing data. It was found that the MDS-reconstructed distances remain highly correlated ($r^2 = .95$) with the original distances if one-third of the proximities are randomly eliminated (i.e., defined as missing data) and the error component in the proximities is about 15%. For high error (30%), r^2 is still .75, which compares well with $r^2 = .83$ for complete data. A significantly greater loss is incurred if two-thirds of the data are missing. However, if the error level is low, excellent recovery is possible even with 80% (!) missing data, given that we scale in the "true" dimensionality t, and given that the number of points is high relative to the dimensionality of the MDS space (see Figure 6.1, upper panels, curves for "no" and "medium" error).

Graef and Spence (1979) showed, moreover, that MDS configurations are poorly recovered if the proximities for the largest distances are missing, whereas missing data for intermediate or short distances are not that crucial. Hence, a missing data design could be improved by making sure that missing data are rare among the proximities for the most dissimilar objects.

These simulation studies show that robust MDS is possible even with many missing data. The user is well advised, nevertheless, to make sure that the missing cells do not form clusters in the proximity matrix.

One should keep in mind, however, that the above simulation results rest on some conditions (many points, reasonable error in the data, known "true" dimensionality, etc.) which are, in practice, often rather difficult to assess. It may be easiest to determine the error level of the data. For direct proximities, it could be estimated by replicating the data collection for some subjects; for correlations, one could consider statistical confidence intervals. Other conditions, however, are less easily diagnosed. For example, the very notion of "true" dimensionality remains obscure in most applications, except in rare cases such as, for example, perceptual studies in a psychophysical context (see Chapter 17). This makes it impossible to come up with a simple answer to the question of how many missing data can be accommodated in MDS.

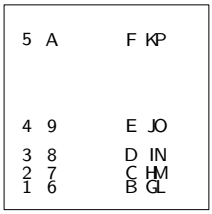


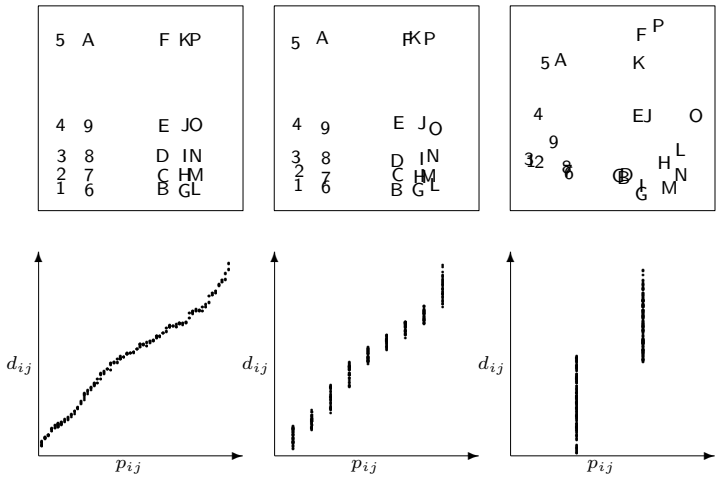
FIGURE 6.2. Synthetic configuration (after Green & Wind, 1973).

Collecting Coarse Data

Another possibility to make the task of collecting proximities simpler in case of direct proximities is to ask the respondents for simpler judgments. One extreme case is the "same" and "different" judgments on the Morse codes (see Chapter 4). Rothkopf (1957) aggregated these judgments over respondents and then analyzed the confusion probabilities as proximities. But is aggregation necessary? Would it make sense to do an MDS on the same–different data of a single individual? At first sight, such data seem "too coarse," but are they?

Green and Wind (1973) report a simulation study that throws some light on this question. They measure the distances of a 2D MDS configuration consisting of 25 points (Figure 6.2). These distances are classified into a small set of intervals. The same ranking number is substituted for all distances within the same interval. The resulting "degraded" distances are taken as proximities in MDS. Using the primary approach to ties (see Sections 3.1, p. 40, and 9.4), it is found that degrading distances into nine ranking numbers still allows one to recover almost perfectly the original configuration (Figure 6.3, Panel b). Even under the most extreme degradation, where the distances are mapped into only two ranking numbers, the original configuration is roughly recovered. One can conclude, therefore, that data that only represent the true distances in terms of distance groupings or blocks can be sufficient for recovering an underlying MDS configuration.

Of course, the granularity of the data may also be too fine in the sense that the data are not reliable to the same extent. For example, in the case of the above 21-point similarity scale employed by Krantz and Tversky (1975), one may well question that the respondents are able to make such fine-grained distinctions. If they are not, then they may not use all of the 21 categories; or if they do, their ratings may not be very reliable. One should not expect that persons are able to reliably distinguish more than 7 ± 2 categories (Miller, 1956). Confronting the individual with a 21-point similarity scale may then actually make his or her task unreasonably difficult.



a. 58 classes b. 9 classes c. 2 classes

FIGURE 6.3. Ordinal MDS representations of distances derived from Fig. 6.2, with Shepard diagrams (after Green & Wind, 1973): (a) uses distances as proximities; (b) uses distances degraded to nine values; (c) uses distances degraded to two values.

Again, there is no rule by which the issue of an optimal granularity could be decided in general. The issue lies outside of MDS, but it is comforting to know that even coarse data allow one to do an MDS analysis. What is important is the reliability of the data.

6.3 Deriving Proximities by Aggregating over Other Measures

Derived proximities are typically correlations or distances computed for a pair of variables, X and Y. A common way to organize the various coefficients available in this context is to consider the scale levels of X and Y. However, in the following, we do not intend to give an encyclopedic overview, but rather present some of the coefficients found most often in the MDS literature. We also discuss a few of the more exotic cases, because they help us to show some of the considerations involved in choosing a proper proximity measure. The obvious scale-level issues are largely ignored.

Correlations over Individuals

Probably the most common case of derived proximities is the one illustrated by the item intercorrelations in Table 5.1. The correlation between item X and item Y is computed over N individuals; that is,

$$r = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{(\sum_{i=1}^{N}(x_i - \bar{x})^2)^{1/2}(\sum_{i=1}^{N}(y_i - \bar{y})^2)^{1/2}},$$

where $\bar{x}$ (resp. $\bar{y}$) is the average over all x_is (resp. y_is). A correlation expresses the extent to which the individuals' responses to two items tend to have a similar pattern of relatively high and low scores.

Correlation coefficients exist for assessing different types of trends. The Pearson correlation measures the extent to which two items are linearly related. Substituting ranks for the raw data yields a rank-linear coefficient, Spearman's ρ. It assesses the monotonic relationship of two items. An alternative that responds more smoothly to small changes of the data is the μ_2 coefficient (Guttman, 1985). It is often used in combination with ordinal MDS (see, e.g., Levy & Guttman, 1975; Elizur et al., 1991; Shye, 1985) because (weak) monotonic coefficients are obviously more consistent with ordinal MDS than linear ones. The formula for μ_2 is

$$\mu_2 = \frac{\sum_{i=1}^{N}\sum_{j=1}^{N}(x_i - x_j)(y_i - y_j)}{\sum_{i=1}^{N}\sum_{j=1}^{N}|x_i - x_j||y_i - y_j|}.$$

The relationship of μ_2 to the usual product-moment coefficient r becomes most transparent if we express r as

$$r = \frac{\sum_{i=1}^{N}\sum_{j=1}^{N}(x_i - x_j)(y_i - y_j)}{\left(\sum_{i=1}^{N}\sum_{j=1}^{N}(x_i - x_j)^2\right)^{1/2}\left(\sum_{i=1}^{N}\sum_{j=1}^{N}(y_i - y_j)^2\right)^{1/2}}$$

(Daniels, 1944). One notes that the denominator of r is never smaller than the denominator of μ_2, which follows from the Cauchy–Schwarz inequality for nonnegative arguments: $\sum_k a_k b_k \leq (\sum_k a_k^2)^{1/2}(\sum_k b_k^2)^{1/2}$. Hence, $|\mu_2| \geq |r|$. One obtains $\mu_2 = r$ exactly if X and Y are linearly related (Staufenbiel, 1987).

Proximities from Attribute Profiles

Correlations typically are computed over individuals; that is, the data in the typical person × variables data matrix are correlated over the rows to yield the intercorrelations of the variables.

Assume now that we want to assess the perceived similarities among a number of cars. One way of doing this is to ask N respondents to assess each of the cars with respect to, say, its attractiveness. Proximities could then be

computed by correlating over the respondents' scores. One notes, however, that this approach completely hinges on the criterion of attractiveness. We may get more meaningful proximities if we do not rely that much on just one criterion but rather on a large selection of attributes on which cars are differentiated. Thus, we could ask the respondents to scale each car with respect to several criteria such as performance, economy, luxury, and so on. (In order to avoid redundancy, one could first factor-analyze these attributes and replace them by supposedly independent criteria or factors.) This would yield a person × cars × attributes matrix. The similarities of cars would then be derived as some function of how similar these cars are over the various attributes.

One possibility is to correlate the attribute profiles of the cars, either for each person in turn (yielding one similarity matrix per person) or over all attributes and all persons (yielding but one global similarity matrix).

An alternative is to measure dissimilarity by computing distances among attribute vectors. Assume, for example, that $\mathbf{X}$ is a cars × attributes matrix that contains average attribute assessments of N persons for each car on m attributes. For example, an element of $\mathbf{X}$ could be the average of the subjective prestige ratings that N persons gave car i. A "simple" distance of any two cars, i and j, in this m-dimensional attribute space is the *city-block distance*,

$$d_{ij}^{(1)}(\mathbf{X}) = \sum_{a=1}^{m} |x_{ia} - x_{ja}|,$$

where i and j are two objects of interest, and x_{ia} and x_{ja} are the scores of these objects on attribute a. Other distances (e.g., the Euclidean distance) are also conceivable but probably less attractive for deriving proximities because they all involve some kind of weighting of the intraattribute differences $x_{ia} - x_{ja}$. For example, in the Euclidean distance,

$$d_{ij}^{(2)}(\mathbf{X}) = \left(\sum_{a=1}^{m} (x_{ia} - x_{ja})^2 \right)^{1/2},$$

the difference terms $x_{ia} - x_{ja}$ are weighted quadratically into the distance function.

An overview of popular proximity measures is given in Table 6.2. To see how the coefficients are related to the attributes, Figure 6.4 shows various isoproximity contours for the case where point x_j is fixed at position $(1, 2)$ and point x_i takes on different positions in the attribute space. The contour lines show the sets of positions where x_i has the same proximity to x_j. In the case of the Euclidean distance, these contours correspond to the usual notion of circles. In the case of the city-block distance, these circles look unfamiliar (see Section 17.2 for more details). On the other hand, the composition rule by which the differences of i and j are aggregated into

TABLE 6.2. Summary of measures of proximities derived from attribute data. The symbol δ_{ij} denotes a dissimilarity and s_{ij} a similarity.

Measure		Formula
P1	Euclidean distance	$\delta_{ij} = \left(\sum_{a=1}^{m}(x_{ia} - x_{ja})^2\right)^{1/2}$
P2	City-block distance	$\delta_{ij} = \sum_{a=1}^{m} \lvert x_{ia} - x_{ja}\rvert$
P3	Dominance distance	$\delta_{ij} = \max_{a=1}^{m} \lvert x_{ia} - x_{ja}\rvert$
P4	Minkowski distance	$\delta_{ij} = \left(\sum_{a=1}^{m}(x_{ia} - x_{ja})^p\right)^{1/p}$ with $p \geq 1$
P5	Canberra distance	$\delta_{ij} = \sum_{a=1}^{m} \frac{\lvert x_{ia} - x_{ja}\rvert}{\lvert x_{ia} + x_{ja}\rvert}$
P6	Bray–Curtis distance	$\delta_{ij} = \frac{\sum_{a=1}^{m} \lvert x_{ia} - x_{ja}\rvert}{\sum_{a=1}^{m}(x_{ia} + x_{ja})}$
P7	Chord distance	$\delta_{ij} = \left(\sum_{a=1}^{m}(x_{ia}^{1/2} - x_{ja}^{1/2})\right)^{1/2}$
P8	Angular separation, congruence coefficient	$s_{ij} = \frac{\sum_{a=1}^{m} x_{ia}x_{ja}}{\left(\sum_{a=1}^{m} x_{ia}^2\right)^{1/2}\left(\sum_{a=1}^{m} x_{ja}^2\right)^{1/2}}$
P9	Correlation	$s_{ij} = \frac{\sum_{a=1}^{m}(x_{ia} - \bar{x}_i)(x_{ja} - \bar{x}_j)}{\left(\sum_{a=1}^{m}(x_{ia} - \bar{x}_i)^2\right)^{1/2}\left(\sum_{a=1}^{m}(x_{ja} - \bar{x}_j)^2\right)^{1/2}}$
P10	Monotonicity coefficient μ_2	$s_{ij} = \frac{\sum_{i=1}^{N}\sum_{j=1}^{N}(x_i - x_j)(y_i - y_j)}{\sum_{i=1}^{N}\sum_{j=1}^{N}\lvert x_i - x_j\rvert\lvert y_i - y_j\rvert}$

the overall distance is extremely simple: the distance is just the sum of the intradimensional differences. The dominance distance, in contrast, is completely determined by just one intradimensional difference of i and j, the largest one. Note that P1 to P3 are special cases of the Minkowski distance P4: $p = 1$ gives the city-block distance P2, $p = 2$ the Euclidean distance P1, and $p = \infty$ the dominance distance P3. The distances P1 to P4 combine dimensional differences directly. Consequently, if the dimensions are attributes measured on different scales, the attributes with the largest variance will dominate the distance measure. Therefore, it is usually better to standardize the attributes so that their variances become equal by converting each attribute to z-scores. Alternatively, each attribute can be divided by another measure for dispersion such as the range (the difference of maximum and minimum).

The proximity measures P5 to P10 all have some provision for controlling the dispersion either for each variable separately or for all variables simultaneously. The Canberra distance corrects the absolute difference along each

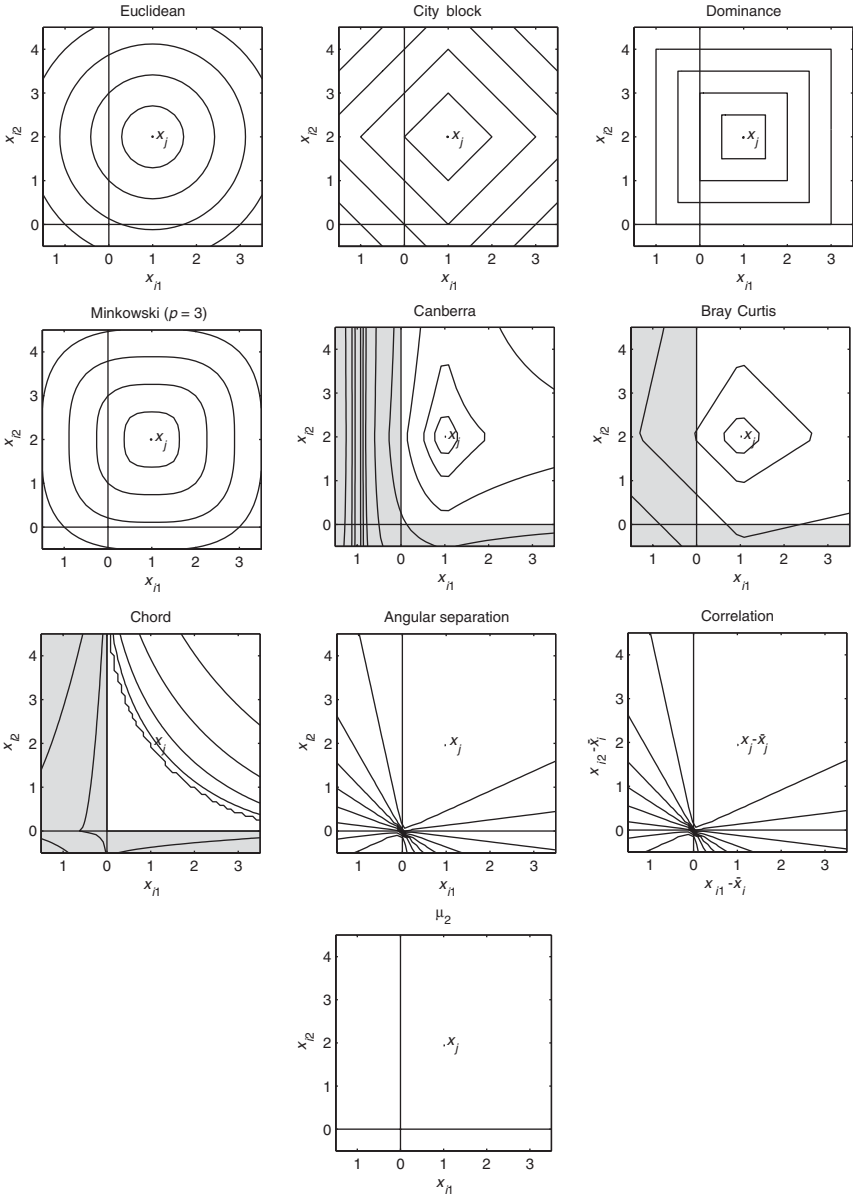


FIGURE 6.4. Contour plots for the different proximity measures defined in Table 6.2, setting $x_j = (1, 2)$. Contour lines close to x_j have low values, whereas further away they have higher values. For the contour lines of the Minkowski distance, the value $p = 3$ was used. Note that μ_2 has no contour lines in this grossly simplified example, because all values are exactly one. The grey areas correspond to negative x_{i1} or x_{i2} which are usually excluded for these measures.

dimension for the size of the coordinates along the axis. In addition, if negative values of x_{ia} are allowed, then δ_{ij} reaches an asymptote of infinity, in Figure 6.4 at $x_{i1} = -1$. Therefore, the Canberra distance is best used when all x_{ia} are positive. The Bray–Curtis distance is often used in ecology and corrects the sum of absolute differences along the axes by the sum of all coordinates over which the differences are taken. Again, this measure seems most useful for nonnegative x_{ia}. In this case, the Bray–Curtis distance corrects large absolute differences when the coordinates are large, too. The chord distance requires positive x_{ia}. Usually, x_{ia} equals the frequency, so that it is positive by nature. Note that for drawing the contour lines in Figure 6.4 for the chord distance, the absolute values of x_{ia} were used. The angular separation is a similarity index between -1 and 1 because it computes the cosine of the angle between the lines from the origin to x_i and the origin to x_j. The contour lines for the correlation are exactly the same as for the angular separation because we changed the axes to $x_{ia} - \bar{x}_i$. Note that both the correlation and μ_2 are best used when the number of dimensions m is reasonably large, certainly larger than in the simplified case of $m = 2$ in Figure 6.4. For μ_2 this simplification leads to $\mu_2 = 1$ for all x_{ia} which explains why there are no contour lines for μ_2. Thus, μ_2 is only meaningful if $m \geq 3$.

Another type of distance function often used in the literature is to count the number of *common elements* in the data profiles and subtract this sum from the total number of attributes on which observations were made. This distance function could be employed, for example, where attributes are coded as either present or absent. An example from archaeology is data on sites where certain artifacts such as pottery, jewelry, bones, and the like, are or are not found (e.g., Kendall, 1971). Sites are considered similar if they share many artifacts.

Restle (1959) suggested this distance function in order to model the perception of similarity: conceiving stimuli X and Y in terms of "feature" sets (i.e., as collections of the things associated with them), we have the distance $d_{XY} = m(X \cup Y) - m(X \cap Y)$, where m is a measure function.[3] Hence, the dissimilarity of X and Y, d_{XY}, is the number of their noncommon features, $m(Y - X) + m(X - Y)$.

When collecting object $\times$ attribute data sets in real life, some attributes may be binary; others may be numerical. The general similarity measure of Gower (1971) is particularly suited for this situation. Let s_{ija} be the similarity between objects i and j on variable a. For binary attributes, we assume that only values $x_{ia} = 0$ and $x_{ia} = 1$ occur. In this case, $s_{ija} = 1$ if x_{ia} and x_{ja} fall in the same category and $s_{ija} = 0$ if they do not. If

[3] A simple measure function is, for example, the number of elements in the set. $X \cup Y$ is the union of X and Y; $X \cap Y$ is the intersection of X and Y; $X - Y$ is the set consisting of the elements of X that are not elements of Y.

the attribute is numerical, then we compute $s_{ija} = 1 - |x_{ia} - x_{ja}|/r_k$ with r_k being the range of attribute a. This definition ensures again that $0 \leq s_{ija} \leq 1$ for all combinations of i, j, and a. The general similarity measure can be defined by

$$s_{ij} = \frac{\sum_a w_{ija} s_{ija}}{\sum_a w_{ija}},$$

where the w_{ija} are given nonnegative weights. Usually w_{ija} is set to one for all i, j, and a. However, if either x_{ia} or x_{ja} is missing (or both), then w_{ija} should be set to zero so that the missing values do not influence the similarity. Here, too, $0 \leq s_{ij} \leq 1$ so that dissimilarities can be obtained by taking $1 - s_{ij}$. However, Gower (1971) suggests to use $(1 - s_{ij})^{1/2}$ as it can be shown that these values can be perfectly represented in a Euclidean space of high dimensionality.

6.4 Proximities from Converting Other Measures

Derived proximities are not always computed by aggregating over individuals or from aggregating over attribute vectors associated with the objects of interest. They can also be generated by appropriate conversion of given scale values for the objects. The conversion is arrived at by theoretical considerations.

Consider the following case. Glushko (1975) was interested in the "goodness" of patterns. He constructed a set of different dot patterns and printed each possible pair on a separate card. Twenty subjects were then asked to indicate which pattern in each pair was the "better" one. The pattern judged better in a pair received a score of 1, the other one a 0. These scores were summed over the subjects, and a dissimilarity measure was constructed on the basis of the following logic. "Since dissimilar goodness between two patterns is implied by frequent choice of either one over the other, the absolute value of the difference between the observed and the expected frequency of a goodness preference represents the dissimilarity of the pattern of goodness of the two patterns ..." (Glushko, 1975, p. 159). Because there were 20 subjects, the expected (random) preference value is 10 for each pair. Hence, proximities were derived by subtracting 10 from each summation score and taking its absolute value.

A similar conversion is the following. Thurstone (1927), Coombs (1967), and Borg (1988) asked N students to indicate in a pair-comparison design which of two offenses (such as murder, arson, or theft) was more "serious." Scoring the more serious one as 1 and the other one as 0, adding these scores over individuals, and dividing by N, one obtains a matrix of dominance probabilities (P_{ij}). These data typically are scaled by Thurstone's Law of Comparative Judgment model, which relates the P_{ij}s to scale values by a cumulative normal density function. However, one can also convert

the probabilities into dissimilarities δ_{ij} and then use ordinal MDS. [Ordinal MDS does not assume a particular (monotonic) model function and, thus, leaves it to the data to exhibit the exact shape of the transformation function.] The conversion formula is $\delta_{ij} = |P_{ij} - 0.5|$.

Tobler and Wineburg (1971) report another interesting proximity, a measure of social interaction between towns or "places" called the *gravity model*: $I_{ij} = kP_iP_j/d_{ij}^2$, where "$I_{ij}$ is the interaction between places i and j; k is a constant, depending on the phenomena; P_i is the population of i; P_j is the population of j; and d_{ij} is the distance between places i and j. Distance may be in hours, dollars, or kilometers; populations may be in income, numbers of people, numbers of telephones, and so on; and the interaction may be in numbers of letters exchanged, number of marriages, similarity of artifacts or cultural traits, and so on." (p. 2). With measures for I_{ij}, P_i, and P_j, the gravity model can be used to solve for the distance d_{ij}. Tobler and Wineburg (1971) report an application from archaeology. Cuneiform tables from Assyria were the database. The number of occurrences of a town's name on these tables was taken as P_i, the number of co-occurrences on the tables as a measure of I_{ij}. The resulting distance estimates were taken as input for a 2D ordinal MDS in an effort to find the (largely unknown) geographical map of these towns.

6.5 Proximities from Co-Occurrence Data

An interesting type of proximities is co-occurrence data. Coxon and Jones (1978), for example, studied the categories that people use to classify occupations. Their subjects were asked to sort a set of 32 occupational titles (such as barman, statistician, and actor) into as many or as few groups as they wished. The result of this sorting can be expressed, for each subject, as a 32×32 *incidence matrix*, with an entry of 1 wherever its row and columns entries are sorted into the same group, and 0 elsewhere. The incidence matrix can be considered a proximity matrix of dichotomous (same–different) data.[4]

Are such co-occurrence data direct proximities? The answer depends on how one wants to define "direct". In the above study on occupation titles, the criterion of similarity should have been obvious to the respondents. Hence, by sorting the occupation titles into groups, they were directly ex-

[4]Burton (1975) further suggests some forms of weighting such as replacing 1 by the number of objects in the category to which a given object pair belongs, or by replacing 1 by the inverse of this number. The former is supposed to emphasize gross discrimination, the latter fine discrimination. Such weightings of global and local discriminations are, however, better introduced as part of the MDS modeling criteria, rather than building them into the data.

pressing their notions of pairwise similarity relations for these stimuli. But consider another case.

England and Ruiz-Quintanilla (1994) asked respondents to check those characteristics in a list that would define work for them. The characteristics were "if it is not pleasant", "if it is physically strenuous", "if you have to do it", and so on. The co-occurrences of these characteristics were defined as the characteristics' proximities. Its seems that this definition is more an interpretation of the researcher, because the respondents never directly assessed the similarity of the characteristics in the context of work, but their relevance with respect to the notion of work. Hence, these proximities seem somewhat more derived than the former ones, which shows that the direct-derived distinction denotes more a continuum than a dichotomy.

Studies that use co-occurrence data typically aggregate incidence matrices over individuals. The most natural way to do this is simply to add these matrices so that the aggregate proximity matrix contains in its cells the frequencies with which two objects were sorted into the same group.

However, it is well worth the effort to consider whether it would be better if these raw frequencies were normed. Let X and Y be two items of interest. An item X can be empirically present or absent, denoted as $X = 1$ and $X = 0$, respectively. With X and Y, there are four possible present–absent combinations. Let $z = f(X, Y)$ be the frequency of an event (X, Y). In particular, let $a = f(1, 1)$ be the frequency of the event where both X and Y are present. Similarly, $b = f(1, 0)$, $c = f(0, 1)$, and $d = f(0, 0)$ (see also Table 6.3). Gower (1985) distinguishes a variety of possible similarity coefficients, all of which vary between 0 and 1. One possibility is

$$s_2 = a/(a + b + c + d),$$

the frequency of events where both X and Y occur relative to the total frequency of all present–absent combinations of X and Y. Another possibility is

$$s_3 = a/(a + b + c),$$

the proportion of events where both X and Y occur, given at least one of them occurs (*Jaccard similarity measure*).

To see the consequences of choosing s_2 or s_3, consider the following example. Bilsky, Borg, and Wetzels (1994) studied forms of conflict tactics among family members, ranging from calm debates over throwing things to physical violence inflicting injuries to other persons. A survey asked the respondents to indicate which forms of behavior had occurred among members of their families in the last five years. If co-occurrence of behavior forms is assessed by s_3, MDS yields a one-dimensional solution where the different behavior forms are simply arrayed in terms of their aggressiveness, with a major gap between behaviors that involve shouting, throwing things, and the like, and those that involve any form of physical violence. Using s_2

TABLE 6.3. Types of combinations of two events X and Y, together with their frequencies (cells entries).

	$X=1$	$X=0$	Total
$Y=1$	a	b	$a+b$
$Y=0$	c	d	$c+d$
Total	$a+c$	$b+d$	$a+b+c+d$

coefficients, however, has the effect that the behaviors that involve physical violence drop dramatically in similarity because they are so rare, that is, because d is so great. This essentially wipes out the clear scale obtained for s_3 proximities.

There are amazingly many ways to combine the four frequencies $a, \ldots, d$ into an overall proximity measure for each pair of objects (see, e.g., Gower, 1985; Gower & Legendre, 1986; Cox & Cox, 1994). However, most of these proximities make sense only in highly specific contexts, so that it serves no purpose to discuss all of them here. It may suffice to consider just one further proximity, the *simple matching coefficient*,

$$s_4 = (a+d)/(a+b+c+d),$$

which counts both co-occurrence and co-nonoccurrence as indices of similarity. In the case of the forms of violent behaviors, s_4 would bring up the question of whether rare forms of behavior, in particular, should be considered very similar simply because of their high rate of co-nonoccurrence. More details about many of the possible binary coefficients and their scalability in MDS can be found in Gower and Legendre (1986).

An small overview of the most frequently used co-occurence measures is presented in Table 6.4, together with the range for each of these indexes. It is easy to convert these similarity measures into dissimilarities by computing $\delta_{ij} = 1 - s_k$, for $k = 2, \ldots, 6$.

6.6 Choosing a Particular Proximity

The availability of so many varieties of proximities seems to make life confusing for the user. Which proximity should be chosen? An answer to this question depends on many considerations, but is typically not that difficult.

An important decision criterion is usually the practical feasibility of a particular data collection method. Consider surveys, for example, where respondents are asked by questionnaires about their attitudes towards various political issues. It would be inconceivable to replace the usual item-by-item ratings by a task where the respondent has to compare the $n(n-1)/2$ pairs of items, because this is simply too time consuming. Moreover, it would be difficult to explain to the respondents what exactly they are supposed to

TABLE 6.4. Overview of some popular co-occurrence measures.

Measure			Bounds of s_k
s_2		$s_2 = \dfrac{a}{a+b+c+d}$	$0 \leq s_2 \leq 1$
s_3	Jacard similarity measure	$s_3 = \dfrac{a}{a+b+c}$	$0 \leq s_3 \leq 1$
s_4	Simple matching coefficient	$s_4 = \dfrac{a+d}{a+b+c+d}$	$0 \leq s_4 \leq 1$
s_5	Hamman	$s_5 = \dfrac{(a+d)-(b+c)}{a+b+c+d}$	$-1 \leq s_5 \leq 1$
s_6	Yule	$s_6 = \dfrac{ad-bc}{ad+bc}$	$-1 \leq s_5 \leq 1$

do in such a task, that is, in which sense they are supposed to compare the items.

Another case is the proximity of intelligence test items, assessed above in terms of how similarly the testees perform on the items. Here, it remains unclear how direct proximities could be defined at all without changing the research question. Assume that we would ask test psychologists to evaluate directly the global similarity of the test items. Such a question, obviously, studies the perception of test psychologists and not the structure of the test item performance of testees.

Direct proximities are more a task for laboratory studies on perceptual structures than, for example, for survey studies. Most of the examples discussed earlier (e.g., Morse code confusions, color similarities) belong to this category. The card-sorting procedures often used by market researchers is another example.

In the context of such research questions, direct proximities typically are collected to explain how they are generated. If the subjects were asked to first assess the objects of interest on scales invented by the researcher, the proximities would be based on these scales, not on criteria freely chosen by subjects themselves. In the facial expressions study by Engen et al. (1958), the direct proximities were, therefore, collected along with ratings on certain dimensions in order to check whether the structure of the former could be explained by the latter (see Section 4.3).

So, the question of what proximity to choose typically is decided to a large extent by the research question and its context. However, this is more true for direct proximities. If one decides to derive proximities, one has a less substantive foothold for choosing a particular measure.

Deriving proximities requires one to decide, first of all, if one wants a correlation coefficient or a distance measure on the observations on two

variables, X and Y. The former assesses the similarity of X and Y in terms of their "profiles", the latter the (dis-)similarity in terms of their element-by-element differences. That is, if $X = 2 \cdot Y$, for example, then $r_{XY} = 1$, but the distance of X and Y is not zero. On the other hand, if the distance of X and Y is zero, then $r_{XY} = 1$ always.

However, the choice between these measures is not that important in practice. The reason is that if proximities are computed by aggregating over attribute scales, it usually makes sense to first standardize the different attribute scales rather than using raw scores. In this case, Euclidean distances are related to Pearson's r by a monotonic function. This can be seen as follows. Assume that we have two variables, X and Y, that are both standardized so that their means are zero and their sum-of-squares is equal to 1. As a result, $r_{XY} = \sum_i x_i y_i$. Then, the Euclidean distance between X and Y is

$$\begin{aligned} d_{XY} &= \left(\sum_{i=1}^{N} (x_i - y_i)^2 \right)^{1/2} \\ &= \left(\sum_{i=1}^{N} x_i^2 + \sum_{i=1}^{N} y_i^2 - 2 \sum_{i=1}^{N} x_i y_i \right)^{1/2} \\ &= (2 - 2r_{XY})^{1/2} . \end{aligned} \tag{6.1}$$

Hence, when using ordinal MDS, it becomes irrelevant which proximity is used, because both yield (inversely) *equivalent* rank-orders.

City-block distances, moreover, are typically highly correlated with Euclidean distances, so that they, too, are monotonically closely related to r in practice. It is also true that Pearson correlations and monotonic correlations such as ρ or μ_2 are highly correlated if the relationship of the items is not extremely nonlinear. Moreover, the *structural* information contained in a matrix of proximities is very robust against variations in the individual proximity coefficients. For that reason, Pearson rs are often chosen in practice rather than the formally more attractive μ_2s. In summary, then, the user need not worry that much about the particular choice for computing proximities from score vectors: the usual measures, such as r or the Euclidean distance, are most often quite appropriate in an MDS context.

6.7 Exercises

Exercise 6.1 Consider the matrix of dominance probabilities P_{ij} below (Borg, 1988). It shows the relative frequencies with which a group of male students judged that the crime/offense in row i is more serious than the crime/offense in column j. Thurstone (1927) and Coombs (1967) report similar data. They analyze them with the Law-of-Comparative-Judgment

model. This model maps dominance probabilities P_{ij} into scale value differences $x_i - x_j$ by the inverse normal distribution ogive; that is, $N^{-1}(P_{ij}) = x_i - x_j$, where N^{-1} denotes the function that maps probabilities into z-scores.

Item	1	2	3	4	5	6	7	8	9	10
1 Abortion	.50	.65	.32	.30	.42	.12	.20	.36	.45	.49
2 Adultery	.35	.50	.20	.19	.25	.02	.11	.28	.31	.33
3 Arson	.68	.80	.50	.41	.62	.13	.22	.45	.61	.67
4 Assault/battery	.70	.81	.59	.50	.67	.16	.29	.51	.70	.72
5 Burglary	.58	.75	.38	.33	.50	.09	.14	.40	.58	.58
6 Homicide	.88	.98	.87	.84	.91	.50	.59	.74	.87	.90
7 Rape	.80	.89	.78	.71	.86	.41	.50	.63	.83	.83
8 Seduction	.64	.72	.55	.49	.60	.26	.37	.50	.66	.69
9 Theft	.55	.69	.39	.30	.42	.13	.17	.34	.50	.53
10 Receiving stolen goods	.51	.67	.33	.28	.42	.10	.17	.31	.47	.50

(a) Davison (1983) suggests that these data can be modeled by ordinal MDS. In fact, he claims that one can solve for a more general class of models called Fechner models. All Fechner models require that (1) $P_{ij} = 0.5 \mapsto d_{ij} = |x_i - x_j| = 0$ and that (2) $d_{ij} = |x_i - x_j|$ grows strictly monotonically as a function of $\delta_{ij} = |P_{ij} - 0.5|$.] Thurstone's model is but one particular Fechner model that relies on the normal function. Use ordinal MDS to find one-dimensional scales for the crime/offense data sets without relying on any particular monotonic function.

(b) Study the empirical relation of dominance probabilities to the corresponding scale differences (=signed distances) and discuss whether the normal mapping function used in the Law-of-Comparative-Judgment model is empirically supported here.

(c) Repeat the MDS analysis with five different random starting configurations. Compare the five solutions. What does your finding imply for unidimensional scaling?

Exercise 6.2 Consider Table A.1 on page 545 in Appendix A that compares several properties of MDS programs. Drop the rows "max. number of objects", "min. number of objects", and "max. dimensionality" as computer constraints that have little to do with the substance of the different MDS programs described here. Turn the remaining matrix into a 1–0 incidence matrix. Then compute at least three different types of similarity coefficients for the set of MDS programs and discuss your choices. Finally, scale these similarity data in 2D MDS spaces and compare the resulting solutions.

Exercise 6.3 Consider Table 1.5 used in Exercise 1.7.

(a) Derive proximity matrices for the row entries by using (1) monotone correlations, (2) city-block distances, and (3) Euclidean distances.

(b) For each set of proximities, find 2D ordinal and interval MDS solutions.

(c) Compare the solutions: How similar are they? Give reasons for their relative similarities or dissimilarities.

Exercise 6.4 Pick ten countries from at least four different continents. For these countries, derive a proximity matrix by card sorting, where you are the respondent yourself. Discuss which problems you encountered in sorting the cards. Replicate the experiment with a different respondent and compare the outcomes.

Exercise 6.5 Consider the data matrix below. It shows the results of a free sorting experiment reported by Dunn-Rankin (1983, p.47). Fifteen persons clustered 11 words that all begin with the letter "a". The entries in the data matrix are cluster numbers.

Person	A	Ad-mits	Aged	Al-most	Aim-ing	And	As	At	Areas	Army	Away
1	1	2	3	2	4	3	1	1	5	6	6
2	1	2	3	2	2	1	1	1	3	2	2
3	1	2	1	2	2	3	1	1	3	3	3
4	1	2	3	4	4	1	5	6	7	8	8
5	1	2	3	4	4	1	5	6	7	8	8
6	1	2	3	3	4	5	1	6	7	8	8
7	1	2	3	2	2	3	1	1	2	2	2
8	1	2	2	4	5	6	7	7	8	9	9
9	1	2	3	2	4	5	1	6	4	4	4
10	1	2	3	4	5	2	1	1	2	6	6
11	1	2	3	2	4	1	1	1	3	5	5
12	1	2	3	4	2	3	1	1	3	3	3
13	1	2	3	2	4	5	1	1	6	7	5
14	1	2	3	2	4	5	1	1	6	7	7
15	1	2	3	2	2	3	1	1	3	2	3

(a) Do the persons sort the words into the same number of clusters? Which person makes the finest distinctions and which person the coarsest?

(b) Compute a matrix of reasonable proximity indices for the 11 words. Analyze the similarities by MDS.

(c) Compute proximity indices for the 15 persons and analyze the indices by MDS. (Hint: Make a list of all pairs of words. If person x throws word i and word j into the same cluster, assign a proximity score of 1. Else, score 0.)

Exercise 6.6 Merkle (1981) studied the frequencies with which product x is bought together with product y, as measured by the sales registry in a set of German clothing stores. He reports the following co-occurrence data.

Product	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1 Expensive suit	28													
2 Expensive trad. shirt	18	68												
3 Expensive tie	13	17	0											
4 Cheap tie	6	8	0	13										
5 Imported shirt	10	25	10	0	20									
6 Medium-priced shirt	2	23	0	15	3	0								
7 Cheap suit	2	27	6	22	6	13	26							
8 Cheap shirt	3	9	10	25	25	13	26	57						
9 Cheap knitwear	17	46	22	24	5	109	222	275	487					
10 Stylish shirt	10	0	4	8	1	48	146	88	57	109				
11 Colored socks	24	21	3	18	7	281	197	117	178	8	273			
12 Jeans	25	10	23	9	5	43	167	146	46	8	46	110		
13 Modern jacket	1	14	3	33	0	3	12	21	87	42	15	14	508	
14 Modern pants	0	0	0	46	16	0	18	67	12	19	20	24	45	88

(a) Discuss how the values on the main diagonal of this matrix are to be interpreted. Are the data similarities or dissimilarities?

(b) Some products are bought more often than others. Discuss what effects this has if one were to submit these data to an MDS analysis. In which ways would the result be influenced by buying frequencies? Where in the MDS plot would a product move that people tend to buy very often?

(c) Merkle (1981) suggests normalizing these data for their different basic frequencies by using Yule's coefficient of colligation: $Y_{xy} = [\sqrt{ad} - \sqrt{bc}]/[\sqrt{ad} + \sqrt{bc}]$, where a denotes the frequency of all sales that contain both x and y, d is the frequency of sales that contain neither x nor y, b are sales of x but not y, and c are sales of y without x. Compute the Y_{xy} coefficients for co-sales of products 1 through 4.

(d) The coefficient Y_{xy} is not easily interpretable. If, however, one skips the square roots in the formula for Y, another coefficient due to Yule results, called Q (see s_6 in Table 6.4). What does Q assess? How can this be expressed in words?

(e) Assume we wanted to do an ordinal MDS of the normalized data. Would it make much, or any, difference whether we use Y or Q?

(f) Describe alternatives for normalizing the data matrix for different overall sales frequencies of the different products.

(g) Compute MDS solutions for these data, both raw and normalized. Discuss the solutions in terms of what features of these products determine whether they tend to be bought jointly or not.

(h) Make a proposal of how the different values of the main diagonal could be represented graphically in the MDS plots.

Part II

MDS Models and Solving MDS Problems

7
Matrix Algebra for MDS

In this chapter, we build a basis for a more technical understanding of MDS. Matrices are of particular importance here. They bring together, in one single mathematical object, such notions as a whole configuration of points, all of the distances among the points of this configuration, or a complete set of proximities. Mathematicians developed a sophisticated algebra for matrices that allows one to derive, for example, how a configuration that represents a matrix of distances can be computed, or how the distances among all points can be derived from a configuration. Most of these operations can be written in just a few lines, in very compact notation, which helps tremendously to see what is going on. The reader does not have to know everything in this chapter to read on in this book. It suffices to know the main concepts and theorems and then later come back to this chapter when necessary. Proofs in this chapter are meant to better familiarize the reader with the various notions. One may opt to skip the proofs and accept the respective theorems, as is common practice in mathematics ("It can be shown that ...").

7.1 Elementary Matrix Operations

The term *matrix* denotes a *rectangular* array of objects such as numbers. A data matrix, for example, may consist of measurement scores for n persons on m items. Usually, a data matrix is written so that the persons form the

rows and the items the columns. A simple example is the 3×2 matrix $\mathbf{A}$,

$$\mathbf{A} = \begin{bmatrix} 1 & 2 \\ 3 & 5 \\ 4 & 7 \end{bmatrix}.$$

It is customary to denote a matrix by a boldface capital letter (such as $\mathbf{A}$) and to use brackets around its elements. Sometimes, it is useful to characterize a matrix by a typical element, which is written as $\mathbf{A} = (a_{ij})$. The symbol a_{ij} denotes the element in row i and column j of $\mathbf{A}$.

The number of rows, n, and the number of columns, m, of a matrix define its *order*. The matrix $\mathbf{A}$ above has order 3 by 2. Occasionally, an $n \times m$ matrix $\mathbf{A}$ is denoted by $\mathbf{A}_{n \times m}$ to show its order explicitly. If $n = m$, we have a *square* or *quadratic* matrix.

Matrices where $m = 1$ or $n = 1$ are also called *vectors*. They are denoted by small boldface letters such as $\mathbf{a}$. A $k \times 1$ vector is called a *column vector* and a $1 \times k$ vector a *row vector*. For example, the matrix $\mathbf{A}$ above consists of two column vectors and three row vectors. A row vector typically is written with a prime sign (e.g., as $\mathbf{a}'$), a column vector without the prime. The third row vector of $\mathbf{A}$ is $\mathbf{r}_3' = [4\ 7]$, and the first column vector of $\mathbf{A}$ is

$$\mathbf{c}_1 = \begin{bmatrix} 1 \\ 3 \\ 4 \end{bmatrix}.$$

A row vector $\mathbf{x}'$ is also written as the m-tuple $(x_1, x_2, \ldots, x_m)$. Thus $\mathbf{x}' = (3, 2, 5)$ is equivalent to $\mathbf{x}' = [3\ 2\ 5]$.

Transposing, Adding, and Multiplying Matrices

One obtains the row vector $\mathbf{x}'$ from the column vector $\mathbf{x}$ simply by writing it as a row vector, an operation called *transposition*. More generally, one can also form the *transpose* of a matrix $\mathbf{A}$ by writing its rows as columns. The transpose is written as $\mathbf{A}'$. For the matrix $\mathbf{A}$ from above, we get

$$\mathbf{A}' = \begin{bmatrix} 1 & 3 & 4 \\ 2 & 5 & 7 \end{bmatrix}.$$

Obviously, $(\mathbf{A}')' = \mathbf{A}$.

A matrix $\mathbf{A}$ is *symmetric* if $a_{ij} = a_{ji}$ for all i, j, or, equivalently, if $\mathbf{A}' = \mathbf{A}$. In data analysis, symmetric matrices (e.g., correlation matrices) are commonplace.

Elementary matrix algebra is concerned with when and how matrices and vectors can be added, subtracted, multiplied, and divided. Addition and subtraction are easily defined. Matrices are added (subtracted) by simply adding (subtracting) corresponding elements. Expressed formally for addition, $\mathbf{A} + \mathbf{B} = (a_{ij} + b_{ij}) = (c_{ij}) = \mathbf{C}$. Table 7.1 gives an example.

Addition (subtraction) is possible only if $\mathbf{A}$ and $\mathbf{B}$ have the same order, because otherwise there are elements in one matrix for which there are no corresponding elements in the other matrix. Table 7.1 also shows how the product of a matrix with a simple number (called a *scalar* in matrix algebra) is defined: $k\mathbf{A} = (k \cdot a_{ij})$; that is, each element of $\mathbf{A}$ is multiplied by the scalar k. (Note that the scalar k differs from the 1×1 matrix $\mathbf{M} = [k]$ whose only element is k.)

In contrast to multiplying a matrix by a scalar, multiplying a matrix by another matrix is quite complicated. It would seem natural to define $\mathbf{AB} = \mathbf{C}$ as $[a_{ij} \cdot b_{ij}] = [c_{ij}]$, but this type of product plays only a very minor role in most applications of matrix algebra. Rather, what is known as "the" product of two matrices is defined as $\mathbf{AB} = [\sum_k a_{ik} \cdot b_{kj}] = [c_{ij}]$. The formula says that each element of row i in $\mathbf{A}$ is to be multiplied by the corresponding element of column j in $\mathbf{B}$, and then all of these products are to be summed to yield c_{ij}. Table 7.1 shows a concrete case, where c_{21} results from $1 \cdot 2 + 2 \cdot 0 + 0 \cdot 1 = 2$.

Matrix multiplication requires that $\mathbf{A}$ has as many columns as $\mathbf{B}$ has rows; that is, if $\mathbf{A}$'s order is $n \times r$, then $\mathbf{B}$'s order must be $r \times m$. $\mathbf{C}$'s order is given directly by canceling r; that is, $\mathbf{C}$ is of order $n \times m$. Hence, if $\mathbf{A}$ and $\mathbf{B}$ are both square matrices, then both $\mathbf{AB}$ and $\mathbf{BA}$ exist and are of the same order. It is important, however, to realize that $\mathbf{AB} \neq \mathbf{BA}$ in general, as can be checked easily by trying some cases. We therefore use special terms and speak of *premultiplication* or *multiplication from the left* and *postmultiplication* or *multiplication from the right*. For example, in $\mathbf{AB}$, $\mathbf{A}$ premultiplies $\mathbf{B}$ or, expressed differently, $\mathbf{B}$ multiplies $\mathbf{A}$ from the right.

Matrix Inverses

We now come to division. To begin, consider a real number k. If k is divided by k, then 1 results: $k/k = (k)(k^{-1}) = (k^{-1})(k) = 1$. The number 1 plays a special role in the multiplication of real numbers: it is the *neutral element* for multiplication, because $1 \cdot k = k \cdot 1 = k$, for all k. Similarly, the *inverse* of a matrix $\mathbf{A}$, $\mathbf{A}^{-1}$, should neutralize $\mathbf{A}$ in a product expression so that $\mathbf{A}^{-1}\mathbf{AB} = \mathbf{B}$ and $\mathbf{AA}^{-1}\mathbf{B} = \mathbf{B}$. But then both $\mathbf{A}^{-1}\mathbf{A}$ and $\mathbf{AA}^{-1}$ should be equal to a matrix that plays the role of the neutral element in matrix multiplication. This matrix is called the *identity matrix* and is denoted by $\mathbf{I}$. Because pre- and postmultiplying $\mathbf{A}$ by $\mathbf{A}^{-1}$ is possible only if both $\mathbf{A}$ and $\mathbf{A}^{-1}$ are square matrices, it follows that $\mathbf{I}$ is square, too. Furthermore, as could be checked by some numerical examples, $\mathbf{I}$ must consist of 0s everywhere, except for the main diagonal, which contains only

1s. For example, the 3×3 identity matrix is

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{7.1}$$

It is easily verified that, for any 3×3 matrix, $\mathbf{IA} = \mathbf{AI} = \mathbf{A}$, which shows that $\mathbf{I}$ is a neutral element in matrix multiplication.

As to the existence of $\mathbf{A}^{-1}$, we have already noted that $\mathbf{A}$ must be square. Moreover, $\mathbf{A}$ must have *full rank*. The rank r of an $n \times m$ matrix is the number of linearly independent rows or columns of this matrix. It cannot be greater than the number of rows or columns, whichever is less. That is, $r \leq \min(n, m)$. A set of rows (columns) is linearly independent if no row (column) is equal to a weighted sum of the other rows (columns). Whether this is true for all rows (columns) of a matrix is generally not easy to diagnose without doing some computations (see Section 7.4).[1]

For some special matrices it is easy to compute the inverse. One case is the *diagonal* matrix whose off-diagonal elements are all equal to zero; that is, $\mathbf{A}$ is diagonal if $a_{ij} = 0$ for all $i \neq j$. An example of a diagonal matrix is the matrix $\mathbf{I}$ in (7.1). One can check that if $\mathbf{A}$ is diagonal, then $\mathbf{A}^{-1}$ is also diagonal, with $1/a_{ii}$ as its diagonal elements. Obviously, $\mathbf{A}^{-1}$ exists only if $a_{ii} \neq 0$, for all i. If this is true and $\mathbf{A}$ is diagonal, then $\mathbf{A}$ has full rank.

A second type of matrix whose inverse is easily found is an $n \times n$ matrix $\mathbf{A}$ that satisfies $\mathbf{A}'\mathbf{A} = \mathbf{I}$. A matrix with that property is called *orthonormal*.[2] But if $\mathbf{A}'\mathbf{A} = \mathbf{I}$, then $\mathbf{A}' = \mathbf{A}^{-1}$ and, because $\mathbf{A}$ is square, we also have $\mathbf{A}\mathbf{A}^{-1} = \mathbf{A}\mathbf{A}' = \mathbf{I}$. Hence, a square matrix with orthonormal columns also has orthonormal rows. A special case of an orthonormal matrix is the identity matrix $\mathbf{I}$.

In Table 7.2, we list some properties of matrix addition and scalar multiplication of a matrix, and in Table 7.3 we summarize properties of matrix multiplications, transposes, and inverses.

---

[1] $\mathbf{A}^{-1}$ denotes, strictly speaking, "the" inverse or the *regular* inverse. There also exist specialized inverses that possess some but not all of the properties of the regular inverse. Examples are the "left" and the "right" inverses. They solve the equations $\mathbf{LA} = \mathbf{I}$ and $\mathbf{AR} = \mathbf{I}$, respectively, for $\mathbf{A}$-matrices that need not be quadratic. Yet, $\mathbf{L}$ and $\mathbf{R}$ require that $\mathbf{A}$ has full column rank or full row rank, respectively. There are even more general types of inverses that do not require such full-rank properties (see below, Section 7.7). Operating with a nonregular inverse on a given matrix always entails loss of information, so that the operation cannot be undone.

[2] Mathematicians typically speak of *orthogonal* matrices. For example, Strang (1976, p. 119) writes: "An orthogonal matrix is simply a *square matrix with orthonormal columns* ... Perhaps *orthonormal* matrix would have been a better name, but it is too late to change." Data analysts build on much less tradition and are perhaps allowed more freedom in their choice of terms.

TABLE 7.1. Examples of matrix addition, scalar multiplication, and multiplication.

$$\mathbf{A}+\mathbf{B} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$$

$$= \begin{bmatrix} a_{11}+b_{11} & a_{12}+b_{12} \\ a_{21}+b_{21} & a_{22}+b_{22} \end{bmatrix} = \mathbf{C}$$

$$\begin{bmatrix} 3 & 6 \\ 7 & 2 \end{bmatrix} + \begin{bmatrix} 1 & -6 \\ 4 & -3 \end{bmatrix} = \begin{bmatrix} 4 & 0 \\ 11 & -1 \end{bmatrix}$$

$$k\mathbf{A} = k \cdot \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = \begin{bmatrix} ka_{11} & ka_{12} \\ ka_{21} & ka_{22} \end{bmatrix}$$

$$2 \cdot \begin{bmatrix} 3 & 5 \\ 7 & 2 \end{bmatrix} = \begin{bmatrix} 6 & 10 \\ 14 & 4 \end{bmatrix}$$

$$\mathbf{AB} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ \mathbf{a}_{31} & \mathbf{a}_{32} & \mathbf{a}_{33} \end{bmatrix} \begin{bmatrix} b_{11} & \mathbf{b}_{12} \\ b_{21} & \mathbf{b}_{22} \\ b_{31} & \mathbf{b}_{32} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \\ c_{31} & \mathbf{c}_{32} \end{bmatrix} = \mathbf{C}$$

$$\begin{bmatrix} 3 & 0 & 2 \\ 1 & 2 & 0 \\ \mathbf{0} & \mathbf{0} & \mathbf{-1} \end{bmatrix} \begin{bmatrix} 2 & \mathbf{1} \\ 0 & \mathbf{1} \\ 1 & \mathbf{1} \end{bmatrix} = \begin{bmatrix} 8 & 5 \\ 2 & 3 \\ -1 & \mathbf{-1} \end{bmatrix}$$

TABLE 7.2. Some properties of matrix addition and scalar multiplication of matrices.

$\mathbf{A} = \mathbf{B}$	$a_{ij} = b_{ij}$ for all i, j
$\mathbf{A} + \mathbf{B} = \mathbf{C}$	$c_{ij} = a_{ij} + b_{ij}$ for all i, j
$\mathbf{A} + \mathbf{B} = \mathbf{B} + \mathbf{A}$	Commutative property
$(\mathbf{A} + \mathbf{B}) + \mathbf{C} = \mathbf{A} + (\mathbf{B} + \mathbf{C})$	Associative property
$c\mathbf{A}$	Has elements $c \cdot a_{ij}$ for all i, j
$c(k\mathbf{A}) = (ck)\mathbf{A} = (kc)\mathbf{A} = k(c\mathbf{A})$	Associative property
$c(\mathbf{A} + \mathbf{B}) = c\mathbf{A} + c\mathbf{B}$	Distributive property for matrices
$(c + k)\mathbf{A} = c\mathbf{A} + k\mathbf{A}$	Distributive property for scalars
$\mathbf{A} + \mathbf{0} = \mathbf{A}$	Adding a null matrix

TABLE 7.3. Some properties of matrix multiplication, transposes, and matrix inverses.

$\mathbf{A}_{n\times r}\mathbf{B}_{r\times m}$	=	$\mathbf{C}_{n\times m}$ if and only if $c_{ij} = \sum_{k=1}^{r} a_{ik}b_{kj}$
$(\mathbf{AB})\mathbf{C}$	=	$\mathbf{A}(\mathbf{BC})$
$\mathbf{AA}$	=	$\mathbf{A}^2$
$(\mathbf{A}+\mathbf{B})(\mathbf{C}+\mathbf{D})$	=	$\mathbf{A}(\mathbf{C}+\mathbf{D})+\mathbf{B}(\mathbf{C}+\mathbf{D})$
	=	$\mathbf{AC}+\mathbf{AD}+\mathbf{BC}+\mathbf{BD}$
$(\mathbf{A}')'$	=	$\mathbf{A}$
$(\mathbf{AB})'$	=	$\mathbf{B}'\mathbf{A}'$
$(\mathbf{ABC})'$	=	$\mathbf{C}'\mathbf{B}'\mathbf{A}'$
$(\mathbf{A}+\mathbf{B})'$	=	$\mathbf{A}'+\mathbf{B}'$
$\mathbf{IA}$	=	$\mathbf{A} = \mathbf{AI}$
$\mathbf{B}$	=	$\mathbf{A}^{-1}$ if and only if $\mathbf{BA} = \mathbf{I} = \mathbf{AB}$
$(\mathbf{A}^{-1})^{-1}$	=	$\mathbf{A}$
$(\mathbf{A}')^{-1}$	=	$(\mathbf{A}^{-1})'$
$(\mathbf{AB})^{-1}$	=	$\mathbf{B}^{-1}\mathbf{A}^{-1}$

7.2 Scalar Functions of Vectors and Matrices

One can take a matrix or a vector and assign to it, by some rule, a simple number. In mathematics, such a rule is called a function. There are infinitely many functions, and each of them serves a different purpose. Here we discuss some functions that are important in the MDS context.

Functions that have many arguments but only one value are frequently used in all fields of science. A familiar example is the product-moment correlation, which has two vector-valued arguments $\mathbf{x}$ and $\mathbf{y}$, and a value r that lies in the interval $[-1, +1]$. The correlation is closely related to the *scalar product* of two vectors. Given two real-valued vectors $\mathbf{x}$ and $\mathbf{y}$, both of the same order, their scalar product is

$$< \mathbf{x}, \mathbf{y} >= x_1y_1 + \cdots + x_ny_n.$$

One notes that this is computationally the same as $\mathbf{x}'\mathbf{y}$. The difference is that the vector product is algebraically a 1×1 matrix with element $< \mathbf{x}, \mathbf{y} >$ and not just a number. In an applied context, however, one does not run into problems by ignoring this distinction. Thus, for example,

$$< \mathbf{x}, \mathbf{y} >= \mathbf{x}'\mathbf{y} = \begin{bmatrix} 1 & 3 & 4 \end{bmatrix} \begin{bmatrix} 2 \\ 5 \\ 7 \end{bmatrix} = 45.$$

Scalar products arise naturally in matrix multiplication. In $\mathbf{A}'\mathbf{B} = \mathbf{C}$, each element c_{ij} of the product matrix $\mathbf{C}$ is the scalar product of the ith row vector of $\mathbf{A}$ and the jth column vector of $\mathbf{B}$.

Of particular importance is the case where $\mathbf{x}'\mathbf{y} = 0$. Vectors whose scalar product is zero are called *orthogonal*. For example, the vectors $(2, 0)$ and

$(0, 1)$ in the usual Euclidean plane are orthogonal. Geometrically, these two vectors correspond to points on the X- and Y-axes, respectively. If one connects these points with line segments to the origin, one notes that these lines are perpendicular, just like the coordinate axes with which they coincide. Perpendicularity of the lines that connect the points x and y with the origin is the geometric interpretation of orthogonality of two coordinate vectors $\mathbf{x}$ and $\mathbf{y}$.

Another example of a function with more than one argument is the distance between two points. Distances are closely related to the *norm* of a vector, a notion that captures the intuitive meaning of length,

$$\|\mathbf{x}\| = \sqrt{\mathbf{x}'\mathbf{x}} = (x_1^2 + \ldots + x_n^2)^{1/2}. \tag{7.2}$$

A whole family of norms arises by first substituting x_i^r for x_i^2 and replacing $1/2$ by $1/r$ and then choosing other positive numbers instead of $r = 2$. For $r = 1$, for example, we obtain the absolute norm $\|\mathbf{x}\|_1 = |x_1| + \ldots + |x_n|$. For a large r, the greatest absolute x_i dominates the norm, so that $\|\mathbf{x}\|_\infty = \max_i |x_i|$. The natural norm, however, is the *Euclidean norm*, where $r = 2$ as in the formula above. Without any special comments to the contrary, the term norm always refers to the Euclidean norm.

All norms satisfy four properties:

$$\begin{aligned}
\|\mathbf{x}\| &\geq 0 \text{ for } \mathbf{x} \neq \mathbf{0} \text{ and} \\
\|\mathbf{x}\| &= 0 \text{ precisely when } \mathbf{x} = \mathbf{0} \text{ (nonnegativity)}, \\
\|k\mathbf{x}\| &= |k|\|\mathbf{x}\|, \text{ for any scalar } k, \\
\|\mathbf{x} + \mathbf{y}\| &\leq \|\mathbf{x}\| + \|\mathbf{y}\| \text{ (triangle inequality)}.
\end{aligned}$$

The norm of a vector is used, for example, to *normalize* a given vector to unit length. If $\mathbf{x}$ is any real-valued vector, then $\mathbf{u} = (1/\|\mathbf{x}\|)\mathbf{x}$ is a *normal* or *unit* vector so that $\|\mathbf{u}\| = 1$.

Norms can be used to express the distance between two points in vector terms. Let $\mathbf{x}$ and $\mathbf{y}$ be the coordinate vectors of some points x and y. Then, $\|\mathbf{x} - \mathbf{y}\|$, the norm of the difference vector $\mathbf{x} - \mathbf{y}$, is equal to the Euclidean distance between x and y. This is easy to see by checking formula (3.3) for the Euclidean distance.

Norms are closely related to loss functions, as we will see. Here, the natural extension of vector norms to matrices is also helpful. The norm of a matrix $\mathbf{A}$ is simply the square root of its sum-of-squares. Thus, the function $\|\mathbf{A}\|$ is a familiar measure of $\mathbf{A}$.

Another matrix function often found in the context of optimization problems is the trace. The *trace* function of an $n \times n$ matrix $\mathbf{A}$ is defined as

$$\text{tr } \mathbf{A} = \sum_{i=1}^{n} a_{ii}, \tag{7.3}$$

TABLE 7.4. Some properties of the trace function.

(1)	tr $\mathbf{A} = \sum_{i=1}^{n} a_{ii}$	Definition of trace function
(2)	tr $\mathbf{A}$ = tr $\mathbf{A}'$	Invariance under transposing $\mathbf{A}$
(3)	tr $\mathbf{ABC}$ = tr $\mathbf{CAB}$ = tr $\mathbf{BCA}$	Invariance under "cyclic" permutation
(4)	tr $(\mathbf{A}'\mathbf{B})$ = tr $(\mathbf{A}'\mathbf{B})'$ = tr $\mathbf{B}'\mathbf{A}$ = tr $\mathbf{AB}'$	Combining properties (2) and (3)
(5)	tr $\mathbf{ab}' = \mathbf{a}'\mathbf{b}$	
(6)	tr $(\mathbf{A} + \mathbf{B})$ = tr $\mathbf{A}$ + tr $\mathbf{B}$	Summation rule

the sum of $\mathbf{A}$'s elements in the main diagonal. This function becomes particularly interesting when we are studying the difference of two corresponding matrices, such as, for example, two configurations $\mathbf{X}$ and $\mathbf{Y}$ whose points have a 1–1 correspondence. A common case is where $\mathbf{X}$ and $\mathbf{Y}$ are two MDS configurations for replicated data. The function tr $(\mathbf{X} - \mathbf{Y})(\mathbf{X} - \mathbf{Y})'$ assesses, then, the sum of squared differences of the coordinates of the corresponding points of $\mathbf{X}$ and $\mathbf{Y}$. This is considered in detail in Chapter 21.

Later on, we need some properties of matrix traces that are conveniently summarized together in Table 7.4. These properties are easy to verify by considering some simple numerical examples.

7.3 Computing Distances Using Matrix Algebra

An important concept in MDS is the distance between two points. Let $\mathbf{X}_{n\times m}$ be the matrix of coordinates of the points. Each row i of $\mathbf{X}$ gives the coordinates of point i on m dimensions, that is, $x_{i1}, x_{i2}, \ldots, x_{im}$. In MDS we are concerned with the distances among all n points. We can use the matrix algebra from the previous section to obtain a compact expression for computing the squared Euclidean distances between all points. The squared Euclidean distance is defined by

$$d_{ij}^2(\mathbf{X}) = d_{ij}^2 = \sum_{a=1}^{m}(x_{ia} - x_{ja})^2 = \sum_{a=1}^{m}(x_{ia}^2 + x_{ja}^2 - 2x_{ia}x_{ja}). \tag{7.4}$$

Suppose that $\mathbf{X}$ contains the coordinates of three points in two dimensions. Now the matrix of squared distances, denoted by $\mathbf{D}^{(2)}(\mathbf{X})$, is

$$\mathbf{D}^{(2)}(\mathbf{X}) = \begin{bmatrix} 0 & d_{12}^2 & d_{13}^2 \\ d_{12}^2 & 0 & d_{23}^2 \\ d_{13}^2 & d_{23}^2 & 0 \end{bmatrix} = \sum_{a=1}^{m} \begin{bmatrix} x_{1a}^2 & x_{1a}^2 & x_{1a}^2 \\ x_{2a}^2 & x_{2a}^2 & x_{2a}^2 \\ x_{3a}^2 & x_{3a}^2 & x_{3a}^2 \end{bmatrix}$$

$$+\sum_{a=1}^{m}\begin{bmatrix} x_{1a}^2 & x_{2a}^2 & x_{3a}^2 \\ x_{1a}^2 & x_{2a}^2 & x_{3a}^2 \\ x_{1a}^2 & x_{2a}^2 & x_{3a}^2 \end{bmatrix} - 2\sum_{a=1}^{m}\begin{bmatrix} x_{1a}x_{1a} & x_{1a}x_{2a} & x_{1a}x_{3a} \\ x_{2a}x_{1a} & x_{2a}x_{2a} & x_{2a}x_{3a} \\ x_{3a}x_{1a} & x_{3a}x_{2a} & x_{3a}x_{3a} \end{bmatrix}$$

$$= \mathbf{c1}' + \mathbf{1c}' - 2\sum_{a=1}^{m}\mathbf{x}_a\mathbf{x}_a' = \mathbf{c1}' + \mathbf{1c}' - 2\mathbf{XX}', \tag{7.5}$$

where $\mathbf{x}_a$ is column a of matrix $\mathbf{X}$, $\mathbf{1}$ is an $n \times 1$ vector of ones, and $\mathbf{c}$ is a vector that has elements $\sum_{a=1}^{m} x_{ia}^2$, the diagonal elements of $\mathbf{XX}'$. The matrix $\mathbf{B} = \mathbf{XX}'$ is called a *scalar product matrix*.

Suppose that

$$\mathbf{X} = \begin{bmatrix} 1 & 2 \\ 3 & 1 \\ 2 & 0 \end{bmatrix} = \begin{array}{c|cc} & \mathbf{x}_1 & \mathbf{x}_2 \\ p_1 & 1 & 2 \\ p_2 & 3 & 1 \\ p_3 & 2 & 0 \end{array} \tag{7.6}$$

is a coordinate matrix. Its rows show the coordinates of three points on dimensions 1 (the first column of $\mathbf{X}$) and 2 (the second column of $\mathbf{X}$), respectively, of Figure 7.1. The distances can be computed using (7.5). The first step is to compute the scalar product matrix $\mathbf{B} = \mathbf{XX}'$; that is,

$$\mathbf{XX}' = \begin{bmatrix} 1 & 2 \\ 3 & 1 \\ 2 & 0 \end{bmatrix}\begin{bmatrix} 1 & 3 & 2 \\ 2 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 5 & 5 & 2 \\ 5 & 10 & 6 \\ 2 & 6 & 4 \end{bmatrix} = \mathbf{B}. \tag{7.7}$$

The second step is to find $\mathbf{c}$. It can be verified that the diagonal elements of $\mathbf{XX}'$ are $\sum_{a=1}^{m} x_{ia}^2$, which are the elements of $\mathbf{c}$. Thus $\mathbf{c}' = (5, 10, 4)$. Inserting these results into (7.5) gives

$$\mathbf{D}^{(2)}(\mathbf{X}) = \begin{bmatrix} 0 & d_{12}^2 & d_{13}^2 \\ d_{12}^2 & 0 & d_{23}^2 \\ d_{13}^2 & d_{23}^2 & 0 \end{bmatrix} = \begin{bmatrix} 5 & 5 & 5 \\ 10 & 10 & 10 \\ 4 & 4 & 4 \end{bmatrix}$$
$$+\begin{bmatrix} 5 & 10 & 4 \\ 5 & 10 & 4 \\ 5 & 10 & 4 \end{bmatrix} - 2\begin{bmatrix} 5 & 5 & 2 \\ 5 & 10 & 6 \\ 2 & 6 & 4 \end{bmatrix} = \begin{bmatrix} 0 & 5 & 5 \\ 5 & 0 & 2 \\ 5 & 2 & 0 \end{bmatrix}.$$

Taking the square root of all elements gives the distance matrix

$$\mathbf{D}(\mathbf{X}) = \begin{bmatrix} 0 & \sqrt{5} & \sqrt{5} \\ \sqrt{5} & 0 & \sqrt{2} \\ \sqrt{5} & \sqrt{2} & 0 \end{bmatrix} \approx \begin{bmatrix} .000 & 2.236 & 2.236 \\ 2.236 & .000 & 1.414 \\ 2.236 & 1.414 & .000 \end{bmatrix}.$$

In Section 7.9, we show how we can solve the reverse problem, that is, how to find the coordinates $\mathbf{X}$ from a given scalar product matrix $\mathbf{B}$.

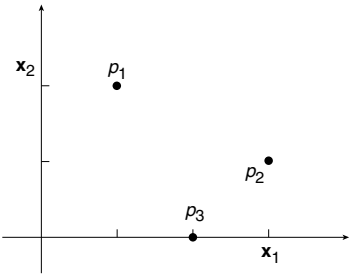


FIGURE 7.1. Geometrical representation of configuration in (7.6).

7.4 Eigendecompositions

Every $n \times n$ matrix $\mathbf{A}$ of real numbers can be decomposed into a product of several matrices. We now consider a particularly useful case, the *eigendecomposition*, which can be constructed for most matrices, but always for symmetric ones. Formally,

$$\mathbf{A} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}', \tag{7.8}$$

with $\mathbf{Q}$ orthonormal (i.e., $\mathbf{Q}'\mathbf{Q} = \mathbf{Q}\mathbf{Q}' = \mathbf{I}$) and $\mathbf{\Lambda}$ diagonal. Equation (7.8) is often written as a system of *eigenequations*

$$\mathbf{A}\mathbf{q}_i = \lambda_i \mathbf{q}_i, \text{ with } \mathbf{q}_i \neq \mathbf{0} \ \ (i = 1, \ldots, n). \tag{7.9}$$

These equations can also be written more compactly as

$$\mathbf{A}\mathbf{Q} = \mathbf{Q}\mathbf{\Lambda}. \tag{7.10}$$

The column vectors of $\mathbf{Q}$ are called the *eigenvectors* of $\mathbf{A}$. The λ_is in the diagonal of $\mathbf{\Lambda}$ are the *eigenvalues* of $\mathbf{A}$. It is customary to order the eigenvalues (and the corresponding eigenvectors) so that $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$. For example, for matrix

$$\mathbf{A} = \begin{bmatrix} 23 & 36 \\ 36 & 2 \end{bmatrix}$$

we get

$$\mathbf{Q} = \begin{bmatrix} 0.8 & -0.6 \\ 0.6 & 0.8 \end{bmatrix} \text{ and } \mathbf{\Lambda} = \begin{bmatrix} 50 & 0 \\ 0 & -25 \end{bmatrix}.$$

A slightly different view of eigendecompositions leads to the important *spectral decomposition* theorem. Consider again equation (7.8). We can think of the product $\mathbf{Q}\mathbf{\Lambda}\mathbf{Q}'$ as a product of two vectors: the row vector

consisting of the column vectors in the product $\mathbf{Q}\mathbf{\Lambda}$, and the column vector made up of the row vectors in $\mathbf{Q}'$,

$$\begin{aligned}\mathbf{A} &= \begin{bmatrix} \lambda_1\mathbf{q}_1 & \lambda_2\mathbf{q}_2 & \dots & \lambda_n\mathbf{q}_n \end{bmatrix} \begin{bmatrix} \mathbf{q}_1' \\ \mathbf{q}_2' \\ \vdots \\ \mathbf{q}_n' \end{bmatrix} \\ &= \lambda_1\mathbf{q}_1\mathbf{q}_1' + \lambda_2\mathbf{q}_2\mathbf{q}_2' + \cdots + \lambda_n\mathbf{q}_n\mathbf{q}_n'. \end{aligned} \tag{7.11}$$

The right-hand side of (7.11) says that $\mathbf{A}$ can be decomposed into a sum of matrices. To illustrate, consider again matrix $\mathbf{A}$ from above. Here, the decomposition is

$$\begin{aligned}\mathbf{A} &= 50 \begin{bmatrix} 0.8 \\ 0.6 \end{bmatrix} \begin{bmatrix} 0.8 & 0.6 \end{bmatrix} - 25 \begin{bmatrix} -0.6 \\ 0.8 \end{bmatrix} \begin{bmatrix} -0.6 & 0.8 \end{bmatrix} \\ &= \begin{bmatrix} 32 & 24 \\ 24 & 18 \end{bmatrix} - \begin{bmatrix} 9 & -12 \\ -12 & 16 \end{bmatrix} = \begin{bmatrix} 23 & 36 \\ 36 & 2 \end{bmatrix}. \end{aligned} \tag{7.12}$$

Some Properties of Spectral Decompositions

Eigenvalues and eigenvectors are important in practice, because they have numerous useful properties. Some of them are listed in the following. Also, some theorems are discussed that should help to better understand such decompositions.

(1) Not every $n \times n$ real matrix possesses an eigendecomposition over the real numbers, even if nonorthogonal eigenvectors are admitted. That is, some matrices can be spectrally decomposed only if one allows for complex eigenvalues and/or eigenvectors, which, in any case, complicates interpretations. An example is the matrix $\mathbf{A}$ in the following. Consider the eigenequation

$$\mathbf{A}\mathbf{q} = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \lambda \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}.$$

This says that $q_1 - q_2 = \lambda q_1$, so that $q_2 = q_1 - \lambda q_1$. Substituting this into the second equation, $q_1 + q_2 = \lambda q_2$, yields $q_1 = 0$, and back-substitution yields $q_2 = 0$. Thus, the only real vector that solves the eigenequation is the null vector $\mathbf{0}$. If one allows for complex numbers, then, for example, $\lambda_1 = 1 + i$ and $\mathbf{q}_1 = (i, 1)$, with $i^2 = -1$, solve the eigenequation $\mathbf{A}\mathbf{q}_1 = \lambda_1\mathbf{q}_1$.

(2) Eigenvectors are not unique. They can, for example, be multiplied by -1, because, if $\mathbf{A}\mathbf{q}_i = \lambda_i\mathbf{q}_i$, then also $\mathbf{A}(-1)\mathbf{q}_i = (-1)\mathbf{A}\mathbf{q}_i = \lambda_i(-1)\mathbf{q}_i = (-1)\lambda_i\mathbf{q}_i$. Therefore, reflections of the eigenvectors are admissible. One also notes that choosing $\mathbf{Q}$ such that $\mathbf{Q}\mathbf{Q}' = \mathbf{I}$ is an arbitrary (although useful) convention. Consider (7.8) and assume that we scale $\mathbf{\Lambda}$ by the factor 3. This is accomplished by replacing $\mathbf{\Lambda}$ in equation (7.8) by $\mathbf{\Lambda}^* = \mathbf{K}\mathbf{\Lambda}$, where

$\mathbf{K} = \text{diag}(3, 3, \ldots, 3)$, a diagonal matrix with all nonnull elements equal to 3. We note that $\mathbf{K}\mathbf{\Lambda}$ can be written as $\mathbf{K}^{1/2}\mathbf{\Lambda}\mathbf{K}^{1/2}$, where $\mathbf{K}^{1/2}$ is the same as raising the diagonal elements of $\mathbf{K}$ to the power $1/2$ because $\mathbf{K}$ is diagonal. Hence, we must replace $\mathbf{Q}$ in equation (7.8) by $\mathbf{Q}^* = \mathbf{Q}\mathbf{K}^{-1/2}$ to compensate for the scaling of the eigenvalues. Thus, $\mathbf{Q}^*\mathbf{\Lambda}^*(\mathbf{Q}^*)'$ is another eigendecomposition of $\mathbf{A}$. One cannot, however, replace $\mathbf{K}$ by a matrix that is not diagonal, because this would destroy the requirement that $\mathbf{\Lambda}$ be diagonal.

(3) The number of eigenvalues that are *not* equal to zero is equal to the *rank* r of a matrix. If no eigenvalue of $\mathbf{A}$ is equal to zero, $\mathbf{A}$ has *full rank*. If there are eigenvalues equal to zero, the matrix has a *null space* with dimensionality greater than zero. It is spanned by the eigenvectors associated with the eigenvalues that are equal to zero.

(4) It can be shown (e.g., Searle, 1982) that if $\mathbf{A}$ is symmetric ($\mathbf{A} = \mathbf{A}'$), its eigenvalues and eigenvectors are always real-valued. Because symmetric matrices are so predominant in MDS, we always assume in the sequel that this condition is satisfied unless stated otherwise. If $\mathbf{A}$ is symmetric, it also has orthogonal eigenvectors. If we assume what is almost always true in practice, namely, that $\lambda_i \neq \lambda_j$, the orthogonality of eigenvectors follows from $\lambda_i\mathbf{q}_j'\mathbf{q}_i = \mathbf{q}_j'\lambda_i\mathbf{q}_i = \mathbf{q}_j'\mathbf{A}\mathbf{q}_i = \mathbf{q}_i'\mathbf{A}'\mathbf{q}_j = \mathbf{q}_i'\mathbf{A}\mathbf{q}_j = \mathbf{q}_i'\lambda_j\mathbf{q}_j = \lambda_j\mathbf{q}_i'\mathbf{q}_j = \lambda_j\mathbf{q}_j'\mathbf{q}_i$. That is, $\lambda_i\mathbf{q}_j'\mathbf{q}_i = \lambda_j\mathbf{q}_j'\mathbf{q}_i$. Because $\lambda_i \neq \lambda_j$, $\mathbf{q}_j'\mathbf{q}_i = 0$, so $\mathbf{q}_j$ and $\mathbf{q}_i$ are orthogonal. If $\lambda_i = \lambda_j$, the eigenvectors can also be constructed orthogonally.

(5) It is natural to ask to what extent the sum-of-squares of $\mathbf{A}$ is accounted for by each of its component matrices, $\lambda_i\mathbf{q}_i\mathbf{q}_i'$. In equation (7.12) we have $\|\mathbf{A}\|^2 = (23^2 + \cdots + 2^2) = 3125$. For the spectral sum of $\mathbf{A}$, we get $\|\lambda_1\mathbf{q}_1\mathbf{q}_1' + \cdots + \lambda_n\mathbf{q}_n\mathbf{q}_n'\|^2 = 3125$. This expression can be split up into $\|\lambda_1\mathbf{q}_1\mathbf{q}_1'\|^2 + \cdots + \|\lambda_n\mathbf{q}_n\mathbf{q}_n'\|^2$. Using (7.3), this is equal to $\lambda_1^2\|\mathbf{q}_1\mathbf{q}_1'\|^2 + \cdots + \lambda_n^2\|\mathbf{q}_n\mathbf{q}_n'\|^2$. But each $\|\mathbf{q}_i\mathbf{q}_i'\|^2 = 1$, which follows as a consequence of choosing $\mathbf{Q}$ so that $\mathbf{Q}\mathbf{Q}' = \mathbf{I}$. Hence, the sum-of-squares of $\mathbf{A}$ is equal to the sum of the squared eigenvalues. In our example in equation (7.12), we therefore have $50^2 + (-25)^2 = 3125$, the same value as before for $\|\mathbf{A}\|^2$. Hence, the first component matrix in (7.12), $\lambda_1\mathbf{q}_1\mathbf{q}_1'$, accounts for $50^2/(50^2+25^2) = .80$ or 80% of $\mathbf{A}$'s sum-of-squares.

(6) The eigendecomposition of $\mathbf{A}$ can be understood in many ways. One way is that it is an attempt to approximate $\mathbf{A}$ by a matrix of lower rank k. The best-possible approximation is the matrix $\lambda_1\mathbf{q}_1\mathbf{q}_1' + \cdots + \lambda_k\mathbf{q}_k\mathbf{q}_k'$. Each component matrix $\lambda_i\mathbf{q}_i\mathbf{q}_i'$ has rank 1, and adding k such matrices leads to a matrix with rank k.

(7) Matrices may not only be understood as configurations but also as transformations. For example, formula (7.9) says that the matrix $\mathbf{A}$ *acts* on the vector $\mathbf{q}_i$ just like a scalar, the eigenvalue λ_i, a particularly simple transformation. Usually, things are not that simple. Consider the case where we want to reflect the vector $\mathbf{x}$ in the plane about the line $x = y$. This is

accomplished by premultiplying $\mathbf{x}$ by a reflection matrix $\mathbf{T}$ so that $\mathbf{Tx} = \mathbf{x}^*$ is the reflected vector:

$$\mathbf{Tx} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ x_1 \end{bmatrix} = \begin{bmatrix} x_1^* \\ x_2^* \end{bmatrix} = \mathbf{x}^*.$$

If we replace $\mathbf{T}$ by its spectral decomposition, we have split up the transformation $\mathbf{T}$ into a sum of operations. We can understand each operation $\lambda_i \mathbf{q}_i \mathbf{q}_i'$ by noting some peculiar properties of its matrix $\mathbf{q}_i \mathbf{q}_i'$. Let $\mathbf{P}_i = \mathbf{q}_i \mathbf{q}_i'$ for short. First, we observe that $(\lambda_i \mathbf{P}_i)(\lambda_i \mathbf{P}_i) = \lambda_i^2 \mathbf{P}_i$. A matrix $\mathbf{A}$ for which $\mathbf{AA}$=$\mathbf{A}$ is called *idempotent* or a *projector*. $\mathbf{P}_i$ projects the vector $\mathbf{x}$ onto the eigenvector $\mathbf{q}_i$. The length of $\mathbf{x}$ on this eigenvector is λ_i. Second, $\mathbf{P}_i \mathbf{P}_j = \mathbf{0}$, for $i \neq j$, because $\mathbf{QQ}' = \mathbf{I}$. Hence, the projections effected by $\mathbf{P}_1, \ldots, \mathbf{P}_r$ are onto r orthogonal dimensions, the eigenvectors. Third, $\mathbf{P}_1 + \ldots + \mathbf{P}_r = \mathbf{I}$, which means that the total length of the projected vector is equal to the original vector. For our example and a vector $\mathbf{x} = (2, 3)$, we get

$$\begin{aligned} \mathbf{Tx} &= \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 2 \\ 3 \end{bmatrix} = (\lambda_1 \mathbf{q}_1 \mathbf{q}_1' + \lambda_2 \mathbf{q}_2 \mathbf{q}_2')\mathbf{x} \\ &= \left((-1) \begin{bmatrix} .5 & -.5 \\ -.5 & .5 \end{bmatrix} + (1) \begin{bmatrix} .5 & .5 \\ .5 & .5 \end{bmatrix} \right) \begin{bmatrix} 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 3 \\ 2 \end{bmatrix} = \mathbf{x}^*. \end{aligned}$$

One can check here geometrically that the transformation $\mathbf{T}$ is such that $\mathbf{x}$ is projected onto the two bisector lines of the plane that can be generated from multiplying the two eigenvectors by all possible real numbers. Postmultiplying the first component matrix by $\mathbf{x}$ means projecting $\mathbf{x}$ onto the eigenvector $\mathbf{q}_1$, which lies on the line $x = -y$, and then reflecting this projection by multiplying it by $\lambda_1 = -1$. The analogous is true for the second component matrix and the second eigenvector. The reflected vector $\mathbf{x}^*$, then, is built from these two vectors that lie on the eigenvalue lines.

(8) An $n \times n$ real symmetric matrix is called *positive definite* if for every $\mathbf{x} \neq \mathbf{0}$ we have $\mathbf{x}'\mathbf{Ax} > 0$. This definition implies that all eigenvalues of $\mathbf{A}$ are strictly greater than 0. This can be seen as follows. If we choose a particular vector $\mathbf{x}$, namely, an eigenvector $\mathbf{q}_i$ of $\mathbf{A}$, then $\mathbf{q}_i'\mathbf{Aq}_i = \lambda_i \mathbf{q}_i'\mathbf{q}_i = \lambda_i$, because $\mathbf{q}_i'\mathbf{q}_i = 1$. Thus, $\lambda_i > 0$ because $\mathbf{q}_i'\mathbf{Aq}_i$ is positive. If $\lambda_i \geq 0$, we call $\mathbf{A}$ *positive semidefinite*.[3] Similarly, a *negative definite* matrix has eigenvalues $\lambda_a < 0$ and consequently $\mathbf{x}'\mathbf{Ax} < 0$ for every $\mathbf{x}$, whereas a negative semidefinite matrix has eigenvalues $\lambda_a \leq 0$ and consequently $\mathbf{x}'\mathbf{Ax} \leq 0$.

[3]Positive definite matrices are closely related to sums-of-squares and, thus, play an important role in multivariate data analysis and in MDS. For example, we can write the sum of squared deviations as $\sum_i (x_i - \bar{x})^2 = \sum_i x_i^2 - n\bar{x}^2 = \mathbf{x}'\mathbf{x} - n\bar{x}^2 = \mathbf{x}'\mathbf{Jx}$, where $\mathbf{J}$ is the "centering" matrix $\mathbf{J} = \mathbf{I} - (1/n)\mathbf{11}'$ and $\mathbf{11}'$ is an $n \times n$ matrix of ones.

Finding a Matrix Inverse via Eigendecomposition

The eigendecomposition can be used for computing the inverse of a matrix. Suppose that we have the eigendecomposition $\mathbf{A} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}'$ and we want to compute the inverse $\mathbf{B} = \mathbf{A}^{-1}$. From Table 7.3, we know that $\mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$, so that $\mathbf{B}\mathbf{A} = \mathbf{I}$. Replacing $\mathbf{A}$ by $\mathbf{Q}\mathbf{\Lambda}\mathbf{Q}'$ gives

$$\mathbf{B}\mathbf{Q}\mathbf{\Lambda}\mathbf{Q}' = \mathbf{I}. \tag{7.13}$$

The unknown $\mathbf{B}$ can be derived by using the orthonormality of $\mathbf{Q}$ and the diagonality of $\mathbf{\Lambda}$. Because $\mathbf{Q}$ is orthonormal and square, we have $\mathbf{Q}'\mathbf{Q} = \mathbf{Q}\mathbf{Q}' = \mathbf{I}$. Hence, postmultiplying (7.13) by $\mathbf{Q}$ gives

$$\mathbf{B}\mathbf{Q}\mathbf{\Lambda} = \mathbf{Q}.$$

The matrix of eigenvalues $\mathbf{\Lambda}$ is diagonal so that its inverse is simply $diag(1/\lambda_1, \ldots, 1/\lambda_n) = \mathbf{\Lambda}^{-1}$. If we postmultiply both sides by $\mathbf{\Lambda}^{-1}$ (using $\mathbf{\Lambda}\mathbf{\Lambda}^{-1} = \mathbf{I}$), we get

$$\mathbf{B}\mathbf{Q} = \mathbf{Q}\mathbf{\Lambda}^{-1}.$$

Using the orthonormality of $\mathbf{Q}$ again and postmultiplying both sides by $\mathbf{Q}'$, we obtain an expression for the inverse of $\mathbf{A}$:

$$\mathbf{A}^{-1} = \mathbf{B} = \mathbf{Q}\mathbf{\Lambda}^{-1}\mathbf{Q}'.$$

From this expression, one can see that if $\mathbf{\Lambda}$ contains zero eigenvalues, $\mathbf{\Lambda}^{-1}$ does not exist, because its diagonal elements $1/\lambda_i$ are undefined for the $\lambda_i = 0$. In other words, if $\mathbf{A}$ is not of full rank, then its inverse does not exist.

7.5 Singular Value Decompositions

A decomposition closely related to the eigendecompositions and even more useful in algebra and for computational purposes is the *singular value decomposition*, SVD, of a matrix. The SVD is also known as the *Eckart–Young theorem*. The main idea of the SVD is that every $n \times m$ matrix $\mathbf{A}$ can be decomposed into

$$\mathbf{A} = \mathbf{P}\mathbf{\Phi}\mathbf{Q}' \tag{7.14}$$

with $\mathbf{P}$ an $n \times m$ matrix of *left singular vectors*, all orthonormal to each other (i.e., $\mathbf{P}'\mathbf{P} = \mathbf{I}$), $\mathbf{\Phi}$ an $m \times m$ diagonal matrix with *singular values* $\phi_i \geq 0$, and $\mathbf{Q}$ an $m \times m$ matrix of *right singular vectors*, all orthonormal to each other (i.e., $\mathbf{Q}'\mathbf{Q} = \mathbf{I}$).

By exploiting the properties of the SVD, it becomes clear how we may compute the SVD. Assume for the moment that we know the SVD of $\mathbf{A}$ as given in (7.14). Then,

$$\mathbf{A}'\mathbf{A} = \mathbf{Q}\mathbf{\Phi}\mathbf{P}'\mathbf{P}\mathbf{\Phi}\mathbf{Q}' = \mathbf{Q}\mathbf{\Phi}\mathbf{\Phi}\mathbf{Q}' = \mathbf{Q}\mathbf{\Phi}^2\mathbf{Q}',$$

which is just the eigendecomposition of $\mathbf{A}'\mathbf{A}$. This proves that the eigenvalues of $\mathbf{A}'\mathbf{A}$ are all nonnegative because they consist of ϕ_i^2 and squared numbers are always nonnegative. Thus, for computing the SVD of $\mathbf{A}$ we start by computing the eigendecomposition of $\mathbf{A}'\mathbf{A} = \mathbf{Q}\mathbf{\Phi}^2\mathbf{Q}'$, which gives us $\mathbf{\Phi}$ and $\mathbf{Q}$ as a result. Using the orthonormality of $\mathbf{Q}$ and the diagonality of $\mathbf{\Phi}$, we obtain $\mathbf{P}$; that is,

$$\begin{aligned} \mathbf{A} &= \mathbf{P}\mathbf{\Phi}\mathbf{Q}' \\ \mathbf{A}\mathbf{Q} &= \mathbf{P}\mathbf{\Phi}\mathbf{Q}'\mathbf{Q} = \mathbf{P}\mathbf{\Phi} \\ \mathbf{A}\mathbf{Q}\mathbf{\Phi}^{-1} &= \mathbf{P}\mathbf{\Phi}\mathbf{\Phi}^{-1} = \mathbf{P}. \end{aligned} \tag{7.15}$$

As an example, we want to find the SVD of

$$\mathbf{X} = \begin{bmatrix} 1 & 2 \\ 3 & 1 \\ 2 & 0 \end{bmatrix}.$$

First, we have to find the eigendecomposition of $\mathbf{X}'\mathbf{X}$; that is,

$$\begin{aligned} \mathbf{X}'\mathbf{X} &= \mathbf{Q}\mathbf{\Phi}^2\mathbf{Q}' = \begin{bmatrix} 14 & 5 \\ 5 & 5 \end{bmatrix} \\ &= \begin{bmatrix} .91 & -.41 \\ .41 & .91 \end{bmatrix} \begin{bmatrix} 16.03 & 0.00 \\ 0.00 & 2.77 \end{bmatrix} \begin{bmatrix} .91 & .41 \\ -.41 & .91 \end{bmatrix}, \end{aligned} \tag{7.16}$$

which gives us $\mathbf{\Phi}$ (with $\phi_1 = 4.03$ and $\phi_2 = 1.67$) and $\mathbf{Q}$. With (7.15) we can compute $\mathbf{P}$; that is,

$$\begin{aligned} \mathbf{P} &= \mathbf{X}\mathbf{Q}\mathbf{\Phi}^{-1} \\ &= \begin{bmatrix} 1 & 2 \\ 3 & 1 \\ 2 & 0 \end{bmatrix} \begin{bmatrix} .91 & -.41 \\ .41 & .91 \end{bmatrix} \begin{bmatrix} 4.03 & 0.00 \\ 0.00 & 1.67 \end{bmatrix}^{-1} \\ &= \begin{bmatrix} .43 & .85 \\ .78 & -.19 \\ .45 & -.49 \end{bmatrix}. \end{aligned}$$

Combining these results shows that the SVD of $\mathbf{X}$ is given by

$$\begin{aligned} \mathbf{X} &= \mathbf{P}\mathbf{\Phi}\mathbf{Q}' \\ &= \begin{bmatrix} .43 & .85 \\ .78 & -.19 \\ .45 & -.49 \end{bmatrix} \begin{bmatrix} 4.03 & 0.00 \\ 0.00 & 1.67 \end{bmatrix} \begin{bmatrix} .91 & .41 \\ -.41 & .91 \end{bmatrix}. \end{aligned} \tag{7.17}$$

It may be verified that the product $\mathbf{P\Phi Q'}$ does indeed reconstruct $\mathbf{X}$. Let us check whether $\mathbf{P'P} = \mathbf{I}$. This means that the columns $\mathbf{p}_1$ and $\mathbf{p}_2$ must satisfy $\mathbf{p}_1'\mathbf{p}_1 = 1$, $\mathbf{p}_2'\mathbf{p}_2 = 1$, and $\mathbf{p}_1'\mathbf{p}_2 = 0$: $p_{11}^2 + p_{21}^2 + p_{31}^2 = .43^2 + .78^2 + .45^2 = 1.00$, $p_{12}^2 + p_{22}^2 + p_{32}^2 = .85^2 + (-.19)^2 + (-.49)^2 = 1.00$, and $p_{11}p_{12} + p_{21}p_{22} + p_{31}p_{32} = .43 \cdot .85 + .78 \cdot (-.19) + .45 \cdot (-.49) = .00$. This shows that $\mathbf{P'P} = \mathbf{I}$. In the same way, the orthonormality of $\mathbf{Q}$ can be checked.

The number of nonzero singular values is equal to the rank of $\mathbf{A}$. Thus, if $\mathbf{A}$ has one or more zero singular values, it is *singular* or *rank deficient*, which means that the columns (rows) are *linearly dependent*. That is, any column (row) of $\mathbf{A}$ is equal to a weighted sum (*linear combination*) of the other columns (rows). If $\mathbf{A}$ has rank 2, for example, then exactly two columns (rows) can be identified, which, with appropriate weights, allows one to reproduce all other columns (rows) of $\mathbf{A}$. Consider the matrix

$$\mathbf{A} = [\mathbf{a}_1|\mathbf{a}_2|\mathbf{a}_3] = \begin{bmatrix} 3 & 2 & 4 \\ 1 & 4 & -2 \\ 4 & 1 & 7 \end{bmatrix},$$

where $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$ are column vectors. The singular values of $\mathbf{A}$ are 9.672, 4.738, 0.000, which implies that any one of the columns is a weighted sum of the other two. For example, $b_1\mathbf{a}_1 + b_2\mathbf{a}_2 = b_3\mathbf{a}_3$. It may be verified that choosing $b_1 = 2$, $b_2 = -1$, and $b_3 = 1$ solves the equation. Note that we might as well have chosen $b_2\mathbf{a}_2 + b_3\mathbf{a}_3 = b_1\mathbf{a}_1$, which gives an equivalent solution for $b_1 = 1$, $b_2 = 1/2$, and $b_3 = 1/2$.

7.6 Some Further Remarks on SVD

In the following, we list some properties of SVD that are useful in the remaining sections of this book.

(1) An SVD of a real $n \times m$ matrix can be written in several ways. The most parsimonious way is called *full rank decomposition*. It uses only those parts of the three component matrices that are needed to reconstruct $\mathbf{A}$. That is, we choose $\mathbf{P}$ and $\mathbf{Q}$ so that $\mathbf{P'P} = \mathbf{Q'Q} = \mathbf{I}_r$, and of $\mathbf{\Phi}$ we only use the upper left-hand corner $r \times r$ submatrix, where $r = \text{rank}(\mathbf{A})$. The version used above in (7.14) or (7.17) is a potentially *rank deficient* case, because here $\mathbf{P}$, for example, may have unnecessary columns if there are zero singular values in $\mathbf{\Phi}$. An often-used rank deficient case is when we augment both $\mathbf{P}$ and $\mathbf{Q}$ with appropriate vectors so that they become $n \times n$ and $m \times m$ orthonormal matrices, respectively. We symbolize this as follows.

$$\mathbf{A} = \mathbf{P}_{n\times n} \begin{bmatrix} \mathbf{\Phi}_{r\times r} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}_{n\times m} \mathbf{Q}_{m\times m}.$$

The leading submatrix $\mathbf{\Phi}_r$ is square and positive definite. As an example, consider equation (7.17), which becomes

$$\begin{aligned}\mathbf{X} &= \mathbf{P}_{3\times 3}\mathbf{\Phi}_{3\times 2}\mathbf{Q}'_{2\times 2} \\ &= \left[\begin{array}{cc|c} .43 & .85 & .30 \\ .78 & -.19 & -.18 \\ .45 & -.49 & .75 \end{array}\right] \left[\begin{array}{cc} 4.03 & 0.00 \\ 0.00 & 1.67 \\ \hline 0 & 0 \end{array}\right] \left[\begin{array}{cc} .91 & .41 \\ -.41 & .91 \end{array}\right].\end{aligned}$$

Obviously, the third column of $\mathbf{P}_{3\times 3}$ is needed to make $\mathbf{P}_{3\times 3}$ orthonormal, but it is not needed to reconstruct $\mathbf{X}$, because it is eliminated by the zero singular value in the SVD matrix product.

The full rank case allows one to reduce the three-matrix SVD product to two matrices, for example, by splitting $\mathbf{\Phi}_{r\times r}$ into two matrices $\mathbf{\Phi}^{1/2}_{r\times r}$ and then setting $\mathbf{L} = \mathbf{P}_{n\times r}\mathbf{\Phi}^{1/2}_{r\times r}$ and $\mathbf{R}' = \mathbf{\Phi}^{1/2}_{r\times r}\mathbf{Q}'_{r\times m}$. Thus, $\mathbf{X} = \mathbf{L}\mathbf{R}'$. The factors $\mathbf{L}$ and $\mathbf{R}'$ are unique up to an arbitrary but full rank transformation $\mathbf{T}_{r\times r}$, because $\mathbf{L}\mathbf{R}' = (\mathbf{L}\mathbf{T})(\mathbf{T}^{-1}\mathbf{R}')$ if $\mathbf{T}$ has full rank r. Factorizations of this sort are used in unfolding and in correspondence analysis, for example. The rank-deficient case of SVD is often useful in algebraic manipulations, because it always has orthogonal matrices $\mathbf{P}$ and $\mathbf{Q}$.

(2) If all singular values are different—which is almost always true with real data—then the singular vectors in $\mathbf{P}$ and $\mathbf{Q}$ are uniquely determined except for reflections.

(3) If $\mathbf{A}$ is symmetric, then its SVD is simply $\mathbf{A} = \mathbf{T}\mathbf{\Phi}\mathbf{T}'$. If $\mathbf{A} = \mathbf{A}'$, we have $\mathbf{P}\mathbf{\Phi}\mathbf{Q}' = \mathbf{Q}\mathbf{\Phi}\mathbf{P}'$, which, after pre- and postmultiplying by $\mathbf{P}$ and $\mathbf{Q}$ and using their orthogonality, yields $\mathbf{P}'\mathbf{Q} = \mathbf{I}$ and thus $\mathbf{P} = \mathbf{Q}$. Thus, if $\mathbf{A}$ is symmetric and positive semidefinite, the SVD corresponds to an eigendecomposition.

(4) The SVD, like the spectral decomposition, provides an optimal least-squares approximation of a matrix $\mathbf{A}$ by a matrix of lower rank. For $\text{rank}(\mathbf{A}) = r \geq k$, the best approximating matrix results from retaining the first k singular values in $\mathbf{\Phi}$ and replacing the remaining $k - r$ by zeros. $\mathbf{A}$ is thus approximated by the matrix sum $\phi_1\mathbf{p}_1\mathbf{q}'_1 + \cdots + \phi_k\mathbf{p}_k\mathbf{q}'_k$, where $\mathbf{p}_i$ and $\mathbf{q}_i$ are the ith column vectors of $\mathbf{P}$ and $\mathbf{Q}$, respectively. This matrix sum has similar properties as the spectral decomposition discussed above. To illustrate, consider the picture in Figure 7.2a. This picture is generated from a 200-by-320 matrix that contains the codes for its pixels. One can approximate this matrix with matrices of much lower rank than 200 in the sense of the above SVD. Figures 7.2b and 7.2c show that some 10 to 20 SVD components suffice to recognize the picture (Gramlich, 2004). Hence, the essential information of the 200-dimensional space of the picture is contained in a space of only about 20 dimensions, and the SVD shows how to obtain this reduced space. This not only provides a solution of a technical compression problem: it also suggests a bottom-up model for the recognition of faces in psychology (see Section 17.7).

a. b. c.

FIGURE 7.2. A 200-by-320 pixel picture (left side), approximated by 10 (center) and 20 (right side) SVD components (Gramlich, 2004).

7.7 Linear Equation Systems

Matrices are closely related to systems of linear equations. Consider an example:

$$\begin{array}{rcrcrcr} -x_1 & + & 2x_2 & + & x_3 & = & -2, \\ 3x_1 & - & 8x_2 & - & 2x_3 & = & 4, \\ x_1 & & & + & 4x_3 & = & -2. \end{array} \tag{7.18}$$

The system is called linear because each equation is a weighted sum of the unknowns x_1, x_2, and x_3. The graph of such an equation in a Cartesian coordinate system corresponds to a straight line. The equations in (7.18) consist of the unknowns x_1, x_2, x_3, the coefficients $-1, 2, \ldots, 4$, and the constants $-2, 4$, and -2. If we remove all symbols from (7.18) except the coefficients, we obtain the matrix

$$\mathbf{A} = \begin{bmatrix} -1 & 2 & 1 \\ 3 & -8 & -2 \\ 1 & 0 & 4 \end{bmatrix}. \tag{7.19}$$

We can also array the unknowns and the constants from (7.18) in vectors:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \text{ and } \mathbf{b} = \begin{bmatrix} -2 \\ 4 \\ -2 \end{bmatrix}. \tag{7.20}$$

Combining (7.19) and (7.20), we can write the equation system (7.18) in matrix notation, very compactly, as

$$\mathbf{Ax} = \mathbf{b}$$

or, more explicitly, as

$$\begin{bmatrix} -1 & 2 & 1 \\ 3 & -8 & -2 \\ 1 & 0 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} -2 \\ 4 \\ -2 \end{bmatrix}. \tag{7.21}$$

That (7.21) is equivalent to (7.18) can be seen by multiplying $\mathbf{A}$ by $\mathbf{x}$ according to the multiplication rule for matrices.

Solving a System of Linear Equations

The linear equation system $\mathbf{Ax} = \mathbf{b}$ can be solved by premultiplying both sides of the equation with $\mathbf{A}^{-1}$ so that $\mathbf{A}^{-1}\mathbf{Ax} = \mathbf{A}^{-1}\mathbf{b}$ or $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$. The vector $\mathbf{A}^{-1}\mathbf{b}$ is a solution, because inserting this vector for $\mathbf{x}$ into $\mathbf{Ax} = \mathbf{b}$ leads to $\mathbf{b} = \mathbf{b}$.

Let the SVD of $\mathbf{A}$ be given by $\mathbf{P\Phi Q}'$, where as usual $\mathbf{P}'\mathbf{P} = \mathbf{I}$, $\mathbf{\Phi}$ is diagonal, and $\mathbf{Q}'\mathbf{Q} = \mathbf{I}$. Making extensive use of these properties allows us to solve the linear system $\mathbf{Ax} = \mathbf{b}$ as follows.

$$\begin{aligned}
\mathbf{Ax} &= \mathbf{b}, \\
(\mathbf{P\Phi Q}')\mathbf{x} &= \mathbf{b}, \\
\mathbf{P}'\mathbf{P\Phi Q}'\mathbf{x} &= \mathbf{P}'\mathbf{b}, \\
\mathbf{\Phi Q}'\mathbf{x} &= \mathbf{P}'\mathbf{b}, \\
\mathbf{\Phi}^{-1}\mathbf{\Phi Q}'\mathbf{x} &= \mathbf{\Phi}^{-1}\mathbf{P}'\mathbf{b}, \\
\mathbf{QQ}'\mathbf{x} &= \mathbf{Q\Phi}^{-1}\mathbf{P}'\mathbf{b}, \\
\mathbf{x} &= \mathbf{Q\Phi}^{-1}\mathbf{P}'\mathbf{b}. \qquad (7.22)
\end{aligned}$$

The linear system $\mathbf{Ax} = \mathbf{b}$ is solved by $\mathbf{x} = \mathbf{Q\Phi}^{-1}\mathbf{P}'\mathbf{b}$. Note that if $\mathbf{A}$ is not square or of full rank, then $\mathbf{\Phi}$ has diagonal elements that are zero, so that $\mathbf{\Phi}^{-1}$ does not exist. If this is true, then there is no unique $\mathbf{x}$ that solves $\mathbf{Ax} = \mathbf{b}$.

Let us apply (7.22) to solve (7.21). The SVD of $\mathbf{A}$ is given by

$$\begin{bmatrix} .27 & .07 & -.96 \\ -.96 & .11 & -.26 \\ .09 & .99 & .10 \end{bmatrix} \begin{bmatrix} 9.12 & .00 & .00 \\ .00 & 4.08 & .00 \\ .00 & .00 & .32 \end{bmatrix} \begin{bmatrix} -.34 & .90 & .28 \\ .30 & -.17 & .94 \\ .89 & .40 & -.22 \end{bmatrix}.$$

For $\mathbf{x} = \mathbf{Q\Phi}^{-1}\mathbf{P}'\mathbf{b}$, we thus find

$$\mathbf{x} = \begin{bmatrix} -.34 & .30 & .89 \\ .90 & -.17 & .40 \\ .28 & .94 & -.22 \end{bmatrix} \begin{bmatrix} .11 & .00 & .00 \\ .00 & .25 & .00 \\ .00 & .00 & 3.10 \end{bmatrix} \begin{bmatrix} .27 & -.96 & .09 \\ .07 & .11 & .99 \\ -.96 & -.26 & .10 \end{bmatrix} \begin{bmatrix} -2 \\ 4 \\ -2 \end{bmatrix} = \begin{bmatrix} 2.0 \\ 0.5 \\ -1.0 \end{bmatrix}.$$

Hence, $\mathbf{x} = (2, 0.5, -1)$ solves (7.18). Here, $\mathbf{Q\Phi}^{-1}\mathbf{P}'$ is equal to its inverse $\mathbf{A}^{-1}$. It may be verified that the condition $\mathbf{A}^{-1}\mathbf{A} = \mathbf{AA}^{-1} = \mathbf{I}$ holds, as is required for the inverse.

Uniqueness, Existence, and g-Inverses

Consider the simple equation $ax = b$, where a, b, and x are scalars. One tends to say that the solution of this equation is $x = b/a$. However, there are three possibilities: (1) if $a \neq 0$, then $x = b/a$ and b/a is the *unique* solution whatever the value of b; (2) if $a = 0$ and $b = 0$, then *any* number x is a solution because $0x = 0$; (3) if $a = 0$ and $b \neq 0$, then $0x \neq 0$ and no solution exists, because the equation is *inconsistent*, implying the

contradiction $0 = b \neq 0$. Exactly the same three possibilities exist for a system of linear equations $\mathbf{Ax} = \mathbf{b}$.

The natural approach to solving $\mathbf{Ax} = \mathbf{b}$ is to ask for the inverse $\mathbf{A}^{-1}$ so that $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$. If this inverse exists, we have a unique solution. But the inverse may not exist because (a) we have "too few" independent equations or (b) because we have "too many" independent equations. Case (a) is illustrated by the following example.

$$\mathbf{A}_1\mathbf{x}_1 = \begin{bmatrix} -1 & 2 & 1 \\ 3 & -8 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} -2 \\ 4 \end{bmatrix} = \mathbf{b}_1. \tag{7.23}$$

Obviously, this system is underdetermined, so that if we solve it for two unknowns, the solutions will always contain the third unknown. For the third unknown, we can pick any value. The system, therefore, is not uniquely solvable. Case (b) is illustrated as follows.

$$\mathbf{A}_2\mathbf{x}_2 = \begin{bmatrix} -1 & 2 \\ 3 & -8 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} -2 \\ 4 \\ -2 \end{bmatrix} = \mathbf{b}_2. \tag{7.24}$$

This system is inconsistent. It has no solution. But consider also the following case.

$$\mathbf{A}_3\mathbf{x}_3 = \begin{bmatrix} -1 & 2 & 1 \\ 3 & -8 & -2 \\ 1 & -2 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} -2 \\ 4 \\ -2 \end{bmatrix} = \mathbf{b}_3. \tag{7.25}$$

Even though this system has three equations and three unknowns, it has no solution. The three equations contain only two different pieces of information, because one notes that the third row in $\mathbf{A}_3$ is just -1 times the first row. Hence, the rank of $\mathbf{A}$ is only 2, and we could, at best, have an under-determined system. It turns out, however, that the system is also inconsistent, because the first and the third equations, being the same except for a multiplier of -1, do not have the same relationship on the side of the coefficients. That is, we do not have $b_1 = -b_3$. This example shows, therefore, that having as many equations as unknowns or, in other words, having a square matrix $\mathbf{A}$ is only necessary but not sufficient for a unique solution to exist.

The case where no solution exists is typical in empirical research. For example, in regression problems where one claims that one dependent variable $\mathbf{y}$ is a linear combination of a set of independent variables $\mathbf{X}$, this is typically only "approximately" true. In this case, the equation system $\mathbf{y} = \mathbf{Xw}$ is inconsistent and we are looking for an optimal approximate solution for $\mathbf{w}$ that minimizes $\|\mathbf{Xw} - \mathbf{y}\|$. Assuming that $\mathbf{X}$ has full column

rank, the best $\mathbf{Xw}$ is $\hat{\mathbf{y}} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$, where $\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$ projects[4] the vector $\mathbf{y}$ onto the space spanned by the columns of $\mathbf{X}$. If $\mathbf{X}$ is rank deficient, however, we cannot directly compute this solution but first must eliminate linear dependencies from the predictors $\mathbf{X}$.

It is not easy to keep track of all of this, but, fortunately, there exists a unified treatment that makes things pleasantly simple. Instead of $\mathbf{A}^{-1}$, one can use a *generalized inverse*, which is equal to the usual inverse if it exists and provides a least-squares solution in that case. One such generalized inverse is the *Moore–Penrose inverse* or *pseudoinverse*, $\mathbf{A}^{+}$. It is the unique matrix that can be computed from the full rank SVD $\mathbf{A} = \mathbf{P}\mathbf{\Phi}_{r\times r}\mathbf{Q}'$ as $\mathbf{A}^{+} = \mathbf{Q}\mathbf{\Phi}_{r\times r}^{-1}\mathbf{P}'$. The above regression problem is solved even if there are linear dependencies in $\mathbf{X}$ by replacing the term $(\mathbf{X}'\mathbf{X})^{-1}$ by $(\mathbf{X}'\mathbf{X})^{+}$. For linear equation systems $\mathbf{Ax} = \mathbf{b}$ in general, optimal solutions are found by setting $\mathbf{x} = \mathbf{A}^{+}\mathbf{b}$. If an exact solution exists—as in (7.23)—then $\mathbf{x} = \mathbf{A}^{+}\mathbf{b}$ will yield it. (One can show that a system $\mathbf{Ax} = \mathbf{b}$ has an exact solution if and only if $\mathbf{AA}^{+}\mathbf{b} = \mathbf{b}$.) If no exact solution exists—as in (7.24) and (7.25)—$\mathbf{x} = \mathbf{A}^{+}\mathbf{b}$ gives the optimal least-squares solution.

There are plenty of generalized inverses. They are usually denoted by $\mathbf{A}^{-}$. They all share the property that $\mathbf{A} = \mathbf{AA}^{-}\mathbf{A}$, which obviously also holds for the regular inverse $\mathbf{A}^{-1}$. The Moore–Penrose has a number of additional properties. They are not always needed, and other forms of generalized inverses may suffice and may be cheaper to compute for a particular purpose. However, not all generalized inverses have the property that they provide least-squares solutions to $\mathbf{Ax} = \mathbf{b}$.

7.8 Computing the Eigendecomposition

We now show how an eigendecomposition can be computed. We consider a typical case, the symmetric matrix $\mathbf{B}$ used previously in (7.7). To find $\mathbf{B}$'s eigenvalues, we can use one of the many sophisticated iterative procedures available in modern computer packages. It would take too much time to explain any of these, but we can convey a sense of how they work by demonstrating the simple *power method*.

For scalar product matrices in the empirical sciences, we can safely assume that their eigenvalues are all positive and distinct so that $\lambda_1 > \cdots > \lambda_k \geq 0$. The number k is either equal to m or is the last eigenvector of interest. We then arbitrarily define some starting vector $\mathbf{q}^{[0]} \neq \mathbf{0}$ and iterate the system

$$\mathbf{q}^{[t+1]} = \|\mathbf{Bq}^{[t]}\|^{-1}\mathbf{Bq}^{[t]}$$

[4]The solution is derived by geometric arguments in Chapter 22. See Figure 22.2 and the accompanying text.

TABLE 7.5. Computing eigenvalues and eigenvectors by the power method. The product $\mathbf{q}^{[t]\prime}\mathbf{B}\mathbf{q}^{[t]}$ estimates the eigenvector λ at iteration t, $\lambda^{[t]}$.

$\mathbf{B}$			$\mathbf{q}^{[0]}$	$\mathbf{q}^{[1]}$	$\mathbf{q}^{[2]}$	$\mathbf{q}^{[3]}$	$\mathbf{q}^{[4]}$	$\mathbf{q}_1$
5	5	2	$1/\sqrt{3}$	.444	.431	.429	.429	.429
5	10	6	$1/\sqrt{3}$	.778	.781	.781	.781	.781
2	6	4	$1/\sqrt{3}$	.444	.452	.453	.454	.454
		$\lambda^{[t]}$	15.000	16.215	16.227	16.227	16.227	16.227

$$\lambda_1\mathbf{q}_1\mathbf{q}_1' = \begin{bmatrix} 2.986 & 5.437 & 3.160 \\ 5.437 & 9.898 & 5.754 \\ 3.160 & 5.754 & 3.345 \end{bmatrix}$$

$\mathbf{B}-\lambda_1\mathbf{q}_1\mathbf{q}_1'$			$\mathbf{q}^{[0]}$	$\mathbf{q}^{[1]}$	$\mathbf{q}^{[2]}$	$\mathbf{q}_2$
2.016	-.437	-1.156	$1/\sqrt{3}$	.853	.853	.853
-.437	.095	.251	$1/\sqrt{3}$	-.185	-.185	-.185
-1.156	.251	.663	$1/\sqrt{3}$	-.489	-.489	-.489
		$\lambda^{[t]}$	.030	2.776	2.776	2.776

$$\lambda_2\mathbf{q}_2\mathbf{q}_2' = \begin{bmatrix} 2.020 & -.438 & -1.158 \\ -.438 & .095 & .251 \\ -1.158 & .251 & .664 \end{bmatrix}$$

$$\lambda_1\mathbf{q}_1\mathbf{q}_1' + \lambda_2\mathbf{q}_2\mathbf{q}_2' = \begin{bmatrix} 2.99 & 5.44 & 3.16 \\ 5.44 & 9.90 & 5.75 \\ 3.16 & 5.75 & 3.34 \end{bmatrix} + \begin{bmatrix} 2.02 & -.44 & -1.16 \\ -.44 & .10 & .25 \\ -1.16 & .25 & .66 \end{bmatrix}$$

$$= \begin{bmatrix} 5 & 5 & 2 \\ 5 & 10 & 6 \\ 2 & 6 & 4 \end{bmatrix}$$

a few times until $\mathbf{q}^{[t+1]}$ remains essentially invariant over the iterations.[5] The scalar factor $\|\mathbf{B}\mathbf{q}^{[t]}\|^{-1}$ normalizes $\mathbf{B}\mathbf{q}^{[t]}$ which prevents the values of $\mathbf{q}$ from becoming extremely large or small over the iterations. After convergence, $\mathbf{q}$ is equal to the first eigenvector and $\mathbf{q}'\mathbf{B}\mathbf{q} = \lambda_1$ is the first eigenvalue. An example is shown in Table 7.5.

Starting with $\mathbf{q}^{[0]} = (1/\sqrt{3}, 1/\sqrt{3}, 1/\sqrt{3})$ in Table 7.5, $\mathbf{B}\mathbf{q}^{[0]} = (6.928, 12.124, 6.928)$ and $\|\mathbf{B}\mathbf{q}^{[0]}\| = \mathbf{q}^{[0]\prime}\mathbf{B}'\mathbf{B}\mathbf{q}^{[0]} = \sqrt{242.986} = 15.588$, so that $\mathbf{q}^{[1]} = (1/15.588)\cdot(6.928, 12.124, 6.928) = (.444, .778, .444)$. Further iterations of the same kind yield $\mathbf{q}^{[2]}$, $\mathbf{q}^{[3]}$, and so on. After four iterations, the results have stabilized. We obtain $\mathbf{q}^{[4]} = (.429, .781, .454)$ and an estimate of the eigenvalue λ_1 of $\mathbf{q}^{[4]\prime}\mathbf{B}\mathbf{q}^{[4]} = 16.227$.

[5]The notation $\mathbf{q}^{[t]}$ indicates that we are dealing with vector $\mathbf{q}$ at time t. Vector $\mathbf{q}^{[0]}$, thus, is $\mathbf{q}$ at time $t = 0$, that is, the starting vector.

How can we find the second eigenvector? Remember that the eigendecomposition of a square 3×3 matrix amounts to

$$\mathbf{B} = \lambda_1 \mathbf{q}_1 \mathbf{q}_1' + \lambda_2 \mathbf{q}_2 \mathbf{q}_2' + \lambda_3 \mathbf{q}_3 \mathbf{q}_3'.$$

At this stage, we know the first eigenvalue λ_1 and the first eigenvector $\mathbf{q}_1$. Moving the known part to the left-hand side of the equations gives

$$\mathbf{B} - \lambda_1 \mathbf{q}_1 \mathbf{q}_1' = \lambda_2 \mathbf{q}_2 \mathbf{q}_2' + \lambda_3 \mathbf{q}_3 \mathbf{q}_3'.$$

To compute the second eigenvalue and eigenvector, we apply the procedure to $\mathbf{B} - \lambda_1 \mathbf{q}_1 \mathbf{q}_1'$. This is shown in the second part of Table 7.5. Eigenvector $\mathbf{q}_2$ is $(.853, -.185, -.489)$ and λ_2 equals 2.776. To find the third eigenvalue, we have to repeat the procedure to $\mathbf{B} - \lambda_1 \mathbf{q}_1 \mathbf{q}_1' - \lambda_2 \mathbf{q}_2 \mathbf{q}_2'$, which in this example is equal to zero everywhere. Therefore, the third eigenvalue must be zero and the first two components suffice to specify $\mathbf{B}$.

Finally, we show why the power method works at all. We started by assuming that $|\lambda_1| > |\lambda_j|, j = 2, \ldots, k$. Also, for scalar product matrices, it holds that $\mathbf{B} = \mathbf{B}'$. The iterations can be written[6] explicitly as

$$\begin{aligned} \mathbf{q}^{[1]} &= \|\mathbf{B}\mathbf{q}^{[0]}\|^{-1} \mathbf{B}\mathbf{q}^{[0]}, \\ \mathbf{q}^{[2]} &= \|\mathbf{B}\mathbf{q}^{[1]}\|^{-1} \mathbf{B}\mathbf{q}^{[1]} \\ &= \|\mathbf{B}\mathbf{B}\mathbf{q}^{[0]}\|^{-1} \mathbf{B}(\mathbf{B}\mathbf{q}^{[0]}), \text{ etc., or as} \\ \mathbf{q}^{[t]} &= \|\mathbf{B}^{\mathbf{t}}\mathbf{q}^{[0]}\|^{-1} \mathbf{B}^t \mathbf{q}^{[0]}. \end{aligned} \tag{7.26}$$

But because $\mathbf{B} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}'$, $\mathbf{B}^2 = (\mathbf{Q}\mathbf{\Lambda}\mathbf{Q}')(\mathbf{Q}\mathbf{\Lambda}\mathbf{Q}') = \mathbf{Q}\mathbf{\Lambda}(\mathbf{Q}'\mathbf{Q})\mathbf{\Lambda}\mathbf{Q}' = \mathbf{Q}\mathbf{\Lambda}^2\mathbf{Q}'$ and, in general, $\mathbf{B}^t = \mathbf{Q}\mathbf{\Lambda}^t\mathbf{Q}'$. If λ_1 dominates all other eigenvalues, then $\mathbf{B}^t$ will be more and more approximated by the additive factor $\lambda_1 \mathbf{q}_1 \mathbf{q}_1'$ in the eigendecomposition as $t \to \infty$. Hence, we get $\mathbf{B}^t \mathbf{q}^{[0]} \approx (\lambda_1^t \mathbf{q}_1 \mathbf{q}_1')\mathbf{q}^{[0]} = \lambda_1^t \mathbf{q}_1 (\mathbf{q}_1' \mathbf{q}^{[0]}) = \lambda_1^t \mathbf{q}_1 k = \textit{ constant } \cdot \mathbf{q}_1$. So, the iterations eventually grind out the first eigenvector, $\mathbf{q}_1$. The irrelevant scaling constant is removed through normalization. The corresponding eigenvalue results from $\mathbf{q}_1' \mathbf{B} \mathbf{q}_1 = \lambda_1$ which follows from equation (7.9).

Apart from its assumptions concerning the distribution of the eigenvalues, the power method is not without problems. Suppose that the matrix to which the power method is applied is not a scalar product matrix, but any square symmetric matrix. Then it may happen that some eigenvalues are negative. Assume that the eigenvalues are ordered decreasingly, so that the largest eigenvalue is λ_1 and the smallest negative eigenvalue is λ_n. If the largest eigenvalue is smaller than minus the smallest eigenvalue, that is, $\lambda_1 < |\lambda_n|$, then the power method converges to the smallest negative eigenvalue λ_n and not to λ_1. A second problem occurs if by accident

[6] $\mathbf{B}^t$ is the product of $\mathbf{B}$ multiplied t times with itself. Thus, $\mathbf{B}^t = \mathbf{B}\mathbf{B}\mathbf{B}\ldots\mathbf{B}$, with t times $\mathbf{B}$.

the start vector $\mathbf{q}^{[0]}$ is chosen exactly equal to an eigenvector. Then, the power method finishes in one iteration, but the obtained eigenvalue is not necessarily the largest one. The third problem of the power method is its slow convergence if two eigenvalues are almost equal. In general, the power method can be accelerated by using $\mathbf{BB}$ instead of $\mathbf{B}$, so that the power method converges to the largest squared eigenvalue. The use of $\mathbf{BB}$ makes differences between the eigenvalues larger.

7.9 Configurations that Represent Scalar Products

We now return to the problem of finding a point configuration that represents a given scalar-product matrix. In matrix notation, this amounts to solving the equation

$$\mathbf{B} = \mathbf{X}\mathbf{X}', \tag{7.27}$$

where $\mathbf{X}$ is the $n \times m$ coordinate matrix of n points in m-dimensional space. Let

$$\mathbf{X} = \begin{bmatrix} 1 & 2 \\ 3 & 1 \\ 2 & 0 \end{bmatrix} \text{ and } \mathbf{B} = \mathbf{X}\mathbf{X}' = \begin{bmatrix} 5 & 5 & 2 \\ 5 & 10 & 6 \\ 2 & 6 & 4 \end{bmatrix}, \tag{7.28}$$

as in Sections 7.3 and 7.5. Suppose that we do an eigendecomposition of $\mathbf{B} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}'$. We know that scalar product matrices are symmetric and have nonnegative eigenvalues (see Section 7.5). Therefore, we may write $\mathbf{B} = (\mathbf{Q}\mathbf{\Lambda}^{1/2})(\mathbf{Q}\mathbf{\Lambda}^{1/2})' = \mathbf{U}\mathbf{U}'$, where $\mathbf{\Lambda}^{1/2}$ is a diagonal matrix with diagonal elements $\lambda_i^{1/2}$. Thus, $\mathbf{U} = \mathbf{Q}\mathbf{\Lambda}^{1/2}$ gives coordinates that reconstruct $\mathbf{B}$. In Table 7.5 the eigendecomposition of matrix $\mathbf{B}$ is given. The coordinates are

$$\begin{aligned} \mathbf{U} &= \mathbf{Q}\mathbf{\Lambda}^{1/2} \\ &= \begin{bmatrix} .43 & .85 \\ .78 & -.19 \\ .45 & -.49 \end{bmatrix} \begin{bmatrix} 4.03 & 0.00 \\ 0.00 & 1.67 \end{bmatrix} = \begin{bmatrix} 1.73 & 1.42 \\ 3.15 & -.31 \\ 1.83 & -.81 \end{bmatrix}. \end{aligned} \tag{7.29}$$

The coordinates in $\mathbf{U}$ differ from those of $\mathbf{X}$ in (7.28). This simply means that they are expressed relative to two different coordinate systems, which, however, can be rotated into each other. For the problem of finding a vector configuration for given scalar products, it is irrelevant how the coordinate axes are rotated. What matters is the configuration.

7.10 Rotations

For the purpose of easy interpretation, some rotations are more useful than others, especially if one wants to check hypotheses about the dimensions. In

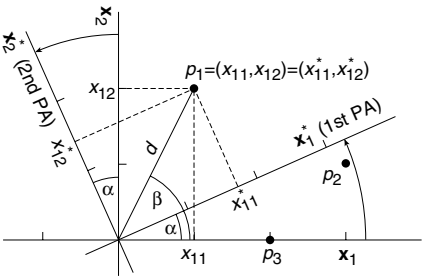


FIGURE 7.3. Rotation of coordinate system by α°.

factor analysis where dimensions play a dominant role, numerous criteria for rotating a configuration have been proposed (see, e.g., Mulaik, 1972). Probably the best known of these criteria is the *varimax* principle. It seeks to rotate a given configuration $\mathbf{X}$ such that the sum of the variances of the x_{ij}^2 in each column j of $\mathbf{X}$ is maximized across all columns. This criterion is designed to make the "loadings" x_{ij} either very small or very large so that each point of $\mathbf{X}$ lies, ideally, on or very close to just one of the dimensions.

This type of *simple structure rotation* is motivated by a particular theory about the dimensional structure of the configuration $\mathbf{X}$ and by considerations about the robustness of this dimensional structure (Kaiser, 1958). Another rotation criterion of a more formal nature is rotation to *principal axes*. Principal axes are the dimensions of a particular orthogonal coordinate system. It has the property that its first dimension (1st principal axis or 1st PA) lies closest to all points of the configuration $\mathbf{X}$. The second PA accounts for most of the points scatter that is orthogonal to the first PA, and so on. If the coordinates in $\mathbf{X}$ refer to a coordinate system whose dimensions are principal axes, then $\mathbf{X}\mathbf{X}'$ is diagonal, and the norm of the first column of $\mathbf{X}$, $\|\mathbf{x}_1\|$, is larger than the norm for any column of any rotation of $\mathbf{X}$. The norm of the second column is similarly the largest one, subject to the condition that $\mathbf{x}_2$ is orthogonal to $\mathbf{x}_1$, and so on.

Let us consider rotations in matrix terms. Rotations can be conceived of in two different ways. (1) The points (say, $p_1, \ldots, p_3$ in Figure 7.1) are transformed, but the coordinate system remains *fixed*. This is called the *alibi* interpretation of the transformation, because the points are moved somewhere else. (2) The points remain fixed, but the coordinate axes are transformed. This is the *alias* interpretation, because the points change their coordinates or *names*.

Consider Figure 7.3. The point p_1 has coordinates (x_{11}, x_{12}) relative to the axes $\mathbf{x}_1$ and $\mathbf{x}_2$. In an alias interpretation of rotation, p_1 is now to be coordinatized relative to new axes, such as the 1st PA and the 2nd PA, which result from $\mathbf{x}_1$ and $\mathbf{x}_2$ by a counterclockwise rotation through the angle α. The new coordinates, x_{11}^* and x_{12}^*, must depend, in some way, on the old coordinates, x_{11} and x_{12}, and the angle α.

First, we note in Figure 7.3 that $x_{11} = d\cos(\beta)$, $x_{12} = d\sin(\beta)$, $x_{11}^* = d\cos(\beta-\alpha)$, and $x_{12}^* = d\sin(\beta-\alpha)$, whence, using the well-known formulas for the sine and the cosine of the difference of two angles,

$$\begin{aligned} x_{11}^* &= d\cos(\beta-\alpha) = d[\cos(\beta)\cos(\alpha)+\sin(\beta)\sin(\alpha)] \\ &= [d\cos(\beta)]\cos(\alpha)+[d\sin(\beta)]\sin(\alpha) \\ &= x_{11}\cos(\alpha)+x_{12}\sin(\alpha), \text{ and} \\ x_{12}^* &= d\sin(\beta-\alpha) = d[\sin(\beta)\cos(\alpha)-\cos(\beta)\sin(\alpha)] \\ &= [d\sin(\beta)]\cos(\alpha)-[d\cos(\beta)]\sin(\alpha) \\ &= x_{12}\cos(\alpha)-x_{11}\sin(\alpha). \end{aligned}$$

Expressing this in matrix notation yields

$$\begin{bmatrix} x_{11}^* & x_{12}^* \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} \end{bmatrix} \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix}. \tag{7.30}$$

The matrix on the right-hand side of (7.30) is the rotation matrix $\mathbf{T}$. If we collect the point coordinates in an $n \times m$ matrix as usual, the new coordinate matrix $\mathbf{X}^*$ is related to the old $\mathbf{X}$ by $\mathbf{X}^* = \mathbf{XT}$. The rotation matrix $\mathbf{T}$ is orthonormal.

A general $m \times m$ rotation matrix can be composed as the product of all planewise rotations. In m-dimensional space, there are $\binom{m}{2} = m(m-1)/2$ such rotations. For example, in 4D the rotation in the plane spanned by the first and the fourth coordinate axes, $\mathbf{T}_{14}$, is

$$\mathbf{T}_{14} = \begin{bmatrix} \cos(\alpha_{14}) & 0 & 0 & -\sin(\alpha_{14}) \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \sin(\alpha_{14}) & 0 & 0 & \cos(\alpha_{14}) \end{bmatrix}.$$

The rotation of the entire 4D space is accomplished by

$$\mathbf{T} = \mathbf{T}_{12}\mathbf{T}_{13}\mathbf{T}_{14}\mathbf{T}_{23}\mathbf{T}_{24}\mathbf{T}_{34}.$$

That rotations leave all of the distances in a configuration unchanged is easy to see. Consider (7.5). Replacing $\mathbf{X}$ by $\mathbf{XT}$ has no effect on $\mathbf{XX}'$, because $\mathbf{XTT}'\mathbf{X}' = \mathbf{XX}'$. Also, the vector $\mathbf{c}$ is simply the collection of the diagonal elements of $\mathbf{XX}'$, and they are not affected by $\mathbf{T}$, as we just saw.

A particular choice of $\mathbf{T}$ is the matrix of $\mathbf{Q}$ from the SVD of $\mathbf{X} = \mathbf{P\Phi Q}'$. With $\mathbf{T} = \mathbf{Q}$, $\mathbf{XT}$ yields a principal axes orientation of the coordinate axes, because $\mathbf{XQ} = \mathbf{P\Phi}$, with orthogonal columns of maximal norm (Gower, 1966). Consider a case of rotating the coordinate axes $\mathbf{x}_1$ and $\mathbf{x}_2$ in Figure 7.1 to principal axes. We begin with the given coordinates

$$\mathbf{X} = \begin{bmatrix} 1 & 2 \\ 3 & 1 \\ 2 & 0 \end{bmatrix}.$$

Using the $\mathbf{Q}$ from (7.16) we have

$$\mathbf{XQ} = \begin{bmatrix} 1 & 2 \\ 3 & 1 \\ 2 & 0 \end{bmatrix} \begin{bmatrix} .91 & -.41 \\ .41 & .91 \end{bmatrix} = \begin{bmatrix} 1.73 & 1.42 \\ 3.15 & -.31 \\ 1.83 & -.81 \end{bmatrix}. \tag{7.31}$$

How does $\mathbf{Q}$ rotate the plane? The answer is found by comparing $\mathbf{Q}$ with the symbolic rotation matrix in formula (7.30). Because .91 corresponds to $\cos(\alpha)$, the rotation angle α is $\arccos(.91) = 24°$. The same α results, for example, from $\arcsin^{-1}(.41)$. Hence, $\mathbf{Q}$ rotates $\mathbf{X}$ by 24° in the positive sense, that is, anticlockwise.

If we compare the coordinates in $\mathbf{XQ}$ of (7.31) with those in Figure 7.3, we note that $\mathbf{XQ} = \mathbf{X}^*$ does indeed contain the PA coordinates of the points. The squared coordinates on $\mathbf{x}_1^*$ now sum to $1.73^2 + 3.15^2 + 1.83^2 = 16.26$. This sum is not only greater than the corresponding sum on $\mathbf{x}_1$ ($1^2 + 3^2 + 2^2 = 14$), but is also the maximum possible for any coordinate axis.

7.11 Exercises

Exercise 7.1 The following exercises cast some additional light on the symmetry and asymmetry of a matrix.

(a) Compute $\mathbf{A} = 0.5(\mathbf{M} + \mathbf{M}')$ and $\mathbf{B} = 0.5(\mathbf{M} - \mathbf{M}')$ for the upper left-hand corner submatrix A, ..., G in Table 4.2.

(b) A square matrix $\mathbf{M}$ is called *skew-symmetric* if $\mathbf{M}' = -\mathbf{M}$. Show that $\mathbf{B} = 0.5(\mathbf{M} - \mathbf{M}')$ is skew-symmetric.

(c) Show that $\mathbf{M} = \mathbf{A} + \mathbf{B}$.

(d) Characterize the decomposition of $\mathbf{M}$ into $\mathbf{A}$ and $\mathbf{B}$ in words. Into what two components is $\mathbf{M}$ decomposed here?

Exercise 7.2 Specify the 2×2 matrix $\mathbf{T}$ that effects a counterclockwise rotation of the 2D plane through an angle of 45 degrees.

Exercise 7.3 The square of a matrix $\mathbf{M}$ is defined by $\mathbf{M}^2 = \mathbf{MM}$.

(a) What properties must $\mathbf{M}$ possess so that $\mathbf{M}^2$ exists?

(b) Assume $\mathbf{T}$ is a rotation matrix. Characterize what $\mathbf{T}^2$ means geometrically.

(c) If $\mathbf{Q}$ is orthogonal, is the same true of $\mathbf{Q}^3$?

Exercise 7.4 Find all 3×3 orthogonal matrices whose entries are zeros and ones.

Exercise 7.5 Use a computer package that does matrix algebra, for example, MatLab, S-plus, R, and Ox. (Note that some statistics packages such as SPSS and SAS can also do matrix algebra.)

(a) Find the pseudoinverse of $\mathbf{A} = \begin{bmatrix} 3 & 2 \end{bmatrix}$ through the SVD components of $\mathbf{A}$.

(b) Find the pseudoinverses for $\mathbf{A}_1$, $\mathbf{A}_2$, and $\mathbf{A}_3$ in (7.23), (7.24), and (7.25).

Exercise 7.6 What 2×2 matrix projects the X–Y plane onto the X-axis?

Exercise 7.7 Let $\mathbf{A} = \begin{bmatrix} 1 & x \\ y & -1 \end{bmatrix}$. Specify x and y so that

(a) $\mathbf{A}\mathbf{A}'$ is symmetric;

(b) $\mathbf{A}\mathbf{A}'$ is skew-symmetric;

(c) $\mathbf{A}$ is orthogonal.

Exercise 7.8 Define $\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 0 & 1 \\ -5 & -2 & 6 \end{bmatrix}$ and $\mathbf{B} = \begin{bmatrix} 7 & -4 & 0 \\ 3 & 2 & 1 \\ 1 & -1 & 6 \end{bmatrix}$.

(a) Compute $(\mathbf{A} + \mathbf{B})^2$.

(b) Compute $(\mathbf{A} + \mathbf{B})(\mathbf{A} - \mathbf{B})$.

Exercise 7.9 Construct a 2×2 matrix with nonzero entries that does not have an inverse.

Exercise 7.10 Find 2×2 matrices $\mathbf{A}$ and $\mathbf{B}$, both unequal to the null matrix $\mathbf{0}$, so that $\mathbf{A}^2 + \mathbf{B}^2 = \mathbf{0}$.

Exercise 7.11 Find 2×2 matrices $\mathbf{A}$ and $\mathbf{B}$ with nonzero entries so that $\mathbf{A}\mathbf{B} = \mathbf{0}$.

Exercise 7.12 Suppose that $\mathbf{X}$ is a matrix in which the third column is equal to twice the first column. Show that the same must be true for any product $\mathbf{Y}\mathbf{X}$.

Exercise 7.13 Let $\mathbf{X}$ be a 3×2 matrix. Try a few cases and demonstrate that $\operatorname{tr} \mathbf{X}'\mathbf{X} = \operatorname{tr} \mathbf{X}\mathbf{X}'$. Show that this property holds in general.

Exercise 7.14 Consider the matrices $\mathbf{A}$ and $\mathbf{B}$ of Exercise 7.8.

(a) Find the eigenvalues and eigenvectors of $\mathbf{AA}'$, $\mathbf{A}'\mathbf{A}$, $\mathbf{BB}'$, and $\mathbf{B}'\mathbf{B}$.

(b) Verify that the trace of these four matrix products is equal to the sum of the respective eigenvalues.

(c) Explain what the traces $\mathbf{A}'\mathbf{A}$ and $\mathbf{B}'\mathbf{B}$ represent geometrically. (Hint: What do the elements in the main diagonal of these product matrices represent? They are measures of what?)

Exercise 7.15 Consider the equation (7.23).

(a) Interpret this equation geometrically in terms of image vectors, preimage vectors, and transformations. What vectors are mapped here onto what images? What affects the mapping?

(b) Can you decompose the transformations into a set of more basic transformations?

Exercise 7.16 For matrix $\mathbf{B}$ of equation (7.28), use the power method with at least five iterations to find the dominant eigenvalue.

Exercise 7.17 Consider matrix $\mathbf{A}_2$ of equation (7.24). How many nonzero eigenvalues exist for $\mathbf{A}_2\mathbf{A}_2'$? Why? (You don't have to do any computations.)

Exercise 7.18 Consider the matrices

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \\ 3 & 4 & 5 \\ 4 & 5 & 6 \end{bmatrix} \text{ and } \mathbf{B} = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 6 \\ 3 & 6 & 9 \\ 4 & 8 & 12 \end{bmatrix}.$$

(a) Plot the first two coordinates of each row of $\mathbf{A}$ and $\mathbf{B}$ as vectors in the X–Y plane.

(b) Find the ranks of $\mathbf{A}$ and of $\mathbf{B}$. Explain why the rank of $\mathbf{A}$ is not equal to 1, even though the second and the third column of $\mathbf{A}$ can be generated from the first column by $\mathbf{a}_2 = \mathbf{a}_1 + 1 \cdot \mathbf{1}$ and by $\mathbf{a}_3 = \mathbf{a}_1 + 2 \cdot \mathbf{1}$, respectively.

(c) Find the linear combinations that generate the third column from the first two columns of $\mathbf{A}$ and of $\mathbf{B}$, respectively.

Exercise 7.19 Matrix $\mathbf{B}$ below is a permutation of matrix $\mathbf{A}$. Therefore, there exists a row permutation matrix $\mathbf{P}$ and a column permutation matrix $\mathbf{Q}$ such that $\mathbf{B} = \mathbf{PAQ}$. Note that any permutation matrix $\mathbf{P}$ has in each

row and column a single value of one and all other values zero. Find the permutation matrices that turn $\mathbf{B}$ back into $\mathbf{A}$. (Hint: Build the desired permutation matrices as products of elementary permutation matrices. You get the permutation matrix $\mathbf{P}$ that exchanges rows i and j of $\mathbf{X}$ in $\mathbf{PX}$ by exchanging columns i and j of an identity matrix.)

$$\mathbf{A} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \quad \text{and} \quad \mathbf{B} = \begin{bmatrix} e & f & d \\ h & i & g \\ b & c & a \end{bmatrix}.$$

Exercise 7.20 Show that $(\mathbf{AB})' = \mathbf{B}'\mathbf{A}'$.

Exercise 7.21 Demonstrate that $(\mathbf{AB})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1}$ for the matrices $\mathbf{A}$ and $\mathbf{B}$ in Exercise 7.8.

Exercise 7.22 Consider the matrices $\mathbf{A}$ and $\mathbf{B}$ in Exercise 7.8.

(a) Normalize the column vectors of $\mathbf{A}$ and $\mathbf{B}$ numerically.

(b) Express this normalization in matrix notation.

Exercise 7.23 Consider the matrices $\mathbf{A}$ in Exercise 7.8.

(a) Compute the correlation matrix of the column vectors of matrix $\mathbf{A}$.

(b) Express the operations that generate this correlation matrix in matrix notation.

(c) Spectrally decompose the correlation matrix as in (7.11).

(d) Specify what sum-of-squares is accounted for by each component.

(e) Check whether the correlation matrix is positive semidefinite or positive definite.

Exercise 7.24 Compute the distances among the rows of matrix $\mathbf{A}$ of Exercise 7.18 by using formula 7.5.

Exercise 7.25 Consider Figure 7.3.

(a) The coordinate axes in this plot are almost optimal in terms of simple structure. Explain why.

(b) The best simple structure orientation of the plane leads to

$$\mathbf{X}^* = \begin{bmatrix} 0.60 & 2.15 \\ 2.76 & 1.56 \\ 1.96 & 0.38 \end{bmatrix}.$$

Show that $\mathbf{X}^*$ more closely satisfies the simple structure criterion than the point coordinates of both the system spanned by $\mathbf{x}_1$ and $\mathbf{x}_2$, and the system of principal axes in Figure 7.3.

(c) Find the rotation matrix that turns the system spanned by $\mathbf{x}_1$ and $\mathbf{x}_2$ so that $\mathbf{X}^*$ results.

Exercise 7.26 Prove that the least-squares solution for $\mathbf{x}$ in the equation system $\mathbf{Ax} = \mathbf{b}$ coincides with the one and only solution for $\mathbf{x}$ if $\mathbf{A}$ is invertible. (Hint: Use theorems of Table 7.2 to simplify the regression projector.)

Exercise 7.27 Find the solution vector $\mathbf{x}$ in the equation system (7.18) by

(a) inverting $\mathbf{A}$;

(b) by solving $\mathbf{Ax} = \mathbf{b}$ as if it were a regression problem with the unknown $\mathbf{x}$, that is, by determining the least-squares solution for $\mathbf{x}$;

(c) by solving the system using the generalized inverse based on the SVD of $\mathbf{A}$.

(d) Discuss the findings.

Exercise 7.28 Assume you have five vectors with four elements each. What can you conclude about their linear dependency?

Exercise 7.29 Let $\mathbf{P}$ be a projector.

(a) Show that $\mathbf{PP} = \mathbf{P}$ (idempotency).

(b) Explain why a projector is idempotent by geometric arguments.

Exercise 7.30 Consider the picture compression problem illustrated in Figure 7.2 on page 154. If you have MatLab, you can easily replicate this example with a few commands. The data for this picture are provided by MatLab under the name "clown.mat". Hence, all you need to do is type the commands

```
load clown          % Load matrix X with pixel codes
image(X)            % Display original picture
[U,S,V]=svd(X);     % SVD of the 200-by-320 pixel matrix
k=10;               % Set compression factor k
image(U(:,1:k)*S(1:k,1:k)*V(:,1:k)') % Approximate picture
colormap(gray)      % Set image to grayscale
```

If you do not have MatLab, download the data from our website and do the exercise in your favorite matrix language.

(a) Test out the performance of some additional ks.

(b) Determine the compression rate accomplished by choosing k SVD components rather than the original matrix of pixels. (Hint: The original matrix contains 64,000 pieces of information; the rank-reduced matrix contains as many pieces as there are elements in its SVD components.)

(c) Try the above problem in color by inserting `colormap(map)` after `image(X)` in the above set of commands.

(d) Measure objectively how well the various matrices of rank k "explain" the original data.

(e) Attempt an interpretation of the SVD approximations. What information is picked up first?

8

A Majorization Algorithm for Solving MDS

An elegant algorithm for computing an MDS solution is discussed in this chapter. We reintroduce the Stress function that measures the deviance of the distances between points in a geometric space and their corresponding dissimilarities. Then, we focus on how a function can be minimized. An easy and powerful minimization strategy is the principle of minimizing a function by iterative majorization. An intuitive explanation for iterative majorization in MDS is given using a simplified example. Then, the method is applied in the SMACOF algorithm for minimizing Stress.

8.1 The Stress Function for MDS

We now place the concepts introduced into a common framework to allow the derivation of mathematically justifiable rather than just intuitively plausible methods for solving the MDS construction problem. The methods can then be extended and generalized to MDS models not considered so far. We need the following six basic definitions, most of which have been introduced before.

D1 n denotes the number of empirical objects (stimuli, variables, items, questions, and so on, depending on the context).

D2 If an observation has been made for a pair of objects, i and j, a proximity value p_{ij} is given. If p_{ij} is undefined, we speak of a *missing value*. The term *proximity* is used in a generic way to denote

both similarity and dissimilarity values. For similarities, a high p_{ij} indicates that the objects i and j are similar.

D3 A *dissimilarity* is a proximity that indicates how dissimilar two objects are. A small score indicates that the objects are similar, a high score that they are dissimilar. A dissimilarity is denoted by δ_{ij}.

D4 $\mathbf{X}$ denotes (a) a point configuration (i.e., a set of n points in m-dimensional space) and (b) the $n \times m$ matrix of the coordinates of the n points relative to m Cartesian coordinate axes. A Cartesian coordinate system is a set of pairwise perpendicular straight lines (coordinate axes). All axes intersect at one point, the *origin*, O. The coordinate of a point on axis a is the directed (signed) distance of the point's perpendicular projection onto axis a from the origin. The m-tuple $(x_{i1}, \ldots, x_{im})$ denotes the coordinates of point i with respect to axes $a = 1, \ldots, m$. The origin has the coordinates $(0, \ldots, 0)$.

D5 The Euclidean distance between any two points i and j in $\mathbf{X}$ is the length of a straight line connecting points i and j in $\mathbf{X}$. It is computed by the value resulting from the formula $d_{ij} = [\sum_{a=1}^{m}(x_{ia} - x_{ja})^2]^{1/2}$, where x_{ia} is the coordinate of point i relative to axis a of the Cartesian coordinate system. We also use $d_{ij}(\mathbf{X})$ for the distance to show explicitly that the distance is a function of the coordinates $\mathbf{X}$.

D6 The term $f(p_{ij})$ denotes a mapping of p_{ij}, that is, the number assigned to p_{ij} according to rule f. This is sometimes written as $f : p_{ij} \mapsto f(p_{ij})$. We also say that $f(p_{ij})$ is a *transformation* of p_{ij}. (The terms function, transformation, and mapping are synonymous in this context.) Instead of $f(p_{ij})$ we often write $\widehat{d}_{ij}$.

So far, the task of MDS was defined as finding a low-dimensional configuration of points representing objects such that the distance between any two points matches their dissimilarity *as closely as possible*. Of course, we would prefer that each dissimilarity should be mapped exactly into its corresponding distance in the MDS space. But that requires too much, because empirical data always contain some component of error (see, e.g., Section 3.2). We define an error of representation by

$$e_{ij}^2 = (d_{ij} - \delta_{ij})^2. \tag{8.1}$$

Summing (8.1) over i and j yields the total error (of approximation) of an MDS representation,

$$\sigma_r(\mathbf{X}) = \sum_{i=1}^{n} \sum_{j=i+1}^{n} (d_{ij} - \delta_{ij})^2, \text{ for all available } \delta_{ij}, \tag{8.2}$$

which is often written as

$$\sigma_r(\mathbf{X}) = \sum_{i<j} (d_{ij} - \delta_{ij})^2, \text{ for all available } \delta_{ij}. \tag{8.3}$$

The relation $i < j$ in (8.3) simply says that it is sufficient, in general, to sum over half of the data, because dissimilarities and distances are symmetric.

What does "for all available δ_{ij}" mean? In practical research, we sometimes have *missing values*, so that some δ_{ij} are undefined. Missing values impose no restriction on any distances in $\mathbf{X}$. Therefore, we define fixed weights w_{ij} with value 1 if δ_{ij} is known and $w_{ij} = 0$ if δ_{ij} is missing. Other values of w_{ij} are also allowed, as long as $w_{ij} \geq 0$. This defines the final version of *raw Stress* (Kruskal, 1964b),

$$\sigma_r(\mathbf{X}) = \sum_{i<j} w_{ij}(d_{ij}(\mathbf{X}) - \delta_{ij})^2. \tag{8.4}$$

We use the notations σ_r and $\sigma_r(\mathbf{X})$ interchangeably to denote raw Stress.

For every set of coordinates $\mathbf{X}$, a Stress value can be computed. Clearly, we do not want just any $\mathbf{X}$, but we want to find an $\mathbf{X}$ such that the errors (8.1) are small or even zero. Mathematically spoken, we want to minimize $\sigma_r(\mathbf{X})$ over $\mathbf{X}$. For that purpose, we first introduce the concept of differentiating a function, which is explained in the next section.

8.2 Mathematical Excursus: Differentiation

Our aim is to find a coordinate matrix $\mathbf{X}$ such that $\sigma_r(\mathbf{X})$ is minimal. This is a rather complex problem because it requires us to pick $n \cdot m$ coordinates optimally with respect to the Stress function. Therefore, we start by looking at a more simple problem, that is, finding the minimum of a function $f(x)$ with one variable x only. This requires some notions of differential calculus. Consider an example. Let y be the dependent variable and x the independent variable in the function

$$f(x) = y = .3x^4 - 2x^3 + 3x^2 + 5, \tag{8.5}$$

and find the x value for which y attains its smallest value. A first rough estimate of the solution can be derived by looking at some points from the graph of this function, that is, points with the coordinates $(x, f(x))$ in a Cartesian coordinate system. A set of such points can be easily found by choosing some x values, plugging them into the right-hand side of (8.5), and solving for y. If we compute the coordinates of some such points on the graph, we arrive at Figure 8.1 and, with more and more points, at Figure 8.2.

It is clear that point E in Figure 8.2 represents the solution of the minimization problem. For $x = 3.6$ the smallest y value of function (8.5) is obtained: $y = 0.96$. However, point B has, in a sense, the same properties as E, provided we consider a limited interval of x values only, such as only those x values to the left of C. B is called a *local minimum* of the function,

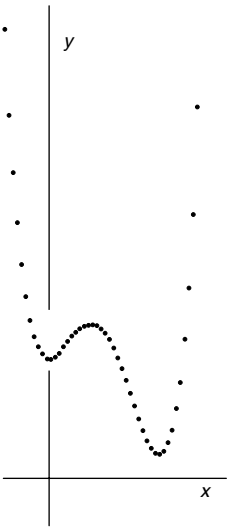


FIGURE 8.1. Some points for $y = 0.3x^4 - 2x^3 + 3x^2 + 5$.

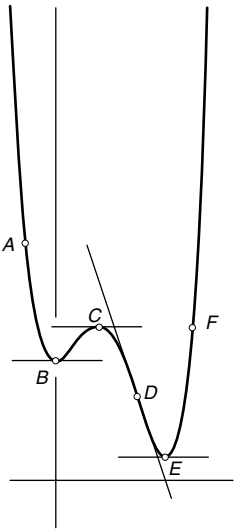


FIGURE 8.2. Graph of $y = 0.3x^4 - 2x^3 + 3x^2 + 5$, with tangent lines at points B, C, D, and E.

and E is the *global minimum*. Analogously, C is a local maximum. Function $f(x)$ has no global maximum.

If we determine the tangents for each point on the graph, it becomes evident that they are horizontal lines at the extrema of the displayed portion of the graph. Figure 8.2 shows this for the minima B and E, and the maximum C. The tangents for other points are not horizontal; that is, their slopes are not zero. This is a property that distinguishes extrema from other points and can be used to find extrema by computation rather than by inspection. If we know all of the extrema, we can select the point with the smallest y-coordinate.

The Slope of a Function

What exactly is a tangent and its slope? Consider Figure 8.3, where the points P and Q are distinguished on the graph for $y = f(x)$. P and Q have the coordinates (x_P, y_P) and (x_Q, y_Q), respectively, or, because $y = f(x)$, $(x_P, f(x_P))$ and $(x_Q, f(x_Q))$, respectively. The straight line through P and Q has the slope

$$\text{slope}(PQ) = \frac{y_Q - y_P}{x_Q - x_P}. \tag{8.6}$$

We now set $x_Q - x_P = \Delta x$. Then (8.6) can be written as

$$\text{slope}(PQ) = \frac{f(x_P + \Delta x) - f(x_P)}{\Delta x}, \tag{8.7}$$

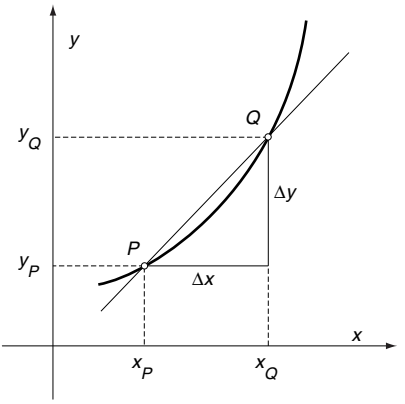


FIGURE 8.3. Some notions for finding tangent line at P.

or, more generally, for any point $P = (x, f(x))$,

$$\text{slope}(PQ) = \frac{f(x + \Delta x) - f(x)}{\Delta x}. \tag{8.8}$$

To find the tangent at point P on the graph, it is necessary to move Q very close to P. However, Q should not become equal to P, because we need two points to uniquely identify the tangent line. This is expressed as follows:

$$\frac{dy}{dx} = \lim_{\Delta x \to 0} \frac{f(x + \Delta x) - f(x)}{\Delta x}, \tag{8.9}$$

where $\lim_{\Delta x \to 0}$ is the *limit operator*. The limit operator makes the difference term Δx in the function $[f(x+\Delta x)-f(x)]/\Delta x$ smaller and smaller, so that Δx approaches 0 without ever reaching it. We say that Δx is made *arbitrarily* or *infinitesimally* small. The symbol dy/dx denotes the resulting *limit* of this operation. Note carefully that the limit dy/dx is *not* generated by setting $\Delta x = 0$, but by approximating $\Delta x = 0$ arbitrarily closely. [Setting $\Delta x = 0$ would turn the right-hand side of (8.9) into 0/0.]

Equations (8.8) and (8.9) are formulated for any point P, not just the particular one in Figure 8.3. Hence, by choosing different Ps, a function of the respective limits is obtained, that is, a function giving the slope of the tangents or the *growth rate* of y relative to x at each point P. This function is called the *derivative* of $y = f(x)$, usually denoted by y'. To illustrate this, let $y = x^2$. The derivative of $y = x^2$ can be found by considering the slope of the tangent at point P:

$$\begin{aligned} \frac{dy}{dx} &= \lim_{\Delta x \to 0} \frac{(x + \Delta x)^2 - (x)^2}{\Delta x} \\ &= \lim_{\Delta x \to 0} \frac{x^2 + (\Delta x)^2 + 2x\Delta x - x^2}{\Delta x} \end{aligned}$$

$$
\begin{aligned}
&= \lim_{\Delta x \to 0} \left(\frac{(\Delta x)^2}{\Delta x} + \frac{2x\Delta x}{\Delta x} \right) \\
&= \lim_{\Delta x \to 0} (\Delta x + 2x) \\
&= \lim_{\Delta x \to 0} (\Delta x) + \lim_{\Delta x \to 0} (2x) = 2x. \qquad (8.10)
\end{aligned}
$$

Because x is not restricted to a particular point P, we have established a function that gives the slope of $y = x^2$ for any x-value. Hence, $y' = 2x$; that is, the slope of the tangent at each point is simply twice its x-coordinate. For $x = 5$, say, we obtain the slope $dy/dx = 10$, which means that $y = x^2$ grows at this point at the rate of 10 y-units per 1 x-unit (compare Figure 8.3). We can check whether these derivations are correct by setting $x = 5$ and $\Delta x = 3$, say, and then making Δx ever smaller; the smaller Δx gets, the more the limiting value $y' = 10$ is approximated.

Finding the Minimum of a Function

The slope at the minimum must be equal to 0. The derivative gives us an expression for the slope, and thus we can find a minimum by checking all points where the derivative is zero. Points with a zero derivative are called *stationary points*. Given the derivative $y' = 2x$, we can find the minimum of $y = x^2$. We first set $y' = 2x = 0$. But $2x = 0$ only if $x = 0$. So we know that $y = x^2$ has a tangent with slope 0 at $x = 0$. Whether this is a minimum can be checked by looking at the graph of the function. Alternatively, we can compute what the function yields at two[1] neighboring points at $x = 0$. For $x_1 = 1$ and $x_2 = -1$, say, we determine $y_1 = 1^2 = 1$ and $y^2 = (-1)^2 = 1$, respectively, both values greater than the y at $x = 0$, which indicates that we have found a minimum at $x = 0$.

The method of setting the derivative of a function equal to zero and then finding the values that solve this equation has identified only one point. This turned out to be a minimum. We might ask where the maxima are. They can be found by considering the bounds of the interval that x should cover. If we do not restrict x, then these bounds are $-\infty$ and $+\infty$, and this is where the maxima are, as we can see by inserting larger and larger x values into $y = x^2$. Therefore, we also must always test the bounds of the x-interval in which we are interested.

Just as we did in equations (8.10) for the function $y = x^2$, we can find the derivative for any other (continuous and smooth) function. Because *differentiation* (i.e., finding the derivative) is useful in many fields of math-

[1] We test two rather than just one neighboring point at $x = 0$ because the tangent has a zero slope not only at extreme points but also in other cases. Consider, for example, a function that first increases, then runs on a plateau, and then increases again. For all of the points on the plateau, the function has a zero slope. Thus, the zero slope condition for stationarity is *only necessary, but not sufficient*, for identifying an extremum.

TABLE 8.1. Some rules of differentiation.

Rule	Function	Derivative
1	$y = \text{constant} = a$	$dy/dx = 0$
2	$y = x$	$dy/dx = 1$
3	$y = a \cdot x$	$dy/dx = a$
4	$y = a \cdot x^n$	$dy/dx = a \cdot n \cdot x^{n-1}$
5	$y = e^x$	$dy/dx = e^x$
6	$y = \sin(x)$	$dy/dx = \cos(x)$
7	$y = \cos(x)$	$dy/dx = -\sin(x)$
Let $u = f(x)$ and $v = h(x)$ be functions of x. Then:		
8	$y = u + v$	$dy/dx = du/dx + dv/dx$
9	$y = u \cdot v$	$dy/dx = u(dv/dx) + v(du/dx)$
10	$y = u/v$	$dy/dx = [v(du/dx) - u(dv/dx)]/v^2$
Let $y = f(z)$ and $z = g(x)$. Then (*chain rule*):		
11	$y = f(g(x))$	$dy/dx = (dy/dz) \cdot (dz/dx)$

ematics, rules have been derived that greatly simplify finding y'. Some such rules are summarized in Table 8.1. Some of them are patent; others are explained later when we need them. For the example above, $y = x^2$, we find y' by applying rule 4: $y' = dy/dx = 1 \cdot 2 \cdot x^{2-1} = 2x$. For (8.5) we find by rules 1, 4, and 8: $dy/dx = (0.3)(4)x^3 - (2)(3)x^2 + (3)(2)x = 1.2x^3 - 6x^2 + 6x$. Setting this derivative equal to 0 yields the equation $1.2x^3 - 6x^2 + 6x = 0$. After factoring, we have $(x)(1.2x^2 - 6x + 6) = 0$. So, the sought x-values result from the equations $x = 0$ and $1.2x^2 - 6x + 6 = 0$. We find $x_1 = 0$ as one solution, which we identify immediately as a local minimum in the graph in Figure 8.2. The quadratic equation yields $x_2 = 3.618$ and $x_3 = 1.382$ for the other solutions. They correspond to points B and E in the graph.

Second- and Higher-Order Derivatives

The derivative of a function $y = f(x)$ is itself a function of x, $y'' = f'(x)$. One therefore can ask for the derivative of y', $y'' = f''(x)$, the derivative of y'', and so on. The second derivative, y'', indicates the rate of change of the rate of change of $f(x)$. For example, for $y = x^3$ we get $y' = 3x^2$. That is, at any point x, the cubic function grows by the factor $3x^2$. Now, differentiating $y' = 3x^2$ with respect to x (using rule 4 in Table 8.1), we get $y'' = 3.2x$. This means that the rate of change of the growth rate also depends on x: it is 6 times the value of x. So, with large x values, the growth of x^3 "accelerates" quite a bit. As a second example, the rate of change of the growth rate of $y = \sqrt{x} = x^{1/2}, x > 0$, is $y'' = f'(1/2 \cdot x^{-1/2}) = (-1/4) \cdot x^{-3/2} = -1/(4\sqrt{x^3})$. So, y' shows that this function has a positive slope at any point x, and y'' indicates that this slope decreases as x becomes

larger. Another way of saying this is that $y = \sqrt{x}$ is concave downwards, whereas $y = x^3$ is convex downwards.

The second derivative is useful to answer the question of whether a stationary point is a minimum or a maximum. Consider Figure 8.2, where we have three stationary points: B, C, and E. C differs from B and E because the speed of growth of $f(x)$ is continuously shrinking when we approach C from the left. To the right of C, the growth rate of $f(x)$ is even negative ("decline"), and becomes more negative as a function of x. The opposite is true for points B and E. This means that if $y'' < 0$ at some stationary point x, then x is a maximum; if $y'' > 0$, x is a minimum. Thus, for the function in Figure 8.2, we have $y'' = 3.6x^2 - 12x + 6$, so that at $x = 0$ (stationary point B) we have $y'' = 6$, for example. Because $6 > 0$, B is a minimum. For $x = 1.382$ (point C), we get -3.708, so that this point is a maximum by the second derivative test.

8.3 Partial Derivatives and Matrix Traces

We often deal with functions that have more than one variable. Such functions are called functions with several variables, multivariable functions, vector functions, or functions with many arguments. An example of such a function is raw Stress, $\sigma_r(\mathbf{X})$. Because we attempt to minimize this function over every single one of its $n \cdot m$ coordinates, we naturally encounter the question of how to find the derivative of multivariable functions. The answer is simple: such functions have as many derivatives as they have arguments, and the derivative for each argument x_i is found by holding all other variables fixed and differentiating the function with respect to x_i as usual. For example, the derivative of the function $f(x, y, z) = x^2y + y^2z + z^2x$ with respect to variable y is $x^2 + 2yz$, using rules 4 and 8 of Table 8.1 and treating the term z^2x as a "constant" (i.e., as not dependent on y). The derivative to one argument of a function of several variables is called the *partial derivative*. The vector of partial derivatives is called the *gradient* vector.

In the following, we focus on one particular multivariable function that becomes important in much of the remainder of this book, the *trace* function, $\text{tr } \mathbf{A} = \sum_{i=1}^{n} a_{ii}$ discussed earlier in Section 7.2 and Table 7.4. The trace can be used to simplify expressing a multiargument linear function such as $f(x_{11}, \ldots, x_{ik}, \ldots, x_{nn}) = \sum_{k=1}^{n} \sum_{i=1}^{n} a_{ki} x_{ik}$, where the a_{ik} terms denote constants and x_{ik} are variables:

$$\sum_{k=1}^{n} \sum_{i=1}^{n} a_{ki} x_{ik} = \text{tr } \mathbf{AX} = f(\mathbf{X}).$$

Here, the constants are collected in the matrix $\mathbf{A}$, the variables in $\mathbf{X}$ (see, e.g., Table 8.2 for an example). Suppose that we want to find the partial

TABLE 8.2. Example of differentiating the linear function tr $\mathbf{AX}$ with respect to an unknown matrix $\mathbf{X}$.

(1)	$\mathbf{AX} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix}$
(2)	$f(\mathbf{X}) = \text{tr } (\mathbf{AX}) = a_{11}x_{11} + a_{12}x_{21} + a_{21}x_{12} + a_{22}x_{22}$
(3)	$\partial f(\mathbf{X})/\partial \mathbf{X} = (\partial f(\mathbf{X})/\partial x_{ij})$
(4)	$\begin{bmatrix} \partial f(\mathbf{X})/\partial x_{11} = a_{11} & \partial f(\mathbf{X})/\partial x_{12} = a_{21} \\ \partial f(\mathbf{X})/\partial x_{21} = a_{12} & \partial f(\mathbf{X})/\partial x_{22} = a_{22} \end{bmatrix} = \mathbf{A}'$
(5)	rule: $\partial \text{tr } (\mathbf{AX})/\partial \mathbf{X} = \mathbf{A}'$

derivative of the linear function $f(\mathbf{X})$ with respect to the matrix $\mathbf{X}$. The partial derivative of $f(\mathbf{X})$ with respect to $\mathbf{X}$ is the matrix consisting of the derivatives of $f(\mathbf{X})$ with respect to each element of $\mathbf{X}$ (i.e., the matrix with elements $\partial f(\mathbf{X})/\partial x_{ik}$). The notation $\partial f(\mathbf{X})/\partial x_{ik}$ denotes the partial derivative. It replaces $df(\mathbf{X})/dx_{ik}$ used previously in Section 8.2 to make clear that we are dealing with a multivariable function f rather than with a function of just one variable, as in Section 8.2. All variables except x_{ik} are considered constant in $\partial f(\mathbf{X})/\partial x_{ik}$. The matrix of partial derivatives is also denoted by $\nabla f(\mathbf{X})$, by $\nabla \text{tr } \mathbf{AX}$, or by $\partial \text{tr } \mathbf{AX}/\partial \mathbf{X}$.

To find $\nabla f(\mathbf{X})$, we have to take the first derivative of $f(\mathbf{X})$ with respect to every x_{ik} separately. That is, $\partial \text{tr } \mathbf{AX}/\partial x_{ik} = a_{ki}$, so that $\partial \text{tr } \mathbf{AX}/\partial \mathbf{X} = \mathbf{A}'$. The steps needed to find the partial derivative of tr $\mathbf{AX}$ are illustrated in Table 8.2. (For properties of matrix traces, see Table 7.4.) More rules for differentiating a matrix trace function are presented in Table 8.3.

Matrix traces are also useful for expressing a quadratic function such as

$$\sum_{i=1}^{n}\sum_{k=1}^{m} x_{ik}^2 = \text{tr } \mathbf{X}'\mathbf{X}.$$

Because tr $(\mathbf{XX}')$ is equal to $\sum_k \sum_i x_{ki}^2$, tr $\mathbf{X}'\mathbf{X}$ = tr $\mathbf{XX}'$. Hence, the gradient of tr $\mathbf{X}'\mathbf{X}$ is equal to $2\mathbf{X}$ by rule 4, Table 8.3, setting $\mathbf{A} = \mathbf{I}$.

As another example, assume that we want to minimize

$$\begin{aligned} f(\mathbf{X}) &= \text{tr } (\mathbf{X} - \mathbf{Z})'(\mathbf{X} - \mathbf{Z}) \\ &= \sum_{i=1}^{n}\sum_{k=1}^{m}(x_{ik} - z_{ik})^2 \end{aligned}$$

by an appropriate choice of $\mathbf{X}$. We solve this problem formally by first finding the gradient $\nabla f(\mathbf{X})$ and then setting $\nabla f(\mathbf{X}) = \mathbf{0}$ and solving for

TABLE 8.3. Some rules for differentiating a matrix trace with respect to an unknown matrix $\mathbf{X}$; matrix $\mathbf{A}$ is a constant matrix; matrices $\mathbf{U}$, $\mathbf{V}$, $\mathbf{W}$ are functions of $\mathbf{X}$ (Schönemann, 1985).

(1)	$\partial \text{tr } (\mathbf{A})/\partial \mathbf{X} = \mathbf{0}$
(2)	$\partial \text{tr } (\mathbf{AX})/\partial \mathbf{X} = \mathbf{A}' = \partial \text{tr } [(\mathbf{AX})']/\partial \mathbf{X}$
(3)	$\partial \text{tr } (\mathbf{X}'\mathbf{AX})/\partial \mathbf{X} = (\mathbf{A} + \mathbf{A}')\mathbf{X}$
(4)	$\partial \text{tr } (\mathbf{X}'\mathbf{AX})/\partial \mathbf{X} = 2\mathbf{AX}$ if $\mathbf{A}$ is symmetric
(5)	$\partial \text{tr } (\mathbf{U} + \mathbf{V})/\partial \mathbf{X} = \partial \text{tr } (\mathbf{U})/\partial \mathbf{X} + \partial \text{tr } (\mathbf{V})/\partial \mathbf{X}$
(6)	$\partial \text{tr } (\mathbf{UVW})/\partial \mathbf{X} = \partial \text{tr } (\mathbf{WUV})/\partial \mathbf{X} = \partial \text{tr } (\mathbf{VWU})/\partial \mathbf{X}$ Invariance under "cyclic" permutations
(7)	$\partial \text{tr } (\mathbf{UV})/\partial \mathbf{X} = \partial \text{tr } (\mathbf{U}_c\mathbf{V})/\partial \mathbf{X} + \partial \text{tr } (\mathbf{UV}_c)/\partial \mathbf{X}$ Product rule: $\mathbf{U}_c$ and $\mathbf{V}_c$ is taken as a constant matrix when differentiating

$\mathbf{X}$. The gradient can be obtained as follows. If we expand $f(\mathbf{X})$, we get

$$f(\mathbf{X}) = \text{tr } \mathbf{X}'\mathbf{X} + \text{tr } \mathbf{Z}'\mathbf{Z} - 2\text{tr } \mathbf{X}'\mathbf{Z},$$

and, by using the rules from Table 7.4,

$$\begin{aligned} \nabla f(\mathbf{X}) &= \nabla \text{tr } \mathbf{X}'\mathbf{X} + \nabla \text{tr } \mathbf{Z}'\mathbf{Z} - \nabla 2\text{tr } \mathbf{X}'\mathbf{Z} \\ &= 2\mathbf{X} + \mathbf{0} - 2\mathbf{Z} = 2\mathbf{X} - 2\mathbf{Z}. \end{aligned}$$

To find the minimum of $f(\mathbf{X})$, its gradient $\nabla f(\mathbf{X}) = 2\mathbf{X} - 2\mathbf{Z}$ must be equal to $\mathbf{0}$, so that $\mathbf{X} = \mathbf{Z}$ at the minimum.

In the sequel, we often make use of trace minimizations. For the difficult problem of minimizing the Stress function, we need an additional minimization method, iterative majorization, which is explained in the next section.

8.4 Minimizing a Function by Iterative Majorization

For finding the minimum of a function $f(x)$, it is not always enough to compute the derivative $f'(x)$, set it equal to zero, and solve for x. Sometimes the derivative is not defined everywhere, or solving the equation $f'(x) = 0$ is simply impossible. For such cases, we have to refer to other mathematical techniques. A useful method consists of trying to get increasingly better estimates of the minimum. We call such a numerical method an *algorithm*. It

consists of a set of computational rules that are usually applied repeatedly, where the previous estimate is used as input for the next cycle of computations which outputs a better estimate. An elegant method is called iterative majorization,[2] which is based on the work of De Leeuw (1977). We first present the main principles of iterative majorization. In the next section, we apply it to the Stress function.

Principles of Majorization

One of the main features of iterative majorization (IM) is that it generates a monotonically nonincreasing sequence of function values. If the function is bounded from below, we usually end up in a stationary point that is a local minimum. An early reference to majorization in the context of line search can be found in Ortega and Rheinboldt (1970, pp. 253–255). Majorization has become increasingly popular as a minimization method; see, for example, Kiers (1990), Bijleveld and De Leeuw (1991), Verboon and Heiser (1992), and Van der Lans (1992). In the field of multidimensional scaling, it has been applied in a variety of settings by, among others, De Leeuw (1977, 1988), De Leeuw and Heiser (1977, 1980), Meulman (1986, 1992), Groenen (1993), Groenen, Mathar, and Heiser (1995), and Groenen, Heiser, and Meulman (1999). Some general papers on iterative majorization are De Leeuw (1994), Heiser (1995), Lange, Hunter, and Yang (2000), Kiers (2002), and Hunter and Lange (2004). Below, we provide an introduction to iterative majorization.

The central idea of the majorization method is to replace iteratively the original complicated function $f(x)$ by an auxiliary function $g(x, z)$, where z in $g(x, z)$ is some fixed value. The function g has to meet the following requirements to call $g(x, z)$ a *majorizing function* of $f(x)$.

- The auxiliary function $g(x, z)$ should be simpler to minimize than $f(x)$. For example, if $g(x, z)$ is a quadratic function in x, then the minimum of $g(x, z)$ over x can be computed in one step (see Section 8.2).
- The original function must always be smaller than or at most equal to the auxiliary function; that is, $f(x) \leq g(x, z)$.
- The auxiliary function should touch the surface at the so-called *supporting point* z; that is, $f(z) = g(z, z)$.

To understand the principle of minimizing a function by majorization, consider the following. Let the minimum of $g(x, z)$ over x be attained at

[2]The term iterative majorization and its abbreviation (IM) was coined by Heiser (1995). Before, the method was called simply majorization. In MDS the method goes back to the work of De Leeuw (1977).

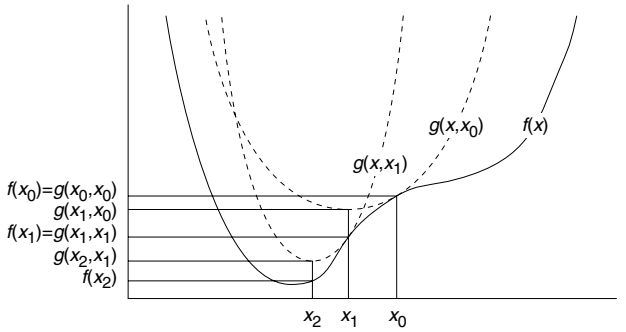


FIGURE 8.4. Illustration of two iterations of the iterative majorization method. The first iteration starts by finding the auxiliary function $g(x, x_0)$, which is located above the original function $f(x)$ and touches at the supporting point x_0. The minimum of the auxiliary function $g(x, x_0)$ is attained at x_1, where $f(x_1)$ can never be larger than $g(x_1, x_0)$. This completes one iteration. The second iteration is analogous to the first iteration.

x^*. The last two requirements of the majorizing function imply the chain of inequalities

$$f(x^*) \leq g(x^*, z) \leq g(z, z) = f(z). \tag{8.11}$$

This chain of inequalities is named the *sandwich* inequality by De Leeuw (1993), because the minimum of the majorizing function $g(x^*, z)$ is squeezed between $f(x^*)$ and $f(z)$. A graphical representation of these inequalities is presented in Figure 8.4 for two subsequent iterations of iterative majorization of the function $f(x)$. The iterative majorization algorithm is given by

1. Set $z = z_0$, where z_0 is a starting value.
2. Find update x^u for which $g(x^u, z) \leq g(z, z)$.
3. If $f(z) - f(x^u) < \varepsilon$, then stop. ($\varepsilon$ is a small positive constant.)
4. Set $z = x^u$ and go to 2.

Obviously, by (8.11) the majorization algorithm yields a nonincreasing sequence of function values, which is an attractive aspect of iterative majorization. If the function $f(x)$ is not bounded from below, and if there are no sufficient restrictions on x, then the stop criterion in step 3 may never be met. In the sequel, this situation does not arise. Although the function value never increases, the majorization principle does not say how fast the function values converge to a minimum. In most applications, an algorithm based on iterative majorization is not very fast. As shown in Section 8.2, a necessary condition for a minimum at point x^* is that the derivative of $f(x)$ at x^* is 0. Using the inequalities of (8.11), this also implies that x^*

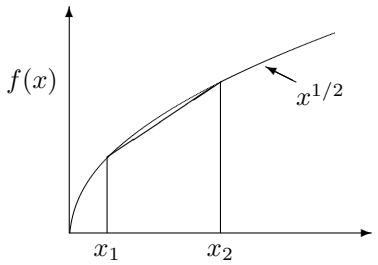


FIGURE 8.5. Graph of the concave function $x^{1/2}$.

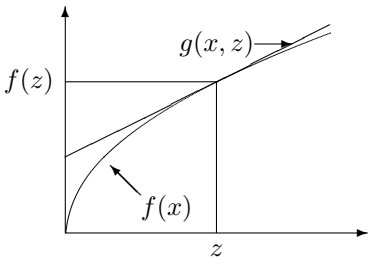


FIGURE 8.6. An example of linear majorization of the concave function $f(x) = x^{1/2}$ by the linear majorizing function $g(x, z)$.

minimizes $g(x, x^*)$ over x, with $g(x^*, x^*)$ as the minimum. Thus, the necessary condition of a zero derivative at a local minimum may be replaced by the weaker condition that $g(x^u, y) = f(y)$ and $x^u = y$. In general, the majorization algorithm can stop at any stationary point, not necessarily at a local minimum. However, Fletcher (1987) notes that, for algorithms that reduce the function value on every iteration, it usually holds that "the stationary point turns out to be a local minimizer, except in rather rare circumstances" (p. 19).

Linear and Quadratic Majorization

We distinguish two particularly useful classes of majorization: linear and quadratic (De Leeuw, 1993). The first one is majorization of a function that is *concave*. A concave function $f(x)$ is characterized by the inequality $f(\alpha x + (1-\alpha)z) \geq \alpha f(x) + (1-\alpha)f(z)$ for $0 \leq \alpha \leq 1$. Thus, the line that connects the function values at $f(x)$ and $f(z)$ remains below the graph of a concave function. An example of the concave function $f(x) = x^{1/2}$ is given in Figure 8.5. But for such a function $f(x)$, it is always possible to have a straight line defined by $g(x, z) = ax + b$ (with a and b dependent on z) such that $g(x, z)$ touches the function $f(x)$ at $x = z$, and elsewhere the line defined by $g(x, z)$ is above the graph of $f(x)$. Clearly, $g(x, z) = ax + b$ is a linear function in x. Therefore, we call this type of majorization *linear majorization*. Any concave function $f(x)$ can be majorized by a linear function $g(x, z)$ at any point z. Thus, $g(x, z)$ satisfies all three requirements of a majorizing function. An example of a linear majorizing function $g(x, z)$ with supporting point z of the concave function $f(x) = x^{1/2}$ is given in Figure 8.6.

The second class of functions that can be easily majorized is characterized by a bounded second derivative. For a function $f(x)$ with a bounded second derivative, there exists a quadratic function that has, compared to $f(x)$, a larger second derivative at any point x. This means that $f(x)$ does not have

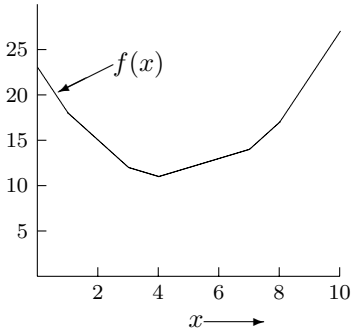


FIGURE 8.7. Graph of the function $f(x) = |x-1| + |x-3| + |x-4| + |x-7| + |x-8|$.

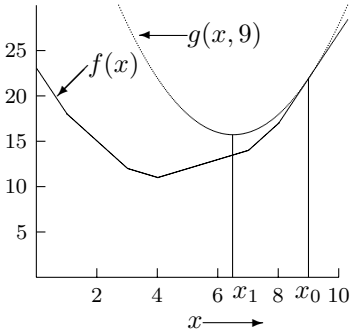


FIGURE 8.8. A quadratic majorizing function $g(x, x_0)$ of $f(x)$ with supporting point $x_0 = 9$.

very steep parts, because there always exists a quadratic function that is steeper. This type of majorization can be applied if the function $f(x)$ can be majorized by $g(x, z) = a(z)x^2 - b(z)x + c(z)$, with $a(z) > 0$, and $a(z)$, $b(z)$, and $c(z)$ functions of z, but not of x. We call this type of majorization *quadratic majorization*.

Example: Majorizing the Median

Heiser (1995) gives an illustrative example of iterative majorization for computing the *median*. The median of the numbers $x_1, x_2, \ldots, x_n$ is the number for which $f(x) = \sum_{i=1}^{n} |x - x_i|$ is a minimum. For example, the median of the numbers $x_1 = 1$, $x_2 = 3$, $x_3 = 4$, $x_4 = 7$, and $x_5 = 8$ is 4. Thus, the median is the value for which 50% of all observations is smaller. The function $f(x)$ is shown in Figure 8.7.

How can we majorize $f(x)$? We begin by noting that $g(x, z) = |z|/2 + x^2/|2z|$ majorizes $|x|$ (Heiser, 1988a). The three majorization requirements are fulfilled by this $g(x, z)$. First, $g(x, z)$ is a simple function because it is quadratic in x. Second, we have $f(x) \leq g(x, z)$ for all x and fixed z. This can be seen by using the inequality $(|x| - |z|)^2 \geq 0$, which always holds, because squares are always nonnegative. Developing this inequality gives

$$\begin{aligned} x^2 + z^2 - 2|x||z| &\geq 0 \\ 2|x||z| &\leq x^2 + z^2 \\ |x| &\leq \frac{1}{2}\frac{x^2}{|z|} + \frac{1}{2}|z|, \end{aligned} \tag{8.12}$$

which proves $|x| \leq g(x, z)$. The third requirement of a majorizing function is that there must be equality in the supporting point; that is, $f(z) = g(z, z)$.

If we substitute $x = z$ in (8.12), we obtain

$$\frac{1}{2}\frac{z^2}{|z|} + \frac{1}{2}|z| = \frac{1}{2}|z| + \frac{1}{2}|z| = |z|,$$

which shows that all three requirements for a majorizing function hold.

$f(x)$ is majorized by replacing x and z in (8.12) by the separate terms in $f(x)$. This means that $|x-1|$ is majorized by $g_1(x,z) \leq |z-1|/2 + (x-1)^2/|2(z-1)|$. Similarly, the second term $|x-3|$ of $f(x)$ is majorized by $g_2(x,z) \leq |z-3|/2 + (x-3)^2/|2(z-3)|$, and so on. Summing the majorization functions for each term in $f(x)$ yields the majorizing function of $f(x)$; that is,

$$\begin{aligned} g(x,z) &= g_1(x,z) + g_2(x,z) + g_3(x,z) + g_4(x,z) + g_5(x,z) \\ &= \frac{1}{2}|z-1| + \frac{(x-1)^2}{|2(z-1)|} + \frac{1}{2}|z-3| + \frac{(x-3)^2}{|2(z-3)|} \\ &\quad + \frac{1}{2}|z-4| + \frac{(x-4)^2}{|2(z-4)|} + \frac{1}{2}|z-7| + \frac{(x-7)^2}{|2(z-7)|} \\ &\quad + \frac{1}{2}|z-8| + \frac{(x-8)^2}{|2(z-8)|}. \end{aligned} \tag{8.13}$$

To start the iterative majorization algorithm, choose the initial value to be $x_0 = 9$, although any other value would be equally valid. This implies that the first supporting point x_0 in the IM algorithm is $z = x_0 = 9$. After substitution of $z = 9$ into (8.13) and simplification, we obtain

$$\begin{aligned} g(x,9) &= \frac{1}{2}|9-1| + \frac{(x-1)^2}{|2(9-1)|} + \frac{1}{2}|9-3| + \frac{(x-3)^2}{|2(9-3)|} + \frac{1}{2}|9-4| \\ &\quad + \frac{(x-4)^2}{|2(9-4)|} + \frac{1}{2}|9-7| + \frac{(x-7)^2}{|2(9-7)|} + \frac{1}{2}|9-8| + \frac{(x-8)^2}{|2(9-8)|} \\ &= \frac{8}{2} + \frac{(x-1)^2}{16} + \frac{6}{2} + \frac{(x-3)^2}{12} + \frac{5}{2} \\ &\quad + \frac{(x-4)^2}{10} + \frac{2}{2} + \frac{(x-7)^2}{4} + \frac{1}{2} + \frac{(x-8)^2}{2} \\ &= \frac{8}{2} + \frac{(x-1)^2}{16} + \frac{6}{2} + \frac{(x-3)^2}{12} + \frac{5}{2} \\ &\quad + \frac{(x-4)^2}{10} + \frac{2}{2} + \frac{(x-7)^2}{4} + \frac{1}{2} + \frac{(x-8)^2}{2} \\ &= \frac{239}{240}x^2 - \frac{517}{40}x + \frac{4613}{80}. \end{aligned} \tag{8.14}$$

This example of quadratic majorization is illustrated in Figure 8.8. Because $g(x, x_0)$ is quadratic in x, its minimum can be easily obtained by setting the derivative equal to zero (see Section 8.2). The minimum of $g(x, x_0)$ is

attained at $x_1 \approx 6.49$. Due to the majorization inequality, we must have that $f(x_1) \leq g(x_1, x_0) \leq g(x_0, x_0) = f(x_0)$. Thus, we have found an x_1 with a lower function value $f(x)$. The next step in the majorization algorithm is to declare x_1 to be the next supporting point, to compute $g(x, x_1)$ and find its minimum x_2, and so on. After some iterations, we find that 4 is the minimum for $f(x)$; hence 4 is the median.

The key problem in quadratic majorization is to find a majorizing inequality such as (8.12). Unlike concave functions, which can always be linearly majorized, it is an art to find quadratic majorizing functions. Note that linear and quadratic majorization can be combined without any problem as long as the majorization conditions hold.

8.5 Visualizing the Majorization Algorithm for MDS

To get an idea what the iterative majorization algorithm does in MDS, we consider a mini example from the data of Exercise 3.3. These data contain the correlations among the returns of 13 stock markets. To analyze these data, we converted the correlations into dissimilarities by (6.1), so that $\delta_{ij} = (2-2r_{ij})^{1/2}$. Then we performed ratio MDS by the SMACOF algorithm (see the next section). The resulting configuration is given in Figure 8.9. We see, for example, that the Dow Jones (dj) and Standard & Poors (sp) indices correlate highly, because they are very close together. We also see that the European indices (brus, dax, vec, cbs, ftse, milan, and madrid) are reasonably similar because they are located together. The Asian markets (hs, nikkei, taiwan, and sing) do not seem to correlate highly among one another as they are lying at quite some distance from one another.

To see how the iterative majorization algorithm for MDS works, consider the situation where the coordinates of all stock indices are kept fixed at the positions of Figure 8.9 except for the point nikkei. To minimize raw Stress, we can only vary the two coordinates x_{i1} and x_{i2} of nikkei. This simplification allows us to visualize the raw Stress function as a surface in 3D with x_{i1} and x_{i2} in the xy plane and the raw Stress value on the z-axis. Figure 8.10 shows the raw Stress surface in both panels. The ground area shows the position of all the fixed points and, for reference, also the optimal position of nikkei. It is clear that in this situation, the coordinates for nikkei where raw Stress finds its global minimum are indeed located at the point with label nikkei. However, a computer is "blind" and cannot "see" where these optimal coordinates of nikkei with the lowest raw Stress function is found. Therefore, it needs an optimization algorithm such as iterative majorization to compute the location of minimal raw Stress.

Iterative majorization for MDS works in this example as follows. Suppose that the initial guess for the coordinates of nikkei is the origin. Then,

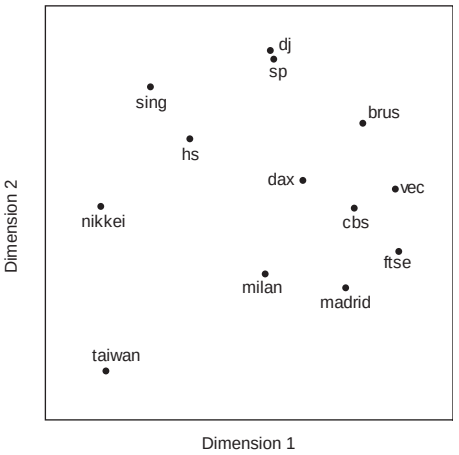


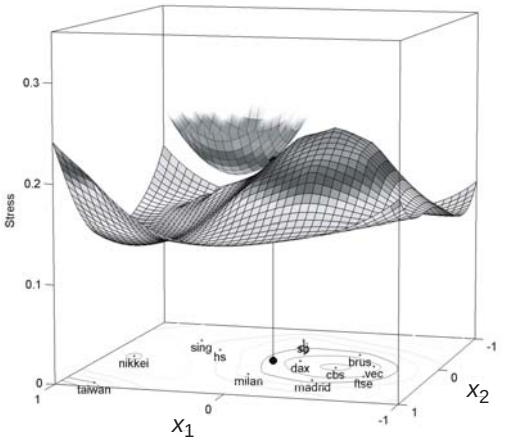
FIGURE 8.9. Ratio MDS solution of correlations between returns of 13 stock markets. The data are given in Exercise 3.3.

the majorizing function must touch the raw Stress function at the origin (with coordinates $x_{i1} = 0$ and $x_{i2} = 0$) and must be located above it (or touch it) at other locations. The parabola in Figure 8.10a satisfies these restrictions and is therefore a valid majorizing function. At the location of the minimum of this majorizing function, the raw Stress function is lower. Thus, choosing this location as the next estimate of the coordinates for nikkei reduces the raw Stress. At this location, a new majorizing function can be found that again touches the raw Stress function at this location and is otherwise located above the raw Stress function. The minimum of this new majorizing function can be determined and will again decrease raw Stress. This process is iterated until the improvement in raw Stress is considered small enough. This final situation is shown in Figure 8.10b with the last majorizing function. We note that the majorizing algorithm has correctly identified the best local minimum possible. The estimates for the location of point nikkei in the different iterations is shown by the trail of points in the xy plane between the origin and the final location of nikkei.

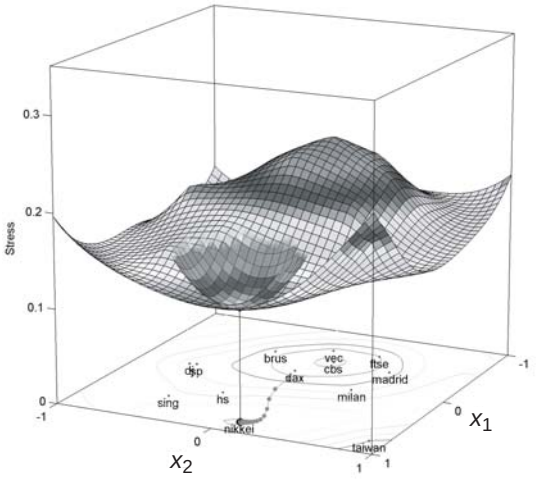
Here, we focused on the special case that only two coordinates need to be estimated and all others are kept fixed. The next section explains how the iterative majorization algorithm works when all coordinates need to be found simultaneously.

8.6 Majorizing Stress

So far, we have discussed the principle of iterative majorization for functions of one variable x only. The same idea can be applied to functions that



a. First majorizing function



b. Final majorizing function

FIGURE 8.10. Visualization of the raw Stress function for the Stock market data where all coordinates are kept fixed except those of nikkei. For reference, the optimal position of nikkei is also shown. The upper panel shows the majorizing function with the origin as current estimate for the location of nikkei. The lower panel shows the final majorizing function and a trail of points in the xy-plane showing the positions of point nikkei in the different iterations.

have several variables. As long as the majorizing inequalities (8.11) hold, iterative majorization can be used to minimize a function of many variables.

We now apply iterative majorization to the Stress function, which goes back to De Leeuw (1977), De Leeuw and Heiser (1977), and De Leeuw (1988). The acronym SMACOF initially stood for "Scaling by Maximizing a Convex Function," but since the mid-1980s it has stood for "Scaling by Majorizing a Complicated Function." Algorithms other than SMACOF have been derived to minimize Stress. For example, using approaches from convex analysis, the same algorithm for minimizing Stress was obtained by De Leeuw (1977), Mathar (1989), and Mathar and Groenen (1991). Earlier, Stress was minimized by steepest descent algorithms by Kruskal (1964b) and Guttman (1968) that use the gradient of Stress. However, the SMACOF theory is simple and more powerful, because it guarantees monotone convergence of Stress. Hence, we pursue the majorization approach and show how to majorize the raw Stress function, $\sigma_r(\mathbf{X})$, following the SMACOF theory.

Components of the Stress Function

The Stress function (8.4) can be written as

$$\begin{aligned}\sigma_r(\mathbf{X}) &= \sum_{i<j} w_{ij}\left(\delta_{ij} - d_{ij}(\mathbf{X})\right)^2 \\ &= \sum_{i<j} w_{ij}\delta_{ij}^2 + \sum_{i<j} w_{ij} d_{ij}^2(\mathbf{X}) - 2\sum_{i<j} w_{ij}\delta_{ij} d_{ij}(\mathbf{X}) \\ &= \eta_\delta^2 + \eta^2(\mathbf{X}) - 2\rho(\mathbf{X}), \end{aligned} \tag{8.15}$$

where $d_{ij}(\mathbf{X})$ is the Euclidean distance between points i and j; see also (3.3). From (8.15) we see that Stress can be decomposed into three parts. The first part, η_δ^2, is only dependent on the fixed weights w_{ij} and the fixed dissimilarities δ_{ij}, and not dependent on $\mathbf{X}$; so η_δ^2 is constant. The second part, $\eta^2(\mathbf{X})$, is a weighted sum of the squared distances $d_{ij}^2(\mathbf{X})$. The final part, $-2\rho(\mathbf{X})$, is a weighted sum of the "plain" distances $d_{ij}(\mathbf{X})$. Before we go on, we have to make one additional assumption: we assume throughout this book that the weight matrix $\mathbf{W}$ is *irreducible*, that is, there exists no partitioning of objects into disjoint subsets, such that $w_{ij} = 0$ whenever objects i and j are in different subsets. If the weight matrix is reducible, then the problem can be decomposed into separate smaller multidimensional scaling problems, one for each subset. Let us consider $\eta^2(\mathbf{X})$ and $\rho(\mathbf{X})$ separately to obtain our majorization algorithm.

A Compact Expression for the Sum of Squared Distances

We first look at $\eta^2(\mathbf{X})$, which is a sum of the squared distances. For the moment, we consider only one squared distance $d_{ij}^2(\mathbf{X})$. Let $\mathbf{x}_a$ be column

a of the coordinate matrix $\mathbf{X}$. Furthermore, let $\mathbf{e}_i$ be the ith column of the identity matrix $\mathbf{I}$. Thus, if $n = 4$, $i = 1$, and $j = 3$, then $\mathbf{e}_i' = [1\ 0\ 0\ 0]$ and $\mathbf{e}_j' = [0\ 0\ 1\ 0]$, so that $(\mathbf{e}_i - \mathbf{e}_j)' = [1\ 0\ -1\ 0]$. But this means that $x_{ia} - x_{ja} = (\mathbf{e}_i - \mathbf{e}_j)'\mathbf{x}_a$, which allows us to express the squared distance $d_{13}^2(\mathbf{X})$ as

$$\begin{aligned} d_{13}^2(\mathbf{X}) &= \sum_{a=1}^m \mathbf{x}_a'(\mathbf{e}_1 - \mathbf{e}_3)(\mathbf{e}_1 - \mathbf{e}_3)'\mathbf{x}_a \\ &= \sum_{a=1}^m \mathbf{x}_a' \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{x}_a = \sum_{a=1}^m \mathbf{x}_a'\mathbf{A}_{13}\mathbf{x}_a \\ &= \text{tr } \mathbf{X}'\mathbf{A}_{13}\mathbf{X}. \end{aligned} \tag{8.16}$$

The matrix $\mathbf{A}_{ij}$ is simply a matrix with $a_{ii} = a_{jj} = 1$, $a_{ij} = a_{ji} = -1$, and all other elements zero. Note that $\mathbf{A}_{ij}$ is row and column centered, so that $\mathbf{A}_{ij}\mathbf{1} = \mathbf{0}$ and $\mathbf{1}'\mathbf{A}_{ij} = \mathbf{0}'$. But $\eta^2(\mathbf{X})$ is a weighted sum of these squared distances. One term of $\eta^2(\mathbf{X})$ is

$$\begin{aligned} w_{ij}d_{ij}^2(\mathbf{X}) &= w_{ij}\text{tr } \mathbf{X}'\mathbf{A}_{ij}\mathbf{X} \\ &= \text{tr } \mathbf{X}'(w_{ij}\mathbf{A}_{ij})\mathbf{X}, \end{aligned}$$

and summing over all $i < j$ terms gives

$$\begin{aligned} \eta^2(\mathbf{X}) &= \sum_{i<j} w_{ij}d_{ij}^2(\mathbf{X}) = \text{tr } \mathbf{X}' \left(\sum_{i<j} w_{ij}\mathbf{A}_{ij} \right) \mathbf{X} \\ &= \text{tr } \mathbf{X}'\mathbf{V}\mathbf{X}. \end{aligned} \tag{8.17}$$

In a 3×3 example, the matrix $\mathbf{V}$ defined in (8.17) becomes

$$\begin{aligned} \mathbf{V} &= \sum_{i<j} w_{ij}\mathbf{A}_{ij} \\ &= \begin{bmatrix} w_{12} + w_{13} & -w_{12} & -w_{13} \\ -w_{12} & w_{12} + w_{23} & -w_{23} \\ -w_{13} & -w_{23} & w_{13} + w_{23} \end{bmatrix}, \end{aligned} \tag{8.18}$$

or, in general, $v_{ij} = -w_{ij}$ if $i \neq j$ and $v_{ii} = \sum_{j=1, j\neq i}^n w_{ij}$ for the diagonal elements of $\mathbf{V}$. By (8.17) we have obtained a compact matrix expression for $\eta^2(\mathbf{X})$. Furthermore, $\eta^2(\mathbf{X})$ is a quadratic function in $\mathbf{X}$, which is easy to handle. Because $\mathbf{V}$ is the weighted sum of row and column centered matrices $\mathbf{A}_{ij}$, it is row and column centered itself, too. Because of our assumption that the weights are irreducible, the rank of $\mathbf{V}$ is $n - 1$, the zero eigenvalue corresponding to the eigenvector $n^{-1/2}\mathbf{1}$.

Majorizing Minus a Weighted Sum of Distances

We now switch to $-\rho(\mathbf{X})$, which is minus a weighted sum of the distances; that is,

$$-\rho(\mathbf{X}) = -\sum_{i<j}(w_{ij}\delta_{ij})d_{ij}(\mathbf{X}).$$

For the moment, we focus on minus the distance. To obtain a majorizing inequality for $-d_{ij}(\mathbf{X})$, we use the *Cauchy–Schwarz* inequality,

$$\sum_{a=1}^{m} p_a q_a \leq \left(\sum_{a=1}^{m} p_a^2\right)^{1/2}\left(\sum_{a=1}^{m} q_a^2\right)^{1/2}. \tag{8.19}$$

Equality of (8.19) occurs if $q_a = cp_a$. If we substitute p_a by $(x_{ia} - x_{ja})$ and q_a by $(z_{ia} - z_{ja})$ in (8.19), we obtain

$$\begin{aligned}\sum_{a=1}^{m}(x_{ia} - x_{ja})(z_{ia} - z_{ja}) &\leq \left(\sum_{a=1}^{m}(x_{ia} - x_{ja})^2\right)^{1/2}\left(\sum_{a=1}^{m}(z_{ia} - z_{ja})^2\right)^{1/2} \\ &= d_{ij}(\mathbf{X})d_{ij}(\mathbf{Z}),\end{aligned} \tag{8.20}$$

with equality if $\mathbf{Z} = \mathbf{X}$. Dividing both sides by $d_{ij}(\mathbf{Z})$ and multiplying by -1 gives

$$-d_{ij}(\mathbf{X}) \leq -\frac{\sum_{a=1}^{m}(x_{ia} - x_{ja})(z_{ia} - z_{ja})}{d_{ij}(\mathbf{Z})}. \tag{8.21}$$

If points i and j have zero distance in configuration matrix $\mathbf{Z}$, then (8.21) becomes undefined, but because of the positivity of $d_{ij}(\mathbf{X})$ it is still true that $-d_{ij}(\mathbf{X}) \leq 0$. Proceeding as in (8.16)–(8.18), a simple matrix expression is obtained:

$$\sum_{a=1}^{m}(x_{ia} - x_{ja})(z_{ia} - z_{ja}) = \text{tr } \mathbf{X}'\mathbf{A}_{ij}\mathbf{Z}. \tag{8.22}$$

Combining (8.21) and (8.22), multiplying by $w_{ij}\delta_{ij}$, and summing over $i < j$ gives

$$\begin{aligned}-\rho(\mathbf{X}) &= -\sum_{i<j}(w_{ij}\delta_{ij})d_{ij}(\mathbf{X}) \\ &\leq -\text{tr } \mathbf{X}'\left(\sum_{i<j} b_{ij}\mathbf{A}_{ij}\right)\mathbf{Z} \\ &= -\text{tr } \mathbf{X}'\mathbf{B}(\mathbf{Z})\mathbf{Z},\end{aligned} \tag{8.23}$$

where $\mathbf{B}(\mathbf{Z})$ has elements

$$\begin{aligned} b_{ij} &= \begin{cases} -\dfrac{w_{ij}\delta_{ij}}{d_{ij}(\mathbf{Z})} & \text{for } i \neq j \text{ and } d_{ij}(\mathbf{Z}) \neq 0 \\ 0 & \text{for } i \neq j \text{ and } d_{ij}(\mathbf{Z}) = 0 \end{cases} \\ b_{ii} &= -\sum_{j=1, j\neq i}^{n} b_{ij}. \end{aligned} \tag{8.24}$$

Because equality occurs if $\mathbf{Z} = \mathbf{X}$, we have obtained the majorization inequality

$$-\rho(\mathbf{X}) = -\text{tr } \mathbf{X}'\mathbf{B}(\mathbf{X})\mathbf{X} \leq -\text{tr } \mathbf{X}'\mathbf{B}(\mathbf{Z})\mathbf{Z}.$$

Thus, $-\rho(\mathbf{X})$ can be majorized by the function $-\text{tr } \mathbf{X}'\mathbf{B}(\mathbf{Z})\mathbf{Z}$, which is a linear function in $\mathbf{X}$.

Consider an example for the computation of $\mathbf{B}(\mathbf{Z})$. Let all $w_{ij} = 1$, the dissimilarities be equal to

$$\mathbf{\Delta} = \begin{bmatrix} 0 & 5 & 3 & 4 \\ 5 & 0 & 2 & 2 \\ 3 & 2 & 0 & 1 \\ 4 & 2 & 1 & 0 \end{bmatrix}, \tag{8.25}$$

and the matrix of coordinates $\mathbf{Z}$ and their distances be

$$\mathbf{Z} = \begin{bmatrix} -.266 & -.539 \\ .451 & .252 \\ .016 & -.238 \\ -.200 & .524 \end{bmatrix} \text{ and } \mathbf{D}(\mathbf{Z}) = \begin{bmatrix} .000 & 1.068 & .412 & 1.065 \\ 1.068 & .000 & .655 & .706 \\ .412 & .655 & .000 & .792 \\ 1.065 & .706 & .792 & .000 \end{bmatrix}. \tag{8.26}$$

The elements of the first row $\mathbf{B}(\mathbf{Z})$ are given by

$$\begin{aligned} b_{12} &= -w_{12}\delta_{12}/d_{12}(\mathbf{Z}) = -5/1.068 = -4.682 \\ b_{13} &= -w_{13}\delta_{13}/d_{13}(\mathbf{Z}) = -3/0.412 = -7.273 \\ b_{14} &= -w_{14}\delta_{14}/d_{14}(\mathbf{Z}) = -4/1.065 = -3.756 \\ b_{11} &= -(b_{12} + b_{13} + b_{14}) = -(-4.682 - 7.273 - 3.756) = 15.712. \end{aligned}$$

In the same way, all elements of $\mathbf{B}(\mathbf{Z})$ can be computed, yielding

$$\mathbf{B}(\mathbf{Z}) = \begin{bmatrix} 15.712 & -4.682 & -7.273 & -3.756 \\ -4.682 & 10.570 & -3.052 & -2.835 \\ -7.273 & -3.052 & 11.588 & -1.263 \\ -3.756 & -2.835 & -1.263 & 7.853 \end{bmatrix}.$$

The SMACOF *Algorithm for Majorizing Stress*

Combining (8.17) and (8.25) gives us the majorization inequality for the Stress function; that is,

$$\begin{aligned} \sigma_r(\mathbf{X}) &= \eta_\delta^2 + \text{tr } \mathbf{X}'\mathbf{V}\mathbf{X} - 2\text{tr } \mathbf{X}'\mathbf{B}(\mathbf{X})\mathbf{X} \\ &\leq \eta_\delta^2 + \text{tr } \mathbf{X}'\mathbf{V}\mathbf{X} - 2\text{tr } \mathbf{X}'\mathbf{B}(\mathbf{Z})\mathbf{Z} = \tau(\mathbf{X}, \mathbf{Z}). \end{aligned} \tag{8.27}$$

Thus $\tau(\mathbf{X}, \mathbf{Z})$ is a simple majorizing function of Stress that is quadratic in $\mathbf{X}$. Its minimum can be obtained analytically by setting the derivative of $\tau(\mathbf{X}, \mathbf{Z})$ equal to zero; that is,

$$\nabla\tau(\mathbf{X}, \mathbf{Z}) = 2\mathbf{V}\mathbf{X} - 2\mathbf{B}(\mathbf{Z})\mathbf{Z} = \mathbf{0},$$

so that $\mathbf{V}\mathbf{X} = \mathbf{B}(\mathbf{Z})\mathbf{Z}$. To solve this system of linear equations for $\mathbf{X}$, we would usually premultiply both sides by $\mathbf{V}^{-1}$. However, the inverse $\mathbf{V}^{-1}$ does not exist, because $\mathbf{V}$ is not of full rank. Therefore, we revert to the Moore–Penrose[3] inverse. The Moore–Penrose inverse of $\mathbf{V}$ is given by $\mathbf{V}^+ = (\mathbf{V} + \mathbf{1}\mathbf{1}')^{-1} - n^{-2}\mathbf{1}\mathbf{1}'$. The last term, $-n^{-2}\mathbf{1}\mathbf{1}'$, is irrelevant in SMACOF as $\mathbf{V}^+$ is subsequently multiplied by a matrix orthogonal to $\mathbf{1}$, because $\mathbf{B}(\mathbf{Z})$ also has eigenvector $\mathbf{1}$ with eigenvalue zero. This leads us to the update formula of the SMACOF algorithm,

$$\mathbf{X}^u = \mathbf{V}^+\mathbf{B}(\mathbf{Z})\mathbf{Z}. \tag{8.28}$$

If all $w_{ij} = 1$, then $\mathbf{V}^+ = n^{-1}\mathbf{J}$ with $\mathbf{J}$ the *centering matrix* $\mathbf{I} - n^{-1}\mathbf{1}\mathbf{1}'$, so that the update simplifies to

$$\mathbf{X}^u = n^{-1}\mathbf{B}(\mathbf{Z})\mathbf{Z}. \tag{8.29}$$

De Leeuw and Heiser (1980) call (8.28) the *Guttman transform*, in recognition of Guttman (1968).

The majorization algorithm guarantees a series of nonincreasing Stress values. When the algorithm stops, the stationary condition $\mathbf{X} = \mathbf{V}^+\mathbf{B}(\mathbf{X})\mathbf{X}$ holds. Note that after one step of the algorithm $\mathbf{X}$ is column centered, even if $\mathbf{Z}$ is not column centered.

The SMACOF algorithm for MDS can be summarized by

1. Set $\mathbf{Z} = \mathbf{X}^{[0]}$, where $\mathbf{X}^{[0]}$ is some (non)random start configuration. Set $k = 0$. Set ε to a small positive constant.
2. Compute $\sigma_r^{[0]} = \sigma_r(\mathbf{X}^{[0]})$. Set $\sigma_r^{[-1]} = \sigma_r^{[0]}$.
3. While $k = 0$ or $(\sigma_r^{[k-1]} - \sigma_r^{[k]} > \varepsilon$ and $k \leq$ maximum iterations) do
4. Increase iteration counter k by one.
5. Compute the Guttman transform $\mathbf{X}^{[k]}$ by (8.29) if all $w_{ij} = 1$, or by (8.28) otherwise.
6. Compute $\sigma_r^{[k]} = \sigma_r(\mathbf{X}^{[k]})$.

[3]Gower and Groenen (1991) report some computationally very efficient Moore–Penrose inverses for some special weight matrices, such as those of a cyclic design and a block design (see Table 6.1).

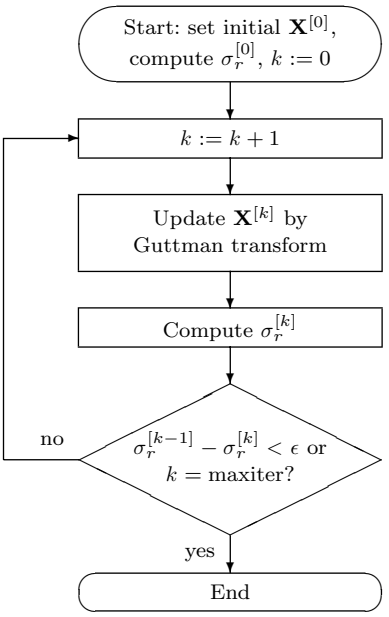


FIGURE 8.11. The flow of the majorization algorithm (SMACOF) for doing MDS.

7. Set $\mathbf{Z} = \mathbf{X}^{[k]}$.

8. End while

A flowchart of the SMACOF algorithm is given in Figure 8.11.

An Illustration of Majorizing Stress

To illustrate the SMACOF algorithm, consider the following example. We assume that all $w_{ij} = 1$, that the dissimilarities $\mathbf{\Delta}$ are those in (8.25), and the starting configuration $\mathbf{X}^{[0]} = \mathbf{Z}$ by (8.26). The first step is to compute $\sigma_r(\mathbf{X}^{[0]})$, which is 34.29899413. Then, we compute the first update $\mathbf{X}^u$ by the Guttman transform (8.29),

$$\begin{aligned}\mathbf{X}^u &= n^{-1}\mathbf{B}(\mathbf{Z})\mathbf{Z} \\ &= \frac{1}{4}\begin{bmatrix} 15.712 & -4.683 & -7.273 & -3.756 \\ -4.683 & 10.570 & -3.052 & -2.835 \\ -7.273 & -3.052 & 11.588 & -1.263 \\ -3.756 & -2.835 & -1.263 & 7.853 \end{bmatrix}\begin{bmatrix} -.266 & -.539 \\ .451 & .252 \\ .016 & -.238 \\ -.200 & .524 \end{bmatrix},\end{aligned}$$

$$\mathbf{X}^u = \begin{bmatrix} -1.415 & -2.471 \\ 1.633 & 1.107 \\ .249 & -.067 \\ -.468 & 1.431 \end{bmatrix} \text{ with } \mathbf{D}(\mathbf{X}^u) = \begin{bmatrix} .000 & 4.700 & 2.923 & 4.016 \\ 4.700 & .000 & 1.815 & 2.126 \\ 2.923 & 1.815 & .000 & 1.661 \\ 4.016 & 2.126 & 1.661 & .000 \end{bmatrix}.$$

TABLE 8.4. The Stress values and the difference between two iterations k of the SMACOF algorithm.

k	$\sigma_r^{[k]}$	$\sigma_r^{[k-1]} - \sigma_r^{[k]}$	k	$\sigma_r^{[k]}$	$\sigma_r^{[k-1]} - \sigma_r^{[k]}$
0	34.29899413		21	.01747237	.00001906
1	.58367883	33.71531530	22	.01745706	.00001531
2	.12738894	.45628988	23	.01744477	.00001229
3	.04728335	.08010560	24	.01743491	.00000986
4	.02869511	.01858823	25	.01742700	.00000791
5	.02290353	.00579158	26	.01742066	.00000634
6	.02059574	.00230779	27	.01741557	.00000509
7	.01950236	.00109338	28	.01741150	.00000408
8	.01890539	.00059698	29	.01740823	.00000327
9	.01853588	.00036951	30	.01740561	.00000262
10	.01828296	.00025292	31	.01740351	.00000210
11	.01809735	.00018561	32	.01740183	.00000168
12	.01795518	.00014217	33	.01740048	.00000135
13	.01784363	.00011155	34	.01739941	.00000108
14	.01775498	.00008866	35	.01739854	.00000086
15	.01768406	.00007092			
16	.01762716	.00005690			
17	.01758144	.00004572			
18	.01754469	.00003675			
19	.01751516	.00002953			
20	.01749143	.00002373			

The next step is to set $\mathbf{X}^{[1]} = \mathbf{X}^u$ and compute $\sigma_r(\mathbf{X}^{[1]}) = 0.58367883$, which concludes the first iteration. The difference of $\sigma_r(\mathbf{X}^{[0]})$ and $\sigma_r(\mathbf{X}^{[1]})$ is large, 33.71531530, so it makes sense to continue the iterations. The second update is

$$\mathbf{X}^{[2]} = \begin{bmatrix} 1.473 & -2.540 \\ 1.686 & 1.199 \\ .154 & .068 \\ -.366 & 1.274 \end{bmatrix},$$

with $\sigma_r(\mathbf{X}^{[2]}) = .12738894$. We continue the iterations until the difference in subsequent Stress values is less than 10^{-6}. With this value, it can be expected that the configuration coordinates are accurate up to the third decimal. The history of iterations is presented in Table 8.4. After 35 iterations, the convergence criterion was reached with configuration

$$\mathbf{X}^{[35]} = \begin{bmatrix} -1.457 & -2.575 \\ 1.730 & 1.230 \\ -0.028 & 0.160 \\ -0.245 & 1.185 \end{bmatrix}.$$

Various nice results can be derived from the SMACOF algorithm. For example, De Leeuw (1988) showed that $\mathbf{X}^{[k]}$ converges linearly to a stationary point. In technical terms, linear convergence means that $||\mathbf{X}^{[\infty]} - \mathbf{X}^{[k-1]}||/||\mathbf{X}^{[\infty]} - \mathbf{X}^{[k]}|| \to \lambda$, where $0 < \lambda < 1$ is the largest eigenvalue not equal to 1 of the matrix of the second derivatives of the Guttman transform. Another attractive aspect of SMACOF is that zero distances are unproblematic, because of the definition of b_{ij} in (8.24). In gradient-based algorithms, ad hoc strategies have to be applied if zero distances occur. If no zero distances are present, then it can be shown that the Guttman transform is a steepest descent step with a fixed stepsize parameter.

8.7 Exercises

Exercise 8.1 Consider the function $f(x) = 2x^3 - 6x^2 - 18x + 9$.

(a) Tabulate the values of the function $f(x)$, its derivative $f'(x)$, and $f''(x)$ for x equal to $-4, -3, -2, 1, 0, 1, \ldots, 6$.

(b) Plot all three functions in the same diagram.

(c) Find the minima and maxima of $f(x)$ in the interval $[-4, +6]$ through inspection of the function graph and through computation, respectively.

(d) Interpret $f''(x)$.

Exercise 8.2 Find local and absolute maxima and minima of the following functions.

(a) $y = x^2 - 3x$, for $0 \leq x \leq 5$.

(b) $v = 1 + 2t + 0.5t^2$, for $-3 \leq t \leq 3$.

(c) $u = 1/(2v + 3)$, for $1 \leq v \leq 3$.

(d) $y = x^3 - 3x$, for $-3 \leq x \leq 3$.

Exercise 8.3 Repeat Exercise 8.2 for

(a) $f(x, y) = 4xy - x^2 - y^2$.

(b) $f(x, y) = x^2 - y^2$.

(c) $f(x, y) = x^2 + 2xy + 2y^2 - 6y + 2$.

Exercise 8.4 Use a computer program that does function plots.

(a) Plot $f(x, y) = x^2 + xy - y$.

(b) Find the minimum value of $f(x, y)$ by graphical means.

(c) Find the minimum of $f(x, y)$ by differentiation techniques. [Hint: Use partial differentiation with respect of $f(x, y)$ with respect to x and y, respectively, to obtain the x- and y-coordinates of the minimal point of the function.]

Exercise 8.5 Use matrix differentiation to solve the regression problem $\mathbf{y} \approx \mathbf{Xb}$, where $\mathbf{y}$ is the criterion vector, $\mathbf{X}$ is the battery of predictor vectors (columns), and $\mathbf{b}$ is the vector of unknown weights. Find $\mathbf{b}$ such that $||\mathbf{y} - \mathbf{Xb}||^2 = \min$. (Hint: Express the norm as a trace function.)

Exercise 8.6 Use the solution from Exercise 8.5 to solve the following problems.

(a) Find the vector $\mathbf{x}_1$ that solves (7.23) on p. 156 in a least-squares sense. That is, minimize $f(\mathbf{x}_1) = ||\mathbf{A}_1\mathbf{x}_1 - \mathbf{b}_1||^2$ by an appropriate $\mathbf{x}_i$.

(b) What is the value of $f(\mathbf{x}_1)$ at the optimal $\mathbf{x}_1$?

(c) Repeat (a) for $\mathbf{A}_2, \mathbf{b}_2$, and $\mathbf{x}_2$ from (7.24).

(d) Repeat (a) for $\mathbf{A}_3, \mathbf{b}_3$, and $\mathbf{x}_3$ from (7.25).

Exercise 8.7 Suppose that we want to approximate the list of values $0, 2, 6, 5$, and 9 by a single value x. One option is to put these values in the vector $\mathbf{z} = [0\ 2\ 6\ 5\ 9]'$ and minimize the least-squares function $f(x) = \|\mathbf{z} - x\mathbf{1}\|^2$ over x.

(a) Derive $f'(x)$ and express the result in the matrix algebra. [Hint: Start by expanding $f(x)$ into separate terms. Then apply the rules for differentiation to the individual terms.]

(b) Equate the derivative to zero. Can an analytic solution for x be obtained?

(c) For what value of x is $f(x)$ at its minimum?

(d) What can you say about the x that minimizes $f(x)$?

Exercise 8.8 Consider the matrix $\mathbf{V}$ in (8.18) on p. 188.

(a) What do you expect to be the outcome of $\mathbf{V1}$ and $\mathbf{1'V}$? Compute the results for the small example of (8.18). Does this result hold for $\mathbf{V}$ being of any size?

(b) Suppose $\mathbf{Y} = \mathbf{Z} + \mathbf{1a}'$. Explain why $\mathbf{VZ} = \mathbf{VY}$.

(c) Show that $\mathbf{V}$ is double centered.

(d) Is the matrix $\mathbf{B}(\mathbf{Z})$ in (8.24) also double centered? Explain why or why not.

(e) As a consequence of (d), how do you expect that a translation of the type $\mathbf{Z} + \mathbf{1a}'$ changes a single iteration (8.29) of the SMACOF algorithm?

Exercise 8.9 In the so-called Median-center problem, the objective is to find a point such that the Euclidean distance to all other points is minimal. Let $\mathbf{Y}$ be the matrix of n given points. The function that needs to be minimized is

$$f(\mathbf{x}) = \sum_{i=1}^{n} d_i(\mathbf{x}),$$

where $d_i(\mathbf{x}) = \|\mathbf{x} - \mathbf{y}_i\|$ and $\mathbf{y}_i$ is row i of $\mathbf{Y}$.

(a) Use the results from the section on majorizing the median to find a majorizing function $g(\mathbf{x}, \mathbf{z})$ that is a weighted sum of $d_i^2(\mathbf{x})$ and where the weights are dependent on the $d_i^2(\mathbf{z})$, where $\mathbf{z}$ is the vector with the previous estimates of $\mathbf{x}$.

(b) Determine the derivative of $g(\mathbf{x}, \mathbf{z})$.

(c) Set the derivative of $g(\mathbf{x}, \mathbf{z})$ equal to zero. Solve this equation for $\mathbf{x}$. (Hint: you will have to use results from Section 7.7.)

(d) Use a program that can do matrix computations and program your majorization algorithm. Choose a random $\mathbf{Y}$ and apply your algorithm to this $\mathbf{y}$. Verify that every subsequent iteration reduces $f(\mathbf{x})$.

9
Metric and Nonmetric MDS

In the previous chapter, we derived a majorization algorithm for fixed dissimilarities. However, in practical research we often have only rank-order information of the dissimilarities (or proximities), so that transformations that preserve the rank-order of the dissimilarities become admissible. In this chapter, we discuss optimal ways of estimating this and other transformations. One strategy for ordinal MDS is to use monotone regression. A different strategy, rank-images, is not optimal for minimizing Stress, but it has other properties that can be useful in MDS. An attractive group of transformations are spline transformations, which contain ordinal and linear transformations as special cases.

9.1 Allowing for Transformations of the Proximities

So far, we have assumed that the proximities are ratio-scaled values. However, in the social sciences often only the *rank-order* of the proximities is considered meaningful. In such cases, the dissimilarities δ_{ij} are replaced in the Stress function by *disparities*, $\widehat{d}_{ij}$ (d-hats)[1]. Disparities are an admissible transformation of the proximities, chosen in some optimal way. For example, if only the rank-order of the proximities is considered informative,

[1] Other frequently used terminology for disparities is *pseudo distances* (Kruskal, 1977; Heiser, 1990) or *target distances*.

then the disparities must have the same rank-order as the proximities. In this case, we speak of *ordinal* MDS or *nonmetric* MDS. If the disparities are related to the proximities by a specific continuous function, we speak of *metric* MDS. The proximities are in both cases transformed into disparities. In this chapter, we discuss various metric and nonmetric transformations of the proximities, when to use them, and how to calculate them. To simplify the presentation, we assume throughout this chapter that the proximities are dissimilarities, unless stated otherwise.

Stress with d-Hats

Disparities are incorporated in the Stress function as

$$\begin{aligned}\sigma_r(\widehat{\mathbf{d}},\mathbf{X}) &= \sum_{i<j} w_{ij}(d_{ij}(\mathbf{X})-\widehat{d}_{ij})^2 \\ &= \sum_{i<j} w_{ij}\widehat{d}_{ij}^2 + \sum_{i<j} w_{ij}d_{ij}^2(\mathbf{X}) - 2\sum_{i<j} w_{ij}\widehat{d}_{ij}d_{ij}(\mathbf{X}) \\ &= \eta_{\hat{d}}^2 + \eta^2(\mathbf{X}) - 2\rho(\widehat{\mathbf{d}},\mathbf{X}), \end{aligned} \tag{9.1}$$

where $\widehat{\mathbf{d}}$ denotes the $s \times 1$ vector of disparities with $s = n(n-1)/2$. In Section 8.6, we saw how to minimize Stress over the configuration matrix $\mathbf{X}$ by the SMACOF algorithm. We follow De Leeuw (1977), De Leeuw and Heiser (1977), and De Leeuw (1988) in extending this algorithm to include disparities by iteratively alternating an update of $\mathbf{X}$ with an update of $\widehat{\mathbf{d}}$. Clearly, if we optimize over both $\widehat{\mathbf{d}}$ and $\mathbf{X}$, a trivial solution is $\widehat{\mathbf{d}} = \mathbf{0}$ and $\mathbf{X}$=$\mathbf{0}$, which makes (9.1) equal to zero. To avoid this degenerated solution, we norm $\widehat{\mathbf{d}}$ to some fixed length, such as

$$\eta_{\hat{d}}^2 = n(n-1)/2. \tag{9.2}$$

Metric MDS Models

We now formulate several types or *models* of MDS. In the simplest case (*absolute* MDS), proximities (here dissimilarities) and disparities are related by $p_{ij} = \widehat{d}_{ij}$. Thus,

$$\sigma_r(\widehat{\mathbf{d}},\mathbf{X}) = \sum_{i<j} w_{ij}(d_{ij}(\mathbf{X})-\widehat{d}_{ij})^2 = \sum_{i<j} w_{ij}(d_{ij}(\mathbf{X})-p_{ij})^2, \tag{9.3}$$

so that each proximity value p_{ij} should correspond exactly to the distance between points i and j in the m-dimensional MDS space.

Absolute MDS is, from an applications point of view, irrelevant, because it is of no interest, for example, to exactly reconstruct from Table 2.1 the European map in its original size. Instead, we settled on *ratio* MDS, where

$\widehat{d}_{ij} = b \cdot p_{ij}$. In this case, the proximities must be dissimilarities. Then, Stress equals

$$\begin{aligned}\sigma_r(\widehat{\mathbf{d}}, \mathbf{X}) &= \sum_{i<j} w_{ij}(d_{ij}(\mathbf{X}) - \widehat{d}_{ij})^2 = \sum_{i<j} w_{ij}(d_{ij}(\mathbf{X}) - bp_{ij})^2 \\ &= \sum_{i<j} w_{ij} d_{ij}^2(\mathbf{X}) + b^2 \sum_{i<j} w_{ij} p_{ij}^2 - 2b \sum_{i<j} w_{ij} p_{ij}^2 d_{ij}^2(\mathbf{X}) \\ &= \eta^2(\mathbf{X}) + b^2 \eta_p^2 - 2b\rho(\mathbf{X}). \end{aligned} \tag{9.4}$$

We see that it is not very difficult to optimize (9.4) over b. Setting the derivative of $\sigma_r(\widehat{\mathbf{d}}, \mathbf{X})$ with respect to b equal to zero yields

$$\begin{aligned}\frac{\partial \sigma_r(\widehat{\mathbf{d}}, \mathbf{X})}{\partial b} &= 2b\eta_p^2 - 2\rho(\mathbf{X}) = 0, \\ b &= \frac{\rho(\mathbf{X})}{\eta_p^2}, \end{aligned}$$

which gives the update of b for ratio MDS.

It is easy to generate further MDS models from $\widehat{d}_{ij} = f(p_{ij})$ by defining f in different ways. One generalization of ratio MDS is *interval* MDS,

$$\widehat{d}_{ij} = a + b \cdot p_{ij}, \tag{9.5}$$

where an additive constant, a, has been added. Ratio and interval MDS are *linear* MDS models, because the $f(p_{ij})$s are linear transformations of the p_{ij}s. This carries certain linear properties of the data into the corresponding distances. If the p_{ij}s are dissimilarities, we require that $b > 0$, because larger dissimilarities should correspond to larger distances. Conversely, if the p_{ij}s represent similarities, then $b < 0$, because a large similarity corresponds to a small distance. In *ratio* MDS, the ratio of any two disparities should be equal to the ratio of the corresponding proximities, because $\widehat{d}_{ij}/\widehat{d}_{kl} = (b \cdot p_{ij})/(b \cdot p_{kl}) = p_{ij}/p_{kl}$. Thus, although it is always possible to assess the ratio of distances in any MDS space and to note that, say, d_{ij} is twice as large as d_{kl}, in ratio MDS such relations should mirror corresponding ratios of the data. In interval MDS, then, the ratio of *differences* ("intervals") of distances should be equal to the corresponding ratio of differences in the data.

Naturally, f does not have to be linear. In principle, we may choose any function we like. However, some functions have been found to be particularly useful in various contexts of psychology. Among them are the logarithmic function

$$\widehat{d}_{ij} = b \cdot \log(p_{ij}), \tag{9.6}$$

or, more generally,

$$\widehat{d}_{ij} = a + b \cdot \log(p_{ij}),$$

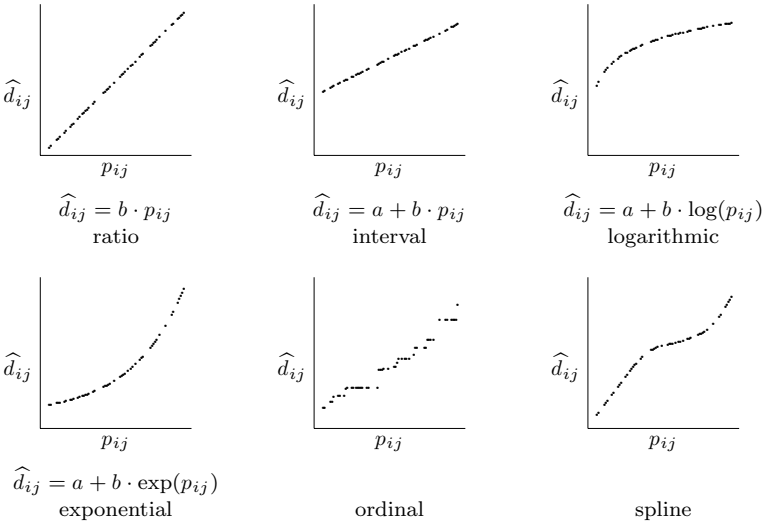


FIGURE 9.1. Transformation plot of several transformations.

and the exponential function

$$\widehat{d}_{ij} = a + b \cdot \exp(p_{ij}). \tag{9.7}$$

Sometimes, we might even consider nonmonotonic functions such as a polynomial function of second degree,

$$\widehat{d}_{ij} = a + b \cdot p_{ij} + c \cdot p_{ij}^2. \tag{9.8}$$

There are no limits to the variety of MDS models that can be constructed in this way. These functions can be viewed in a transformation plot, where the horizontal axis is defined by the proximities (p_{ij}) and the vertical axis is defined by the transformed proximities ($\widehat{d}_{ij}$). Some of the transformations discussed so far are graphed in Figure 9.1.

One problem may occur when fitting some of these models (Heiser, 1990). In the step for finding optimal disparities $\widehat{d}_{ij}$, *negative* disparities can occur. For example, this happens in (9.6) when some p_{ij}s are smaller then 1 and some larger than 1, because $\log(x) < 0$ for $0 < x < 1$. More importantly, in interval MDS, model (9.5), negative disparities can and do occur. Because distances can never be negative, a zero residual in the Stress function is unreachable for negative disparities. Moreover, the majorization algorithm may fail to converge because the inequalities that are used to derive (8.23) are reversed for negative disparities, thereby destroying the convergence proof. This problem can be repaired in two ways: first, on top of the restrictions implied by the model, the disparities are restricted to be positive (which makes updating the disparities more complicated), or, second, the SMACOF algorithm is extended to deal with negative disparities. For more details on this issue, we refer to Heiser (1990).

TABLE 9.1. Some MDS models ordered by the scale level of the proximities (from strong to weak).

Transformation	$\widehat{d}_{ij}$
Absolute	p_{ij}
Ratio	$b \cdot p_{ij}$ with $b > 0$
Interval	$a + b \cdot p_{ij}$ with $a \geq 0, b \geq 0$
Spline	A sum of polynomials of p_{ij}
Ordinal	Preserve the order of p_{ij}s in $\widehat{d}_{ij}$s

Nonmetric MDS

All of the models from (9.3) to (9.8) are *metric*; that is, they represent various properties of the data related to algebraic operations (addition, subtraction, multiplication, division). In contrast, *nonmetric* models represent only the ordinal properties of the data. For example, if $p_{12} = 5$ and $p_{34} = 2$, an ordinal model reads this only as $p_{12} > p_{34}$ (assuming here that the data are dissimilarities) and constructs the distances d_{12} and d_{34} so that $d_{12} > d_{34}$.

Ordinal models typically require that

$$\text{if } p_{ij} < p_{kl}, \text{ then } \widehat{d}_{ij} \leq \widehat{d}_{kl}, \tag{9.9}$$

and no particular order of the distances for $p_{ij} = p_{kl}$ (weak monotonicity[2] and the primary approach to ties). Notice that the models (9.3) to (9.7) also lead to distances ordered in the same way as the corresponding proximities. But they are all special cases of (9.9), where no particular function f is required for the monotone relation. In Table 9.1 some common MDS models are ordered by the scale level of the proximities.

Even weaker MDS models are conceivable. If, for example, we had proximities coded as a, b, or c, we only may require that there be three classes of distances, one for each data code. All that the distances represent then is the qualitative distinctness, and the model could be called *nominal* MDS, where the disparities are restricted by

$$\text{if } p_{ij} = p_{kl}, \text{ then } \widehat{d}_{ij} = \widehat{d}_{kl},$$

which is implemented in the program ALSCAL. However, we discourage the use of nominal MDS because when interpreting an MDS solution we usually assume that the closer two points are, the more similar the objects they represent. The nominal MDS model thwarts this interpretation. Moreover, it admits transformations that may radically change the appearance of the

[2]Requiring strong monotonicity or $\widehat{d}_{ij} < \widehat{d}_{kl}$ rather than just $\widehat{d}_{ij} \leq \widehat{d}_{kl}$ does not lead to stronger models in practice, because one can always turn an equality into an inequality by adding a very small number ϵ to one side of the equation.

MDS configuration. Finally, strict equality in empirical proximities is often rather exceptional, and, indeed, it is just the case that is *excluded* in the usual ordinal MDS (primary approach to ties) because of its presumed empirical unreliability.

Ad Hoc MDS Models

In addition to such textbook models of MDS, more complicated models are occasionally necessary in real applications. Typically, they involve a function $\widehat{\mathbf{d}} = f(\mathbf{p})$ that is itself a combination of several component functions. Consider, for example, the case of ordinal MDS in (9.9). We may not be satisfied with simply requiring that the data be mapped by "some" monotonic function into distances. We may also want to insist that this function be negatively accelerated, say, because we have a theory about what is going on behind the data. We then have to restrict the $\widehat{d}_{ij}$s to be negatively accelerating. Such additional restrictions on f come from substantive considerations and, therefore, are without limit in their number and variety.

SMACOF *with Admissibly Transformed Proximities*

The SMACOF algorithm with transformation of the proximities can be summarized by

1. Set $\mathbf{Z} = \mathbf{X}^{[0]}$, where $\mathbf{X}^{[0]}$ is some (non)random start configuration. Set iteration counter $k = 0$. Set ε to a small positive constant.
2. Find optimal disparities $\widehat{d}_{ij}$ for fixed distances $d_{ij}(\mathbf{X}^{[0]})$.
3. Standardize $\widehat{d}_{ij}$ so that $\eta^2_{\hat{d}} = n(n-1)/2$.
4. Compute $\sigma_r^{[0]} = \sigma_r(\widehat{\mathbf{d}}, \mathbf{X}^{[0]})$. Set $\sigma_r^{[-1]} = \sigma_r^{[0]}$.
5. While $k = 0$ or $(\sigma_r^{[k-1]} - \sigma_r^{[k]} > \varepsilon$ and $k \leq$ maximum iterations) do
6. Increase iteration counter k by one.
7. Compute Guttman transform $\mathbf{X}^{[k]}$ by (8.29) if all $w_{ij} = 1$, or by (8.28) otherwise, where δ_{ij} is replaced by $\widehat{d}_{ij}$.
8. Find optimal disparities $\widehat{d}_{ij}$ for fixed distances $d_{ij}(\mathbf{X}^{[k]})$.
9. Standardize $\widehat{d}_{ij}$ so that $\eta^2_{\hat{d}} = n(n-1)/2$.
10. Compute $\sigma_r(\widehat{\mathbf{d}}, \mathbf{X}^{[k]})$.
11. Set $\mathbf{Z} = \mathbf{X}^{[k]}$.

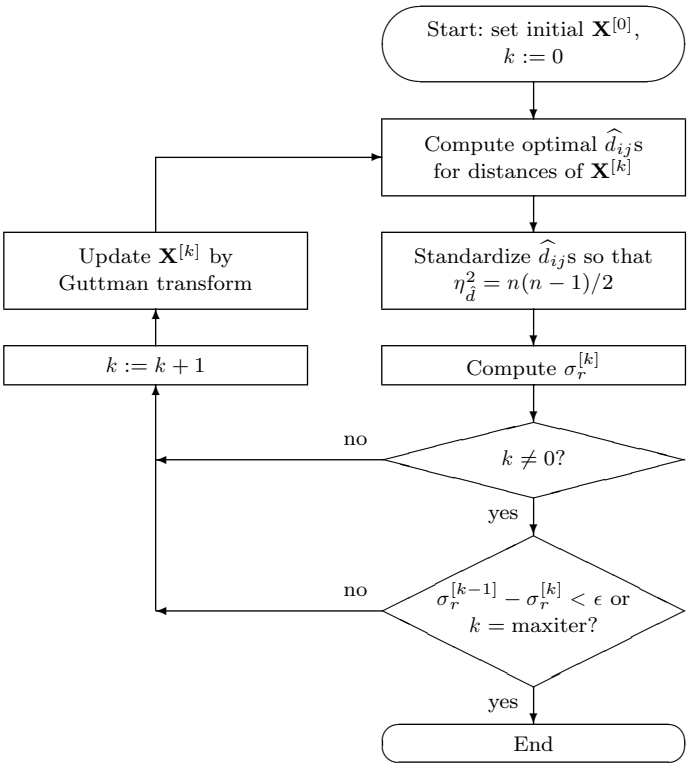


FIGURE 9.2. The flow of the majorization algorithm (SMACOF) for doing MDS with optimal transformations.

12. End while.

A flowchart of this algorithm is presented in Figure 9.2. Note that when computing the Guttman transform the places of the δ_{ij}s are taken by $\widehat{d}_{ij}$s. The allowed transformation of the disparities determines how the update for the disparities in Steps 2 and 7 should be calculated. In the next sections, we discuss the optimal update for ordinal MDS and MDS with splines.

9.2 Monotone Regression

In ordinal MDS, we have to minimize $\sigma_r(\widehat{\mathbf{d}}, \mathbf{X})$ over both $\mathbf{X}$ and $\widehat{\mathbf{d}}$, where the disparities must have the same order as the proximities p_{ij}; that is,

$$\text{if } p_{ij} < p_{kl}, \text{ then } \widehat{d}_{ij} \leq \widehat{d}_{kl} \tag{9.10}$$

if the proximities are dissimilarities, and an inverse order relationship if they are similarities. We switch to Step 7 in the SMACOF algorithm, where

TABLE 9.2. Pairs, ranks, symbolic proximities, numeric proximities (=ranks), numeric distances of starting configuration $\mathbf{X}$, symbolic distances, and target distances for $\mathbf{X}$.

Pair	Rank	Sym. p_{ij}	p_{ij}	d_{ij}	Sym. d_{ij}	$\widehat{d}_{ij}$
Humphrey–McGovern	1	p_{HM}	1	7.8	d_{HM}	3.38
McGovern–Percy	2	p_{MP}	2	3.2	d_{MP}	3.38
Nixon–Wallace	3	p_{NW}	3	0.8	d_{NW}	3.38
Nixon–Percy	4	p_{NP}	4	1.7	d_{NP}	3.38
Humphrey–Percy	5	p_{HP}	5	9.1	d_{HP}	5.32
Humphrey–Nixon	6	p_{HN}	6	7.9	d_{HN}	5.32
Humphrey–Wallace	7	p_{HW}	7	7.4	d_{HW}	5.32
McGovern–Nixon	8	p_{MW}	8	2.3	d_{MW}	5.32
Percy–Wallace	9	p_{PW}	9	2.3	d_{PW}	5.32
McGovern–Wallace	10	p_{MW}	10	2.9	d_{MW}	5.32

better-fitting $\widehat{d}_{ij}$s with respect to fixed $d_{ij}(\mathbf{X})$ have to be found, subject to the constraints (9.10). Suppose that the order of the $d_{ij}(\mathbf{X})$s is exactly the same as the order of the proximities p_{ij}. Then, simply choosing $\widehat{d}_{ij} = d_{ij}(\mathbf{X})$ defines the optimal update. If the fixed $d_{ij}(\mathbf{X})$s are *not* in the same order as the proximities, the optimal update is found by *monotone regression* of Kruskal (1964b).

The Up-and-Down-Blocks Algorithm

We discuss the solution of minimizing $\sigma_r(\widehat{\mathbf{d}})$ by monotone regression with Kruskal's up-and-down-blocks algorithm. Consider an example. Rabinowitz (1975) describes a hypothetical experiment where a subject was asked to rank-order all possible pairs of the following politicians from most to least similar: Humphrey (H), McGovern (M), Percy (P), Nixon (N), and Wallace (W). The subject generated the ranking numbers exhibited in the second column of Table 9.2. They are shown in the form of the familiar proximity matrix in Table 9.3.

Now, assume that we have a first configuration $\mathbf{X}$, which leads to the distances in Table 9.2. How are the $\widehat{d}_{ij}$s computed? Consider the distances d_{ij} for the pairs Humphrey–McGovern and McGovern–Percy, d_{HM} and d_{MP}. The corresponding proximities are ordered as $p_{HM} < p_{MP}$. Because the proximities are dissimilarities (i.e., the smaller the p-value, the larger the similarity), $d_{HM} \leq d_{MP}$ should hold in a perfect MDS representation. This is obviously not true for the configuration $\mathbf{X}$, because it yields $d_{HM} = 7.8$ and $d_{MP} = 3.2$. Thus, the points of $\mathbf{X}$ must be moved so that d_{HM} becomes smaller and d_{MP} larger. Now, given two numbers, the arithmetical mean yields the number that is closest to both of them in the least-squares sense. Thus, setting $(d_{HM} + d_{MP})/2 = \widehat{d}_{HM} = \widehat{d}_{MP}$ defines target values that satisfy the requirements.

TABLE 9.3. Proximity matrix for politicians.

	H	M	P	W	N
H	–	1	5	7	6
M	1	–	2	10	8
P	5	2	–	9	4
W	7	10	9	–	3
N	6	8	4	3	–

TABLE 9.4. Derivation of the disparities in Table 9.2 by monotone regression.

			Trial Solutions for $\widehat{d}_{ij}$							Final
Pair	p_{ij}	$\widehat{d}_{ij}$	I	II	III	IV	V	VI	VII	$\widehat{d}_{ij}$
Humphrey–McGovern	1	7.8	5.5	3.93	3.38	3.38	3.38	3.38	3.38	3.38
McGovern–Percy	2	3.2	5.5	3.93	3.38	3.38	3.38	3.38	3.38	3.38
Nixon–Wallace	3	0.8	0.8	3.93	3.38	3.38	3.38	3.38	3.38	3.38
Nixon–Percy	4	1.7	1.7	1.7	3.38	3.38	3.38	3.38	3.38	3.38
Humphrey–Percy	5	9.1	9.1	9.1	9.1	8.5	8.13	6.68	5.8	5.32
Humphrey–Nixon	6	7.9	7.9	7.9	7.9	8.5	8.13	6.68	5.8	5.32
Humphrey–Wallace	7	7.4	7.4	7.4	7.4	7.4	8.13	6.68	5.8	5.32
McGovern–Nixon	8	2.3	2.3	2.3	2.3	2.3	2.3	6.68	5.8	5.32
Percy–Wallace	9	2.3	2.3	2.3	2.3	2.3	2.3	2.3	5.8	5.32
McGovern–Wallace	10	2.9	2.9	2.9	2.9	2.9	2.9	2.9	2.9	5.32

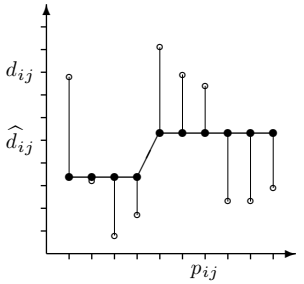


FIGURE 9.3. Shepard diagram of monotone regression as calculated in Tables 9.2 and 9.4. The open points represent pairs of corresponding proximities and distances (p_{ij}, d_{ij}), the solid points disparities $\widehat{d}_{ij}$. The solid line is the best-fitting monotone regression curve.

Beginning with the first pair of distances in Table 9.4, we get a first trial solution for the disparities by setting $5.5 = \widehat{d}_{HM} = \widehat{d}_{MP}$ and $\widehat{d}_{ij} = d_{ij}$ for all remaining distances. This yields the values in column I of Table 9.4. This trial solution, however, satisfies the monotonicity requirement only for its first two elements, and the third disparity value is too small, because $d_{NW} = 0.8$ is smaller than both of the preceding values. So, we create a new block by computing the average of the first three distances $(5.5 + 5.5 + 0.8)/3 = 3.93$. We then use 3.93 for $\widehat{d}_{HM}, \widehat{d}_{MP}$, and $\widehat{d}_{NW}$, and again hope that everything else is in order, thus setting $\widehat{d}_{ij} = d_{ij}$ for all other distances. This yields the second trial solution for the disparities (column II). This sequence still violates the monotonicity requirement in row 4. Hence, a new block is formed by joining the previous block and d_{NP}. The resulting disparities in column III form a weakly monotonic sequence up to and including row 5. In row 6, a value 7.9 turns up, however, that is smaller than the preceding one, 9.1. So, we join 9.1, 7.9, and all preceding values into one block, average these values, and so on. Table 9.4 shows all of the steps leading to the final disparity sequence of $\widehat{d}_{ij}$s in the last column. This completes monotone regression for the first iteration.

A Shepard diagram is given in Figure 9.3. In the main algorithm, we then have to normalize the $\widehat{d}_{ij}$ such that their sum-of-squares is equal to $n(n-1)/2$. Then, we start the second iteration by computing an update for the configuration $\mathbf{X}$. This gives new distances for which we can compute new disparities by monotone regression, as we have done above.

Smoothed Monotone Regression

A more restrictive version of ordinal MDS is *smoothed* monotone regression (Heiser, 1985, 1989a). Apart from the order restrictions implied by ordinal MDS, we also impose the restriction that the difference between differences

of adjacent disparities is never larger than the average disparity. Thus, if the $s = n(n-1)/2$ elements of vector $\widehat{\mathbf{d}}$ are ordered as the proximities, then smoothed monotone regression requires

$$\begin{array}{rcll} |(\widehat{d}_k - 0) - (0 - 0)| & \leq & s^{-1}\sum_{l=1}^{s}\widehat{d}_l, & \text{for } k = 1, \\ |(\widehat{d}_k - \widehat{d}_{k-1}) - (\widehat{d}_{k-1} - 0)| & \leq & s^{-1}\sum_{l=1}^{s}\widehat{d}_l, & \text{for } k = 2, \\ |(\widehat{d}_k - \widehat{d}_{k-1}) - (\widehat{d}_{k-1} - \widehat{d}_{k-2})| & \leq & s^{-1}\sum_{l=1}^{s}\widehat{d}_l, & \text{for } k = 3, \ldots, s. \end{array} \tag{9.11}$$

Thus, the restrictions are imposed on the difference of subsequent differences. The advantage of this internally bounded form of monotone regression is that the steps between two adjacent disparities can never get large. Therefore, the Shepard diagram always shows a smooth relation of p_{ij}s and $\widehat{d}_{ij}$s without irregular steps in the curve. For $k = 1$, the first restriction of (9.11) implies that $\widehat{d}_k$ should be between zero and the average d-hat. Therefore, a smoothed monotone transformation has a first d-hat that is quite close to zero and will be increasing in a smooth way. It can be verified that a quadratically increasing transformation and a logarithmically increasing transformation satisfy the maximal stepsizes as defined in (9.11). Unfortunately, Heiser reports that it is not easy to compute optimal disparities for given distances using smoothed monotone regression. Also, the smoothed monotone regression problem tends to become computationally demanding if n is large (say $n > 25$).

9.3 The Geometry of Monotone Regression

In the previous section, we saw how monotone regression is performed. Here, we give a geometrical explanation of monotone regression. Consider an example. Suppose that we have

$$\mathbf{P} = \begin{bmatrix} - & 1 & 3 \\ 1 & - & 2 \\ 3 & 2 & - \end{bmatrix}, \text{ and } \mathbf{D}(\mathbf{X}) = \begin{bmatrix} - & 1 & 2 \\ 1 & - & 3 \\ 2 & 3 & - \end{bmatrix},$$

and $w_{ij} = 1$ for each pair i, j. Let us reformulate the problem in simpler notation. Denote the unknown $\widehat{d}_{ij}$ as x_l, and the known $d_{ij}(\mathbf{X})$ as a_l. Also, we order the a_l in the rank-order of the proximities. This leads to Table 9.5. Rewriting σ_r accordingly, the monotone regression problem becomes minimizing

$$\sigma_r(\mathbf{x}) = \sum_{l=1}^{s}(x_l - a_l)^2$$

under the restriction that $0 \leq x_1 \leq x_2 \leq \cdots \leq x_s$, where $s = n(n-1)/2$.

TABLE 9.5. Reformulation of the monotone regression problem.

s	Pair i,j	Proximity	$a_s = d_{ij}(\mathbf{X})$	$x_s = \widehat{d}_{ij}$
1	12	p_{12}	$a_1 = 1$	$x_1 = \widehat{d}_{12}$
2	23	p_{23}	$a_2 = 3$	$x_2 = \widehat{d}_{23}$
3	13	p_{13}	$a_3 = 2$	$x_3 = \widehat{d}_{13}$

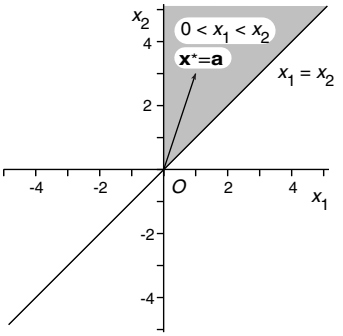


FIGURE 9.4. The area for which $0 \leq x_1 \leq x_2$.

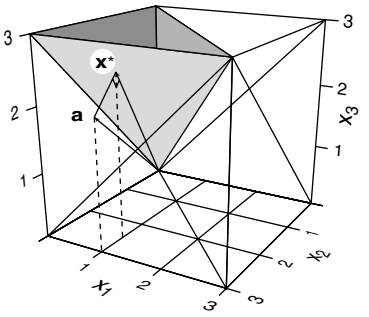


FIGURE 9.5. The area for which $0 \leq x_1 \leq x_2 \leq x_3$.

To see what these restrictions imply geometrically, consider the case where we have the restrictions $0 \leq x_1 \leq x_2$. The shaded area in Figure 9.4 shows the area in which these inequalities hold. Here, each axis denotes one of the variables x_l. The first part of the inequalities implies that all x_l should be nonnegative, because we do not want the disparities to become negative. The elements $a_1 = 1$ and $a_2 = 3$ fall in the shaded area, so that choosing $x_1^* = a_1 = 1$ and $x_2^* = a_2 = 3$ gives $\sigma_r(\mathbf{x})$ where the order restriction on x_1 and x_2 is not violated. If $\mathbf{a}$ were outside the shaded area, then we would have to find an $\mathbf{x}$ on the border of the shaded area that is closest to $\mathbf{a}$ by the up-and-down-blocks algorithm. The triple of inequalities $0 \leq x_1 \leq x_2 \leq x_3$ of our simple example can be represented graphically as in Figure 9.5. After orthogonal projection on each pair of axes, the area in which the inequalities hold is similar to that of Figure 9.4. The three inequalities combined give the inner part of the *ordered cone* in Figure 9.5. Monotone regression amounts to projecting $\mathbf{a}$ onto this cone. If $\mathbf{a}$ is ordered with increasing values, then it is located inside the cone. In this example, the $\mathbf{x}$ with the shortest distance to $\mathbf{a}$ that is in or on the ordered cone equals $\mathbf{x}^* = [1, 2.5, 2.5]'$.

Geometrically, monotone regression amounts to finding the $\widehat{\mathbf{d}}$ that is in the ordered cone (defined by the proximities) and as close as possible to the vector of distances.

TABLE 9.6. Calculation of the primary and the secondary approaches to ties in ordinal MDS for given distances d_{ij}.

			Primary Approach				Secondary Approach		
Pair	p_{ij}	d_{ij}	I	II	III	$\widehat{d}_{ij}$	I	II	$\widehat{d}_{ij}$
3,2	1	3	3	2.50	2.50	2.50	3	2.50	2.50
4,1	2	2	2	2.50	2.50	2.50	2	2.50	2.50
3,1	3	6	6	6	4.50	4.50	6	6	4.66
4,2	4	5	3	3	4.50	5	5	4	4.66
2,1	4	3	5	5	5	4.50	3	4	4.66
4,3	5	7	7	7	7	7	7	7	7

9.4 Tied Data in Ordinal MDS

In ordinal MDS, the relevant data information is the rank-order of the proximities. But consider the rank-order of the proximities in the following matrix.

$$\mathbf{P} = \begin{bmatrix} - & 4 & 3 & 2 \\ 4 & - & 1 & 4 \\ 3 & 1 & - & 5 \\ 2 & 4 & 5 & - \end{bmatrix}.$$

We see that the proximities p_{21} and p_{42} have the same ranks; that is, they are *tied*. How should such ties be represented in an MDS configuration? It would seem natural to represent them by equal distances in an MDS solution, but this is known as the *secondary approach to ties*. For our simple example, it means that $\widehat{d}_{21} = \widehat{d}_{42}$, so that $\widehat{d}_{31} \leq \widehat{d}_{21} = \widehat{d}_{42} \leq \widehat{d}_{43}$. In the *primary approach*, tied proximities impose *no* restrictions on the corresponding distances. In other words, it is not necessary to map tied data into equal distances. For our example, the primary approach to ties implies $\widehat{d}_{31} \leq \widehat{d}_{21} \leq \widehat{d}_{43}$ and $\widehat{d}_{31} \leq \widehat{d}_{42} \leq \widehat{d}_{43}$. Nothing is required of the distances representing equal proximities, except that they must be smaller (larger) than the distances corresponding to smaller (larger) proximities. Ties in the data thus can be *broken* in the representing distances.

In Table 9.6, an example is presented of the calculation for the primary and secondary approaches to ties for given distances. The resulting Shepard diagrams are shown in Figure 9.6. In the primary approach, the first estimate of the disparities is obtained by setting $\widehat{d}_{ij} = d_{ij}$ and then reordering these $\widehat{d}_{ij}$ wherever they correspond to tied p_{ij} values so that they increase monotonically. Then, standard monotone regression is applied (see Section 9.2). Finally, the resulting disparities are permuted back into the original order of the distances. The secondary approach to ties follows the same strategy as monotone regression, except that the first disparity estimates for tied data are replaced by their average values.

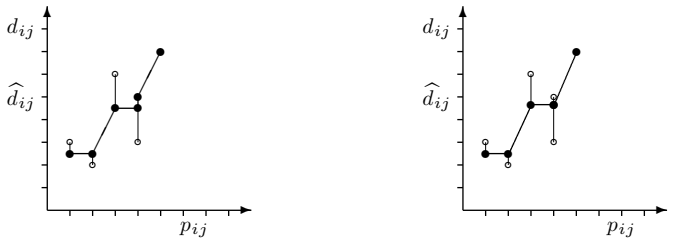


FIGURE 9.6. Shepard diagram of monotone regression calculated with the primary approach to ties (a), or the secondary approach to ties (b) (see Table 9.6). The solid points are the disparities $\widehat{d}_{ij}$, and the open points are the distances $d_{ij}(\mathbf{X})$.

How do ties arise in proximity data? There are several possibilities. Consider the following case. Assume that we want to find out how people perceive cars by studying their similarity impressions. A stack of cards is prepared, where each card shows one pair of cars. The subject is asked to split the stack into two piles, one containing the more similar pairs, the other the more dissimilar pairs. Then, the subject is asked to repeat this exercise for each pile in turn, and repeat again, and so on, until he or she feels that it is not possible to discriminate any further between the pairs of cars in any pile. If the subject stops when each pile has only one card left, then we get a complete similarity order of the pairs of cars and no ties occur. It is more likely, however, that some of the final piles will have more than one card. Most likely, the piles for the extremely similar pairs will be quite small, whereas those for pairs with intermediate similarity will be larger. This means that if we assign the same proximity value to all pairs in a pile, ties will arise for every pile containing one or more cards. However, we would not want to assume that these data are tied because the subject feels that the respective pairs of cars are exactly equal. Rather, the subject stops the card sorting only because the pairs in some piles do not appear to be sufficiently different for a further meaningful or reliable ordering. Hence, the primary approach to ties should be chosen in analyzing these data.

Consider another example, a pilot study on the perception of nations (Wish, Deutsch, & Biener, 1970; Wish, 1971), where the respondents had to judge the degree of similarity between each pair of 12 nations on a 9-point rating scale with endpoints labeled as "very different" and "very similar", respectively. Here, the proximities for each respondent must have ties, because there are 66 pairs of nations, and, thus, it would require a rating scale with at least 66 categories in order to be able to assign a different proximity value to every stimulus pair. The 9-point rating scale works as a relatively coarse sieve on the true similarities, so the data would be best interpreted as indicators for intervals on a continuum of similarity.

The primary approach to ties is again indicated, inasmuch ties must result due to the data collection method.

A further way for ties to occur is when the proximities are derived from other data. Consider the correlation matrix of intelligence tests in Table 5.1. Several ties occur here, so that with the primary approach to ties, the distances d_{17}, d_{24}, and d_{27}, say, are merely required to be less than d_{37} and greater than d_{26}. However, we can compute the correlation coefficients to more decimal places. Assume that we get, by using three decimal places, $r_{17} = .261$, $r_{24} = .263$, and $r_{27} = .259$. In ordinal MDS, it should then hold that $d_{24} < d_{17} < d_{27}$, and so the MDS solution must satisfy additional properties. But is it worthwhile to place such stronger demands on the solution? Clearly not. The correlations may not even be reliable to three decimal places. Even the value of $r = .26$ should be read as $r \approx .26$. Hence, the secondary approach to ties makes no sense here.

9.5 Rank-Images

A completely different way of computing disparities in *ordinal* MDS is based on *rank-images*. The basic idea is that if a perfect fit exists in ordinal MDS, then the rank-order of the distances must be equal to the rank-order of the proximities. To compute the disparities, a switch is made to a loss function that is different from Stress; that is,

$$\tau(\widehat{\mathbf{d}}) = (\mathbf{R}_p\widehat{\mathbf{d}} - \mathbf{R}_d\mathbf{d})'(\mathbf{R}_p\widehat{\mathbf{d}} - \mathbf{R}_d\mathbf{d}), \tag{9.12}$$

where we assume for simplicity that all the weights w_{ij} are one in the Stress function. $\mathbf{R}_p$ is a permutation matrix (that has only a single one in each row and column, and zeros elsewhere) such that $\mathbf{R}_p\mathbf{p}$ is the vector of proximities ordered from small to large. Similarly, $\mathbf{R}_d$ is a permutation matrix that orders the distances $\mathbf{d}$ from small to large. $\mathbf{R}_p$ is known, the vector of distances $\mathbf{d}$ is known, and thus $\mathbf{R}_d$ is known. The only unknown vector is the vector of disparities $\widehat{\mathbf{d}}$ that we intend to find. To find the minimum of (9.12) we use the fact that $\mathbf{R}'\mathbf{R} = \mathbf{I}$ for any permutation matrix $\mathbf{R}$. Equation (9.12) is a quadratic function in $\widehat{\mathbf{d}}$, so that its minimum can be found in one step by setting the gradient (first derivative)

$$\nabla\tau(\widehat{\mathbf{d}}) = 2\mathbf{R}_p'\mathbf{R}_p\widehat{\mathbf{d}} - 2\mathbf{R}_p'\mathbf{R}_d\mathbf{d} = 2\widehat{\mathbf{d}} - 2\mathbf{R}_p'\mathbf{R}_d\mathbf{d}$$

equal to zero for all elements: $\nabla\tau(\widehat{\mathbf{d}}) = \mathbf{0}$ implies $\widehat{\mathbf{d}} = \mathbf{R}_p'\mathbf{R}_d\mathbf{d}$ (and $\mathbf{R}_p\widehat{\mathbf{d}} = \mathbf{R}_d\mathbf{d}$). If the proximities are already ordered increasingly, then $\mathbf{R}_p = \mathbf{I}$ and the rank-image transformation amounts to setting the disparities equal to the ordered distances.

A flaw of using rank-images for ordinal MDS is that convergence of the overall algorithm cannot be guaranteed. This is caused by the switch from

TABLE 9.7. Derivation of rank-image disparities from the politicians data given in Tables 9.2 and 9.3.

Pair	$\mathbf{R}_p\mathbf{p}$	$\mathbf{R}_p\mathbf{d}$	$\mathbf{R}_d\mathbf{d}$	$\mathbf{R}_p\widehat{\mathbf{d}}$
Humphrey–McGovern	1	7.8	0.8	0.8
McGovern–Percy	2	3.2	1.7	1.7
Nixon–Wallace	3	0.8	2.3	2.3
Nixon–Percy	4	1.7	2.3	2.3
Humphrey–Percy	5	9.1	2.9	2.9
Humphrey–Nixon	6	7.9	3.2	3.2
Humphrey–Wallace	7	7.4	7.4	7.4
McGovern–Nixon	8	2.3	7.8	7.8
Percy–Wallace	9	2.3	7.9	7.9
McGovern–Wallace	10	2.9	9.1	9.1

the Stress loss function to the loss function (9.12). This could be solved by trying to minimize the same function (9.12) for updating the configuration $\mathbf{X}$. However, because $\mathbf{R}_d$ is dependent on the distances and thus on $\mathbf{X}$, it is very hard to minimize (9.12) over $\mathbf{X}$. Nevertheless, we can still use rank-images in the SMACOF algorithm, although convergence is no longer guaranteed. As De Leeuw and Heiser (1977) remark: "It is, of course, perfectly legitimate to use the rank-images ... in the earlier iterations (this may speed up the process, cf. Lingoes & Roskam, 1973). As long as one switches to [monotone regression] in the final iterations convergence will be achieved" (p. 742). Lingoes and Roskam (1973) do exactly this in their MINISSA program, because they claim that "the rank-image transformation is more robust against trivial solutions and local minima" (Roskam, 1979a, p. 332).

As an example of the calculation of rank-images, we again use the data on the similarity of politicians from Table 9.2. The proximities are already ordered from small to large, so that $\mathbf{R}_p = \mathbf{I}$. The disparities according to the rank-image transformation are given in Table 9.7.

In Guttman (1968) and in some computer programs, rank-images are denoted by d^*_{ij} (d-star) as opposed to $\widehat{d}_{ij}$ (d-hat) obtained by monotone regression. Here, we retain the notation of $\widehat{d}_{ij}$ for a disparity, even if the disparity is a rank-image.

9.6 Monotone Splines

Quite flexible transformations are obtained by using splines. We show that special cases of (monotone) splines include interval transformations, polynomial transformations, and ordinal transformations. In this section, we limit ourselves to the class of monotone splines, which are also called *I-splines* (integrated splines) in the literature. Whenever we refer to a spline

in the sequel, we mean a monotone spline. One of its main characteristics is that the resulting transformation is smooth. For a good review of applications of monotone splines in statistics, we refer to Ramsay (1988). For more general references on splines, see De Boor (1978) and Schumaker (1981).

There are three reasons for wanting a smooth transformation in MDS. First, ordinal MDS can result in a crude transformation. For example, in Figure 9.3 the rank-order of ten different proximities was transformed in only two different disparities. Such crude transformations neglect much of the variation in the proximities. A second reason is that we want to retain more than ordinal information of the data. For example, if the proximities are correlations, we may want to consider more than just the rank-orders of the correlations (as in ordinal MDS), but less than the interval information (as in interval MDS). Third, degenerate solutions (see Chapter 13) can be avoided by imposing smooth transformations. In general, a spline transformation yields a much smoother transformation curve than an ordinal transformation. Compare, for example, the nonsmooth ordinal transformation in Figure 9.1 and the smooth spline transformation in the same figure. Thus, splines can be used to obtain smooth transformation curves, while keeping the ordinal information of the proximities intact.

Characterization of Monotone Splines

What does a spline transformation look like? In general, the transformation is a smooth monotone increasing curve. The conceptual idea is that it is not possible to map *all* proximities into disparities by one simple function (such as the linear transformation in interval MDS). Then, splines can be used to specify such simple mappings for several intervals. The additional restriction on the separate transformation of each interval is that they should be smoothly connected and monotone increasing. We discuss later that interval and ordinal transformations are two extreme cases of monotone spline transformations. Hence, other spline mappings can be seen as more restrictive than ordinal mappings and less restrictive than linear mappings.

The endpoints (extrema) of the intervals are called *knots*. Because splines are required to be smooth, the endpoint of one interval coincides with an extremal point of the adjacent interval, so that a knot ties together the two intervals. The size of the intervals is characterized by the *knot* sequence of the knots t_i. As before in this chapter, we string out the $s = n(n-1)/2$ proximities in the vector $\mathbf{p}$ and index its elements by i, where $i = 1, 2, \ldots, s$. We also assume that the elements in $\mathbf{p}$ are ordered increasingly. Two knots are reserved, one for the smallest value of the proximities $t_0 = p_{\min}$ and the other for the largest value $t_m = p_{\max}$. The other knots, if present, are called *interior knots*, because they must be greater than t_0 and smaller than t_m. Thus, the ordered knot sequence of the m knots $t_0 = p_{\min}, t_1, t_2, \ldots, t_m = p_{\max}$ defines the intervals $[t_0, t_1]$, $[t_1, t_2]$, ... , $[t_{m-1}, t_m]$, so that every observed value p_{ij} falls into one of these intervals.

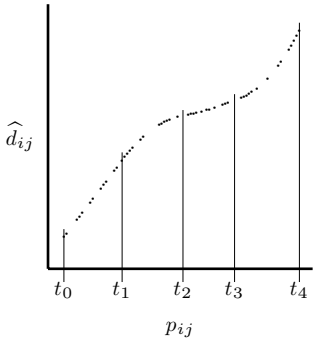


FIGURE 9.7. Example of a spline transformation with three interior knots.

Usually, the interior knots are placed at $K-1$ quantiles so that the K intervals are equally filled with proximities. Figure 9.7 shows an example of a spline transformation with three interior knots that define four intervals.

The smoothness within an interval is guaranteed by choosing the transformation as a polynomial function of the proximities. Examples of polynomial functions are $f(p) = 3p^2 - 2p + 1$ (a second degree polynomial), and $f(p) = 6p - 3$ (a first degree polynomial). In general, a polynomial function of degree r is defined as $f(p) = \sum_{k=0}^{r} a_k p^k$, where a_k are weights and $p^0 = 1$. The degree r of the polynomial is specified by the *order* of the spline, or the *degree* of the spline. Because the entire spline transformation must be smooth, we must also have smoothness between the intervals at the knots. The smoothness at the knots is also determined by the order of the spline in the following way: at knot t_i, the first $r-1$ derivatives of two polynomials of the adjacent intervals $[t_{i-1}, t_i]$ and $[t_i, t_{i+1}]$ must be equal. For a spline of order 1, this property implies that the lines are joined at each interior knot, so that the transformation is continuous. A quadratic spline has—apart from continuity—equal first derivatives at each interior knot. A third-order spline has continuity up to the second derivatives at the interior knots, and so on. Note that a spline of order 0 is not even continuous.

It remains to be seen how a spline transformation can be computed. Suppose that we specify a spline of degree r with k interior knots. It turns out that the spline transformation (with the properties outlined above) can be computed by using a special $s \times (r + k)$ matrix $\mathbf{M}$ that can be derived from $\mathbf{p}$. The spline transformation is defined simply as $\widehat{\mathbf{d}} = \mathbf{Mb}$ for any vector of nonnegative weights $\mathbf{b}$. Viewed this way, finding a spline transformation is nothing more than solving a multiple regression problem for optimal weights $\mathbf{b}$. These weights are used to predict the fixed distances $\mathbf{d}$ by the weighted sum $\widehat{\mathbf{d}} = \mathbf{Mb}$. We restrict $\mathbf{b}$ to be nonnegative, which ensures that the transformation is monotone increasing.

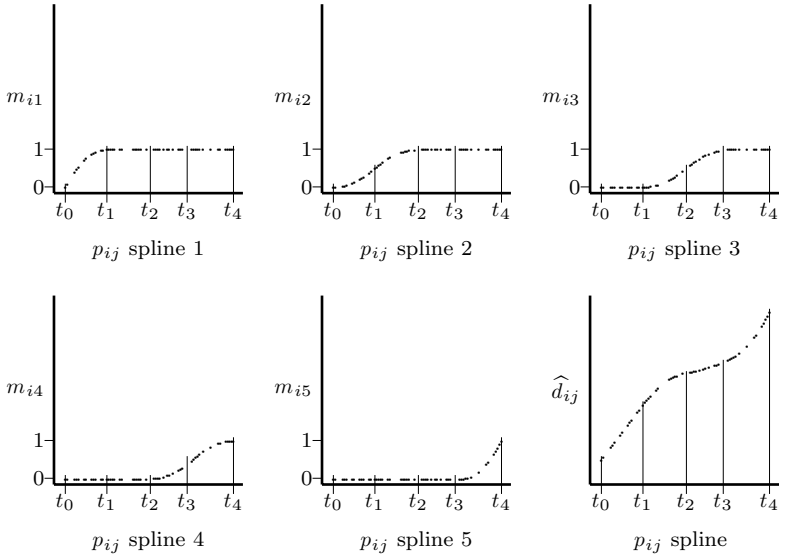


FIGURE 9.8. Separate columns of spline bases $\mathbf{M}$ belonging to a monotone spline of order 2 with three interior knots. The plot on the bottom right is a transformation resulting from a weighted sum of the previous five columns of $\mathbf{M}$ plus an intercept.

Specifying the Matrix $\mathbf{M}$

The crux of a spline transformation lies in how the matrix $\mathbf{M}$ is set up. It turns out that for monotone splines each column is a piecewise polynomial function of $\mathbf{p}$, in such a way that any linear combination satisfies the required smoothness restrictions at the knots. To accomplish this, matrix $\mathbf{M}$ has a special form. The elements of column j of $\mathbf{M}$ in the first $\max(0, j-r)$ intervals are equal to 0, and the elements in the last $\max(0, k-j+1)$ intervals are equal to 1. The remaining intervals contain a special polynomial function of degree r, which we specify below for splines of orders zero, one, and two. Figure 9.8 shows an example of the columns of $\mathbf{M}$ as a function of $\mathbf{p}$ for $k = 3$ and $r = 2$. The first column $\mathbf{m}_1$ has elements equal to 1 in the last three intervals, the second column $\mathbf{m}_2$ has elements 1 in the last two intervals, the third column $\mathbf{m}_3$ has 0s in the first interval and 1s in the last interval, the fourth column $\mathbf{m}_4$ has 0s in the first and second intervals, and the fifth and final column has 0s in the first, second, and third intervals. The values in the intervals that are not 0 or 1 are a quadratic function in p_{ij} that is continuous and has equal derivatives at the knots.

We now come to explicit expressions for splines of orders zero, one, and two. The columns of $\mathbf{M}$ for an order-zero spline are defined by an indicator function that is 0 if p_i is smaller than knot j and 1 otherwise; that is, the

spline basis $\mathbf{M}$ is an $s \times k$ matrix with elements

$$m_{ij} = \begin{cases} 0 & \text{if } t_0 \le p_i < t_j, \\ 1 & \text{if } t_j \le p_i < t_{k+1}. \end{cases}$$

If the number of interior knots k is 0, then $\mathbf{M}$ is not defined in a zero-order spline, because all values p_i fall in the same interval $[t_0, t_1]$, so that $m_i = 1$ for all i. Clearly, for our purpose, the transformation $\hat{d}_{ij} = 1$ for all i, j is not acceptable, because it ignores the variability in the observed proximities.

The columns of $\mathbf{M}$ of an order-one spline are defined by a piecewise linear function; that is,

$$m_{ij} = \begin{cases} 0 & \text{if } t_0 \le p_i < t_{j-1}, \\ \dfrac{p_i - t_{j-1}}{t_j - t_{j-1}} & \text{if } t_{j-1} \le p_i < t_j, \\ 1 & \text{if } t_j \le p_i \le t_{k+1}. \end{cases}$$

For a monotone spline of order two, we can write a direct formulation of the elements of $\mathbf{M}$; that is,

$$m_{ij} = \begin{cases} 0 & \text{if } t_0 \le p_i < t_{j-2}, \\ \dfrac{(t_{j-2} - p_i)^2}{(t_{j-1} - t_{j-2})(t_j - t_{j-2})} & \text{if } t_{j-2} \le p_i < t_{j-1}, \\ 1 - \dfrac{(t_j - p_i)^2}{(t_j - t_{j-1})(t_j - t_{j-2})} & \text{if } t_{j-1} \le p_i < t_j, \\ 1 & \text{if } t_j \le p_i < t_{k+1}, \end{cases}$$

after Ramsay (1988). Note that for $j = 1$ we have reference to $t_{j-2} = t_{-1}$, which we define as $t_{-1} = t_0$. Equivalently, for $j = k + 1$ we define knot $t_j = t_{k+1}$. The lower-right plot in Figure 9.8 plots the proximities against $\hat{\mathbf{d}} = \mathbf{Mb}$ for some given vector $\mathbf{b}$. For the calculation of monotone splines of higher order and for more general information on splines, we refer to Ramsay (1988) and De Boor (1978).

Let the proximity matrix $\mathbf{P}$ be given by

$$\mathbf{P} = \begin{bmatrix} 0 & 1.0 & 1.5 & 3.2 \\ 1.0 & 0 & 2.0 & 3.8 \\ 1.5 & 2.0 & 0 & 4.5 \\ 3.2 & 3.8 & 4.5 & 0 \end{bmatrix},$$

or in vector notation $\mathbf{p}' = (1.0, 1.5, 2.0, 3.2, 3.8, 4.5)$. Let the knots be given by $t_0 = 1.0$, $t_1 = 3.0$, $t_2 = 4.5$, so that the number of interior knots k equals 1. For these data Table 9.8 shows $\mathbf{M}$ for a zero-order spline, a first-order spline, and a second-order spline.

The spline basis $\mathbf{M}$ of $\mathbf{p}$ is invariant under linear transformation of $\mathbf{p}$. It turns out that by choosing the two extrema as knots, we obtain a row

TABLE 9.8. Example of spline bases $\mathbf{M}$ for a zero-order spline, a first-order spline, and a second-order spline. The knots are $t_0 = 1.0$, $t_1 = 3.0$, $t_2 = 4.5$.

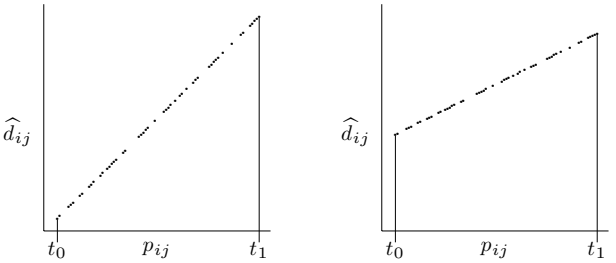
	$r = 0$	$r = 1$		$r = 2$		
$\mathbf{p}$	$\mathbf{m}_1$	$\mathbf{m}_1$	$\mathbf{m}_2$	$\mathbf{m}_1$	$\mathbf{m}_2$	$\mathbf{m}_3$
1.0	0	0.00	0.00	0.00	0.00	0.00
1.5	0	0.25	0.00	0.44	0.04	0.00
2.0	0	0.50	0.00	0.75	0.14	0.00
3.2	1	1.00	0.13	1.00	0.68	0.02
3.8	1	1.00	0.53	1.00	0.91	0.28
4.5	1	1.00	1.00	1.00	1.00	1.00

of 0s for the smallest proximity and a row of 1s for the largest proximity, as can be verified in the examples of Table 9.8. This implies that whatever the weights $\mathbf{b}$, the disparity of the smallest proximity will be 0. This is not desirable in MDS, because the smallest proximity does not necessarily have to be represented by a zero distance. Therefore, we include a positive intercept in our spline transformation; that is, $\widehat{\mathbf{d}} = b_0\mathbf{1} + \mathbf{Mb}$. For MDS, we need the intercept, so that the disparity corresponding to the smallest proximity can be transformed into any nonnegative value.

Special Cases of Monotone Splines

Let us look at two special cases of monotone splines. The first case is a spline with order larger than zero ($r > 0$) and no interior knots ($k = 0$), so that there are only two knots, one at the smallest value of $\mathbf{p}$ and one at the largest value of $\mathbf{p}$. For this case, monotone splines have the property that the row sum of $\mathbf{M}$ is equal to $c\mathbf{p}$ (with $c > 0$ an arbitrary factor); that is, $cp_i = \sum_j m_{ij}$. An example of a transformation plot for this case is given in Figure 9.9a. A second property is that such spline transformations are equivalent to transformations obtained by polynomial regression of the same degree. If we deal with a first-order spline, then $\mathbf{M}$ consists of one column only that is linearly related to p. Therefore, a first-order spline with two knots and an intercept is equivalent to an interval transformation, as can be seen in Figure 9.9b.

The second special case of a monotone spline occurs if exactly $k = n - 1$ interior knots and the order $r = 0$ are specified. If an intercept is included, then this is equivalent to performing monotone regression. A small example clarifies this statement. Let the proximities be $\mathbf{p}' = (1, 2, 3, 4, 5)$ and the knots be at $\mathbf{t}' = (0.5, 1.5, 2.5, 3.5, 4.5, 5.5)$. Then, the matrix $\mathbf{M}$ of a



a. Sum of splines with two knots.

b. Order-one spline with two knots.

FIGURE 9.9. Special cases of spline transformation: (a) all weights $b_i = 1$, two knots, and order larger than zero; (b) spline with two knots and order one, which is equal to an interval transformation if an intercept is included.

zero-order spline is equal to

$$\mathbf{M} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}.$$

For monotone splines, we require weights $\mathbf{b}$ to be larger or at most equal to 0, so that $\mathbf{Mb}$ plus an intercept $b_0\mathbf{1}$ (with $b_0 \geq 0$) is always larger than 0. The matrix multiplication plus the intercept results in

$$\begin{aligned} \widehat{\mathbf{d}} &= b_0\mathbf{1} + \mathbf{Mb} = b_0 \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} \\ &= \begin{bmatrix} b_0 \\ b_0 + b_1 \\ b_0 + b_1 + b_2 \\ b_0 + b_1 + b_2 + b_3 \\ b_0 + b_1 + b_2 + b_3 + b_4 \end{bmatrix}. \end{aligned} \tag{9.13}$$

But the restrictions $b_j \geq 0$ for $j = 0$ to 4 in (9.13) imply that $0 \leq \widehat{d}_1 \leq \widehat{d}_2 \leq \widehat{d}_3 \leq \widehat{d}_4 \leq \widehat{d}_5$, which is exactly the same restriction as in monotone regression. Thus, a zero-order monotone spline transformation with appropriately chosen knots is exactly equal to a monotone regression transformation.

Therefore, a monotone spline transformation can be seen as a general transformation with linear and ordinal transformations as extreme cases.

Solving the Nonnegative Least-Squares Problem for Monotone Splines

How can we calculate the disparities $\widehat{\mathbf{d}}$ for a monotone spline transformation? Remember that the disparities for splines with intercept are defined by $\widehat{\mathbf{d}} = \mathbf{Mb}$, where $\mathbf{M}$ here is augmented with a column of 1s for the intercept and the weight vector $\mathbf{b}$ is augmented with element b_0 for the intercept. We have to find weights b_j such that they are as close as possible to the (fixed) distance vector $\mathbf{d}$, subject to the constraints that $b_j \geq 0$. Thus, we have to minimize

$$\tau(\mathbf{b}) = (\mathbf{d} - \widehat{\mathbf{d}})'(\mathbf{d} - \widehat{\mathbf{d}}) = (\mathbf{d} - \mathbf{Mb})'(\mathbf{d} - \mathbf{Mb}), \tag{9.14}$$

subject to $b_j \geq 0$. Minimizing $\tau(\mathbf{b})$ over $\mathbf{b}$ is a *nonnegative least-squares* problem. It can be solved by alternating least squares (ALS), which, in this case, amounts to the following strategy. First, start with an initial weight vector $\mathbf{b}$, with $b_j \geq 0$. Then, fix all weights except b_j. Then, compute $\mathbf{r} = \mathbf{d} - \sum_{l \neq j} b_l \mathbf{m}_l$, where $\mathbf{m}_j$ denotes column j of matrix $\mathbf{M}$. Problem (9.14) simplifies into

$$\tau(b_j) = (\mathbf{r} - b_j \mathbf{m}_j)'(\mathbf{r} - b_j \mathbf{m}_j) = \mathbf{r}'\mathbf{r} + b_j^2 \mathbf{m}_j'\mathbf{m}_j - 2b_j \mathbf{m}_j'\mathbf{r},$$

which reaches its unconstrained minimum at $b_j = \mathbf{m}_j'\mathbf{r}/\mathbf{m}_j'\mathbf{m}_j$. If $b_j < 0$, then we set $b_j = 0$. Then, we update the next weight, while keeping the other weights fixed, compute the unconstrained minimum (if negative, then set it to zero), and so on, until we have updated all of the weights once. These steps define one iteration of the alternating least-squares algorithm, because every weight b_j has been updated once. Iterate over this process until the weights $\mathbf{b}$ do not change anymore. It can be proved that this alternating least-squares algorithm always reaches a global minimum of the nonnegative least-squares problem. A different strategy for solving (9.14) under nonnegative constraints is described in Lawson and Hanson (1974, p. 161).

9.7 A Priori Transformations Versus Optimal Transformations

In data analysis it is not uncommon to preprocess the data to make their distribution more "normal." The researcher may want to preprocess his or her dissimilarity data with a similar goal in mind. It may appear more attractive from a theoretical point-of-view not to optimally transform dissimilarities into d-hats by "some" monotonic function, but to apply a fixed a priori transformation on them. One such choice was suggested by Buja and Swayne (2002): they recommend using a power transformation of the

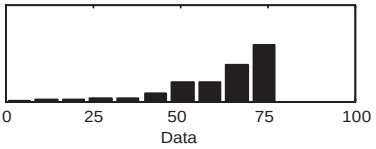
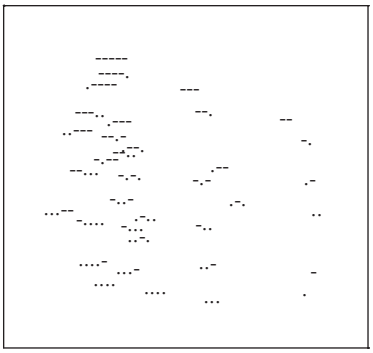


FIGURE 9.10. Distribution of the Morse code dissimilarities.

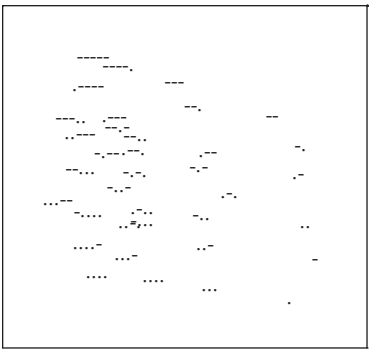
dissimilarities; that is, $\widehat{d}_{ij} = \delta_{ij}^q$ with q any positive or negative value. A positive value of q yields a convex transformation that stretches the larger dissimilarities and shrinks the smaller ones. In the case where the distribution of the dissimilarities is negatively skewed (thus with relatively many large values and few small values), then a positive q will make the $\widehat{d}_{ij}$s more evenly distributed. For negative q, the power transformation has a concave form thereby shrinking the larger dissimilarities and stretching the smaller ones. For dissimilarities that have a positively skewed distribution (i.e., data with few large and many small values), a negative q stretches the larger values and shrinks the smaller ones. The larger (or smaller) q, the stronger the shrinking and stretching. Values of q close to zero or exactly equal to zero are not very informative as all d-hats become the same; that is, $\widehat{d}_{ij} = \delta_{ij}^0 = 1$ for all ij. These d-hats can be seen as totally uninformative because they do not depend on the data (see also Section 13.3).

Let us consider the Morse code data from Section 4.2. To apply MDS, we first have to symmetrize the similarities in Table 4.2. To apply the power transformation, we also need to transform the similarities into dissimilarities. This was done by setting $\delta_{ij} = \max_{ij}((s_{ij} + s_{ji})/2) - (s_{ij} + s_{ji})/2$ thereby ensuring that the smallest δ_{ij} is zero and the largest is equal to $\max_{ij}((s_{ij} + s_{ji})/2)$. Note that Buja and Swayne (2002) also extensively discuss the Morse code data but use a different way of constructing the dissimilarities. Figure 9.10 shows the distribution of the dissimilarities obtained this way. This distribution has a tail to the left (a negatively skewed distribution), so that there are more large dissimilarities than small dissimilarities. A power transformation using $q = 3.1$ yields the distribution in Figure 9.11e which seems to be reasonably evenly distributed. One way to find out how to choose the value of q is simply trying out different values and see which one gives the best Stress value. In our case, the optimal value for q was 3.1.

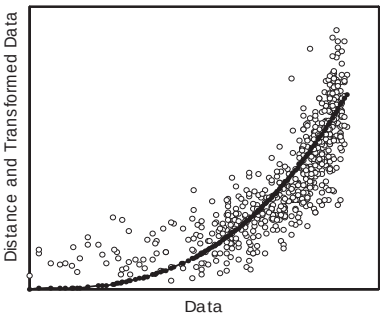
We now compare the power transformation to an ordinal MDS on these data. Figure 9.11 exhibits the results for both analyses with the left panels showing the results of a power transformation and the right panels the ordinal MDS results. The Stress-1 for the power transformation is .2290 and for ordinal MDS 0.2102 indicating that only a little information is lost by switching from ordinal to a power transformation. Looking at the distributions of the $\widehat{d}_{ij}$s, the ordinal transformation seems to be better able to stretch the smaller values than the the power transformation. Thus, the



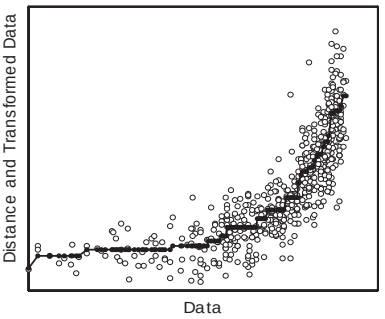
a. Solution power transformation.



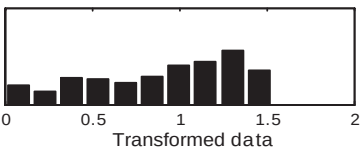
b. Solution ordinal MDS.



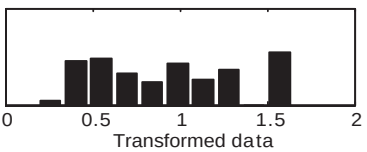
c. Shepard diagram power transformation.



d. Shepard diagram ordinal MDS.



e. Distribution of d-hats of the power transformation.



f. Distribution of d-hats of ordinal MDS.

FIGURE 9.11. MDS of the Morse code data using the power transformation with $q = 3.1$ (left panels) and an ordinal MDS (right panels).

gain in fit is due to the better ability of the ordinal MDS to properly represent the smaller dissimilarities, because their errors in the Shepard plot are smaller for ordinal MDS than for the power transformation. The solutions in Figures 9.11a and 9.11b are highly similar. Close inspection reveals small differences in location, perhaps most notably so for points "– – – – ." (9), "." (e), and "–" (t).

The example shows that a power transformation using only a single parameter can yield an MDS solution that is close to ordinal MDS. Clearly, a power transformation is a more parsimonious function than an ordinal transformation. A strong point of Buja and Swayne (2002) is to consider the distribution of the $\widehat{d}_{ij}$s. We conjecture that good transformations tend to give d-hats that are evenly distributed. For dissimilarities that have an "irregular" shape (e.g., a bimodal shape), we expect that the power transformation will not be able to yield a solution close to an ordinal one. For regularly shaped but skewed distributions, we expect the power transformation to work fine.

Applying the power transformation in MDS is easily done in the GGVIS software discussed extensively in Buja and Swayne (2002) (see also, Appendix A). In an interactive way, GGVIS allows you to determine the optimal q. Note that SYSTAT has a special option to find the optimal q by the program itself.

9.8 Exercises

Exercise 9.1 Consider the dissimilarity data in Exercise 2.4 and the MDS coordinates for these data in Exercise 3.2. For convenience, they are both reproduced in the table below.

Dissimilarities

Color	Red	Orange	Green	Blue
Red	-	1	3	5
Orange	1	-	2	6
Green	3	2	-	4
Blue	5	6	4	1

MDS Coordinates

Dim.1	Dim. 2
0	2
0	0
4	0
6	6

(a) String out the dissimilarities for the different pairs of colors in a column vector.

(b) Compute the MDS distances from the points' coordinates, and append a column with these distances to the vector of dissimilarities from above.

(c) Derive the $\widehat{d}_{ij}$s for the data-distance pairs, proceeding as we did above in Table 9.4.

(d) Plot a Shepard diagram for the data, distances, and $\widehat{d}_{ij}$s.

(e) Find the rank-images of the distances.

(f) Make a scatter plot of the distances vs. the rank-images. What does that plot tell you about the MDS solution?

Exercise 9.2 Discuss the Lingoes–Roskam conjecture that rank-images are less prone to degenerated solutions than monotone regression in Kruskal's sense. What is the rationale for this conjecture?

Exercise 9.3 Consider the notion of primary and secondary approaches to ties in ordinal MDS.

(a) List arguments or describe circumstances where the primary approach makes more sense than the secondary approach.

(b) Collect and discuss arguments in favor of the secondary approach.

Exercise 9.4 Consider the transformation plots in Figure 9.1. Sketch some monotone functions that satisfy the primary approach to ties. How do they differ from functions for the secondary approach to ties? (Hint: Consider Figure 3.3.)

10
Confirmatory MDS

If more is known about the proximities or the objects, then additional restrictions (or constraints) can be imposed on the MDS model. This usually means that the MDS solutions must satisfy certain additional properties of the points' coordinates or the distances. These properties are derived from substantive considerations. The advantage of enforcing such additional properties onto the MDS model is that one thus gets direct feedback about the validity of one's theory about the data. If the Stress of a confirmatory MDS solution is not much higher than the Stress of a standard ("unconstrained") MDS solution, the former is accepted as an adequate model. Several procedures that allow one to impose such external constraints are described and illustrated.

10.1 Blind Loss Functions

In most MDS applications discussed so far, we did not just represent the data geometrically and then interpret the solutions, but started by formulating certain predictions on the structure of the MDS configuration. For example, in Section 4.1, it was conjectured that the similarity scores on the colors would lead to a circular point arrangement in a plane (color circle). In Section 4.3, it was predicted that the similarity data on facial expressions could be explained by a 3D coordinate system with specified axes. However, these predictions had no influence on the MDS solution. Rather, structural hypotheses were dropped when the data were handed over to

an MDS computer program. The MDS program optimizes Stress, which is substantively blind; that is, it is not tailored to the particular questions that are being asked. The program mechanically grinds out "some" optimal distance representation under a few general restrictions such as the dimensionality m and the admissible transformations on the proximities.

Minimizing Stress gives a solution that is locally optimal. Yet, other local minimum solutions may exist with a similar Stress, or possibly even with lower Stress (see also Section 13.4). The question is which solution should be preferred. If a hypothesis for the data is available, then, of course, we would be particularly interested in the solution that most directly speaks to this hypothesis. This is obviously the solution that most closely satisfies the hypothesis, even if its Stress is somewhat higher than the Stress for other solutions.

Assume, for example, that we had not obtained the color circle in Figure 4.1 because the computer program succeeded in finding a solution with a lower Stress value. Assume further that the formally optimal but theoretically unintelligible solution had Stress .05, but the one matching our predictions had .06. Having had only the Stress-optimal solution, we probably would have concluded—incorrectly—that the predictions were wrong. Thus, what we want is a method that guarantees that the solution satisfies our expectations. Once we have it, we can decide whether this solution has an acceptable fit to the data.

10.2 Theory-Compatible MDS: An Example

Consider an example. Noma and Johnson (1977) asked subjects to assess the similarity of 16 ellipses having different shapes and sizes. The ellipses were constructed according to the design shown in Figure 10.1. The horizontal dimension of this design configuration is eccentricity ("shape"), and the vertical, area ("size").[1] The design shows, for example, that ellipse 4 is very flat and long, but 13 is more circular and also larger.[2] The subjects had to rate each pair of ellipses on a scale from 1 ("most similar") to 10 ("least similar or most different"). This rating was replicated three times, with the pairs presented in different random orders. Table 10.1 gives the aggregated scores for one individual.

---

[1]Eccentricity is defined as $[1-(h/n)^2]^{1/2}$ and area is $\pi/4 \cdot h \cdot n$, where h is the length of the ellipse's major axis and n is the length of its minor axis. Hence, eccentricity is a function of the ratio of h and n, and area depends on the product of h and n.

[2]Noma and Johnson (1977, p. 31) characterize the design as follows: "A factorial design with four equally spaced levels of area crossed with four equally spaced levels of eccentricity was employed in constructing the stimuli. The largest ellipse was in a 3:1 ratio to the smallest, and the most eccentric was in a 1.66:1 ratio to the least eccentric."

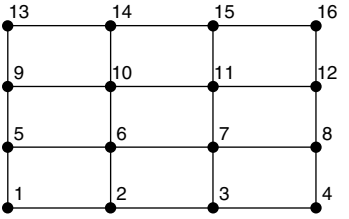


FIGURE 10.1. Design configuration for ellipses in Noma–Johnson study. X-axis is eccentricity ("shape"); Y-axis is area ("size").

TABLE 10.1. Dissimilarities for 16 ellipses; summed over three replications of subject DF (Noma & Johnson, 1977).

No.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	–															
2	18	–														
3	26	9	–													
4	23	14	12	–												
5	4	17	17	25	–											
6	15	6	13	13	18	–										
7	16	17	10	6	21	17	–									
8	24	20	13	10	24	17	12	–								
9	8	16	20	22	4	16	24	24	–							
10	16	9	16	14	21	8	11	10	14	–						
11	20	13	11	9	18	12	8	11	22	12	–					
12	22	18	17	12	21	21	12	6	23	13	11	–				
13	16	16	21	24	13	16	22	23	4	16	21	22	–			
14	17	14	16	19	20	9	14	17	13	4	14	17	17	–		
15	21	20	15	9	25	14	8	11	19	19	4	16	22	17	–	
16	26	19	14	15	24	16	11	12	22	16	10	6	30	17	9	–

From related research (see Section 17.4) it could be expected that an MDS configuration similar to the design configuration would result from the proximities. That is, the MDS configuration should form a rectangular grid as in Figure 10.1, although not necessarily with the same spacing of the vertical and horizontal lines. This would allow us to explain the similarity judgments by the dimensions "shape" and "area". Ordinal MDS of the data in Table 10.1 yields, however, a configuration (Figure 10.2) that is in definite disagreement with these predictions. But, then, a theory-conforming configuration does not necessarily have to have the *lowest-possible Stress*. Rather, it would be sufficient if it had an *acceptably low Stress*. Indeed, such a solution exists. It is shown in Figure 10.3. Its Stress is .185, as compared to .160 for the theory-incompatible solution in Figure 10.2. This example shows that there can be different MDS configurations that all represent a given set of data with roughly the same precision.

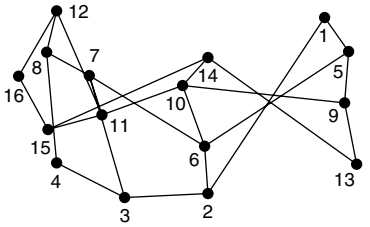


FIGURE 10.2. Minimal-Stress MDS representation for data in Table 10.1.

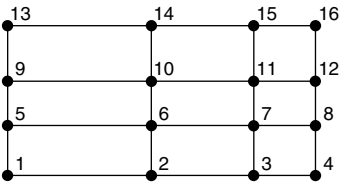


FIGURE 10.3. Minimal-Stress theory-compatible MDS.

10.3 Imposing External Constraints on MDS Representations

We now show how a confirmatory MDS procedure can be constructed. We begin by considering the task of constructing an ordinal MDS representation of the facial expression data from Table 4.4 of Abelson and Sermat (1962) in 2D so that (a) the Stress is as low as possible, and (b) all points can be coordinated by dimensions that are a linear combination of the three external scales of Engen et al. (1958). Condition (b) is an additional requirement imposed on the MDS representation. It is called a side constraint or an *external constraint* to distinguish it from the *internal* constraints due to the data and the general representation function.

The restriction that is imposed on the configuration is

$$\mathbf{X} = \mathbf{YC},$$

where $\mathbf{Y}$ is the 13×3 matrix with the three external scales of Table 4.3, and $\mathbf{C}$ is a 3×2 matrix of unknown weights. This matrix equation is shown explicitly in Table 10.2. The mathematical problem to be solved is to minimize $\sigma_r^2(\mathbf{X})$ by an appropriate choice of $\mathbf{C}$, subject to the condition $\mathbf{X} = \mathbf{YC}$. Solutions for this problem were given by Bentler and Weeks (1978), Bloxom (1978), and De Leeuw and Heiser (1980). We follow the approach of De Leeuw and Heiser (1980), because they show that this and more general constrained MDS models can be handled easily within the majorization framework (see Chapter 8).

As shown in (8.27), raw Stress can be majorized by

$$\tau(\mathbf{X}, \mathbf{Z}) \quad = \quad \eta_\delta^2 + \text{tr } \mathbf{X}'\mathbf{V}\mathbf{X} - 2\text{tr } \mathbf{X}'\mathbf{B}(\mathbf{Z})\mathbf{Z}, \tag{10.1}$$

which is equal to $\sigma_r(\mathbf{X})$ if $\mathbf{Z} = \mathbf{X}$; that is, $\sigma_r(\mathbf{X}) = \tau(\mathbf{X}, \mathbf{X})$. Let $\overline{\mathbf{X}} = \mathbf{V}^+\mathbf{B}(\mathbf{Z})\mathbf{Z}$ be the Guttman transform (8.28) of $\mathbf{Z}$, where $\mathbf{Z}$ satisfies the imposed constraints. Then (10.1) equals

$$\begin{aligned} \tau(\mathbf{X}, \mathbf{Z}) \quad &= \quad \eta_\delta^2 + \text{tr } \mathbf{X}'\mathbf{V}\mathbf{X} - 2\text{tr } \mathbf{X}'\mathbf{V}\overline{\mathbf{X}} \\ &= \quad \eta_\delta^2 + \text{tr } (\mathbf{X} - \overline{\mathbf{X}})'\mathbf{V}(\mathbf{X} - \overline{\mathbf{X}}) - \text{tr } \overline{\mathbf{X}}'\mathbf{V}\overline{\mathbf{X}}. \end{aligned} \tag{10.2}$$

TABLE 10.2. Matrix equation $\mathbf{X} = \mathbf{YC}$ in (10.1), with $\mathbf{Y}$ taken from Table 4.3; $\mathbf{X}$ is the desired 13×2 MDS configuration; $\mathbf{C}$ is an unknown matrix of weights.

$$\begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \\ x_{31} & x_{32} \\ x_{41} & x_{42} \\ x_{51} & x_{52} \\ x_{61} & x_{62} \\ x_{71} & x_{72} \\ x_{81} & x_{82} \\ x_{91} & x_{92} \\ x_{10,1} & x_{10,2} \\ x_{11,1} & x_{11,2} \\ x_{12,1} & x_{12,2} \\ x_{13,1} & x_{13,2} \end{bmatrix} = \begin{bmatrix} 3.8 & 4.2 & 4.1 \\ 5.9 & 5.4 & 4.8 \\ 8.8 & 7.8 & 7.1 \\ 7.0 & 5.9 & 4.0 \\ 3.3 & 2.5 & 3.1 \\ 3.5 & 6.1 & 6.8 \\ 2.1 & 8.0 & 8.2 \\ 6.7 & 4.2 & 6.6 \\ 7.4 & 6.8 & 5.9 \\ 2.9 & 3.0 & 5.1 \\ 2.2 & 2.2 & 6.4 \\ 1.1 & 8.6 & 8.9 \\ 4.1 & 1.3 & 1.0 \end{bmatrix} \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \\ c_{31} & c_{32} \end{bmatrix}$$

For a given configuration $\mathbf{Z}$, only the second term of $\tau(\mathbf{X}, \mathbf{Z})$ is dependent on $\mathbf{X}$, whereas the first and last terms are constant with respect to $\mathbf{X}$. Using $\sigma_r(\mathbf{X}) = \tau(\mathbf{X}, \mathbf{X})$, (10.2) shows that, for every configuration $\mathbf{X}$, raw Stress is the sum of *lack of model fit*, $\eta_\delta^2 - \text{tr } \overline{\mathbf{X}}'\mathbf{V}\overline{\mathbf{X}}$, and *lack of confirmation fit*, $\text{tr } (\mathbf{X} - \overline{\mathbf{X}})'\mathbf{V}(\mathbf{X} - \overline{\mathbf{X}})$. The latter is best expressed as a percentage of the total raw Stress. For example, if there are no constraints on $\mathbf{X}$, then the lack of confirmation fit is 0%.

Finding a constrained update amounts to minimizing

$$\text{tr } (\mathbf{X} - \overline{\mathbf{X}})'\mathbf{V}(\mathbf{X} - \overline{\mathbf{X}}), \tag{10.3}$$

subject to the restrictions on $\mathbf{X}$, in each iteration. The $\mathbf{X}$ that minimizes (10.3) *and* satisfies the constraints is used as the update. Thus, step 6 (computation of the Guttman transform) in the majorization algorithm of Chapter 9 (see also Figure 9.2) is followed by step 6a, in which (10.3) is minimized over $\mathbf{X}$, subject to the constraints on $\mathbf{X}$. De Leeuw and Heiser (1980) note that it is not necessary to find the global minimum of (10.3). The decrease of Stress is guaranteed as long as

$$\text{tr } (\mathbf{X}^u - \overline{\mathbf{X}})'\mathbf{V}(\mathbf{X}^u - \overline{\mathbf{X}}) \leq \text{tr } (\mathbf{Y} - \overline{\mathbf{X}})'\mathbf{V}(\mathbf{Y} - \overline{\mathbf{X}}) \tag{10.4}$$

holds for the update $\mathbf{X}^u$.

For the faces data, we simply substitute $\mathbf{X}$ by $\mathbf{YC}$ in (10.3), which yields

$$\begin{aligned} L(\mathbf{C}) &= \text{tr } (\mathbf{YC} - \overline{\mathbf{X}})'\mathbf{V}(\mathbf{YC} - \overline{\mathbf{X}}) \\ &= \text{tr } \overline{\mathbf{X}}'\mathbf{V}\overline{\mathbf{X}} + \text{tr } \mathbf{C}'\mathbf{Y}'\mathbf{VYC} - 2\text{tr } \mathbf{C}'\mathbf{Y}'\mathbf{V}\overline{\mathbf{X}}. \end{aligned}$$

$L(\mathbf{C})$ needs to be minimized over $\mathbf{C}$, because $\mathbf{Y}$ is fixed (see also Section 8.3). Finding the optimal weights $\mathbf{C}^u$ is a simple regression problem that is solved by

$$\mathbf{C}^u = (\mathbf{Y}'\mathbf{VY})^{-1}\mathbf{Y}'\mathbf{V}\overline{\mathbf{X}},$$

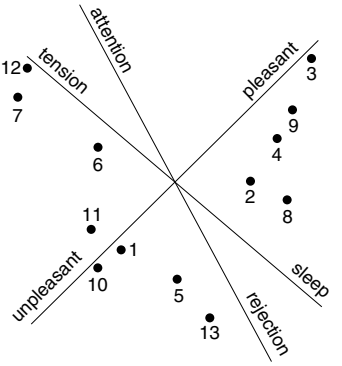


FIGURE 10.4. Theory-consistent solution by constrained MDS of the facial expression data of Abelson and Sermat (1962) with Stress .14.

so that

$$\mathbf{X}^u = \mathbf{Y}\mathbf{C}^u.$$

The unconstrained 2D solution of the faces data in Figure 4.9 has $\sigma_1 = .11$. The theory-compatible solution (where the dimensions are constrained to be linear combinations of three external scales) has $\sigma_1 = .14$ (see Figure 10.4). The unconstrained and constrained solutions only differ with respect to point 8. The raw Stress of the constrained solution is 0.0186 ($= \sigma_r$), of which .0182 (=97%) is the lack of model fit and .0004 (=3%) is Stress due to the constraints. Therefore, the theory-consistent solution can be accepted at the cost of a slightly higher Stress value. The optimal $\mathbf{C}$ equals

$$\mathbf{C} = \begin{bmatrix} .219 & .031 \\ -.035 & .137 \\ -.024 & .053 \end{bmatrix}.$$

What does $\mathbf{C}$ do to $\mathbf{Y}$? One way of interpreting $\mathbf{C}$ as an operator on $\mathbf{Y}$ is to decompose $\mathbf{C}$ into its singular value components:

$$\mathbf{C} = \mathbf{P}\mathbf{\Phi}\mathbf{Q}' = \begin{bmatrix} .985 & .168 \\ -.140 & .919 \\ -.102 & .357 \end{bmatrix} \begin{bmatrix} .224 & .000 \\ .000 & .150 \end{bmatrix} \begin{bmatrix} 1.000 & -.029 \\ .029 & 1.000 \end{bmatrix}.$$

$\mathbf{C}$ first rotates $\mathbf{Y}$ by $\mathbf{P}$, because $\mathbf{P}$ is orthonormal,[3] and takes out the third dimension. Then, $\mathbf{\Phi}$ multiplies the X-axis by .224 and the Y-axis by .150.

[3]In fact, the orthonormality of $\mathbf{P}$ only implies that $\mathbf{P}'\mathbf{P} = \mathbf{I}$, but not that $\mathbf{P}'\mathbf{P} = \mathbf{I}$, as required for a rotation matrix. However, $\mathbf{P}$ can be interpreted as a matrix that rotates $\mathbf{Y}$ to two dimensions.

Finally, $\mathbf{Q}'$ rotates the space somewhat, but only in the X–Y-plane. Note that the final rotation by $\mathbf{Q}'$ could as well be omitted, because it does not change the distances. Thus, $\mathbf{C}$ can be understood as a transformation of the 3D space of the external scales that rotates this space into a plane and also stretches this plane differentially along its coordinate axes so that the resulting configuration has minimal Stress.

The three external variables are plotted as lines such that the angles of the lines correspond to correlations between the variables and the X- and Y-axes, respectively (Figure 10.4). In comparison to Figure 4.9 (where the external variables were fitted *afterwards*, not simultaneously), variable pleasant/unpleasant has about the same direction, whereas the variables sleep/tension and attention/rejection have been interchanged. Because the latter two variables are highly intercorrelated, however, this interchange does not lead to a much different interpretation.

External Constraints and Optimal Scaling

Instead of the linear constraints used above, a multitude of other constraints can be used for which the least-squares solution of (10.4) can be computed, each constraint leading to a different model. De Leeuw and Heiser (1980) discuss many sorts of constraints, some of which are shown below. Apart from the general majorization result that Stress is reduced in every iteration, they also prove several other convergence results if the global minimum of (10.4) can be established. A more accessible overview of constrained MDS and its applications is given by Heiser and Meulman (1983b).

The facial expression data were analyzed by Heiser and Meulman (1983b) in a slightly different way. They used only the ordinal information of the three external variables of the faces data. Let $\mathbf{Y} = [\mathbf{y}_1\ \mathbf{y}_2\ \mathbf{y}_3]$ be the matrix of the three external variables. The constraints on the MDS solution are $\mathbf{X} = \widehat{\mathbf{Y}}\mathbf{C}$, where $\widehat{\mathbf{Y}} = [\widehat{\mathbf{y}}_1\ \widehat{\mathbf{y}}_2\ \widehat{\mathbf{y}}_3]$ and each column $\widehat{\mathbf{y}}_k$ can be *optimally scaled* (see, e.g., Young, 1981; Gifi, 1990). In optimal scaling of an ordinal variable, the original variable $\mathbf{y}_k$ is replaced by a different variable $\widehat{\mathbf{y}}_k$ that has the same order as the original variable *and* reduces the loss function, hence the name optimal scaling. Ordinal transformations on the external variables are computed using monotone regression, and implemented in the programs SMACOF-II (Meulman, Heiser, & De Leeuw, 1983) and PROXSCAL (see Appendix A). Thus, (10.4) was not only optimized over $\mathbf{C}$, but also over $\widehat{\mathbf{Y}}$, where every column of $\widehat{\mathbf{Y}}$ is constrained to have the order of the corresponding external variable $\mathbf{y}_k$. Apart from finding an optimal transformation of the proximities, an optimal transformation of the external variables is also found here. In their analysis of the facial expression data, Heiser and Meulman (1983b) conclude that pleasant–unpleasant is a more basic dimension than attention–rejection, which is a nonlinearly related ef-

fect. Optimal scaling of external variables allows interesting models to be specified, such as the one below.

If only two external variables are involved in a 2D MDS space, then the ordinal restrictions on the two external variables result in dimensional restrictions or in an axial partitioning of the space. For example, the hypothesized grid-like structure in Figure 10.3 was enforced onto the MDS configuration by the two external variables

$$\begin{aligned} \mathbf{y}_1' &= [1,2,3,4,1,2,3,4,1,2,3,4,1,2,3,4] \text{ and} \\ \mathbf{y}_2' &= [1,1,1,1,2,2,2,2,3,3,3,3,4,4,4,4]. \end{aligned}$$

Then, $\mathbf{X}$ is obtained by $\mathbf{X} = \widehat{\mathbf{Y}}\mathbf{C}$, with the secondary approach to ties (i.e., ties remain tied). The external variables $\mathbf{y}_1$ and $\mathbf{y}_2$ are derived from the design configuration in Figure 10.1. Thus, the 2-tuple $(f_1(y_{i1}), f_2(y_{i2}))$ contains the coordinates of each point i, where f_1 and f_2 are Stress-minimizing monotonic functions that use the secondary approach to ties. These requirements come from psychophysics. We do not expect that an ellipse that is twice as eccentric in terms of the ratio of its axes is also perceived as twice as eccentric, for example. Rather, we would expect by the Weber–Fechner law that perceived eccentricity should be related to "objective" eccentricity in a roughly logarithmic way, in general. Indeed, that is exactly what the data show in Figure 10.3. Note that we did not enforce a logarithmic spacing on the horizontal axis. Rather, this function was found by MDS as the best in terms of Stress.[4]

Regionally Constrained MDS

Optimal scaling of the external variables can also be used to impose regional constraints on the MDS solution. For example, we know for each point two facets and use MDS to separate the different classes of points by two sets of parallel lines, where each set corresponds to one facet. This constraint only works if the number of dimensions is equal to the number of external variables and only for axial partitioning of the MDS space. In addition, the facets should be ordered.

Consider the following example. The ordinal MDS solution (Figure 4.7) of the Morse code data of Rothkopf (1957) was interpreted using two physical properties of the signals. The two properties are signal length (varying from .05 to .95 seconds) and signal type (the ratio of long vs. short beeps). Figure 4.7 shows that the unconstrained MDS solution can be partitioned by these two facets. However, the dashed lines (partitioning the plane by signal type) have a rather irregular curvature. We now ask whether an MDS solution

[4] On the dimension "size", in contrast, this spacing is not obvious, which may be due to the size range of the ellipses. See also Figure 17.8 for a similar experiment with rectangles.

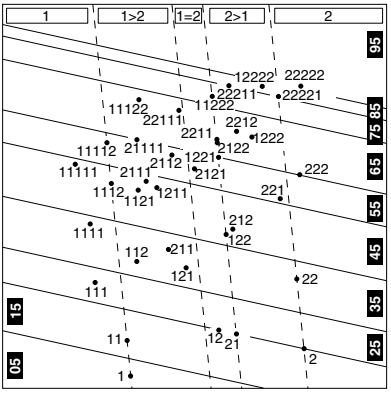


FIGURE 10.5. Theory-consistent solution by constrained MDS of the Morse code data of Rothkopf (1957) with Stress .21.

exists that can be partitioned by straight lines while still being acceptable in terms of Stress. We use PROXSCAL to answer this question.

The external variables for signal type $\mathbf{y}_1$ and the signal length $\mathbf{y}_2$ have a value for every Morse code. The 2D MDS space is constrained to be a linear combination of signal length and signal type; that is, $\mathbf{X} = \widehat{\mathbf{Y}}\mathbf{C}$, with $\widehat{\mathbf{Y}} = [\widehat{\mathbf{y}}_1\ \widehat{\mathbf{y}}_2]$ where $\widehat{\mathbf{y}}_1$ and $\widehat{\mathbf{y}}_2$ are monotonic transformations of $\mathbf{y}_1$ and $\mathbf{y}_2$, respectively. In contrast to the example above, we now allow that tied coordinates can be untied (primary approach to ties). This combination of restrictions implies that there is a direction in the MDS space that represents $\widehat{\mathbf{y}}_1$, and all projections of the points onto this line satisfy the order of the signal lengths, so that perpendicular lines separate the space into regions with equal signal lengths. The same holds for $\widehat{\mathbf{y}}_2$, so that a separation of the space for signal types is obtained. Figure 10.5 shows the solution of the ordinal MDS analysis with the external constraints described above. This theory-consistent configuration has Stress .21 ($\sigma_r = .043$), and the unconstrained solution in Figure 4.7 has Stress .18. The raw Stress can be decomposed into model Stress (=.0429, .997%) and Stress due to the constraints (=.00011, .003%). Apart from theory-consistency, Figures 10.5 and 4.7 differ with respect to the location of the points representing the Morse codes E and T (labeled as 1 and 2, respectively, in both figures). These points are less well represented, which is reflected by their Stress per point (see Section 3.4) of .095 and .082, respectively, the largest contributions to overall Stress. In summary, however, the difference in Stress of the constrained and the unconstrained solutions is rather small, so that the theory-consistent solution seems acceptable.

Cluster Differences Scaling

A different type of constraint was suggested by Heiser (1993), who proposed a mixture of an MDS analysis and a cluster analysis. This method was called *cluster differences scaling*. Every object is assigned to a cluster, and every cluster is represented by a point in the space. In cluster differences scaling, Stress is optimized over the coordinates *and* over the cluster memberships. Groenen (1993) showed that this method can be interpreted as MDS with the restriction that the configuration is of the type $\mathbf{GX}$, where $\mathbf{G}$ is an $n \times k$ indicator matrix (which has a single one in each row, and all other values zero), and $\mathbf{X}$ is a $k \times m$ matrix of the k cluster coordinates. Heiser and Groenen (1997) elaborate on this model, give a decomposition of the dispersion (sum of squared dissimilarities), and present a convergent algorithm. The assignment of objects to clusters (by matrix $\mathbf{G}$) gives rise to many local minima. Groenen (1993) and Heiser and Groenen (1997) managed to avoid such local minima by repeatedly computing a fuzzy form of cluster differences scaling until the fuzzy form yields the same result as the crisp form.

The Extended Euclidean Model

Suppose that the proximities are not very well explained by an MDS in low-dimensional space. One reason could be that some objects are quite unique. One could account for this and allow each object to retain its uniqueness in MDS by assigning a uniqueness dimension to each object, in addition to the low-dimensional space common to all objects (Bentler & Weeks, 1978). A uniqueness dimension $\mathbf{x}_i$ for object i has coordinates of zero for all objects, except for object i. Thus, the matrix of coordinates consists of the usual $n \times m$ matrix of coordinates $\mathbf{X}$ common to all objects, augmented by a diagonal $n \times n$ matrix $\mathbf{U}$ of uniqueness coordinates. The augmented coordinate matrix is denoted by $[\mathbf{X}|\mathbf{U}]$. The distance between objects i and j is

$$d_{ij}(\mathbf{X}|\mathbf{U}) = \left(\sum_{a=1}^{m} (x_{ia} - x_{ja})^2 + u_{ii}^2 + u_{jj}^2 \right)^{1/2},$$

which is called the *extended Euclidean* distance (Winsberg & Carroll, 1989). The distance consists of a common part and a part determined by the uniqueness of the objects i and j.

This special coordinate matrix also can be viewed as an example of a constrained configuration. The constraints simply consist of fixing the off-diagonal elements of $\mathbf{U}$ to zero while leaving the diagonal elements free (Bentler & Weeks, 1978).

10.4 Weakly Constrained MDS

We now consider a weaker form of constraining an MDS solution. It puts additional external restrictions on the configuration that are not strictly enforced. Rather, they may be violated, but any violation leads to higher Stress. Weakly constrained MDS attempts to minimize such violations.

Let us try to represent the color data from Table 4.1 ordinally by distances in a plane so that (a) the Stress is as low as possible, and (b) all points lie on a perfect circle. Condition (b) is the external constraint imposed on the MDS representation.

Figure 4.1 shows that the usual MDS result already satisfies condition (b) very closely, so we use this solution in the following as a starting configuration. The confirmatory scaling problem then consists of finding a projection of the points onto a circle such that the Stress value goes up as little as possible. If we pick a point somewhere close to the center of the color circle in Figure 4.1 and construct a circle around this point such that it encloses all points of the configuration, then an approximate solution to our scaling problem could be found simply by projecting all points radially towards the outside onto this circle. An optimal solution can be constructed in a similar fashion.

First, augment the proximity matrix in Table 4.1 with a "dummy" object Z. Z does not stand for an additional concrete stimulus, but serves an auxiliary purpose here and represents the circle center in the MDS configuration. The proximities between Z and any of the real stimuli $434, 445, \ldots, 674$ are defined as missing data. This leads to the 15×15 data matrix $\mathbf{P}_1$ in Table 10.3.

Second, define another 15×15 proximity matrix $\mathbf{P}_2$ which expresses the side constraints. No external constraints are to be imposed on the distances between any two color points. However, all should lie on a circle and so all must have the same distance to point Z. This gives the constraint pattern $\mathbf{P}_2$ shown in Table 10.4, where all elements except those in row Z and in column Z are missing values. All elements in row and column Z are set equal to 10, but any other number would do as well.

Third, use the configuration in Figure 4.1 as a starting configuration, after adding proper coordinates for one further point, Z. The coordinates of Z should be chosen so that Z lies somewhere in the center of the circular manifold in Figure 4.1. This can be done most easily by centering the MDS configuration in Figure 4.1, that is, shifting it so that the centroid of all 14 points coincides with the origin, or, computationally, by subtracting the mean from the values in each column of $\mathbf{X}$ in turn. Z then has the coordinates (0.0, 0.0).

Fourth, define a loss criterion for the scaling procedure. We choose

$$\sigma_T(\mathbf{X}; \mathbf{P}_1; \mathbf{P}_2) = \sigma_r(\mathbf{X}; \mathbf{P}_1) + a \cdot \sigma_r(\mathbf{X}; \mathbf{P}_2), \tag{10.5}$$

TABLE 10.3. Similarities for colors with wavelengths 434 to 674 nm (Ekman, 1954); Z is a dummy variable; – denotes a missing value; the matrix is called $\mathbf{P}_1$ in the text.

nm	434	445	465	472	490	504	537	555	584	600	610	628	651	674	Z
434	–	86	42	42	18	06	07	04	02	07	09	12	13	16	–
445	86	–	50	44	22	09	07	07	02	04	07	11	13	14	–
465	42	50	–	81	47	17	10	08	02	01	02	01	05	03	–
472	42	44	81	–	54	25	10	09	02	01	00	01	02	04	–
490	18	22	47	54	–	61	31	26	07	02	02	01	02	00	–
504	06	09	17	25	61	–	62	45	14	08	02	02	02	01	–
537	07	07	10	10	31	62	–	73	22	14	05	02	02	00	–
555	04	07	08	09	26	45	73	–	33	19	04	03	02	02	–
584	02	02	02	02	07	14	22	33	–	58	37	27	20	23	–
600	07	04	01	01	02	08	14	19	58	–	74	50	41	28	–
610	09	07	02	00	02	02	05	04	37	74	–	76	62	55	–
628	12	11	01	01	01	02	02	03	27	50	76	–	85	68	–
651	13	13	05	02	02	02	02	02	20	41	62	85	–	76	–
674	16	14	03	04	00	01	00	02	23	28	55	68	76	–	–
Z	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–

TABLE 10.4. Restriction matrix for color data in Table 10.3; – denotes a missing value; the matrix is called $\mathbf{P}_2$ in the text.

nm	434	445	465	472	490	504	537	555	584	600	610	628	651	674	Z
434	–	–	–	–	–	–	–	–	–	–	–	–	–	–	10
445	–	–	–	–	–	–	–	–	–	–	–	–	–	–	10
465	–	–	–	–	–	–	–	–	–	–	–	–	–	–	10
472	–	–	–	–	–	–	–	–	–	–	–	–	–	–	10
490	–	–	–	–	–	–	–	–	–	–	–	–	–	–	10
504	–	–	–	–	–	–	–	–	–	–	–	–	–	–	10
537	–	–	–	–	–	–	–	–	–	–	–	–	–	–	10
555	–	–	–	–	–	–	–	–	–	–	–	–	–	–	10
584	–	–	–	–	–	–	–	–	–	–	–	–	–	–	10
600	–	–	–	–	–	–	–	–	–	–	–	–	–	–	10
610	–	–	–	–	–	–	–	–	–	–	–	–	–	–	10
628	–	–	–	–	–	–	–	–	–	–	–	–	–	–	10
651	–	–	–	–	–	–	–	–	–	–	–	–	–	–	10
674	–	–	–	–	–	–	–	–	–	–	–	–	–	–	10
Z	10	10	10	10	10	10	10	10	10	10	10	10	10	10	–

where $\sigma_r(\mathbf{X};\mathbf{P}_1)$ is the loss of configuration $\mathbf{X}$ relative to $\mathbf{P}_1$, $\sigma_r(\mathbf{X};\mathbf{P}_2)$ the loss relative to $\mathbf{P}_2$, and a is a nonnegative weight. This means that $\sigma_r(\mathbf{X};\mathbf{P}_1)$ is the Stress of a given configuration relative to the proximity matrix $\mathbf{P}_1$, and $\sigma_r(\mathbf{X};\mathbf{P}_2)$ is the Stress of this configuration relative to the constraint matrix $\mathbf{P}_2$. The second term of (10.5) is called a *penalty* term. It penalizes the solution for not satisfying the constraints. The strength of the penalty is determined by the size of the *penalty parameter* a. Of course, $\sigma_r(\mathbf{X};\mathbf{P}_1)$ and $\sigma_r(\mathbf{X};\mathbf{P}_2)$ are computed only over those elements of the matrices $\mathbf{P}_1$ and $\mathbf{P}_2$ that are not defined to be missing data. Thus,

$$\sigma_r(\mathbf{X};\mathbf{P}) = \sum_{i<j}[\widehat{d}_{ij} - d_{ij}(\mathbf{X})]^2, \text{ for all defined } p_{ij},$$

where $\widehat{d}_{ij}$ (dependent on $\mathbf{P}$) is the target distance (disparity) of $d_{ij}(\mathbf{X})$ defined by the chosen MDS model. In the present example, we choose ordinal MDS and the secondary approach to ties on $\mathbf{P}_2$, because all tied data values in the restriction matrix $\mathbf{P}_2$ should be mapped into exactly the same distance. (With the primary approach to ties, $\sigma_r(\mathbf{X};\mathbf{P}_2) = 0$ for any $\mathbf{X}$, because all defined elements of $\mathbf{P}_2$ are equal.) But then the target distances in $\sigma_r(\mathbf{X};\mathbf{P}_2)$ obtained by monotone regression are all equal to the arithmetic mean of the distances from point Z to all other points of the configuration $\mathbf{X}$.

Fifth, find a method to minimize (10.5). This does not pose a new problem. We proceed as in Chapter 8, that is, using the majorizing approach to minimize Stress.

Sixth, given the initial configuration of the unconstrained solution in Figure 4.1, iterate to solve the MDS task. If we start with $a = 1$, the restrictions only slightly determine the final solution. As a is increased, the effect of the side constraints on the configuration is increased. If $a \to \infty$, then every solution strictly satisfies the circular constraint. Because the effect of the constraints on the solution is set by the penalty parameter a, the method in this section that minimizes (10.5) may be called *weakly constrained MDS* (after the weakly constrained regression of Ten Berge, 1991). Often, choosing $a = 100$ generates a theory-conforming solution.

If it is at all possible to impose the side constraints onto a configuration of n points in a space of fixed dimensionality, we should end up with $\sigma_r(\mathbf{X};\mathbf{P}_2) = 0$, provided the iterations do not get stuck in a local minimum. Of course, we can impose conditions that are impossible to satisfy (e.g., attempting to represent the distances among a cube's corners in a ratio MDS plane). The final $\sigma_r(\mathbf{X};\mathbf{P}_1)$ is an index for how well the theory-conforming solution represents the given data. However, the raw measures $\sigma_r(\mathbf{X};\mathbf{P}_1)$ and $\sigma_r(\mathbf{X};\mathbf{P}_2)$ are not very practical, so we express the fit of $\mathbf{X}$ relative to $\mathbf{P}_1$ and $\mathbf{P}_2$ by a more familiar index such as Stress.

A procedure similar to the one described above is the program CMDA (Borg & Lingoes, 1980). Weakly constrained MDS can also be computed

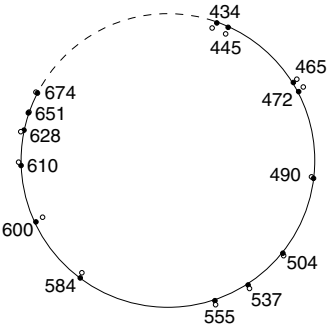


FIGURE 10.6. Stress-optimal (circles) and perfectly circular (points) MDS representation of color proximities in Table 4.1.

by programs that allow for data weights, such as KYST and PROXSCAL. Then data weights w_{ijk} are set to 0 for missing proximities, and $w_{ij1} = 1$ for nonmissing proximities of $\mathbf{P}_1$ and $w_{ij2} = a$ for nonmissing proximities of $\mathbf{P}_2$.

With the matrices given in Tables 10.3 and 10.4, and using ordinal MDS with the secondary approach to ties, we obtain the configuration of the solid points in Figure 10.6. To demonstrate how $\sigma_r(\mathbf{X}; \mathbf{P}_2)$ has affected the MDS solution, Figure 10.6 also shows the MDS configuration (open circles) obtained from a regular MDS analysis. The Stress of the weakly constrained MDS configuration relative to $\mathbf{P}_1$ is $\sigma_1 = 0.0374$, whereas it is $\sigma_1 = 0.0316$ for the unconstrained MDS configuration. The side constraints have led to an increment in Stress so small that both representations should be considered equally good, especially because we can assume that the data are not error-free. We therefore conclude that the color-circle theory is compatible with the given data.

A different approach was followed by Cox and Cox (1991). They forced the configuration onto the surface of a sphere by expressing the point coordinates not as Cartesian but as spherical coordinates, and then minimizing Stress only over the longitude and latitude angles that specify the points' positions in space.

Hubert, Arabie, and Meulman (1997) analyzed a related but different problem, namely to model the dissimilarities by distance between points *along the path* of the circle. This model is essentially the same as unidimensional scaling where the dimension is bent to be circular. Hubert et al. call their model circular unidimensional scaling. In Section 13.5, we show that unidimensional scaling suffers from many local minima and that a combinatorial approach is useful to find a global minimum. For this reason, Hubert et al. use combinatorial optimization together with iterative projection techniques to solve the circular unidimensional scaling problem.

TABLE 10.5. Correlations of eight intelligence tests (lower half); for structuples, see text; upper half contains hypothesized similarities.

Test	NA1	NA2	NI	GI1	GI2	GA1	GA2	GA3
NA1	–	5	4	3	3	4	4	4
NA2	.67	–	4	3	3	4	4	4
NI	.40	.50	–	4	4	3	3	3
GI1	.19	.26	.52	–	5	4	4	4
GI2	.12	.20	.39	.55	–	4	4	4
GA1	.25	.28	.31	.49	.46	–	5	5
GA2	.26	.26	.18	.25	.29	.42	–	5
GA3	.39	.38	.24	.22	.14	.38	.40	–

Enforcing Order Constraints onto MDS Distances

We now look at weakly constrained MDS where certain order relations are imposed onto the MDS distances. Consider the correlation matrix in Table 5.1, repeated for convenience in the lower half of Table 10.5 (Guttman, 1965). The variables here are eight intelligence test items, coded by the facets "language of communication" = {N = numerical, G = geometrical} and "requirement" = {I = inference, A = application}. For example, item 1 and item 2 both were classified as NA or numerical-application items.

One can predict how these items should be correlated among each other by invoking the *contiguity principle*. This principle is based on the (seemingly) plausible idea that "variables which are more similar in their facet structure will also be more related empirically" (Foa, 1965, p.264).[5] Similarity in facet structure is typically defined as the number of structs that two structuples have in common, whereas empirical similarity is assessed by some correlation between items (Foa, 1958). Hence, one predicts here, for example, that item 4 should be at least as similar to item 8 as to item 2, because the former share one definitional element (their language), whereas the latter differ on both facets. Predictions of this kind imply that the MDS configuration should be circular (Figure 10.7).

To test this prediction, we have to set up a restriction matrix $\mathbf{P}_2$ that enforces certain order relations onto the MDS distances. Because $\mathbf{P}_1$ (lower half in Table 10.5) contains similarity coefficients, we choose $\mathbf{P}_2$'s values correspondingly. A $\mathbf{P}_2$ that confirms the theory of Figure 10.7 is given in the upper half of Table 10.5. It is built as follows. The proximities for items with the same structuples, such as p(NA1,NA2) and p(GA1,GA3), all are set to the value 5. The proximities that correspond to the immediate

[5]Upon closer inspection, the contiguity makes sense only if all facets are ordered in the sense of the observational range (see Borg & Shye, 1995). However, we do not study these conditions here but simply use the example to demonstrate how certain constraints can be set up.

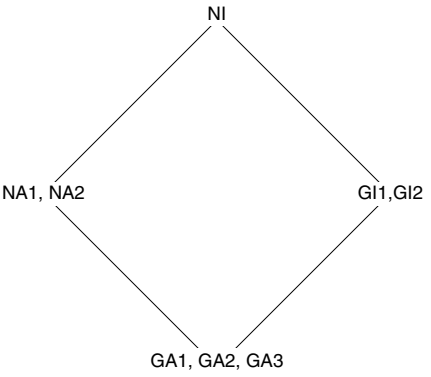


FIGURE 10.7. Hypothesized configuration of points representing intelligence tests with different facet compositions (G = geometrical, N = numerical; A = application, I = inference).

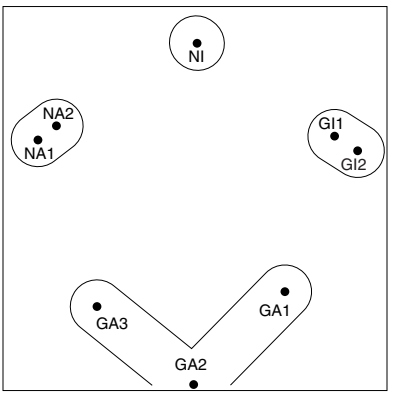


FIGURE 10.8. Best MDS representation that perfectly satisfies regional hypotheses in Figure 10.7.

neighborhood relations (shown in Figure 10.7 by the line segments) are set to the value 4, since none of these distances should be larger than any distance between definitionally equivalent items. Finally, what remain are the large distances between the groups NI, GA and the groups NA, GI, which are set to 3 in $\mathbf{P}_2$. Because we are doing ordinal MDS on similarities, the values 3, 4, and 5 are immaterial and may be replaced by any numbers that have the same order. By using the primary approach to ties on $\mathbf{P}_2$, all distances associated with a, say, 4 in $\mathbf{P}_2$, should not be larger than those associated with a 3 in $\mathbf{P}_2$. However, the distances within either class are not required to be equal.

The weakly constrained MDS representation (with $a = 100$) is shown in Figure 10.8. It satisfies the side constraints perfectly, with an acceptably small overall Stress ($\sigma_1 = .0404$). What remains, though, is some scatter among the items with the same structuples (notably GA and GI), so more and/or modified facets might be considered.

10.5 General Comments on Confirmatory MDS

Confirmatory MDS offers models that are open for theory-driven ad hoc specifications. Standard MDS models, in contrast, are more like closed systems that allow the user only some global specifications, such as choosing the dimensionality of the solution space or the Minkowski metric parameter (see Chapter 17). The purpose of confirmatory MDS is to enforce certain expected relations on an otherwise optimal data representation in order to see how compatible these relations are with the data.

The varieties of confirmatory MDS are, in principle, without bounds. New theories may require new confirmatory MDS procedures. Dimensional theories are best served by the existing procedures, and regional theories worst. We have seen that it is rather easy to enforce certain axial partitionings onto an MDS solution. It is also not too difficult to enforce cluster structures onto the MDS configuration, for example, for strict clustering by cluster differences scaling (Heiser & Groenen, 1997) and for weak clustering by setting up appropriate order constraints on the distances (Borg & Lingoes, 1987). However, with the MDS programs available today, it is difficult to enforce a more intricate regional pattern such as a radex, for example, onto an MDS solution. It is even more difficult, or even impossible, to formulate constraints on *general partitionability* relative to a facet design for the points, as discussed in Chapter 5.[6]

Apart from such problems of enforcing particular types of constraints onto an MDS solution, the general question of how to evaluate such methods and their results within cumulative scientific research remains to be answered. The more theoretically guided researcher may be tempted to always force his or her theory onto an MDS solution and then evaluate the resulting Stress. Unless there is a good estimate for the random noise component in the data (e.g., reliability measures), this is a dangerous strategy, because it does not allow one to separate *errors of approximation* from *errors of estimation*. The latter are due to sampling errors, and Stress in standard (unconstrained) MDS essentially reflects, as we saw in Chapter 3, such random errors in the data. Thus, high Stress values may be discarded as "technical" information only. Errors of approximation, however, would not go away even if the data were perfectly reliable. They simply express the misfit of the model. To separate these two error sources, one should always compute a standard MDS solution and then compare its Stress to the Stress obtained under additional restrictions. What is important, then, is the Stress increment. If strict constraints are used (as opposed to weakly constrained MDS), one should compare the ratio of Stress due to model misfit and Stress due to violation of the constraints. If the latter term is relatively small, then the theory-confirming solution can be accepted.

As a rule of thumb, it holds that if a standard MDS solution is similar to what was predicted theoretically, enforcing the theory onto the MDS solution does not make much difference in terms of Stress. However, if standard MDS does not yield the expected configuration, then it is impossible to say whether confirmatory MDS will make much difference. That depends on the loss function and its local minima.

---

[6] Guttman (1976) suggested combining MDS with *multidimensional scalogram analysis*, using MSA's notions of contiguity. See also Borg and Groenen (1998). These ideas have not yet been studied systematically, however.

Stress increments must be evaluated in any case, and this is a complex matter (Lingoes & Borg, 1983). What must be taken into account here is: (a) the number of points, n, because enforcing external constraints on few points or distances is generally easier than dealing with a large n; (b) the dimensionality of the MDS solution, m, for reasons similar to those for n; (c) the error component in the proximities, because with very noisy data, further constraints have less effect on Stress; (d) the similarity between the standard solution and the confirmatory solution: minor corrections of the standard solution should have little effect on Stress; and (e) the increased theoretical intelligibility of the confirmatory solution over the standard solution: if the latter makes little sense, one may be willing to accept larger increments in Stress, because a theoretically justified solution promises to be more stable over replications, and there is no reason to predict stability for structures that are not understood.

10.6 Exercises

Exercise 10.1 Consider the matrix below. In its lower half it shows similarity coefficients for tonal stimuli (Levelt, Geer, & Plomp, 1966). Each stimulus consisted of two simultaneously heard tones with a fixed ratio between their frequencies. Fifteen stimuli were used: the twelve musical intervals within the octave; and in addition two wider intervals (4:9 and 2:5) and one narrow interval between minor and major second (11:12). To control for pitch effects, the mean frequency for each interval was held constant at 500 Hz. Previous analyses by Levelt et al. (1966) and by Shepard (1974) had shown that the subjective similarities for these tone intervals form a horseshoe-like structure in the two-dimensional MDS plane.

Freq. Ratio	No.	15	13	12	7	6	3	9	2	10	5	8	14	1	11	4
15:16	15	–	14	13	12	11	10	9	8	7	6	5	4	3	2	1
11:12	13	32	–	14	13	12	11	10	9	8	7	6	5	4	3	2
8:9	12	29	32	–	14	13	12	11	10	9	8	7	6	5	4	3
5:6	7	19	22	28	–	14	13	12	11	10	9	8	7	6	5	4
4:5	6	14	17	23	28	–	14	13	12	11	10	9	8	7	6	5
3:4	3	15	10	13	22	25	–	14	13	12	11	10	9	8	7	6
5:7	9	8	8	14	25	24	27	–	14	13	12	11	10	9	8	7
1:2	2	9	10	14	13	18	21	22	–	14	13	12	11	10	9	8
5:8	10	6	7	13	20	21	17	22	27	–	14	13	12	11	10	9
3:5	5	12	11	12	15	20	24	13	25	24	–	14	13	12	11	10
4:7	8	7	11	15	14	18	14	16	13	27	30	–	14	13	12	11
8:15	14	7	10	7	10	17	10	19	18	18	18	26	–	14	13	12
1:2	1	8	3	9	9	14	15	9	14	12	13	22	26	–	14	13
4:9	11	9	14	6	8	8	12	9	10	10	17	20	13	29	–	14
2:5	4	9	14	6	10	7	12	10	16	18	18	14	13	25	30	–

TABLE 10.6. Correlation coefficients (decimal points omitted) for the 30 forms of protest acts described in Table 1.2.

Act	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
2	41	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
3	53	45	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
4	20	30	26	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
5	20	37	33	46	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
6	18	27	28	41	54	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
7	19	29	29	45	58	57	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
8	25	30	37	36	47	46	56	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
9	03	14	06	31	34	40	41	33	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
10	03	09	04	20	25	31	31	27	52	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
11	53	15	31	10	08	07	08	12	-03	05	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
12	37	64	39	26	31	24	25	27	10	07	29	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
13	38	26	57	17	18	19	16	23	03	05	49	39	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
14	19	26	24	58	37	32	34	29	21	15	16	37	24	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
15	21	30	29	31	52	34	37	36	18	12	12	40	24	45	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
16	13	16	21	23	33	49	32	30	23	15	15	23	25	31	43	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
17	17	23	23	26	36	36	46	39	21	20	06	36	21	40	60	44	.	.	.	.	.	.	.	.	.	.	.	.	.	.
18	20	22	25	23	27	32	33	51	17	15	15	32	29	36	50	44	62	.	.	.	.	.	.	.	.	.	.	.	.	.
19	02	08	05	16	18	23	22	20	29	20	00	15	08	31	34	38	43	43	.	.	.	.	.	.	.	.	.	.	.	.
20	00	06	03	12	14	20	18	17	21	37	00	16	07	25	31	31	39	38	70	.	.	.	.	.	.	.	.	.	.	.
21	57	34	44	18	22	18	20	25	-01	02	44	34	36	16	24	14	21	23	04	06	.	.	.	.	.	.	.	.	.	.
22	33	68	37	22	28	21	25	29	09	06	16	63	25	22	28	15	26	27	08	08	45	.	.	.	.	.	.	.	.	.
23	41	35	60	21	29	26	27	34	04	05	30	34	48	20	27	18	24	28	04	04	55	46	.	.	.	.	.	.	.	.
24	23	26	26	54	38	32	39	33	15	17	14	27	14	52	31	21	32	25	16	14	29	30	34	.	.	.	.	.	.	.
25	22	28	30	32	61	38	42	40	20	15	10	27	18	29	51	27	37	31	17	13	28	33	41	45	.	.	.	.	.	.
26	17	21	25	29	36	59	37	34	26	20	09	22	18	28	31	48	35	31	22	20	22	27	32	35	47	.	.	.	.	.
27	20	26	27	31	43	42	52	42	23	22	11	27	18	27	34	28	47	31	17	15	24	33	34	47	54	54	.	.	.	.
28	24	25	32	30	36	36	42	58	20	16	15	27	24	28	33	26	38	46	17	16	31	36	45	44	49	45	58	.	.	.
29	02	08	03	18	20	23	20	17	32	24	03	09	05	18	15	16	17	13	24	15	07	15	09	22	22	30	32	32	.	.
30	02	06	01	16	18	19	16	14	25	33	03	09	04	15	10	12	15	12	17	28	09	12	09	22	22	26	31	30	57	.

(a) Replicate Shepard's MDS analysis and verify that the order of the stimuli on the horseshoe corresponds to the order of the entries of the table.

(b) Use confirmatory ordinal MDS and try to unbend the horseshoe such that it does not bend back upon itself. The upper half of the table indicates a pattern of values that satifies such a simple structure. You may use these pseudodata to impose the "unbending" constraints, but note that a simplex is a biconditional order structure. Impose only minimal constraints.

(c) Compare the constrained MDS solution with the one that does not use external constraints and discuss the findings.

Exercise 10.2 Consider the lower-half matrix in Table 17.7. Its unconstrained city-block MDS representation is shown in Figure 17.8. Try to force a perfect "rectangular" structure onto this solution so that, for example, points 1, 2, 3, and 4 lie on a straight vertical line, and points 1, 5, 9, and 13 lie on a straight horizontal line (see dashed grid in Figure 17.8).

Exercise 10.3 Table 10.6 shows the intercorrelations of the 30 forms of protest behavior (Levy, 1983) analyzed before in Section 1.2 (West German data of early 1974, $N = 2307$).

(a) Use confirmatory MDS to enforce a solution where the points are perfectly separated in 3D space in the sense of their design facets that are shown in Table 1.2.

(b) Compare the solution to an "unconstrained" solution as discussed in Section 1.2.

(c) Discuss any amount of additional Stress due to the external constraints.

11

MDS Fit Measures, Their Relations, and Some Algorithms

A problem in MDS is how to evaluate the Stress value. Once a solution is found, how good is it? In Chapter 3, several statistical simulation studies were reported. Here we give an interpretation of normalized Stress in terms of the proportion of the explained sum-of-squares of the disparities. We also show that normalized Stress is equal to Stress-1 at a minimum and that the configurations only differ by a scale factor. Then, other common measures of fit for MDS are discussed. For these fit measures, we refer to some recent algorithmic work. Finally, it is discussed how weights in MDS can be used to emphasize different aspects of the data, to approach other MDS loss functions, or to take the reliability of the data into account.

Throughout this chapter, we refer to the data as being dissimilarities δ_{ij} for notational simplicity. However, all definitions of Stress measures and their relations remain valid when the dissimilarities are replaced by $\widehat{d}_{ij}$ obtained by optimal transformation (see the approach taken in Section 9.1).

11.1 Normalized Stress and Raw Stress

In Section 3.2, we saw that σ_r depends on the "size" of $\mathbf{X}$. Changing the scale of the coordinates of $\mathbf{X}$ changes σ_r accordingly. To avoid this scale dependency, one can use the implicit normalization used in Kruskal's Stress-1. Here, we elaborate on a different measure, which we call *normalized Stress*. This coefficient shows (after convergence) the proportion of the sum-

of-squares of the δ_{ij}s that is *not* accounted for by the distances. We define normalized Stress $\sigma_n(\mathbf{X})$ as

$$\sigma_n(\mathbf{X}) = \frac{\sigma_r(\mathbf{X})}{\eta_\delta^2} = \frac{\sum_{i<j} w_{ij}(\delta_{ij} - d_{ij}(\mathbf{X}))^2}{\sum_{i<j} w_{ij}\delta_{ij}^2}. \tag{11.1}$$

Clearly, if $\sum_{i<j} w_{ij}\delta_{ij}^2 = 1$, then $\sigma_n(\mathbf{X}) = \sigma_r(\mathbf{X})$.

De Leeuw (1977) (and, among others, Commandeur, 1993) show how $\sigma_n(\mathbf{X})$ is related to the square of Tucker's coefficient of congruence.[1] This relation can be explained as follows. Suppose that $\mathbf{X}^*$ is a local minimum of $\sigma_r(\mathbf{X})$. This implies that $b\mathbf{Y}^* = \mathbf{X}^*$ (with $b > 0$) also must be a local minimum. Note that $\mathbf{Y}^*$ has coordinates that are proportional to $\mathbf{X}^*$. We show that for optimal b normalized Stress is equal to one minus the square of Tucker's coefficient of congruence.

To find an optimal b, we use the property that the Euclidean distance is a positively homogeneous function in $\mathbf{X}$; that is, $d_{ij}(b\mathbf{Y}^*) = bd_{ij}(\mathbf{Y}^*)$ for $b \geq 0$. Then $\sigma_r(b\mathbf{Y}^*)$ can be written as

$$\begin{aligned}
\sigma_r(b\mathbf{Y}^*) &= \sum_{i<j} w_{ij}(\delta_{ij} - d_{ij}(b\mathbf{Y}^*))^2 \\
&= \sum_{i<j} w_{ij}\delta_{ij}^2 + b^2 \sum_{i<j} w_{ij} d_{ij}^2(\mathbf{Y}^*) - 2b \sum_{i<j} w_{ij}\delta_{ij} d_{ij}(\mathbf{Y}^*) \\
&= \eta_\delta^2 + b^2\eta^2(\mathbf{Y}^*) - 2b\rho(\mathbf{Y}^*).
\end{aligned} \tag{11.2}$$

The minimum of (11.2) over b is obtained by setting the first derivative of $\sigma_r(b\mathbf{Y}^*)$ with respect to b equal to zero, $2b\eta^2(\mathbf{Y}^*) - 2\rho(\mathbf{Y}^*) = 0$. Thus, the optimal b is $b^* = \rho(\mathbf{Y}^*)/\eta^2(\mathbf{Y}^*)$ (see, e.g., Mathar & Groenen, 1991). Inserting b^* in $\sigma_r(b\mathbf{Y}^*)$ gives

$$\sigma_r(b^*\mathbf{Y}^*) = \eta_\delta^2 - \left(\frac{\rho(\mathbf{Y}^*)}{\eta(\mathbf{Y}^*)}\right)^2. \tag{11.3}$$

Dividing both sides by η_δ^2 yields

$$\sigma_n(b^*\mathbf{Y}^*) = \frac{\sigma_r(b^*\mathbf{Y}^*)}{\eta_\delta^2} = 1 - \left(\frac{\rho(\mathbf{Y}^*)}{\eta_\delta\eta(\mathbf{Y}^*)}\right)^2, \tag{11.4}$$

where the last term is equal to the square of Tucker's congruence coefficient with distances and dissimilarities. The congruence coefficient is always between -1 and 1, due to the Cauchy–Schwarz inequality. Moreover,

[1] The congruence coefficient of two variables X and Y, c, is the correlation of these variables about their origin or "zero", not about their means (as in Pearson's correlation coefficient). The coefficient c was first used by Tucker (see, e.g., Tucker, 1951) to assess the similarity of corresponding factors resulting from factor analyses of different samples. It is defined as $c = (\sum_i (x_i y_i)/[(\sum_i x_i^2)(\sum_i y_i^2)]^{1/2}$.

negative congruence coefficients are impossible because distances and dissimilarities are nonnegative. Hence, at a stationary point $\mathbf{X}^*$, it holds that $0 \leq \sigma_n(\mathbf{X}^*) \leq 1$. The value of $\sigma_n(\mathbf{X}^*)$ is the proportion of variation of the dissimilarities not accounted for by the distances, and $1 - \sigma_n(\mathbf{X}^*)$ is the fitted proportion, a *coefficient of determination*. Because distances and dissimilarities both are positive, congruence coefficients tend to be close to 1 in practice. Therefore, values of $\sigma_n(\mathbf{X}^*) < .10$ are usually not difficult to obtain.

Using the normalized Stress (as defined in this section) gives a clear interpretation that does not depend on the scale of the dissimilarities.

Relation Between Normalized Stress and Stress-1

Fortunately, there exists a simple relation between the normalized Stress σ_n and Stress-1 σ_1. In fact, we show here that $\sigma_1^2 = \sigma_n$ at a local minimum if we allow for a rescaling of the solution. Note that Raw Stress σ_r and normalized Stress σ_n differ from most other Stress measures in that no square root is taken.

Let $\mathbf{X}^*$ be a local minimum obtained by minimizing σ_n. De Leeuw and Heiser (1980) and De Leeuw (1988) proved that for $\mathbf{X}^*$ it holds that $\eta^2(\mathbf{X}^*) = \rho(\mathbf{X}^*)$. This result implies that

$$\sigma_n(\mathbf{X}^*) = 1 - \frac{\eta^2(\mathbf{X}^*)}{\eta_\delta^2}. \tag{11.5}$$

Now, for the same configuration, Stress-1 can be expressed as

$$\begin{aligned} \sigma_1^2(\mathbf{X}^*) &= \frac{\eta_\delta^2 + \eta^2(\mathbf{X}^*) - 2\rho(\mathbf{X}^*)}{\eta^2(\mathbf{X}^*)} = \frac{\eta_\delta^2 - \eta^2(\mathbf{X}^*)}{\eta^2(\mathbf{X}^*)} \\ &= \frac{\eta_\delta^2}{\eta^2(\mathbf{X}^*)} - 1. \end{aligned}$$

From (11.5) we have $\eta_\delta^2/\eta^2(\mathbf{X}^*) = 1/(1 - \sigma_n(\mathbf{X}^*))$, so that

$$\sigma_1^2(\mathbf{X}^*) = \frac{\sigma_n(\mathbf{X}^*)}{1 - \sigma_n(\mathbf{X}^*)}.$$

However, the scale of $\mathbf{X}^*$ is not optimal for Stress-1. By allowing for a scaling factor b, Stress-1 becomes

$$\sigma_1^2(b\mathbf{X}^*) = \frac{\eta_\delta^2 + b^2\eta^2(\mathbf{X}^*) - 2b\rho(\mathbf{X}^*)}{b^2\eta^2(\mathbf{X}^*)} = \frac{\eta_\delta^2 + (b^2 - 2b)\eta^2(\mathbf{X}^*)}{b^2\eta^2(\mathbf{X}^*)}.$$

An optimal b can be found by differentiating $\sigma_1^2(b\mathbf{X}^*)$ with respect to b; that is,

$$\frac{\partial \sigma_1^2(b\mathbf{X}^*)}{\partial b} = \frac{2b^2[b-1]\eta^4(\mathbf{X}^*) - 2b\eta^2(\mathbf{X}^*)[\eta_\delta^2 + (b^2 - 2b)\eta^2(\mathbf{X}^*)]}{b^4\eta^2(\mathbf{X}^*)}$$

$$= \frac{2b\eta^2(\mathbf{X}^*) - 2\eta_\delta^2}{b^3},$$

which is equal to zero for $b^* = \eta_\delta^2/\eta^2(\mathbf{X}^*)$. Inserting b^* into $\sigma_1^2(b\mathbf{X}^*)$ yields

$$\begin{aligned}
\sigma_1^2(b^*\mathbf{X}^*) &= \frac{\eta_\delta^2 + \frac{\eta_\delta^4}{\eta^4(\mathbf{X}^*)}\eta^2(\mathbf{X}^*) - 2\frac{\eta_\delta^2}{\eta^2(\mathbf{X}^*)}\eta^2(\mathbf{X}^*)}{\frac{\eta_\delta^4}{\eta^4(\mathbf{X}^*)}\eta^2(\mathbf{X}^*)} \\
&= \frac{\eta_\delta^2/\eta^2(\mathbf{X}^*) - 1}{\eta_\delta^2/\eta^2(\mathbf{X}^*)} \\
&= 1 - \frac{\eta^2(\mathbf{X}^*)}{\eta_\delta^2} = \sigma_n(\mathbf{X}^*).
\end{aligned}$$

This proves that Stress-1 is equal to normalized Stress at a local minimum if the scale is calibrated properly.

11.2 Other Fit Measures and Recent Algorithms

A whole variety of MDS loss functions have been proposed in the literature. In this section, we describe some of them. A summary of different fit measures and their relations is given by Heiser (1988a). Here, we restrict ourselves to the most commonly used MDS loss functions. One of the reasons for our emphasis on using Stress in MDS is that the majorization algorithm is a simple procedure for which nice theoretical convergence results have been derived (De Leeuw, 1988). In this section, we assume that the weights $w_{ij} = 1$, for all i, j. We start with a brief overview of other algorithms for minimizing Stress.

Algorithms for Minimizing Raw Stress

Let us first turn to raw Stress. Apart from majorization, several other approaches for minimizing raw Stress have been reported in the literature. Some of these approaches are equivalent to the majorization algorithm discussed in Section 8.6. For example, De Leeuw (1993) reparameterized the raw Stress function, where the coordinates are restricted to be a sum of some other fixed coordinate matrices. The algorithm is also based on majorization. A convex analysis approach for minimizing raw Stress (De Leeuw, 1977; Mathar, 1989; Mathar & Groenen, 1991; Meyer, 1993) leads to the same algorithm as the majorization approach. A relation between the convex analysis approach and the majorization approach (for the more general case of Minkowski distances) was discussed by Mathar (1994). A genetic algorithm to minimize raw Stress was proposed by Mathar and Žilinskas (1993), who found this a promising approach for small MDS

problems. Glunt, Hayden, and Raydan (1993) proposed a spectral gradient algorithm, which was, in one example, 10 times faster than the majorizing algorithm.

Implicitly Normalized Stress

In Section 3.2, it was indicated that raw Stress $\sigma_r(\mathbf{X})$ can be misleading, because it is dependent on the normalization of the dissimilarities. To circumvent this inconvenience, normalized Stress $\sigma_n(\mathbf{X})$ was introduced in Section 11.1. A different solution is to require explicitly $\eta_\delta^2 = c$, with c a positive constant (e.g., $\eta_\delta^2 = n(n-1)/2$), as was imposed in nonmetric MDS by (9.2). This solution is called *explicit* normalization. A third (but historically earlier) solution was pursued by Kruskal (1964a), which is called *implicit normalization*. Here, Stress is expressed in relation to the size of $\mathbf{X}$. More concretely, σ is divided by the sum of the squared distances in $\mathbf{X}$ and the root is taken of the total fraction; that is,

$$\sigma_1(\mathbf{X}) = \left(\frac{\sigma(\mathbf{X})}{\eta^2(\mathbf{X})}\right)^{1/2} = \left(\frac{\sum_{i<j}[\delta_{ij} - d_{ij}(\mathbf{X})]^2}{\sum_{i<j} d_{ij}(\mathbf{X})}\right)^{1/2}.$$

This expression, proposed by Kruskal (1964a) is called *Stress formula 1*. Note that often Stress-1 is expressed using disparities $\widehat{d}_{ij}$ to allow for transformations. Throughout this chapter, we use dissimilarities δ_{ij} instead of $\widehat{d}_{ij}$ for reasons of notational simplicity. Kruskal and Carroll (1969) proved that implicitly or explicitly normalized Stress gives the same configuration up to scaling constant. A different form of implicit normalization is *Stress formula 2*; that is,

$$\sigma_2(\mathbf{X}) = \left(\frac{\sum_{i<j}[\delta_{ij} - d_{ij}(\mathbf{X})]^2}{\sum_{i<j}[d_{ij}(\mathbf{X}) - \bar{d}]^2}\right)^{1/2},$$

with $\bar{d}$ the average distance. This version of Stress was introduced to avoid a particular type of degeneracy in unfolding, that is, solutions where all distances are equal.

The Alienation Coefficient and the Guttman–Lingoes Programs

Another error measure, the *alienation coefficient*, abbreviated as K, is used only in combination with rank-images as target distances. K can be derived from normalized Stress $\sigma_n(\mathbf{X})$ as defined in (11.4) by setting $\delta_{ij} = d_{ij}^*$, where d_{ij}^* denotes the disparity obtained by the rank-image transformation (see Section 9.5). Thus, the alienation coefficient is defined as

$$K = \left(1 - \frac{[\sum_{i<j} d_{ij}^* d_{ij}(\mathbf{X})]^2}{\sum_{i<j}(d_{ij}^*)^2 \sum_{i<j} d_{ij}^2(\mathbf{X})}\right)^{1/2}. \tag{11.6}$$

The quotient term in (11.6) is known as the *monotonicity coefficient*, μ (Guttman, 1981). It is similar to a correlation coefficient, which is easier to see if we rewrite it as

$$\mu = \frac{\sum_{i<j} d_{ij}^* d_{ij}(\mathbf{X})}{[\sum_{i<j}(d_{ij}^*)^2 \sum_{i<j} d_{ij}^2(\mathbf{X})]^{1/2}}. \tag{11.7}$$

Hence, μ differs from the usual Pearson correlation coefficient on the variables *distances* and *rank-images* in not subtracting the means from the variables. The regression line, therefore, runs through the origin and not the centroid of the *image diagram*, the plot of all points with coordinates (d_{ij}, d_{ij}^*). Note that in an image diagram all points are exactly on the bisector if and only if the solution is perfect. In that case, $\mu = 1$. Furthermore, μ is equal to Tucker's congruence coefficient of the distances and their rank-images.

For practical purposes, μ has the disadvantage that it takes on values close to 1 even if the MDS solution is far from perfect. We can, however, convert μ into the coefficient of alienation

$$K = (1 - \mu^2)^{1/2},$$

which yields values that vary over a greater range and, thus, are easier to distinguish. K is a measure for the "unexplained" variation of the points in the image diagram, whereas μ^2 is a *coefficient of determination*, that is, a measure for the "explained" variance. The smaller K, the more precise is the representation, or, conversely, the greater K, the worse the fit of the MDS model to the proximities. The squared alienation coefficient is equal to normalized Stress $\sigma_n(\mathbf{X})$ if rank-images are used instead of disparities. The Guttman–Lingoes programs and various other programs (see Appendix A) do ordinal MDS by attempting to minimize K rather than Stress.

Minimizing S-Stress

The S-Stress loss function of Takane, Young, and De Leeuw (1977),

$$\sigma_{AL}(\mathbf{X}) = \sum_{i<j}(d_{ij}^2(\mathbf{X}) - \delta_{ij}^2)^2, \tag{11.8}$$

is minimized by ALSCAL (see Appendix A). This loss function sums the differences of squared dissimilarities and squared distances. One of the reasons for using squared distances is that $\sigma_{AL}(\mathbf{X})$ is differentiable everywhere, even if $d_{ij}(\mathbf{X}) = 0$ for some pair i, j. Squaring distances and dissimilarities causes S-Stress to emphasize larger dissimilarities more than smaller ones, which may be viewed as a disadvantage of S-Stress. A fast Newton–Raphson procedure to minimize S-Stress was proposed by Browne (1987). An alternative algorithm was presented by Glunt, Hayden, and Liu (1991). For the full-dimensional case of $m = n - 1$, Gaffke and Mathar (1989) developed an algorithm that always yields a global minimum.

Maximum Likelihood MDS and MULTISCALE

The MULTISCALE loss function of Ramsay (1977) is based on the sum of the squared difference of the logarithm of the dissimilarities and the distances; that is,

$$\sigma_{MU}(\mathbf{X}) = \sum_{i<j} \left[\log(d_{ij}(\mathbf{X})) - \log(\delta_{ij})\right]^2 .$$

This loss function is used in a *maximum likelihood* (ML) framework. The likelihood is the probability that we find the data given $\mathbf{X}$. This probability is maximized in ML-MDS. For ML estimation, we need to assume independence among the residuals and a *lognormal* distribution of the residuals. In many cases, these assumptions are too rigid. However, if they do hold, then σ_{MU} has the advantage that confidence regions of the points can be obtained and that different models can be tested. If the residuals are assumed to be *normally* distributed, then MULTISCALE reduces to minimizing Stress. An advantage of using a logarithm in σ_{MU} is that the large dissimilarities do not determine the solution as much as when Stress is minimized. Conversely, dissimilarities close to zero are relatively important for the solution. The MULTISCALE program is discussed in Appendix A.

Further Algorithms and Developments

Groenen, De Leeuw, and Mathar (1996) discussed a least-squares loss function for MDS that includes Stress, S-Stress, and MULTISCALE as special cases. They used

$$\sigma_G(\mathbf{X}) = \sum_{i<j} w_{ij} \left[f(\delta_{ij}^2) - f(d_{ij}^2(\mathbf{X}))\right]^2 ,$$

where $f(z)$ is an increasing scalar function. For example, choosing $f(z) = z^{1/2}$ gives Stress, $f(z) = z$ gives S-Stress, and $f(z) = \log(z)$ gives the MULTISCALE loss function. They derive several properties of the gradient and hessian (the matrix of second derivatives) of this function. For example, it can be shown that S-Stress is differentiable everywhere (Takane et al., 1977) and that at a local minimum Stress has no zero distances (and thus is differentiable) if $w_{ij}\delta_{ij} > 0$ for all i, j (De Leeuw, 1984). Kearsley, Tapia, and Trosset (1998) provide an algorithm for the Stress and S-Stress versions of σ_G based on a globalized Newton's method, which they claim uses fewer iterations than the majorizing algorithm and yields lower Stress solutions.

To minimize Stress, Luengo, Raydan, Glunt, and Hayden (2002) have elaborated on the so-called spectral gradient algorithm. In a small comparison study, Groenen and Heiser (2000) found that the spectral gradient algorithm was the fastest algorithm, outperforming SMACOF and KYST. This may be of importance for MDS with a large number of objects.

A special case of σ_G occurs in applications in chemistry, where the objective is to find stable molecules. The energy function used is essentially equal to $\sigma_G(\mathbf{X})$ with $f(z) = z^6$ and $\delta_{ij} = 1$ for all i, j. The gradient becomes so steep that this problem turns out to be combinatorial in nature (see, e.g., Xue, 1994).

De Leeuw and Groenen (1997) considered the problem of finding those dissimilarity matrices for which a given $\mathbf{X}$ is a local minimum (or has a zero gradient) for Stress. This problem is called *inverse MDS*. If this set of dissimilarities is large, then the local minimum is not very informative. After all, many dissimilarity matrices have $\mathbf{X}$ as a (possible) local minimum. Groenen et al. (1996) discuss the problem of inverse MDS for the loss function $\sigma_G(\mathbf{X})$.

An overview of various algorithmic approaches in MDS is given by Mathar (1997).

11.3 Using Weights in MDS

So far, we have used the weights w_{ij} only to indicate nonmissing dissimilarities. Choosing $w_{ij} = 1$ indicates that for object pair ij a dissimilarity has been observed, whereas $w_{ij} = 0$ is used for pairs ij where a dissimilarity is "missing". As zero weights lead to zero error terms in the Stress loss function, the distance that corresponds to a missing data value cannot be assessed in terms of fit. Hence, it contributes nothing to the Stress, whatever its value. But this also means that this distance cannot be interpreted directly, but only in terms of what is implied by the distances that represent given data. If the number of missing dissimilarities gets large or if they form special block patterns (as in Table 6.1, e.g.), we should take care in interpreting distances that "represent" missing data. Then, one should emphasize the interpretation of distances that represent observed data values.

Using Particular Weighting Schemes

Instead of using w_{ij}'s that are zero or one in the minimization of Raw Stress, we can apply any positive value for w_{ij}. Heiser (1988a) exploited this powerful idea and distinguished several weighting schemes of which we discuss a few below.

Consider the S-Stress loss function. Instead of (11.8), S-Stress may also be written as

$$\sigma_{AL}(\mathbf{X}) = \sum_{i<j} (\delta_{ij} + d_{ij}(\mathbf{X}))^2 (\delta_{ij} - d_{ij}(\mathbf{X}))^2,$$

which shows that each S-Stress error term consists of two factors: the square of the ordinary Stress residual $(\delta_{ij} - d_{ij}(\mathbf{X}))^2$ and a weighting term $(\delta_{ij} +$

$d_{ij}(\mathbf{X}))^2$ that is also dependent on $d_{ij}(\mathbf{X})$. Assume that the residuals are reasonably small. Then, $(\delta_{ij} + d_{ij}(\mathbf{X}))^2$ can be approximated by replacing $d_{ij}(\mathbf{X})$ by δ_{ij} so that

$$(\delta_{ij} + d_{ij}(\mathbf{X}))^2 \approx 4\delta_{ij}^2.$$

Therefore, the minimization of S-Stress can be approximated by minimizing Stress choosing $w_{ij} = 4\delta_{ij}^2$. This approximation shows that optimizing S-Stress tends to lead to small errors for the large dissimilarities and large errors for the smaller dissimilarities. In other words, large dissimilarities are much better represented than the small ones.

McGee (1966) proposed the idea of *elastic scaling*. This form of MDS fits relative residuals so that the proper representation of small dissimilarities is equally important as fitting large dissimilarities. The loss function minimized in elastic scaling is

$$\sigma_{EL}(\mathbf{X}) = \sum_{i<j}(1 - d_{ij}(\mathbf{X})/\delta_{ij})^2 = \sum_{i<j}\delta_{ij}^{-2}(\delta_{ij} - d_{ij}(\mathbf{X}))^2.$$

Thus, choosing $w_{ij} = \delta_{ij}^{-2}$ makes minimizing raw Stress do the same as McGee's elastic scaling.

An MDS method popular in the pattern recognition literature is called *Sammon mapping* after Sammon (1969). The loss function can be expressed as

$$\sigma_{SAM}(\mathbf{X}) = \sum_{i<j}\delta_{ij}^{-1}(\delta_{ij} - d_{ij}(\mathbf{X}))^2,$$

which is identical to raw Stress for $w_{ij} = \delta_{ij}^{-q}$, with $q = -1$. The objective is somewhat similar to that of elastic scaling of McGee (1966), although larger dissimilarities still are somewhat more emphasized in the MDS solution.

The MULTISCALE loss function of Ramsay (1977) can be written as

$$\sigma_{MU}(\mathbf{X}) = \sum_{i<j}\log^2(d_{ij}(\mathbf{X})/\delta_{ij}),$$

showing that the squared logarithm of the relative error is minimized. Provided that the relative error is close to one, $\log(a)$ can be approximated by $a - 1$; that is,

$$\sigma_{MU}(\mathbf{X}) = \sum_{i<j}\log^2(d_{ij}(\mathbf{X})/\delta_{ij}) \approx \sum_{i<j}(1 - d_{ij}(\mathbf{X})/\delta_{ij})^2 = \sigma_{EL}(\mathbf{X}).$$

Thus, the objective of MULTISCALE and elastic scaling scaling coincides in that errors are corrected for the size of the dissimilarities.

The examples above show that choosing w_{ij} as a power of δ_{ij} leads to (approximations) of other loss functions. For this reason, Buja and Swayne

(2002) incorporated the weights $w_{ij} = \delta_{ij}^q$ in their GGVIS software (see Appendix A). Figure 11.1 shows solutions for ratio MDS of the facial expression data of Table 4.4 using several values of q. The middle panels show the standard solution with $q = 0$ and all weights being one as $w_{ij} = \delta_{ij}^0 = 1$. The Shepard diagram in the middle-right panel shows that the size of the errors does not depend on the size of the dissimilarities. Note that the solution for $q = 0$ is the same as Figures 4.8 and 4.9 up to a rotation.

In contrast, for $q = -5$, the large dissimilarities show much error and thus are not well represented. For example, the two worst fitting large dissimilarities are between faces 12 and 13 ("Knows plane will crash" and "Light sleep") and faces 3 and 7 ("Very pleasant surprise" and "Anger at seeing dog beaten"). Both distances are too small in this representation. In this case, the small dissimilarities have little error, and thus can be safely interpreted.

The reverse situation occurs for $q = 5$ where the large dissimilarities are fitted with almost no error and there is quite some error in the representation of the smaller errors. The Shepard plot shows three or four bad-fitting small dissimilarities, which turn out to be connected with face 12. However, face 12 is located so far away because it has several large dissimilarities with other faces (2, 3, 4, 5, 8, 9, and 13) that are all large and represented with almost no error. This compromise is typical for choosing large q. Hence, only large distances can be properly interpreted and small distances should be interpreted with care. If the dissimilarities have some clustering, then choosing a large q may reveal a clearer clustering structure than choosing all $w_{ij} = 1$.

Summarizing, to emphasize the representation of small dissimilarities, choose a large negative q. For a proper representation of the large dissimilarities, choose a large q. If you want to use relative errors to penalize small deviations for small dissimilarities equally heavy as large deviations for large dissimilarities, choose $q = -2$. To measure the error directly without any modification, choose $q = 1$.

Using Weights on Substantive Grounds

All of the above schemes for picking weights w_{ij} had in common that the weights were specified on the basis of general and rather formal considerations. We conclude this discussion about using weights in MDS by pointing out that weights can also be picked on a substantive basis. One particular choice for w_{ij} would be to set it equal to the empirically assessed reliability of the proximity p_{ij}. This means that highly reliable proximities have more impact on the MDS solution than unreliable ones.

The problem, of course, is that reliabilities are seldom collected, because to collect one set of proximities is typically demanding enough. Estimating reliabilities from other information is not that simple either. Consider, for example, the Morse code data in Table 4.2. We may come to the conclusion

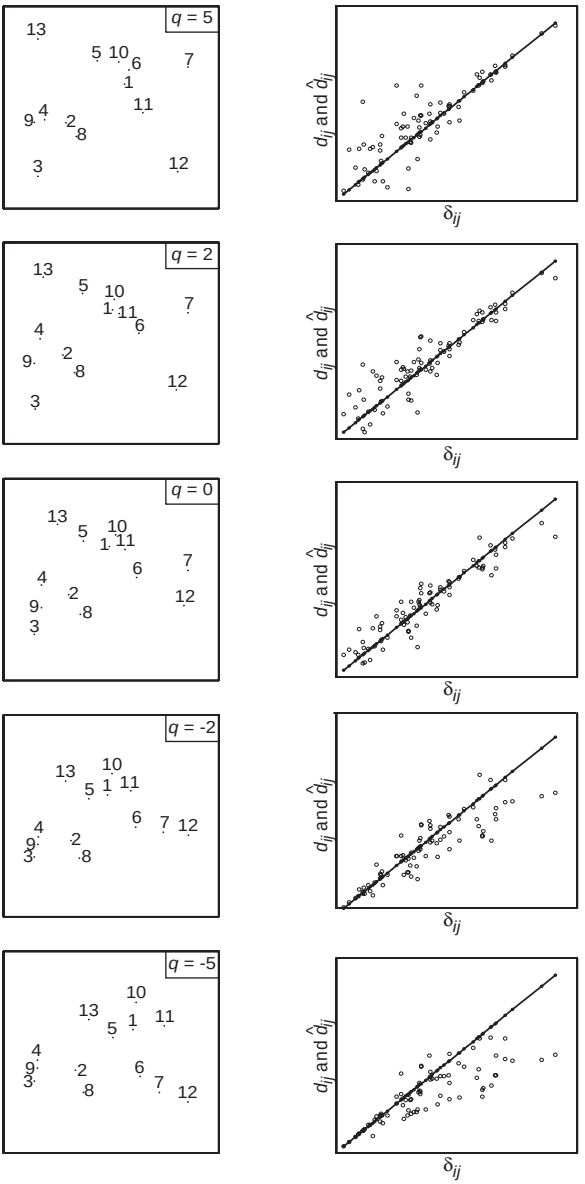


FIGURE 11.1. Ratio MDS of facial expression data of Table 4.4 where $w_{ij} = \delta_{ij}^q$ for q is $-5, -2, 0, 2$, and 5. The left panels show the configurations, the right panels the corresponding Shepard plots.

that these data are essentially symmetric, symmetrize the data, and use the degree of asymmetry as a measure of the unreliability of the confusion probability for each pair. This approach sounds plausible but a closer study of the asymmetries in Chapter 23 reveals that the asymmetries are clearly not just random. Other solutions to obtain reliabilities from the data we have could be considered. For example, one may feel that confusing a signal with itself relates to reliability, and then compute a reliability measure for a pair of signals on the basis of their individual reliabilities. Obviously, many such measures could be considered, and there are many ways to collect reliabilities directly such as, for example, simply replicating the proximity observations at least twice. What is and what is not a good reliability estimate must be decided within the substantive context of the particular data.

Note also that proximities are often data that are collapsed over individuals. This is true too for the Morse code data in Table 4.2. But different individuals can agree on the similarity of some pairs, and disagree on others. This information could also be used to weight the data so that the respondents' common perceptual space relies more on data where interindividual agreement is relatively high.

11.4 Exercises

Exercise 11.1 Compute, by hand, the alienation coefficient for the p_{ij} and d_{ij} in Table 9.2, p. 206.

Exercise 11.2 Consider the data in Table 1.3, p. 10. One may attempt to weight these data somehow to account for possible differences in their reliability. For example, the students who generated these similarity ratings were certainly less familiar with (what was then) "Congo" than with the U.S.A. or the U.K.

(a) Develop a scheme that generates reliability estimates for each proximity in Table 1.3 on the basis of simple ratings of the different nations in terms of their assumed familiarity to the students in this experiment. (Hint: One way of rating the reliability of the proximity p_{ij} is to multiply the familiarity ratings for i and for j.)

(b) Use these estimates to weight the proximities, and redo the (ordinal) MDS with these weights.

(c) Discuss any differences (configuration, Stress, pointwise Stress, interpretation) of the weighted MDS solution and the "unweighted" (or, rather, unit-weights) solution in Figure 1.5.

Exercise 11.3 There are many ways to generate weights δ_{ij} for proximities p_{ij}.

(a) MDS is often used to analyze the structure of correlation matrices (see, e.g., Tables 1.1, 5.1, and 20.1). Discuss some ways to sensibly weight correlations for potentially more robust MDS analyses of such data.

(b) Consider the similarity judgments on facial expressions described in Section 4.3. The respondents may make these judgments with different degrees of confidence. How could this information be collected and incorporated into the MDS analysis?

(c) Even the similarities on the colors in Table 4.1 could be weighted. One possible way is to assume that primary colors (red, blue, green) generate more reliable judgments. Devise a method to generate weights on that basis.

Exercise 11.4 Consider the data in Table 4.1, p. 65. Their Shepard diagram in Figure 4.2 exhibits a slightly nonlinear trend. Find a transformation on the similarities that linearizes the relationship of these data to their MDS distances. Justify this transformation in terms of psychophysics, if possible. Redo the MDS analysis with the rescaled data and a linear MDS model.

Exercise 11.5 Dissimilarities may be related to nonlinear manifolds that are embedded in very high-dimensional space. For example, a constant face that an observer looks at from different angles in space corresponds to different points in the space of its image pixels on the retina. This space has thousands of dimensions, but the points that represent the faces still lie on some nonlinear manifolds (with the angles as parameters) within this space. MDS does not necessarily uncover such manifolds, because of "using greedy optimization techniques that first fit the large-scale (linear) structure of the data, before making small-scale (nonlinear) refinements" (Tenenbaum, 1998, p. 683). One suggestion to solve this problem is to use a "bottom-up" approach that computes distances for points in small local environments only, and then build up large distances by concatenating such distances over geodesics within the manifolds (given that these manifolds are densely packed with points).

(a) Construct a so-called Swiss roll of points in 3D as in the left panel of Figure 11.2. A Swiss roll can be made as follows. Generate two uniformly distributed vectors $\mathbf{u}$ and $\mathbf{v}$ of n points (say, choose $n = 1000$). Then, the coordinates are $x_i = \frac{1}{2} v_i \sin(4\pi v_i)$, $y_i = u_i - \frac{1}{2}$, and $z_i = \frac{1}{2} v_i \cos(4\pi v_i)$.

(b) Compute Euclidean distances for the points in your manifold, and then use metric MDS in an attempt to recover the original Swiss roll configuration.

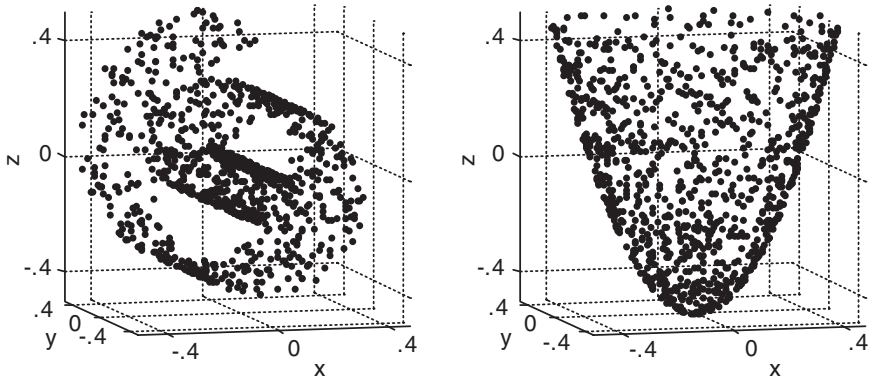


FIGURE 11.2. Manifolds described in Exercise 11.5. The left panel shows the "Swiss roll" manifold, the right panel a "bowl".

(c) Now focus predominantly on small distances by a suitable weighting pattern, and repeat the MDS analyses with small or even zero weights on large distances. Check to what extent this approach manages to unroll the Swiss roll into a plane. [Shepard and Carroll (1966) call this the "intrinsic" dimensionality of the manifold.] Compare the resulting MDS configuration to the one obtained in Exercise (b) above.

(d) Repeat (a) to (c), but now for a "bowl" of points in 3D. A bowl is generated similarly as the Swiss roll in (a), except that $x_i = \frac{1}{2}v_i^{1/2}\cos(2\pi u_i)$, $y_i = \frac{1}{2}v_i^{1/2}\sin(2\pi u_i)$, and $z_i = v_i - \frac{1}{2}$. Can you "flatten" the bowl-like manifold by appropriate weighting into a 2D MDS configuration?

12
Classical Scaling

Because the first practical method available for MDS was a technique due to Torgerson (1952, 1958) and Gower (1966), *classical scaling* is also known under the names *Torgerson scaling* and *Torgerson–Gower scaling*. It is based on theorems by Eckart and Young (1936) and by Young and Householder (1938). The basic idea of classical scaling is to assume that the dissimilarities are distances and then find coordinates that explain them. In (7.5) a simple matrix expression is given between the matrix of squared distances $\mathbf{D}^{(2)}(\mathbf{X})$ (we also write $\mathbf{D}^{(2)}$ for short) and the coordinate matrix $\mathbf{X}$, which shows how to get squared Euclidean distances from a given matrix of coordinates and then scalar products from these distances. In Section 7.9, the reverse was discussed, that is, how to find the coordinate matrix given a matrix of scalar products $\mathbf{B} = \mathbf{X}\mathbf{X}'$. Classical scaling uses the same procedure but operates on squared dissimilarities $\boldsymbol{\Delta}^{(2)}$ instead of $\mathbf{D}^{(2)}$, because the latter is unknown. This method is popular because it gives an analytical solution, requiring no iterations.

12.1 Finding Coordinates in Classical Scaling

We now explain some fundamental issues in classical scaling. How do we arrive at a scalar product matrix $\mathbf{B}$, given a matrix of squared distances $\mathbf{D}^{(2)}$? Because distances do not change under translations, we assume that $\mathbf{X}$ has column means equal to 0. Remember from (7.5) that the squared

distances are computed from $\mathbf{X}$ by

$$\mathbf{D}^{(2)} = \mathbf{c1}' + \mathbf{1c}' - 2\mathbf{XX}' = \mathbf{c1}' + \mathbf{1c}' - 2\mathbf{B}, \tag{12.1}$$

where $\mathbf{c}$ is the vector with the diagonal elements of $\mathbf{XX}'$. Multiplying the left and the right sides by the centering matrix $\mathbf{J} = \mathbf{I} - n^{-1}\mathbf{11}'$ and by the factor $-\frac{1}{2}$ gives

$$\begin{aligned} -\tfrac{1}{2}\mathbf{JD}^{(2)}\mathbf{J} &= -\tfrac{1}{2}\mathbf{J}(\mathbf{c1}' + \mathbf{1c}' - 2\mathbf{XX}')\mathbf{J} \\ &= -\tfrac{1}{2}\mathbf{Jc1}'\mathbf{J} - \tfrac{1}{2}\mathbf{J1c}'\mathbf{J} + \tfrac{1}{2}\mathbf{J}(2\mathbf{B})\mathbf{J} \\ &= -\tfrac{1}{2}\mathbf{Jc0}' - \tfrac{1}{2}\mathbf{0c}'\mathbf{J} + \mathbf{JBJ} = \mathbf{B}. \end{aligned} \tag{12.2}$$

The first two terms are zero, because centering a vector of ones yields a vector of zeros ($\mathbf{1}'\mathbf{J} = \mathbf{0}$). The centering around $\mathbf{B}$ can be removed because $\mathbf{X}$ is column centered, and hence so is $\mathbf{B}$. The operation in (12.2) is called *double centering*. To find the MDS coordinates from $\mathbf{B}$, we factor $\mathbf{B}$ by eigendecomposition, $\mathbf{Q\Lambda Q}' = (\mathbf{Q\Lambda}^{1/2})(\mathbf{Q\Lambda}^{1/2})' = \mathbf{XX}'$. The method of classical scaling only differs from this procedure in that the matrix of squared distances $\mathbf{D}^{(2)}$ is replaced by the squared dissimilarities $\mathbf{\Delta}^{(2)}$.

The procedure for classical scaling is summarized in the following steps.

1. Compute the matrix of squared dissimilarities $\mathbf{\Delta}^{(2)}$.

2. Apply double centering to this matrix:

$$\mathbf{B}_{\mathbf{\Delta}} = -\tfrac{1}{2}\mathbf{J\Delta}^{(2)}\mathbf{J}. \tag{12.3}$$

3. Compute the eigendecomposition of $\mathbf{B}_{\mathbf{\Delta}} = \mathbf{Q\Lambda Q}'$.

4. Let the matrix of the first m eigenvalues *greater than zero* be $\mathbf{\Lambda}_+$ and $\mathbf{Q}_+$ the first m columns of $\mathbf{Q}$. Then, the coordinate matrix of classical scaling is given by $\mathbf{X} = \mathbf{Q}_+\mathbf{\Lambda}_+^{1/2}$.

If $\mathbf{\Delta}$ happens to be a Euclidean distance matrix, then classical scaling finds the coordinates up to a rotation. Note that the solution $\mathbf{Q}_+\mathbf{\Lambda}_+^{1/2} = \mathbf{X}$ is a principal axes solution (see Section 7.10). In step 4, negative eigenvalues can occur but not if $\mathbf{\Delta}$ is a Euclidean distance matrix (see Chapter 19). In classical scaling, the negative eigenvalues (and its eigenvectors) are simply ignored as error.

Classical scaling minimizes the loss function

$$\begin{aligned} L(\mathbf{X}) &= ||-\tfrac{1}{2}\mathbf{J}[\mathbf{D}^{(2)}(\mathbf{X}) - \mathbf{\Delta}^{(2)}]\mathbf{J}||^2 \\ &= ||\mathbf{XX}' + \tfrac{1}{2}\mathbf{J\Delta}^{(2)}\mathbf{J}||^2 \\ &= ||\mathbf{XX}' - \mathbf{B}_{\mathbf{\Delta}}||^2, \end{aligned} \tag{12.4}$$

sometimes called *Strain* (see Carroll & Chang, 1972). Gower (1966) proved that choosing the classical scaling solution solves (12.4).[1]

A nice property of classical scaling is that the dimensions are nested. This means that, for example, the first two dimensions of a 3D classical scaling solution are the same as the two dimensions of a 2D classical scaling solution. Note that MDS by minimizing Stress does not give nested solutions.

It remains to be seen what dimensionality one should choose. Sibson (1979) suggests that the sum of the eigenvalues in $\mathbf{\Lambda}_+$ should approximate the sum of all eigenvalues in $\mathbf{\Lambda}$, so that small negative eigenvalues cancel out small positive eigenvalues. For a rationale of this proposal, see Chapter 19.

12.2 A Numerical Example for Classical Scaling

As an example, we use the faces data from Table 4.4. Here, we consider the first four items only; that is,

$$\mathbf{\Delta} = \begin{bmatrix} 0 & 4.05 & 8.25 & 5.57 \\ 4.05 & 0 & 2.54 & 2.69 \\ 8.25 & 2.54 & 0 & 2.11 \\ 5.57 & 2.69 & 2.11 & 0 \end{bmatrix}, \text{ so that } \mathbf{\Delta}^{(2)} = \begin{bmatrix} .00 & 16.40 & 68.06 & 31.02 \\ 16.40 & .00 & 6.45 & 7.24 \\ 68.06 & 6.45 & .00 & 4.45 \\ 31.02 & 7.24 & 4.45 & .00 \end{bmatrix}.$$

The second step in classical scaling is to compute

$$\begin{aligned} \mathbf{B}_{\mathbf{\Delta}} &= -\tfrac{1}{2}\mathbf{J}\mathbf{\Delta}^{(2)}\mathbf{J} \\ &= -\frac{1}{2}\begin{bmatrix} \frac{3}{4} & -\frac{1}{4} & -\frac{1}{4} & -\frac{1}{4} \\ -\frac{1}{4} & \frac{3}{4} & -\frac{1}{4} & -\frac{1}{4} \\ -\frac{1}{4} & -\frac{1}{4} & \frac{3}{4} & -\frac{1}{4} \\ -\frac{1}{4} & -\frac{1}{4} & -\frac{1}{4} & \frac{3}{4} \end{bmatrix} \begin{bmatrix} .00 & 16.40 & 68.06 & 31.02 \\ 16.40 & .00 & 6.45 & 7.24 \\ 68.06 & 6.45 & .00 & 4.45 \\ 31.02 & 7.24 & 4.45 & .00 \end{bmatrix} \begin{bmatrix} \frac{3}{4} & -\frac{1}{4} & -\frac{1}{4} & -\frac{1}{4} \\ -\frac{1}{4} & \frac{3}{4} & -\frac{1}{4} & -\frac{1}{4} \\ -\frac{1}{4} & -\frac{1}{4} & \frac{3}{4} & -\frac{1}{4} \\ -\frac{1}{4} & -\frac{1}{4} & -\frac{1}{4} & \frac{3}{4} \end{bmatrix} \\ &= \begin{bmatrix} 20.52 & 1.64 & -18.08 & -4.09 \\ 1.64 & -.83 & 2.05 & -2.87 \\ -18.08 & 2.05 & 11.39 & 4.63 \\ -4.09 & -2.87 & 4.63 & 2.33 \end{bmatrix}. \end{aligned}$$

In the third step, we compute the eigendecomposition of $\mathbf{B}_{\mathbf{\Delta}}$; that is, $\mathbf{B}_{\mathbf{\Delta}} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}'$ with

$$\mathbf{Q} = \begin{bmatrix} .77 & .04 & .50 & -.39 \\ .01 & -.61 & .50 & .61 \\ -.61 & -.19 & .50 & -.59 \\ -.18 & .76 & .50 & .37 \end{bmatrix} \text{ and } \mathbf{\Lambda} = \begin{bmatrix} 35.71 & .00 & .00 & .00 \\ .00 & 3.27 & .00 & .00 \\ .00 & .00 & .00 & .00 \\ .00 & .00 & .00 & -5.57 \end{bmatrix}.$$

There are two positive eigenvalues, one zero eigenvalue due to the double centering and one negative eigenvalue.[2] For this example, we can construct

[1] Note that Kloek and Theil (1965) also derived the classical scaling solution, but more in the sense of a how-to-do construction scheme than in terms of algebra.

[2] Double centering introduces a linear dependency, because if the columns of a matrix add up to the zero vector, then any column can be expressed as a linear combination

at most two dimensions in Euclidean space. Step 4 tells us that the configuration $\mathbf{X}$ is found by

$$\begin{aligned}\mathbf{X} &= \mathbf{Q}_+\boldsymbol{\Lambda}_+^{1/2} \\ &= \begin{bmatrix} .77 & .04 \\ .01 & -.61 \\ -.61 & -.19 \\ -.18 & .76 \end{bmatrix} \begin{bmatrix} 5.98 & .00 \\ .00 & 1.81 \end{bmatrix} = \begin{bmatrix} 4.62 & .07 \\ .09 & -1.11 \\ -3.63 & -.34 \\ -1.08 & 1.38 \end{bmatrix}.\end{aligned}$$

12.3 Choosing a Different Origin

Usually, $\mathbf{X}$ is constructed so that its columns sum to zero. This means that the origin of configuration $\mathbf{X}$ coincides with the center of gravity of its points (centroid). Choosing this origin is, however, not necessarily the best choice. In psychological research, for example, some objects may be less familiar to the respondents, and thus lead to less reliable distance estimates than others. In such a case, it is wiser to pick as an origin a point that is based more on the points associated with less error. How could this be accomplished?

For a general solution, consider picking some arbitrary point s as the new origin, with the restriction that s lies in the space of the other points. That is, in terms of algebra, point s should lie in the row space of $\mathbf{X}$; that is, the coordinate vector of s is a weighted sum of the rows of $\mathbf{X}$, $\mathbf{s}' = \mathbf{w}'\mathbf{X}$, where $\mathbf{w}'$ is an m-element row vector of weights. With s as the new origin, the point coordinates become

$$\mathbf{X}_s = \mathbf{X} - \mathbf{1}\mathbf{s}' = \mathbf{X} - \mathbf{1}\mathbf{w}'\mathbf{X} = (\mathbf{I} - \mathbf{1}\mathbf{w}')\mathbf{X} = \mathbf{P}_w\mathbf{X}. \tag{12.5}$$

If the weight vector $\mathbf{w}$ is chosen such that $\mathbf{w}'\mathbf{1} = 1$, then $\mathbf{P}_w$ is a *projector*[3]. If $\mathbf{B} = \mathbf{X}\mathbf{X}'$, one obtains $\mathbf{B}_s = \mathbf{X}_s\mathbf{X}_s'$ after projecting $\mathbf{X}$ to a new origin s. In terms of the old origin, $\mathbf{B}_s = \mathbf{P}_w\mathbf{B}\mathbf{P}_w'$.

If one chooses a particular object i as the origin, then $\mathbf{w}' = [0, \ldots, 1, \ldots, 0]$, where the 1 is in the ith position. If one picks the centroid as the origin, then $\mathbf{w}' = [1/n, \ldots, 1/n]$. Another choice is to pick the weights in $\mathbf{w}$ so that they reflect the reliability of the objects. In this case, unreliable elements should have a weight close to zero and reliable elements a high value. In this way, the origin will be attracted more towards the reliable points.

---

of the other columns. Hence, a doubly centered matrix does not have full rank, and, therefore, it has at least one zero eigenvalue (see Chapter 7). The negative eigenvalue shows that $\boldsymbol{\Delta}$ is not a matrix of Euclidean distances (see Section 19.1).

[3]For every projector matrix $\mathbf{P}$ it holds that $\mathbf{P}\mathbf{P} = \mathbf{P}$ (*idempotency*). In the given case, it is also true that $\mathbf{P}_s\mathbf{1} = \mathbf{0}$ (Schönemann, 1970).

Instead of using $\mathbf{J}$ in the double-centering formula, we can also use the projector $\mathbf{P}_w$. Then, step 2 in classical scaling becomes $\mathbf{B}_{\boldsymbol{\Delta}} = -\frac{1}{2}\mathbf{P}_w\boldsymbol{\Delta}^{(2)}\mathbf{P}_w'$. The zero eigenvalue of $\mathbf{B}_{\boldsymbol{\Delta}}$ has eigenvector $\mathbf{w}$, so that the weighted average (using weights $\mathbf{w}$) of the classical scaling coordinate matrix $\mathbf{X}$ is equal to zero.

12.4 Advanced Topics

A solution for classical scaling with linear constraints was discussed by Carroll, Green, and Carmone (1976), De Leeuw and Heiser (1982), and Ter Braak (1992). The linear constraints imposed on $\mathbf{X}$ require $\mathbf{X} = \mathbf{YC}$, where $\mathbf{Y}$ is an $n \times r$ matrix of r external variables, and $\mathbf{C}$ are weights to be optimized by classical scaling. (This type of constraint was also discussed in Section 10.3 for constrained MDS with Stress.)

How can the weights in $\mathbf{C}$ be computed? Let $\mathbf{Y} = \mathbf{P}\boldsymbol{\Phi}\mathbf{Q}'$ be the singular value decomposition. Then $\mathbf{X} = \mathbf{YC} = \mathbf{P}\boldsymbol{\Phi}\mathbf{Q}'\mathbf{C} = \mathbf{PC}_*$, where $\mathbf{P}$ is orthonormal ($\mathbf{P}'\mathbf{P} = \mathbf{I}$). The Strain loss function (12.4) used by classical scaling can be written as

$$\begin{aligned} L(\mathbf{C}) &= ||\mathbf{B}_{\boldsymbol{\Delta}} - \mathbf{YCC}'\mathbf{Y}'||^2 = ||\mathbf{B}_{\boldsymbol{\Delta}} - \mathbf{PC}_*\mathbf{C}_*'\mathbf{P}'||^2 \\ &= ||\mathbf{B}_{\boldsymbol{\Delta}}||^2 - ||\mathbf{P}'\mathbf{B}_{\boldsymbol{\Delta}}\mathbf{P}||^2 + ||\mathbf{P}'\mathbf{B}_{\boldsymbol{\Delta}}\mathbf{P} - \mathbf{C}_*\mathbf{C}_*'||^2, \quad (12.6) \end{aligned}$$

which can be verified by writing out all of the terms in the equation. Only the last term of (12.6) is dependent on $\mathbf{C}_*$. $L(\mathbf{C})$ is solved for $\mathbf{C}$ by the eigendecomposition of $\mathbf{P}'\mathbf{B}_{\boldsymbol{\Delta}}\mathbf{P} = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}'$ and choosing $\mathbf{C}_* = \mathbf{Q}_+\boldsymbol{\Lambda}_+^{1/2}$ (as in Step 4 in Section 12.1), so that $\mathbf{C} = \mathbf{Q}\boldsymbol{\Phi}^{-1}\mathbf{C}_*$.

To illustrate constrained classical scaling, we reanalyze the constrained MDS of the facial expression data in Section 10.3. The external constraint matrix $\mathbf{Y}$ is defined as in Table 10.2. The total loss of the constrained 2D classical scaling solution is 8366.3, which explains 75% of the sum-of-squares of $\mathbf{B}_{\boldsymbol{\Delta}}$, against 92% for the unconstrained classical scaling solution (with loss 2739.9). The corresponding solution is shown in Figure 12.1, where the external variables are represented by lines. The optimal weight matrix $\mathbf{C}$ obtained by constrained classical scaling is

$$\mathbf{C} = \begin{bmatrix} 1.283 & .482 \\ -.219 & .300 \\ -.445 & .782 \end{bmatrix}.$$

To get the coordinates $\mathbf{X}$ in Figure 12.1, we compute $\mathbf{X} = \mathbf{YC}$. This solution does not differ much from the constrained MDS solution in Figure 10.4. The main difference lies in the location of point 8.

Even for loss functions other than $L(\mathbf{X})$, classical scaling is optimal. Let $\mathbf{E} = \mathbf{XX}' - \mathbf{B}_{\boldsymbol{\Delta}}$, so that $L(\mathbf{X}) = ||\mathbf{E}||^2$. The loss can also be expressed as the

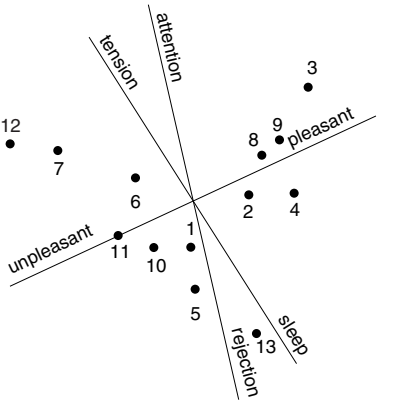


FIGURE 12.1. Constrained classical scaling of the facial expression data of Abelson and Sermat (1962).

sum of the squared eigenvalues of $\mathbf{E}$; that is, if $\mathbf{E} = \mathbf{K}\mathbf{\Phi}\mathbf{K}'$, then $||\mathbf{E}||^2 = \sum_i \phi_i^2$. This loss function is an example of an orthonormal invariant norm, because the value of the loss function remains invariant under pre- and postmultiplication of the orthonormal matrix $\mathbf{K}$. Mathar and Meyer (1993) prove that the classical scaling solution is also optimal for the minimization of any orthonormal invariant norm on $\mathbf{E}$. For example, the classical scaling solution is optimal if the loss is defined as $L(\mathbf{X}) = \sum_i |\phi_i|$.

In contrast to the MDS method discussed in Chapter 9, it is difficult to incorporate transformations of the proximities in classical scaling. An algorithm was proposed by Trosset (1993) that optimally transforms the proximities for Strain.

Classical scaling can even be used to study or discover the "intrinsic geometry" of highly nonlinear structures contained in high-dimensional spaces. For example, a point configuration that forms a helix in 3D space is intrinsically one-dimensional in the sense that if you move back and forth on this helix, the distances along the helix are additive. As long as you stay on the helix, Euclidean distances d_{ij} are only approximately correct measures for the length of the path from point i to point j if i and j are close (or, in the case of highly nonlinear structures, "very close") to each other. For points that are far apart, Euclidean distances can grossly underestimate the intrinsic distance of i and j. To study such geometries and to unroll them into low-dimensional Euclidean geometries, Tenenbaum, De Silva, and Langford (2000) first define the radius of a small neighborhood, ϵ, and then set $\delta_{ij} = d_{ij}$ for all ij where $d_{ij} < \epsilon$, and $\delta_{ij} = \infty$ otherwise. Then, in a second cycle, these values are replaced by computing distances over the network of point triples as follows: $\delta_{ij} = \min_k(\delta_{ij}, \delta_{ik} + \delta_{jk})$, for all k. If there are many points that are well spread out, this generates graph

distances that approximate the lengths of the paths within the curved structure. Applying classical scaling to dissimilarities generated in such a way from nonlinear structures allowed Tenenbaum et al. (2000) to unroll these structures successfully.

12.5 Exercises

Exercise 12.1 Use classical scaling on the data in Table 4.1, p. 65. (Note: You first have to transform the similarity data into reasonable dissimilarities.) Compare the solution to the one obtained by ordinal MDS (Figure 4.1).

Exercise 12.2 Use matrix $\mathbf{X}$ computed in Section 12.2, p. 264, to reconstruct both $\mathbf{B}_{\boldsymbol{\Delta}}$ and $\boldsymbol{\Delta}$. Assess how well this $\mathbf{X}$ "explains" $\boldsymbol{\Delta}$.

Exercise 12.3 Take matrix $\boldsymbol{\Delta}$ from Section 12.2. Instead of centering this matrix, choose one of its entries as the element serves as the origin of the MDS space.

(a) Compute $\mathbf{B}_{\boldsymbol{\Delta}}$ relative to this particular origin.

(b) Find the classical scaling representation for this $\mathbf{B}_{\boldsymbol{\Delta}}$.

(c) Compare this solution to the solution $\mathbf{X}$ found in Section 12.2.

13

Special Solutions, Degeneracies, and Local Minima

In this chapter, we explain several technical peculiarities of MDS. First, we discuss degenerate solutions in ordinal MDS, where Stress approaches zero even though the MDS distances do not represent the data properly. Then we consider MDS of a constant dissimilarity matrix (all dissimilarities are equal) and indicate what configurations are found in this case. Another problem in MDS is the existence of multiple local minima solutions. This problem is especially severe for unidimensional scaling. For this case, several strategies are discussed that are less prone to local minima. For full-dimensional scaling, in contrast, it is shown that the majorization algorithm always finds a globally optimal solution. For other dimensionalities, several methods for finding a global minimum exist, for example, the tunneling method and distance smoothing.

13.1 A Degenerate Solution in Ordinal MDS

In the various MDS applications discussed so far in this book, we assumed that the loss function employed to find the MDS configuration $\mathbf{X}$ would actually work in the desired sense. In particular, a low Stress value was interpreted as an index that the given proximities were well represented by the distances of $\mathbf{X}$. But is that always true? In Section 3.2, we noticed, for example, that if one minimizes raw Stress, a trivial solution is possible: if $\mathbf{X}$ is made smaller and smaller over the iterations, raw Stress can be arbitrarily reduced, even though proximities and distances are not systematically

TABLE 13.1. Correlations of some KIPT subtests of Guthrie (1973). The lower triangular elements contain the correlations, the upper triangular the rank-order of the correlations in decreasing order.

Subtest	NP	LVP	SVP	CCP	NR	SLP	CCR	ILR
Nonsense word production (NP)	-	9	4	1	6	19	10	12
Long vowel production (LVP)	.78	-	1	7	5	21	20	22
Short vowel production (SVP)	.87	.94	-	3	2	17	16	23
Consonant cluster production (CCP)	.94	.83	.90	-	7	14	11	16
Nonsense word recognition (NR)	.84	.85	.91	.83	-	17	15	18
Single letter production (SLP)	.53	.47	.56	.60	.56	-	13	16
Consonant cluster recognition(CCR)	.72	.48	.57	.69	.59	.62	-	8
Initial letter recognition (ILR)	.66	.45	.44	.57	.55	.57	.82	-

related. Therefore, one has to avoid this outcome (e.g., by using normalized Stress as a loss criterion), because it may—and usually does—lead to a pseudo solution that does not represent the data in the desired sense.

MDS configurations where the loss criterion can be made arbitrarily small irrespective of the relationship of data and distances are called *degenerate* solutions of the particular loss function. They can be avoided, in general, by imposing additional constraints onto the loss function. One example was shown above for raw Stress, where the constraint is a normalization of raw Stress or the requirement that $\mathbf{X}$ must not shrink.

In ordinal MDS, there exist further degenerate solutions, even when using normalized Stress. These solutions arise for particular data. Consider an example. Table 13.1 presents a matrix of correlation coefficients on eight subtests of the Kennedy Institute Phonics Test (KIPT), a reading skills test (Guthrie, 1973). If we scale these data by ordinal MDS in a plane, we obtain the configuration shown in Figure 13.1a. There are just three groups of points. One contains the subtests NP, LVP, SVP, CCP, and NR in a very tight cluster in the upper right-hand corner; a second contains CCR and ILR, also very close together in the upper left-hand corner; finally, point SLP is clearly separated from both of these clusters, with essentially the same distance to either one of them. We find, furthermore, that the MDS solution appears to be almost perfect, because its Stress value is practically equal to zero (i.e., smaller than the stopping criterion for the MDS algorithm).

The Shepard diagram in Figure 13.1b reveals, however, some peculiarities. The data, which are relatively evenly distributed over an interval from $r = .44$ to $r = .94$ (see Table 13.1), are not represented by distances with a similar distribution but rather by two clearly distinct classes of distances. In fact, the MDS procedure maps all correlations $r \geq .78$ into almost the same small distance, and all correlations $r < .72$ into almost the same large distance. Even though the resulting step function is perfectly admissible within ordinal MDS, we would probably be reluctant to accept it as a

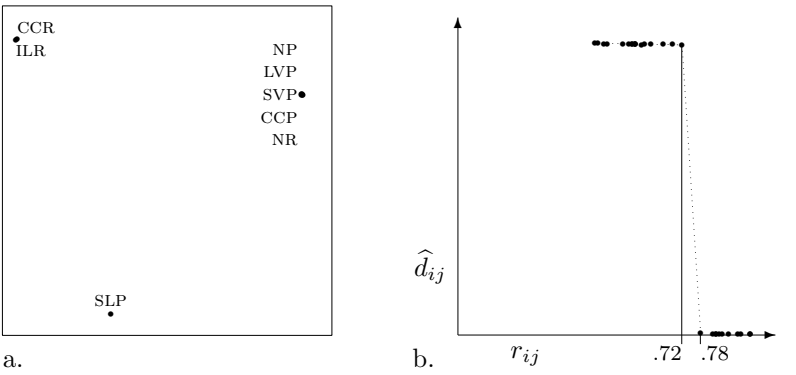


FIGURE 13.1. Ordinal MDS solution (a) and Shepard diagram (b) for correlations in Table 13.1.

sensible transformation of the empirical data, because the transformation simply dichotomizes our data. Using ordinal MDS does not mean that we are indifferent to which monotonic function is chosen as being optimal for the procedure. For our correlations in Table 13.1, it appears reasonable to assume that their differences are also meaningful to some extent, even though their order relations may be more reliable. Hence, we should insist that the correlations be mapped into distances by a more smoothly increasing monotone function. The regression line in the Shepard diagram could then be approximated by a parametric function, for example, a power function or a monotone spline. However, the exact type of the regression function is not known a priori. Otherwise, we would simply choose it and specify a metric MDS model.

On closer analysis, one finds that the solution in Figure 13.1 does not only have an odd transformation function, but it also possesses a peculiar relationship to the data. We can see this as follows. Table 13.1 has been arranged so that the subtests are lumped together into three blocks, where one cluster consists of only one element, SLP. This reveals that: (1) the five subtests in the block {NP, ... , NR} correlate higher with each other than with any subtest in the other blocks, CCR, ILR, or SLP; the lowest *within-block* correlation is r(NR,LVP) = .78, but the highest correlation with any other subtest is r(NR,CCR) = .72; (2) for the block {CCR, ILR}, the within-block correlation is r(CCR, ILR) = .82, which is higher than any of the *between-block* correlations; (3) the same holds trivially for the block {SLP}, where r(SLP, SLP) = 1.00. Because all correlations $r \geq .78$ are mapped into (almost) the same very small distance and all $r < .78$ into (almost) the same much larger distance, the MDS procedure shrinks all within-block distances to almost zero and makes all between-block distances almost equally large. This represents a formal solution to

TABLE 13.2. The rank-order of KIPT subtests in the upper half, the optimal disparities in ordinal MDS in the lower half.

Subtest	NP	LVP	SVP	CCP	NR	SLP	CCR	ILR
NP	-	9	4	1	6	19	10	12
LVP	0	-	1	7	5	21	20	22
SVP	0	0	-	3	2	17	16	23
CCP	0	0	0	-	7	14	11	16
NR	0	0	0	0	-	17	15	18
SLP	1	1	1	1	1	-	13	16
CCR	1	1	1	1	1	1	-	8
ILR	1	1	1	1	1	1	0	-

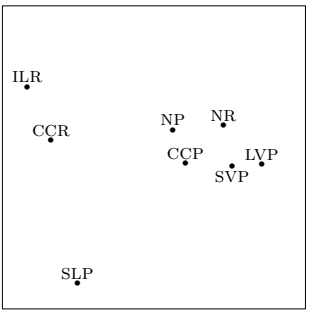
the MDS problem, because it reduces the loss function to a very small value indeed—whether or not the within-block and the between-block distances, respectively, are ordered as the data are! The only aspect of the data that is properly represented, therefore, is that between-block distances are all larger than within-block distances.

It is not difficult to see why Stress is so small in the example above. The lower half of Table 13.2 shows the optimal disparities. One notes that as long as the ranking number in the upper half of the matrix is 9 or smaller, the disparities are all zero, and for rank-order 10 or larger, the disparities are all one. These disparities perfectly match the rank-order information of the data. Ordinal MDS assigns the subtests to three clusters. The within-cluster disparities are zero, so that all points within the cluster have the same coordinates and thus zero distance. Between the cluster points, the distances should be one.

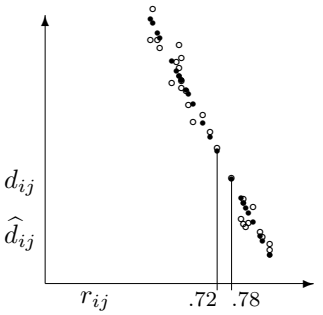
This type of degeneracy can be expected with ordinal MDS when the dimensionality is high compared to the number of objects. It all depends, though, on how many within-blocks of zero exist. In our example, we have three blocks of zero disparities (counting SLP as one cluster). With four within-blocks of zeros, one obtains four clusters for which a perfect solution exists in three dimensions, and so on. The only information that this ordinal MDS solution correctly represents is the partitioning of items in clusters.

13.2 Avoiding Degenerate Solutions

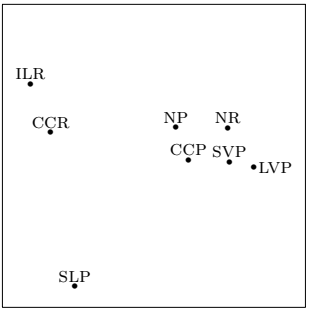
The general solution to degeneracy is to impose stronger restrictions onto the function that maps data into distances. In many instances, a degenerate solution occurs because there are not enough constraints to avoid it. In Table 9.1, we ordered the transformations from strong to weak. Because an ordinal transformation is the weakest possible form of transformation, we can choose any of the stronger transformations as an alternative. We have applied two stronger transformations to the data in Table 13.1, a



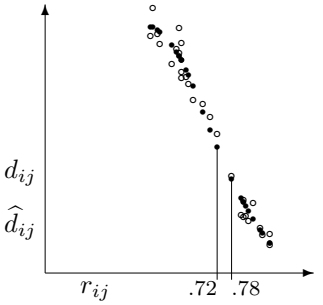
a. linear MDS



b. Shepard plot (linear MDS)



c. MDS with spline



d. Shepard plot (splines)

FIGURE 13.2. Solution of linear MDS with intercept (a) on correlations ($\sigma_n = .0065$) in Table 13.1 and the transformation plot (b), and the solution of MDS with monotonic spline (one interior knot, of order 2, $\sigma_n = .0054$).

linear transformation (with intercept) and a spline transformation (with one interior knot and order 2). The results are in Figure 13.2. Both MDS solutions fit well (interval MDS $\sigma_n = .0065$, monotone spline $\sigma_n = .0054$), and both, of course, map the correlations *smoothly* into distances.

Interval scaling of the data is not the only possibility for arriving at a reasonable MDS configuration when the data possess the peculiar block pattern discussed above. Indeed, any kind of metric representation of the data prevents degenerate solutions. The transformation could also be defined, for example, by $\widehat{d}_{ij} = a + b \cdot \exp(\delta_{ij})$. Depending on the context, such a model may be more attractive a priori, because it specifies a theory about the relation of data and distances that is more precise than to admit just *any* monotone mapping.

13.3 Special Solutions: Almost Equal Dissimilarities

An interesting special case of MDS is concerned with equal dissimilarities. By the *constant dissimilarity case* we mean that $\delta_{ij} = c$, for all i, j, with $c > 0$.[1] We may regard these data as null-data: the differences between all pairs of objects are the same.[2] If we do a ratio MDS on these dissimilarities, the solution has a particular pattern. Consider a simple example of a 3×3 dissimilarity matrix with all dissimilarities equal to 1. An MDS solution with $\sigma_n = 0$ in two dimensions is obtained by placing the points on the corners of an equilateral triangle. It is not hard to extend this result to a solution of a 4×4 constant dissimilarity matrix in three dimensions, where a perfect solution consists of the corner points of a regular tetrahedron (a three-sided pyramid, all sides of equal length). Such a figure is called a *simplex*.[3] The perfect solution for a general $n \times n$ constant dissimilarity matrix is a simplex in $n - 1$ dimensions.

But what happens in lower dimensionality? The optimal MDS solution for constant dissimilarities in one dimension consists of points equally spread on a line. In two dimensions, the points lie on concentric circles (De Leeuw & Stoop, 1984). In three dimensions (or higher), the points lie equally spaced on the surface of a sphere (Buja, Logan, Reeds, & Shepp, 1994). Any permutation of these points gives an equally good fit. Examples of these solutions are shown in Figure 13.3.

For ordinal MDS, we allowed that $p_{ij} \leq p_{kl}$ can be admissibly transformed by a weak monotone function into $\widehat{d}_{ij} = \widehat{d}_{kl}$. Yet, this means that if we choose *all* disparities equal, then the disparities satisfy any rank-order of the proximities, and equal disparities, in turn, ask for an MDS configuration with equal distances. Generally, though, monotone regression should find disparities with a stronger relation to the order of the data (see Section 9.2 and Table 9.4 for an example). However, the equal-disparities scenario can be used to compute a particular upper bound for Stress values in ordinal MDS. Such bounds were determined as follows. We entered a matrix of constant dissimilarities into an MDS program and let the program determine the local minimum Stress. This was done for a range of different ns in one to six dimensions. The Stress values are given in Table 13.3 and can be

---

[1] The dissimilarities do not have to be exactly equal; they may also be approximately equal; that is, $c - \epsilon \leq \delta_{ij} \leq c + \epsilon$ for some small ϵ $(0 \leq \epsilon \leq c)$.

[2] We may regard the constant dissimilarity case as one variety of a *formal* null hypothesis. Another, more common, form of such a null hypothesis is the assumption that the dissimilarities are "random" (see Chapter 3). A *substantively* motivated null hypothesis, in contrast, is derived from the incumbent theory on the domain of interest, whereas the alternative hypothesis relates to the challenging theory.

[3] Note that this simplex (of points) is not equivalent to the simplex of ordered regions discussed in Chapter 5.

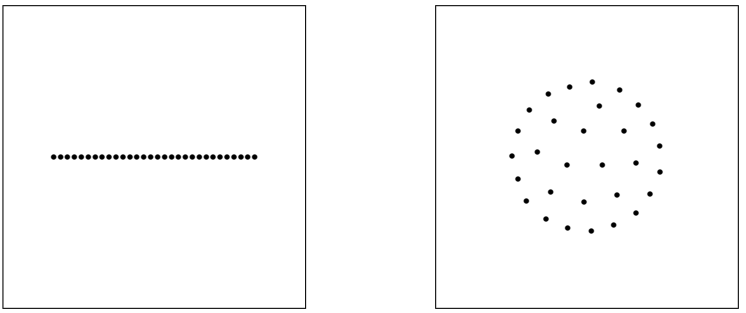


FIGURE 13.3. Solutions for constant dissimilarities with $n = 30$. The left plot shows the unidimensional solution and the right plot a 2D solution.

used as a reference. For example, the Stress found for ordinal MDS on the Morse code data in Chapter 4 was .18 ($n = 36$, 2D). In Table 13.3, we find that for $n = 35$ in 2D the worst expected Stress under the equal-disparity scenario is .3957. Thus, from the Stress value alone, we can safely assume that the 2D solution of the Morse code data shows more structure than the constant dissimilarities case, which was verified by the interpretation.

De Leeuw and Stoop (1984) proved, using theoretical arguments, that for unidimensional scaling Stress could never be larger than $[(n-2)/3n]^{1/2}$, which for large n becomes $1/\sqrt{3} = .5774$. In 2D, they derive the upper bound of Stress by assuming that the points lie equally spaced on a circle (which need not be the optimal solution for constant dissimilarities; see, e.g., the panel on the right-hand side of Figure 13.3). Then, Stress is smaller than $[1 - 2\cot^2(\pi/2n)/(n^2 - n)]^{1/2}$, with the limit $[1 - 8/\pi^2]^{1/2} = .4352$ for large n.

The all-disparities-being-equal degenerate solution seems uncommon in practice. In any case, if it occurs it can be most easily detected by checking the Shepard diagram for numerically highly similar dissimilarities or d-hats. For example, if ratio MDS is used on dissimilarities that fall into the interval [.85, .95] and, thus, have quite similar ratios, a solution is found that is close to the one obtained for constant dissimilarities. Thus, the strong ratio MDS model is not always optimal for showing the data structure. Rather, in such a case we advise redoing the analysis with interval MDS or by using monotone splines. The intercept estimates the constant part of the dissimilarities, and the varying part of the dissimilarities is shown by the MDS configuration. In other words: if the Shepard diagram shows signs of constant dissimilarities or d-hats, the MDS user's strategy should not consist in mechanically choosing a stronger transformation, but rather one that has at least an intercept.

TABLE 13.3. Upper bound values of Stress for ordinal MDS based on MDS of constant dissimilarities.

n	1D	2D	3D	4D	5D	6D
2	.0000	.0000	.0000	.0000	.0000	.0000
3	.3333	.0000	.0000	.0000	.0000	.0000
4	.4083	.1691	.0000	.0000	.0000	.0000
5	.4472	.2598	.1277	.0000	.0000	.0000
6	.4714	.2674	.1513	.1005	.0000	.0000
7	.4880	.2933	.1838	.1265	.0843	.0000
8	.5000	.3084	.2027	.1356	.1091	.0728
9	.5092	.3209	.2145	.1568	.1192	.0949
10	.5164	.3315	.2280	.1688	.1237	.1072
12	.5271	.3473	.2423	.1847	.1473	.1140
14	.5345	.3579	.2555	.1977	.1612	.1334
16	.5401	.3658	.2648	.2069	.1691	.1442
18	.5443	.3719	.2718	.2145	.1780	.1520
20	.5477	.3767	.2777	.2200	.1838	.1572
25	.5538	.3855	.2883	.2311	.1949	.1694
30	.5578	.3914	.2955	.2387	.2022	.1766
35	.5606	.3957	.3007	.2439	.2078	.1822
40	.5628	.3987	.3045	.2480	.2121	.1868
45	.5644	.4012	.3076	.2512	.2154	.1900
50	.5657	.4032	.3100	.2538	.2179	.1926

13.4 Local Minima

MDS algorithms usually end up in a *local minimum*. This property guarantees that any small change of the configuration leads to a higher Stress. In contrast, for a *global minimum* MDS configuration, there is no other configuration with lower Stress. A simplified view of the Stress function is shown in Figure 13.4 for an MDS analysis with two local minima, $\mathbf{X}^*$ and $\mathbf{X}^{**}$, where $\mathbf{X}^{**}$ is a global minimum. The solution found by MDS algorithms is sometimes a global minimum, sometimes only a local minimum.[4] Note that more than one global minimum configuration may exist. Those configurations all have the same global minimum Stress, although the configurations are different (even when the freedom of rotation, translation, and reflection are taken into account). For this reason, we refer to *a* global minimum instead of *the* global minimum.

There are differences between the various MDS algorithms in the effectiveness of locating a global minimum. We limit our discussion of local minima to absolute MDS because for this MDS model the local minimum problem is complicated enough. The local minimum problem can be worse

[4]Local minima in MDS are not necessarily bad. A configuration with a slightly worse fit is acceptable if it has a clearer interpretation than a configuration with a better fit (see also Chapter 10).

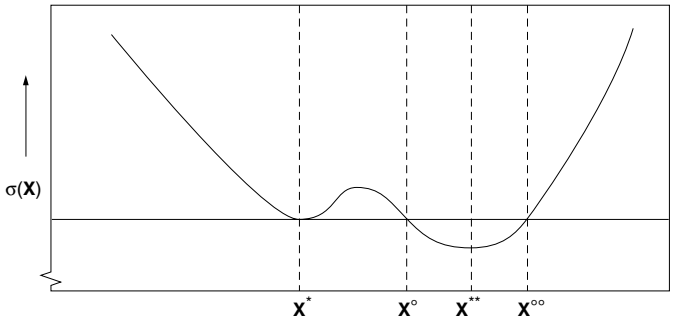


FIGURE 13.4. Example of local minima of a simplified Stress function $\sigma_r(\mathbf{X})$. $\mathbf{X}^*$ is a local minimum, whereas $\mathbf{X}^{**}$ is also a global minimum. $\mathbf{X}^\circ$ and $\mathbf{X}^{\circ\circ}$ have Stress $\sigma_r(\mathbf{X}^*)$.

for nonmetric MDS, or, in the case of nonmetric unidimensional scaling, be less severe.

A simulation study of Groenen and Heiser (1996) showed that local minima are more likely to occur in low dimensionality and hardly occur or are absent in high dimensionality. Below, two special cases are discussed, unidimensional scaling and full-dimensional scaling, for which theoretical results exist concerning local and global optima.

The start configuration of the searching process is of crucial importance for the determination of the final minimum. A random configuration $\mathbf{X}$ is most likely not ideal for finding the lowest-Stress solution by the gradient method, because it does not pay any attention to the data. Therefore, all modern MDS programs use, by default, a *rational* starting configuration derived by some variant of the metric methods discussed in Chapter 12, usually the classical scaling solution of Torgerson (1958) and Gower (1966). Naturally, rationality in the above sense does not guarantee that the starting configuration is best for the particular purpose of an MDS analysis; we may therefore sometimes choose to construct a starting configuration according to given substantive expectations.

Several different methods exist for finding the global minimum. The *method of dimension reduction* repeats the MDS analysis, starting from a high dimensionality (say, 10) and then reducing the dimensionality of the solution space stepwise (down to 2, say). The local minimum configuration of the higher-dimensional analysis is used as a start configuration for the MDS analysis in one dimension lower by dropping the dimension that accounts for the least variance (i.e., the last principal component). Proceeding in this manner, one hopes that the low-dimensional solution is a global minimum.

A different method, called *multiple random starts*, or *multistart*, consists of running the MDS analysis from many (say, 100) different random starting configurations and choosing the one with the lowest Stress. Using multistart

and making some mild assumptions (see Boender, 1984), an estimate for the expected total number of local minima can be given. Let n_s be the number of multistart start configurations and n_m the number of different local minima obtained. Then, the total expected number of local minima n_t is

$$n_t = \frac{n_m(n_s - 1)}{n_s - n_m - 2}. \tag{13.1}$$

If n_s is approximately equal to n_t, then we may assume that all local minima are found. The one with the lowest Stress is the candidate global minimum. Multistart usually gives satisfactory results but is computationally intensive.

Yet another approach is the tunneling method, discussed in Section 13.7. For an overview of other global minimization methods, we refer to Groenen (1993). For a comparison of various global optimization methods on a large empirical data set, see Groenen, Mathar, and Trejos (2000).

13.5 Unidimensional Scaling

It has been noted by De Leeuw and Heiser (1977), Defays (1978), Hubert and Arabie (1986), and Pliner (1996) that minimizing the Stress function with equal weights changes to a combinatorial problem when $m = 1$. It turns out that Stress has many local minima. Therefore, when doing (absolute) MDS in one dimension, one *always* has to be concerned about the local minimum problem. If, however, transformations of the proximities are allowed, then the local minimum problem in unidimensional scaling may be less severe. What follows is a technical discussion of the local minimum problem in unidimensional absolute MDS.

Unidimensional Scaling: A Combinatorial Problem

Inasmuch we are dealing with one dimension, the matrix of coordinates $\mathbf{X}$ has one column and is presented by the $n \times 1$ column vector $\mathbf{x}$ in this section. The distance between two points in one dimension is equal to $d_{ij}(\mathbf{x}) = |x_i - x_j|$. This can be expressed as $d_{ij}(\mathbf{x}) = (x_i - x_j)\text{sign}(x_i - x_j)$, where $\text{sign}(x_i - x_j) = 1$ for $x_i > x_j$, $\text{sign}(x_i - x_j) = 0$ for $x_i = x_j$, and $\text{sign}(x_i - x_j) = -1$ for $x_i < x_j$. An important observation is that only the rank-order of $\mathbf{x}$ determines the $\text{sign}(x_i - x_j)$. In this case, Stress can be expressed as

$$\begin{aligned} \sigma_r(\mathbf{x}) &= \eta_\delta^2 + \eta^2(\mathbf{x}) - 2\rho(\mathbf{x}) \\ &= \sum_{i<j} w_{ij}\delta_{ij}^2 + \sum_{i<j} w_{ij}(x_i - x_j)^2 - 2\sum_{i<j} w_{ij}\delta_{ij}|x_i - x_j| \end{aligned}$$

$$= \eta_\delta^2 + \mathbf{x}'\mathbf{V}\mathbf{x} - 2\sum_{i<j} w_{ij}\delta_{ij}(x_i - x_j)\text{sign}(x_i - x_j). \qquad (13.2)$$

This shows that the cross-product term of Stress, $\rho(\mathbf{x})$, can be factored into a term that is linear in $\mathbf{x}$ and a term that depends only on the rank-order of the elements of $\mathbf{x}$. Therefore, $\rho(\mathbf{x})$ is a piecewise linear function, its pieces being linear within each rank-order of $\mathbf{x}$. For each rank-order, the Stress is consequently quadratic in $\mathbf{x}$. This suggests that the unidimensional scaling problem can be solved by minimizing Stress over all permutations, a *combinatorial* problem. We show that at a local optimum of a function that is only dependent on the rank-order of $\mathbf{x}$, the Guttman transform yields an $\mathbf{x}$ that has the same rank-order. For that rank-order, Stress has a local minimum.

Let ψ denote the rank-order of the vector $\mathbf{x}$, such that $x_{\psi(1)}$ denotes the smallest element of $\mathbf{x}$, and $x_{\psi(i)}$ the element of $\mathbf{x}$ with rank i, so that $x_{\psi(1)} \leq x_{\psi(2)} \leq \cdots \leq x_{\psi(i)} \leq \cdots \leq x_{\psi(n)}$. Let $\mathbf{R}$ be the corresponding permutation matrix, so that $\mathbf{R}\mathbf{x}$ is the vector with the elements ordered nondecreasingly. Define $l_i = \sum_{j<i} w_{\psi(i)\psi(j)}\delta_{\psi(i)\psi(j)}$ and $u_i = \sum_{j>i} w_{\psi(i)\psi(j)}\delta_{\psi(i)\psi(j)}$, which are, respectively, the row sum up to the main diagonal and the row sum from the main diagonal of the matrix with values $w_{\psi(i)\psi(j)}\delta_{\psi(i)\psi(j)}$. Using this notation, (13.2) can be written as

$$\sigma_r(\mathbf{x}) = \eta_\delta^2 + \mathbf{x}'\mathbf{V}\mathbf{x} - 2\mathbf{x}'\mathbf{R}'(\mathbf{l} - \mathbf{u}). \qquad (13.3)$$

For a given rank-order ψ, (13.3) is quadratic in $\mathbf{x}$ and has its minimum when $\mathbf{x}$ is equal to the Guttman transform $\mathbf{V}^+\mathbf{R}'(\mathbf{l}-\mathbf{u})$. The Guttman transform of the majorization approach only uses the rank-order information of the previous configuration, because $\mathbf{R}$, $\mathbf{l}$, and $\mathbf{u}$ only depend on the permutation of $\mathbf{x}$. Therefore, the majorizing algorithm stops if the rank-order of $\mathbf{x}$ does not change, which usually happens in a few iterations. At this point, Stress has a local minimum. Function (13.3) can also be expressed as

$$\sigma_r(\mathbf{x}) = \eta_\delta^2 + \|\mathbf{x} - \mathbf{V}^+\mathbf{R}'(\mathbf{l} - \mathbf{u})\|_{\mathbf{V}}^2 - \|\mathbf{l} - \mathbf{u}\|_{\mathbf{R}\mathbf{V}^+\mathbf{R}'}^2, \qquad (13.4)$$

where the term $t(\psi) = \|\mathbf{l}-\mathbf{u}\|_{\mathbf{R}\mathbf{V}^+\mathbf{R}'}^2$ is a function of the permutation only. Thus, if $t(\psi)$ is maximized, the second term of (13.4) vanishes if $\mathbf{x}$ is chosen equal to the Guttman transform $\mathbf{V}^+\mathbf{R}'(\mathbf{l} - \mathbf{u})$.

Defays (1978) minimizes (13.4) by maximizing $t(\psi)$. Suppose that we have found a permutation ψ that is locally optimal with respect to adjacent pairwise interchanges. That is, any local change of ψ, interchanging $\psi_{(i)}$ and $\psi_{(i+1)}$, does not increase the value of $t(\psi)$. We say that $t(\psi)$ has a local maximum if permutation ψ satisfies this condition. Note that this is a stronger formulation for a local minimum than we used for Stress, because Stress has a local minimum whenever the Guttman transform cannot change the order of $\mathbf{x}$. Groenen (1993) proves that, even for nonconstant w_{ij}, Stress has a local minimum whenever $t(\psi)$ has a local maximum. Suppose that we know how to find a ψ that makes $t(\psi)$ attain the highest

possible value. Then ψ defines the order of $\mathbf{x}$ for a *global* minimum of Stress.

Pliner (1996) gives a $100(1-\alpha)\%$ confidence interval for the number of local minima in unidimensional scaling. Let n_s be the number of (random) sample configurations ψ and n_m be the number of those permutations for which $\sigma_r(\mathbf{x})$ is a local minimum. Then, the confidence interval is given by

$$\left[n! \frac{n_m}{n_m + (n_s - n_m + 1) X_F(2(n_s - n_m + 1), 2n_m)}, \right.$$
$$\left. n! \frac{(n_m + 1) X_F(2(n_m + 1), 2(n_s - n_m))}{(n_s - n_m) + (n_m + 1) X_F(2(n_m + 1), 2(n_s - n_m))} \right],$$

where $X_F(\nu_1, \nu_2)$ is the critical point of an F distribution with (ν_1, ν_2) degrees of freedom such that the probability equals $\alpha/2$ for a similarly distributed t to have t larger or equal to the critical point. Pliner showed that for an 8×8 example (13.5) gave the exact number of local minima of 12770. For another (random data) example, he obtained a 95% confidence interval of $[2.6 \cdot 10^9, 3.4 \cdot 10^9]$ for the number of local minima.

Some Algorithms for Unidimensional Scaling

A whole variety of combinatorial optimization strategies is available for maximizing $t(\psi)$ over ψ. One obvious strategy is simply to try all different orders ψ of n objects, and choose the one for which $t(\psi)$ is maximal. This strategy of *complete search* guarantees a global maximum of $t(\psi)$ and thus a global minimum of Stress. However, because there are $n!$ different permutations, a complete search becomes impractical for $n \geq 10$. Other, more efficient strategies are available. For equal weights, the strategy of *dynamic programming* of Hubert and Golledge (1981) and Hubert and Arabie (1986) is very efficient for moderate n. Their strategy reduces the order of computation from $n!$ to 2^n while still finding a globally optimal solution. Groenen (1993) extended their approach to the case of nonidentical weights but loses the guarantee of reaching a global optimum and some of the computational efficiency. The strategy of *local pairwise interchange* (LOPI) does not guarantee global optimality, but it is very efficient and yields good results. LOPI strategies amount to choosing a pair of objects, interchanging them, and evaluating $t(\psi)$ for the changed rank-order. If $t(\psi)$ is higher than any rank-order we have found so far, then we accept the pairwise interchange. The search is stopped if the pairwise interchanges do not yield a higher $t(\psi)$. The resulting ψ defines a local minimum of Stress. The various implementations of the LOPI strategy result in better local minima of Stress compared to applying the SMACOF algorithm. In a simulation study of Groenen (1993), the LOPI strategies found a global maximum of $t(\psi)$ in the majority of the cases. Poole (1984, 1990) obtained good results in locating the global optimum for unidimensional unfolding. De Soete, Hubert,

and Arabie (1988) found that LOPI performed better than an alternative method called *simulated annealing*. Brusco (2001) studied the use of another implementation of simulated annealing in unidimensional scaling and reported that often a good candidate global minimum was found. Brusco and Stahl (2000) focused on good initial configurations for unidimensional scaling. They proposed to use the results of a related quadratic assignment problem as a start for unidimensional scaling. Their study showed that such an approach can indeed provide effective and efficient initial solutions for large-scale unidimensional scaling problems. A review of unidimensional scaling algorithms minimizing the sum of absolute errors instead of the usual squared errors can be found in Brusco (2002).

Instead of a combinatorial technique, Pliner (1996) used a *smoothing approach* to the local minimum problem in unidimensional scaling. The Stress function is replaced by the function

$$\sigma_\epsilon(\mathbf{X}) = \sum_{i<j} \delta_{ij}^2 + \sum_{i<j} (x_i - x_j)^2 - 2 \sum_{i<j} \delta_{ij} g_\epsilon(x_i - x_j) \text{ with} \tag{13.5}$$

$$g_\epsilon(t) = \begin{cases} t^2(3\epsilon - |t|)/3\epsilon^2 + \epsilon/3, & \text{if } |t| < \epsilon \\ |t|, & \text{if } |t| \geq \epsilon, \end{cases} \tag{13.6}$$

which smooths $-d_{ij}(\mathbf{x})$. The only difference of (13.5) with Stress is that for small distances ($d_{ij} < \epsilon$) the distance in the last term of (13.5) is replaced by a smooth function. Figure 13.5 shows how $g_\epsilon(x_i - x_j)$ smooths $d_{ij}(\mathbf{x}) = |x_i - x_j|$. Pliner recommends starting with the value $\epsilon = 2 \max_{1 \leq i \leq n} n^{-1} \sum_{j=1}^{n} \delta_{ij}$, minimizing $\sigma_\epsilon(\mathbf{X})$ over $\mathbf{X}$, and using the minimizer as a starting configuration for minimizing $\sigma_\epsilon(\mathbf{X})$ again, but with a smaller value of ϵ. This procedure is repeated until ϵ is very small. If we assume that all distances are greater than 0, then there exists an ϵ for which $\sigma_\epsilon(\mathbf{X})$ reduces to raw Stress. Because $-g_\epsilon(t)$ is a concave function in t, it can be linearly majorized, so that a convergent algorithm can be obtained [as proved by Pliner (1996) using a different argumentation]. More important, the smoothing algorithm turns out to yield global minima solutions very often. Numerical experiments of Pliner suggest that in at least 60% (sometimes even 100%) of the runs, a global minimum was found, which makes this smoothing strategy an important aid for finding the global minimum in unidimensional scaling. Section 13.8 discusses an extension of this smoothing strategy to higher dimensionality.

13.6 Full-Dimensional Scaling

To better understand the local minimum problem for Stress, we consider full-dimensional scaling (absolute MDS), where the dimensionality is $m = n - 1$. In full-dimensional scaling, there is only one minimum, a global one (De Leeuw, 1993). This can be seen as follows. Consider the matrix of

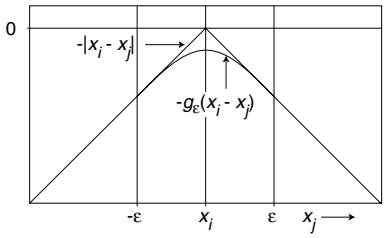


FIGURE 13.5. The function $-d_{ij}(\mathbf{x})$ and the smoothed version $-g_\epsilon(\mathbf{x})$ in (13.5) used by Pliner (1996).

squared distances $\mathbf{D}^{(2)}(\mathbf{X}) = \mathbf{1c}' + \mathbf{c1}' - 2\mathbf{XX}'$, with $\mathbf{X}$ being column centered and where $\mathbf{c}$ contains the diagonal elements of $\mathbf{XX}'$ (see also Section 7.3). Thus, the rank of $\mathbf{XX}'$ can never exceed $n-1$. For $m = n-1$, the cross-product term $\mathbf{XX}'$ is simply a double-centered positive semidefinite (p.s.d.) matrix $\mathbf{B}$, so that the squared distances are equal to $b_{ii} + b_{jj} - 2b_{ij}$. It can be verified that the set of p.s.d. matrices is convex, because for $\mathbf{B}_1, \mathbf{B}_2$ p.s.d. and $0 \leq \alpha \leq 1$, $\alpha\mathbf{B}_1 + (1-\alpha)\mathbf{B}_2$ is p.s.d., too. This allows us to express Stress as

$$\begin{aligned} \sigma_r(\mathbf{B}) &= \sum_{i<j} w_{ij}\delta_{ij}^2 + \sum_{i<j} w_{ij}(b_{ii} + b_{jj} - 2b_{ij}) \\ &\quad -2\sum_{i<j} w_{ij}\delta_{ij}(b_{ii} + b_{jj} - 2b_{ij})^{1/2}. \end{aligned} \tag{13.7}$$

The first term of (13.7) does not depend on $\mathbf{B}$, and the second term is a linear function of $\mathbf{B}$. The third term is minus the square root of the same linear function of $\mathbf{B}$, which is also a convex function in $\mathbf{B}$. It may be verified that the sum of a linear and a convex function is convex, so that $\sigma_r(\mathbf{B})$ is a convex function in $\mathbf{B}$. Thus, minimizing Stress over $\mathbf{B}$ is minimizing a convex function over a convex set, which has a local minimum that is a global minimum. Note that this result does not hold in the case where $\mathbf{B}$ is restricted to have $m < n-1$, because the set of $\mathbf{B}$s restricted to have rank $m < n-1$ is not convex.

Although one would expect $\mathbf{B}$ to be of rank $n-1$ at a minimum, this usually is not the case. In fact, numerical experiments suggest that at a minimum, the rank of $\mathbf{B}$ does not exceed the number of positive eigenvalues in classical scaling. Critchley (1986) and Bailey and Gower (1990) proved this conjecture for S-Stress, but no proof exists for Stress. This result implies that an MDS analysis (with or without transformations) in dimensionality $n-1$ usually ends with a solution of lower rank. De Leeuw and Groenen (1997) prove that at a minimum $\mathbf{B}$ has rank $n-1$ only in the case of a perfect representation of Stress zero with $\mathbf{\Delta}$ a Euclidean distance matrix. The converse is also true: at a minimum with nonzero Stress, $\mathbf{B}$ has rank $n-2$ or smaller.

In confirmatory MDS, the linear constraint $\mathbf{X} = \mathbf{YC}$ is used quite often (see Chapter 10). If, without loss of generality, $\mathbf{Y}$ has $r < n$ columns and is of full rank r, and the dimensionality m of $\mathbf{X}$ equals r, then confirmatory MDS with linear constraints has one minimum, which is global. The same reasoning as above can be used to verify this statement, with the additional constraint that $\mathbf{B} = \mathbf{YCC}'\mathbf{Y}'$, which is also convex if $\mathbf{C}$ is square. In the extreme case where $\mathbf{Y}$ has only one column, $\mathbf{C}$ becomes a scalar, for which the global minimum solution was given in Section 11.1 by b^*.

13.7 The Tunneling Method for Avoiding Local Minima

The problem of local minima is not limited to MDS but is also quite common in numerical optimization. There are many methods for finding a configuration that is not only locally optimal but also has the overall best minimum. One of these methods, called the *tunneling method*, was made suitable for MDS by Groenen and Heiser (1991), Groenen (1993), and Groenen and Heiser (1996). The basic idea of the tunneling method can be described by the following analogy. Suppose that our objective is to find the lowest spot in a mountainous area. First, we try to find the lowest spot in a small area by pouring water and following the water until it forms a small pool. Then, we start drilling a tunnel horizontally. If the tunnel gets out of the mountain, then we are sure that the water flows to a spot that is lower (or remains at the same height). Repeating these steps leads us eventually to the global minimum.

The same idea can be applied for finding the global minimum of the Stress function. Then, the tunneling method alternates over the following two steps.

- Find a local minimum $\mathbf{X}^*$ of the Stress function.
- Find another configuration that has the same Stress as $\mathbf{X}^*$.

The second step is the crux of the method and is called the tunneling step. It is performed by minimizing the *tunneling function* $\tau(\mathbf{X})$. Suppose that the Stress function to be minimized is the one graphed in Figure 13.4. For this Stress function, the tunneling function $\tau(\mathbf{X})$ is shown in Figure 13.6. Near the local minimum $\mathbf{X}^*$, the tunneling function $\tau(\mathbf{X})$ has a *pole* (peak) to avoid finding $\mathbf{X}^*$ as a solution of the tunneling step. Furthermore, $\tau(\mathbf{X})$ becomes zero at $\mathbf{X}^\circ$ and $\mathbf{X}^{\circ\circ}$, which are exactly those points in Figure 13.4 that have the same Stress as $\mathbf{X}^*$. Thus, finding the minimum of $\tau(\mathbf{X})$ gives the solution of the second step of the tunneling method. The precise

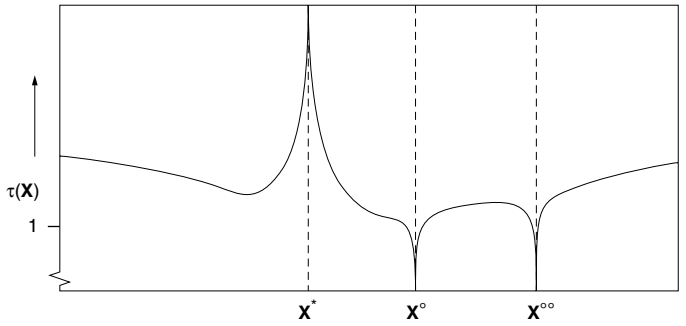


FIGURE 13.6. The tunneling function $\tau(\mathbf{X})$. $\mathbf{X}^*$ is a local minimum of Stress (see also Figure 13.4), $\mathbf{X}^\circ$ and $\mathbf{X}^{\circ\circ}$ are configurations with the same Stress as $\mathbf{X}^*$.

definition of the tunneling function is

$$\tau(\mathbf{X}) = |\sigma_r(\mathbf{X}) - \sigma_r(\mathbf{X}^*)|^\lambda \left(1 + \frac{\omega}{\sum_{i<j} w_{ij}[d_{ij}(\mathbf{X}^*) - d_{ij}(\mathbf{X})]^2}\right). \quad (13.8)$$

Here λ is the *pole strength* parameter that determines how steep the peak is near the local minimum $\mathbf{X}^*$. The *pole width* parameter ω determines the width of activity of the pole. Groenen and Heiser (1996) suggest that $\lambda \leq 1/3$ and $\omega \approx n/2$ are needed to have an effective pole, although the latter seems to depend much on the particular data set.

The effectiveness of the tunneling method is determined by the success of the tunneling step. Clearly, if we start the tunneling step from the global minimum $\mathbf{X}^*$, then $\tau(\mathbf{X})$ cannot become zero (assuming that there is no other global minimum with the same global minimum Stress). Therefore, at some point the tunneling step must be stopped. However, if the tunneling step is stopped too early, then the global minimum can be missed. Experiments of Groenen and Heiser (1996) showed that the tunneling method is able to find the global minimum systematically. However, for some combinations of λ and ω and for certain data sets, the tunneling method fails.

For more details about the tunneling method and the iterative majorization algorithm used for minimizing $\tau(\mathbf{X})$, we refer to Groenen (1993) or Groenen and Heiser (1996). The latter also contains an extension of the tunneling method with Minkowski distances.

13.8 Distance Smoothing for Avoiding Local Minima

In Section 13.5, we discussed the idea of Pliner (1996) to avoid local minima by gradually introducing the rough edges of the Stress function. However,

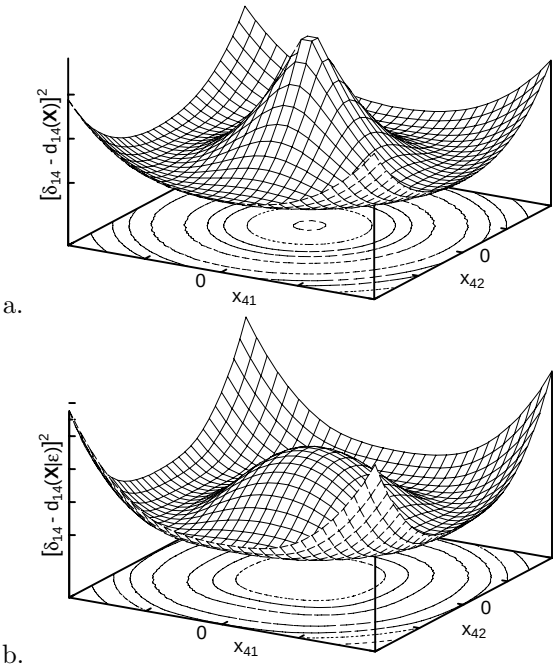


FIGURE 13.7. Surface of the error term $(5 - d_{14}(\mathbf{X}))^2$ in panel (a) and of the corresponding error term $(5 - d_{14}(\mathbf{X}|\epsilon))^2$ in distance smoothing with $\epsilon = 2$.

he only implemented his idea for unidimensional scaling and no algorithm was developed or tested for higher dimensionality. Groenen et al. (1999) continued this line of research by extending this method to more than one dimension. In addition, they also allowed for any Minkowski distance and derived a majorizing algorithm. Their method for avoiding local minima in MDS was called distance smoothing. Here, we explain the basic ideas.

Consider a toy example to visualize the raw Stress function in two dimensions. Suppose that we have $n = 4$ points in 2D, keeping point 1 fixed at $(0, 0)$, point 2 at $(5, 0)$, and point 3 at $(2, -1)$ and leaving the coordinates (x_{41}, x_{42}) for point 4 free, so that

$$\mathbf{X} = \begin{bmatrix} 0 & 0 \\ 5 & 0 \\ 2 & -1 \\ x_{41} & x_{42} \end{bmatrix}.$$

The only relevant dissimilarities are those that involve point 4. Assume that $\delta_{14} = 5, \delta_{24} = 3$, and $\delta_{34} = 2$. Then, minimizing Stress amounts to finding the optimal coordinates x_{41} and x_{42}. For this example, the Stress function can be written as

$$\sigma_r(x_{41}, x_{42}) = (5 - d_{14}(\mathbf{X}))^2 + (3 - d_{24}(\mathbf{X}))^2 + (2 - d_{34}(\mathbf{X}))^2 + c,$$

where $d_{ij}(\mathbf{X})$) are Euclidean distances and c takes all constant terms. The error term $(5 - d_{14}(\mathbf{X}))^2$ is visualized in Figure 13.7a and shows a peak at the origin.

Now, we show what happens if we smooth the peak of the distance. Groenen et al. (1999) do this by using the smoothed distance

$$d_{ij}(\mathbf{X}|\epsilon) = \left(\sum_{s=1}^{p} h_\epsilon^2(x_{is} - x_{js}) \right)^{1/2}, \tag{13.9}$$

where

$$h_\epsilon(t) = \begin{cases} \frac{1}{2}t^2/\epsilon + \frac{1}{2}\epsilon, & \text{if } |t| < \epsilon, \\ |t|, & \text{if } |t| \geq \epsilon, \end{cases} \tag{13.10}$$

Note that $h_\epsilon(t)$ is slightly different from the definition of $g_\epsilon(t)$ in (13.5), but has almost the same form. Now the smoothed Stress becomes

$$\sigma_\epsilon(x_{41}, x_{42}) = (5 - d_{14}(\mathbf{X}|\epsilon))^2 + (3 - d_{24}(\mathbf{X}|\epsilon))^2 + (2 - d_{34}(\mathbf{X}|\epsilon))^2 + c.$$

The effect of distance smoothing on a single error term is shown in Figure 13.7b for $\epsilon = 2$. Clearly, the peak is replaced by a smoothed form. The smoothing is governed by the parameter ϵ: for a large ϵ, there is much smoothing and for ϵ approaching zero no smoothing occurs, so that the error $(5 - d_{14}(\mathbf{X}|\epsilon))^2$ approaches $(5 - d_{14}(\mathbf{X}))^2$.

The effect of the combined error terms for $\sigma_r(x_{41}, x_{42})$ and $\sigma_\epsilon(x_{41}, x_{42})$ with $\epsilon = 2$ and $\epsilon = 5$ are shown in Figure 13.8. The irregularities in the Stress function of Figure 13.8a are caused by the peaks that appear in each of the error terms. Increasing ϵ smooths the irregularity as can be seen in Figures 13.8b and 13.8c. Distance smoothing starts from a large ϵ so that σ_ϵ is very smooth. Then smaller values of ϵ gradually introduce the irregularity. Eventually, for ϵ close to zero, $\sigma_\epsilon(x_{41}, x_{42})$ approaches $\sigma_r(x_{41}, x_{42})$ closely.

The distance smoothing strategy consists of the following steps. Start with a large value of ϵ and minimize σ_ϵ. Then reduce ϵ somewhat and continue minimizing σ_ϵ. Repeat these steps until ϵ is close to zero. Finally, continue minimization σ_r.

Groenen et al. (1999) studied the effectiveness of distance smoothing in comparison to the SMACOF algorithm and KYST. In a simulation study on error-free data using 100 random starts, distance smoothing recovered the true global minimum always for unidimensional scaling and almost always in 2D or 3D. SMACOF and KYST recovered the perfect data only in a small percentage of the random starts. However, for MDS with Minkowski distances close to the dominance distance, distance smoothing did not perform well and KYST yielded the same or better results. Similar results were obtained for error-perturbed data.

To be on the safe side, Groenen et al. (1999) recommend applying the distance smoothing strategy with 10 random starts and choosing the lowest local minimum as the candidate global minimum.

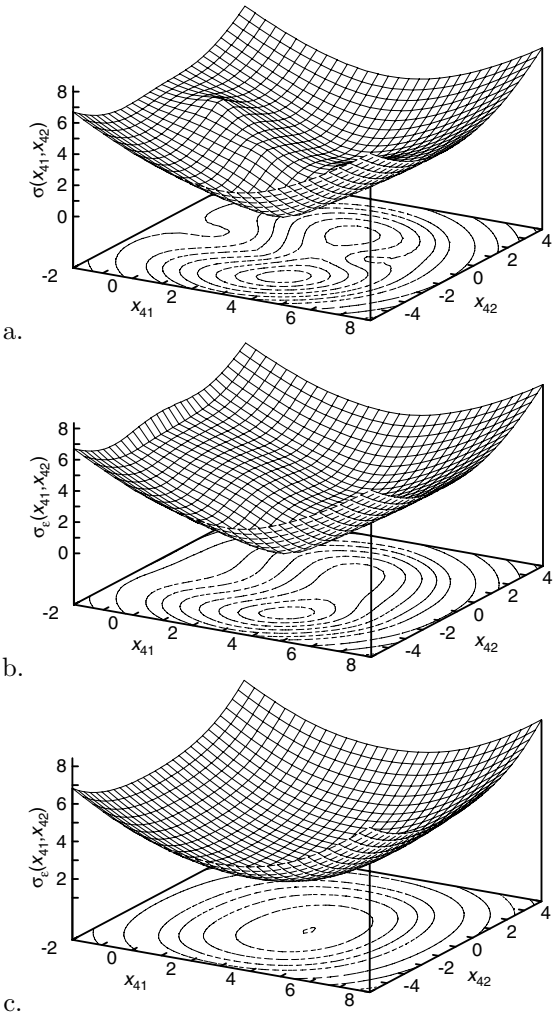


FIGURE 13.8. The surface of the original Stress function $\sigma_r(x_{41}, x_{42})$ panel (a), of the smoothed Stress function $\sigma_\epsilon(x_{41}, x_{42})$ for $\epsilon = 2$ in panel (b) and $\epsilon = 5$ in panel (c).

Note that throughout the discussion on global minima in this chapter, we have assumed ratio MDS. It is not clear how severe the local minimum problem is when we allow for optimal transformations of the d-hats.

13.9 Exercises

Exercise 13.1 Consider the multitrait-multimethod matrix in Exercise 1.6. Do both an ordinal and an interval MDS with these data. Study the Shepard diagrams of both solutions. What would you recommend to a user of MDS, given these findings?

Exercise 13.2 Consider the data matrix in Exercise 1.6.

(a) Use the T- and M-codings of the nine variables to define a starting configuration for MDS and then repeat Exercise 1.6 with this starting configuration. Point 1, thus, gets starting coordinates (1,1); point 2 gets (2,1), and so on.

(b) Study the Shepard diagram of an ordinal MDS and compare it to the Shepard diagram of a linear MDS. Discuss whether these data are better scaled with an ordinal or with an interval MDS (see also Borg & Groenen, 1997; Borg, 1999).

Exercise 13.3 Set up a data matrix (at least 5×5) with $\delta_{ij} = 1$ for all $i \neq j$ and $\delta_{ij} = 0$ for all $i = j$.

(a) Use an interactive MDS program (such as the freeware program PERMAP, see Appendix A) to find a 2D ratio MDS solution for these data.

(b) Click on one point of the solution and move this point to a different position. Then, rerun the MDS analysis with this new starting configuration. Possibly repeat this process, trying to find a different solution from the one obtained above. Compare your results to Figure 13.3.

(c) Find a 1D solution and compare it to Figure 13.3. Test the stability of this solution by the procedure described above. What do you conclude?

(d) Repeat the above analyses with ordinal MDS.

(e) Set up a new data matrix with "nearly equal" but all different dissimilarities ($i \neq j$) from the interval [.85, .95]. Run ratio, interval, and ordinal MDS analyses for these data, using different MDS programs

and forcing the program to do many iterations. Which approach represents the data best not just in terms of Stress, but in terms of describing the structure of the data? Explain why.

Exercise 13.4 Use the data in Table 10.1 on p. 229.

(a) Scale these data with ordinal MDS and compare the solution to the one in Figure 10.3.

(b) Redo the above scaling with two different starting configurations, one that corresponds to Figure 10.2 and one that corresponds to Figure 10.3. Does your MDS program lead to solutions similar to the starting configurations? Can you generate radically different local-minima solutions? How much do they differ in terms of Stress?

(c) Check whether the solutions generated with the different starting configurations remain the same when you force the program to do many (100, say) iterations. (Hint: You may also have to set a very small Stress target value to force your program to actually do that many iterations.)

(d) Use an interactive program (such as PERMAP) and test the stability of the MDS solutions by moving some points and then rerunning MDS from thereon.

Part III

Unfolding

14
Unfolding

The unfolding model is a model for preferential choice. It assumes that different individuals perceive various objects of choice in the same way but differ with respect to what they consider an ideal combination of the objects' attributes. In unfolding, the data are usually preference scores (such as rank-orders of preference) of different individuals for a set of choice objects. These data can be conceived as proximities between the elements of two sets, individuals and choice objects. Technically, unfolding can be seen as a special case of MDS where the within-sets proximities are missing. Individuals are represented as "ideal" points in the MDS space so that the distances from each ideal point to the object points correspond to the preference scores. We indicate how an unfolding solution can be computed by the majorization algorithm. Two variants for incorporating transformations are discussed: the conditional approach, which only considers the relations of the data values within rows (or columns), and the unconditional approach, which considers the relations among all data values as meaningful. It is found that if transformations are allowed on the data, then unfolding solutions are subject to many potential degeneracies. Stress forms that reduce the chances for degenerate solutions are discussed.

14.1 The Ideal-Point Model

To introduce the basic notions of *unfolding* models, we start with an example. Green and Rao (1972) asked 42 individuals to rank-order 15 breakfast

TABLE 14.1. Preference orders for 42 individuals on 15 breakfast items (Green & Rao, 1972). The items are: A=toast pop-up; B=buttered toast; C=English muffin and margarine; D=jelly donut; E=cinnamon toast; F=blueberry muffin and margarine; G=hard rolls and butter; H=toast and marmalade; I=buttered toast and jelly; J=toast and margarine; K=cinnamon bun; L=Danish pastry; M=glazed donut; N=coffee cake; O=corn muffin and butter.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	13	12	7	3	5	4	8	11	10	15	2	1	6	9	14
2	15	11	6	3	10	5	14	8	9	12	7	1	4	2	13
3	15	10	12	14	3	2	9	8	7	11	1	6	4	5	13
4	6	14	11	3	7	8	12	10	9	15	4	1	2	5	13
5	15	9	6	14	13	2	12	8	7	10	11	1	4	3	5
6	9	11	14	4	7	6	15	10	8	12	5	2	3	1	13
7	9	14	5	6	8	4	13	11	12	15	7	2	1	3	10
8	15	10	12	6	9	2	13	8	7	11	3	1	5	4	14
9	15	12	2	4	5	8	10	11	3	13	7	9	6	1	14
10	15	13	10	7	6	4	9	12	11	14	5	2	8	1	3
11	9	2	4	15	8	5	1	10	6	7	11	13	14	12	3
12	11	1	2	15	12	3	4	8	7	14	10	9	13	5	6
13	12	1	14	4	5	6	11	13	2	15	10	3	9	8	7
14	13	11	14	5	4	12	10	8	7	15	3	2	6	1	9
15	12	11	8	1	4	7	14	10	9	13	5	2	6	3	15
16	15	12	4	14	5	3	11	9	7	13	6	8	1	2	10
17	7	10	8	3	13	6	15	12	11	9	5	1	4	2	14
18	7	12	6	4	10	1	15	9	8	13	5	3	14	2	11
19	2	9	8	5	15	12	7	10	6	11	1	3	4	13	14
20	10	11	15	6	9	4	14	2	13	12	8	1	3	7	5
21	12	1	2	10	3	15	5	6	4	13	7	11	8	9	14
22	14	12	10	1	11	5	15	8	7	13	2	6	4	3	9
23	14	6	1	13	2	5	15	8	4	12	7	10	9	3	11
24	10	11	9	15	5	6	12	1	3	13	8	2	14	4	7
25	15	8	7	5	9	10	13	3	11	6	2	1	12	4	14
26	15	13	8	5	10	7	14	12	11	6	4	1	3	2	9
27	11	3	6	14	1	7	9	4	2	5	10	15	13	12	8
28	6	15	3	11	8	2	13	9	10	14	5	7	12	1	4
29	15	7	10	2	12	9	13	8	5	6	11	1	3	4	14
30	15	10	7	2	9	6	14	12	8	11	5	3	1	4	13
31	11	4	9	10	15	8	6	5	1	13	14	2	12	3	7
32	9	3	10	13	14	11	1	2	4	5	15	6	7	8	12
33	15	8	1	11	10	2	4	13	14	9	6	5	12	3	7
34	15	8	3	11	10	2	4	13	14	9	6	5	12	1	7
35	15	6	10	14	12	8	2	4	3	5	11	1	13	7	9
36	12	2	13	11	9	15	3	1	4	5	6	8	10	7	14
37	5	1	6	11	12	10	7	4	3	2	13	9	8	14	15
38	15	11	7	13	4	6	9	14	8	12	1	10	3	2	5
39	6	1	12	5	15	9	2	7	11	3	8	10	4	14	13
40	14	1	5	15	4	6	3	8	9	2	12	11	13	10	7
41	10	3	2	14	9	1	8	12	13	4	11	5	15	6	7
42	13	3	1	14	4	10	5	15	6	2	11	7	12	8	9

items from 1 (= most preferred) to 15 (= least preferred). They obtained the data in Table 14.1, where each row i contains the ranking numbers assigned to breakfast items A, ..., O by individual i. These numbers express some kind of closeness, the proximity of each item to an optimal breakfast item.

In contrast to other examples discussed so far, the row entries of this matrix differ from the column entries: the former are individuals, the latter breakfast items.[1] It is possible, though, to conceive of Table 14.1 as a *submatrix* of the familiar proximity matrix. This is shown in Figure 14.1, where the shaded rectangles stand for the observed scores. Both rectangles contain the same scores: the rows and columns of one rectangle appear

[1] A matrix with different row and column entries is called a *two-mode* matrix, see Section 3.7.

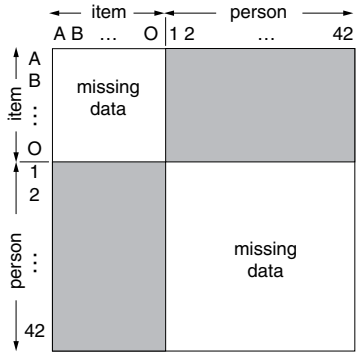


FIGURE 14.1. Schematic view of proximity matrix in Table 14.1 as a submatrix of a complete proximity matrix.

as columns and rows in the other. Each rectangle is called an *off-diagonal corner matrix*. One notes that in this data matrix only *between-sets proximities* are given and no *within-sets proximities*. Hence, one can analyze these proximities by "regular" MDS if the within-sets proximities are treated as missing values.

Ideal Points and Isopreference Contours

Figure 14.2 presents such an unfolding solution for Table 14.1. The resulting configuration consists of 57 points, 42 for the individuals (shown as stars) and 15 for the breakfast items (shown as solid points). Every individual is represented by an *ideal point*. The closer an *object point* lies to an ideal point, the more the object is preferred by the respective individual. For example, Figure 14.2 says that individual 4 prefers K (cinnamon bun) and L (Danish pastry) the most, because the object points of these breakfast items are closest to this individual's ideal point. The circles around point 4 are *isopreference contours*. Each such contour represents a class of choice objects that are preferred equally by individual 4. We note that for individual 4, D and M are slightly less preferred than K and L. Somewhat less preferred is the coffee and cake breakfast (N), whereas A, B, C, E, F, G, H, I, J, and O are more or less equally disliked.

In this way, the preferences for every individual are modeled by relating ideal points to the points representing the choice objects. This defines the *ideal-point model* of unfolding. Note that the model assumes that all individuals share the same psychological space for the choice objects. Individual differences are modeled exclusively by the different ideal points.

The term "unfolding" (Coombs, 1950) was chosen for the following reason. Assume that Figure 14.2 was printed on a thin handkerchief. If this handkerchief is picked up with two fingers at the point representing in-

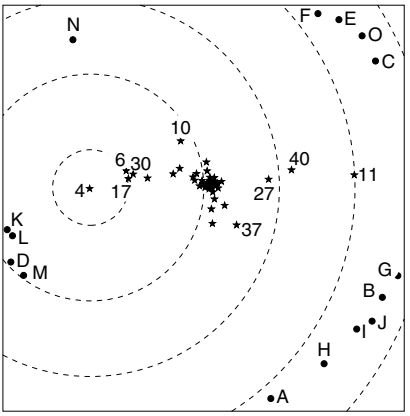


FIGURE 14.2. Unfolding representation of data in Table 14.1. Stars are individuals, solid points are items; the circles show the isopreference contours for individual 4.

dividual i, y_i, and then pulled through the other hand, we have folded it: point y_i is on top, and the farther down the object points, the less preferred the objects they represent. The order of the points in the vertical direction corresponds (if we folded a perfect representation) to how individual i ordered these objects in terms of preference. Picking up the handkerchief in this way at any individual's ideal point yields this individual's empirical rank-order. The MDS process, then, is the inverse of the folding, that is, the unfolding of the given rank-orders into the distances.[2]

Figure 14.2 seems to indicate that none of the breakfast items is particularly attractive to the respondents, because none really comes close to an ideal point (a "star"). Furthermore, we also see that the ideal points scatter quite a bit, indicating considerable interindividual differences in what kind of breakfast item the respondents prefer. Thus, it would be impossible to please everybody with any particular small set of breakfast items. However, before embarking on further interpretations, we should first ask to what extent we can really trust what we see here in the unfolding configuration.

Unfolding: Technical Challenges

An MDS analysis of an off-diagonal proximity matrix poses technical challenges. A lot of data are missing and, moreover, the missing data are not just randomly scattered throughout the data matrix. What does that mean in terms of the model? Consider a case suggested by Green and Carmone (1970). Figure 14.3 shows 35 points, arranged to form an A and an M. Assume that we compute the distances for this configuration, and use them

[2]More precisely, the case just described is conditional unfolding; see below.

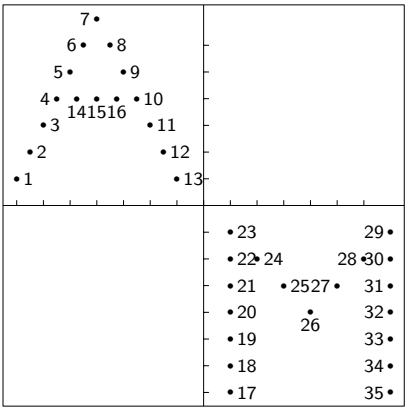


FIGURE 14.3. Synthetic *AM* configuration (after Green & Carmone, 1970).

as data for ordinal MDS. If a 2D representation is computed, it will, no doubt, recover the underlying *AM* configuration almost perfectly. But what happens in the unfolding situation when only those data that correspond to distances *between* the points in the A and the M are employed? If M's points are fixed, then, for example, the order of $d(13, 23)$ to $d(13, 29)$ implies that point 13 must be placed to the left of the perpendicular through the midpoint of the line segment connecting 23 to 29. At the same time, the points in A impose constraints on those in M, and, indeed, those are the only ones imposed on M's points, just as M's points are the only points to constrain the points of A. Note that this involves all distances between A and M. Considering that there are many such order relations, it seems plausible to expect a very good recovery of the *AM* configuration.

In the next sections, we show, however, that blind optimization of Stress (with admissible transformation of the proximities) yields degenerate solutions for unfolding. We discuss why this is so.

14.2 A Majorizing Algorithm for Unfolding

Assume that the proximities are dissimilarities and that no transformations are allowed on the data. Let $\mathbf{W}$ be the partitioned matrix of weights w_{ij},

$$\begin{bmatrix} \mathbf{W}_{11} & \mathbf{W}_{12} \\ \mathbf{W}'_{12} & \mathbf{W}_{22} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{W}_{12} \\ \mathbf{W}'_{12} & \mathbf{0} \end{bmatrix},$$

and let the coordinate matrix $\mathbf{X}$ be partitioned in $\mathbf{X}_1$ for the n_1 individuals and $\mathbf{X}_2$ for the n_2 objects in the unfolding analysis. Because the within-sets proximities are missing, $\mathbf{W}_{11} = \mathbf{0}$ and $\mathbf{W}_{22} = \mathbf{0}$. This weight matrix can be used in any program for MDS that allows missing values to do unfolding.

Heiser (1981) applied this idea for the majorizing algorithm for minimizing Stress (see Chapter 8). The corresponding algorithm is summarized below.

Consider the minimization of raw Stress; that is,

$$\begin{aligned}\sigma_r(\mathbf{X}) &= \sum_{i<j} w_{ij}(\delta_{ij} - d_{ij}(\mathbf{X}))^2 \\ &= \eta_\delta^2 + \text{tr } \mathbf{X}'\mathbf{V}\mathbf{X} - 2\text{tr } \mathbf{X}'\mathbf{B}(\mathbf{X})\mathbf{X},\end{aligned}$$

where $\mathbf{V}$ is defined as in (8.18) and $\mathbf{B}(\mathbf{X})$ as in (8.24). For the moment, assume that all between-sets weights are one, so that the $n_1 \times n_2$ matrix $\mathbf{W}_{12} = \mathbf{1}\mathbf{1}'$, where the vectors $\mathbf{1}$ are of appropriate lengths. Then, the partitioned matrix $\mathbf{V}$ equals

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}_{11} & \mathbf{V}_{12} \\ \mathbf{V}_{12}' & \mathbf{V}_{22} \end{bmatrix} = \begin{bmatrix} n_2\mathbf{I} & -\mathbf{1}\mathbf{1}' \\ -\mathbf{1}\mathbf{1}' & n_1\mathbf{I} \end{bmatrix}.$$

The majorization algorithm of Section 8.6 proves that Stress is reduced by iteratively taking the Guttman transform (8.28), $\mathbf{X}^u = \mathbf{V}^+\mathbf{B}(\mathbf{Y})\mathbf{Y}$, where $\mathbf{Y}$ is the previous estimate of $\mathbf{X}$. Heiser (1981) showed that for unfolding with equal weights $\mathbf{W}_{12} = \mathbf{1}\mathbf{1}'$ we can use instead of the Moore–Penrose inverse $\mathbf{V}^+$ a generalized inverse

$$\mathbf{V}^- = \begin{bmatrix} n_2^{-1}(\mathbf{I} - n^{-1}\mathbf{1}\mathbf{1}') & \mathbf{0} \\ \mathbf{0} & n_1^{-1}(\mathbf{I} - n^{-1}\mathbf{1}\mathbf{1}') \end{bmatrix},$$

where $n = n_1 + n_2$. $\mathbf{B}(\mathbf{Y})$ can be partitioned in the same way as $\mathbf{V}$; that is,

$$\mathbf{B}(\mathbf{Y}) = \begin{bmatrix} \mathbf{B}_{11}(\mathbf{Y}) & \mathbf{B}_{12}(\mathbf{Y}) \\ \mathbf{B}_{12}(\mathbf{Y})' & \mathbf{B}_{22}(\mathbf{Y}) \end{bmatrix};$$

see (8.24).

The update becomes

$$\mathbf{X}_1^u = [\mathbf{V}^-]_{11}[\mathbf{B}_{11}(\mathbf{Y})\mathbf{Y}_1 + \mathbf{B}_{12}(\mathbf{Y})\mathbf{Y}_2], \tag{14.1}$$

$$\mathbf{X}_2^u = [\mathbf{V}^-]_{22}[\mathbf{B}_{12}(\mathbf{Y})'\mathbf{Y}_1 + \mathbf{B}_{22}(\mathbf{Y})\mathbf{Y}_2]. \tag{14.2}$$

As with every majorizing algorithm, the Stress is reduced in every iteration until convergence is reached.

If the between-sets weights have different values, then the update formulas (14.1) and (14.2) do not work anymore. Instead, the update formula (8.28) for MDS with weights should be applied. The SMACOF algorithm needs the computation of the Moore-Penrose inverse $\mathbf{V}^+$ of the $(n_1 + n_2) \times (n_1 + n_2)$ matrix $\mathbf{V}$ which can be computed outside the iteration loop and stored in memory. For reasonable-sized unfolding problems, the memory and computational effort do not pose a problem for current computers.

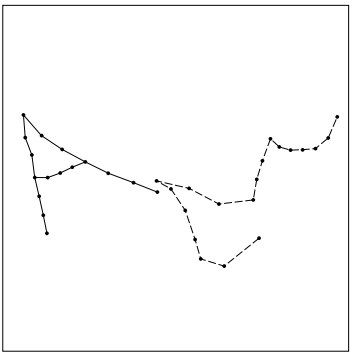


FIGURE 14.4. Ordinal unconditional unfolding representation based on distances between points in A and M in Figure 14.3.

14.3 Unconditional Versus Conditional Unfolding

We now take the 19×16 corner matrix of the between-sets distances of the AM example and check whether ordinal MDS (called "unfolding" under these circumstances) can recover the AM configuration in Figure 14.3. We emphasize ordinal unfolding, but any of the transformations discussed in Chapter 9 for complete MDS can be used.

Unconditional Unfolding

In ordinary MDS, any nonmissing proximity can be compared *unconditionally* to any other nonmissing proximity. For unfolding, this situation is called *unconditional* unfolding.

The unconditional unfolding solution for the AM data is shown in Figure 14.4.[3] Contrary to expectation, this is not a particularly good reconstruction of the original AM configuration. The M is quite deformed, and the A is sheared to the left. Yet, the Stress is only .01, so it seems that the ordinal relations of the between-sets proximities are too weak to guarantee perfect recovery of the underlying configuration.

An MDS analysis for a complete set of proximities on A and M is constrained by many more order relations than doing MDS on an off-diagonal submatrix. In the off-diagonal submatrix, we have $n_A \cdot n_M$ entities, where $n_A = 16$, the number of points in A, and $n_M = 19$ for M. Because we can compare any two entities, we have $\binom{n_A \cdot n_M}{2} = 46,056$ order relations. In

[3]We used the program Minissa-I (Lingoes, 1989), but any other MDS program that allows for missing data could be used as well. Unconditional unfolding can be accomplished by embedding the corner matrix into a complete matrix as shown in Figure 14.1. Programs that allow the user to input off-diagonal matrices directly are only more convenient, but they yield the same solutions as "regular" MDS with missing data.

the complete case (no missing data), we have $\binom{n_A+n_M}{2} = 595$ different entities, and, thus, $\binom{595}{2} = 176,715$ order relations. Yet, the sheer reduction of proximities and thus of relevant order relations between the proximities is, by itself, not of crucial importance: Figure 6.1, for example, shows that almost perfect recovery of the underlying configuration is still possible even when 80% of the proximities are eliminated. This recovery, however, depends critically on a systematic interlocking of the nonmissing proximities. In the unfolding case, such interlocking is not given: rather, there are *no* proximities at all for determining the distances within the two subsets of points.

Conditional Unfolding

The data information is now reduced even further by treating the 19×16 proximity matrix derived from the *AM* configuration *row-conditionally.*[4] A proximity is only compared to other proximities within its own row, not to proximities in other rows. With $n_A = 16$ and $n_M = 19$, row-conditionality reduces the number of order constraints in the MDS representation from 46,056 in the unconditional case to only $n_A \cdot [n_M(n_M - 1)/2] = 2{,}736$ in the conditional case. An early reference of the use of row-conditional unfolding is Gleason (1967).

Why do we consider *conditional unfolding* at all? After all, the unconditional approach already has serious problems. But consider Table 14.1. Each of its rows is generated by a different individual. For such data, it must be asked whether they can be meaningfully compared over individuals. By comparing the ranks unconditionally, we would assume that if individual i ranks beakfast item x higher than individual j ranks item y, then x comes closer to i's ideal item than y is to j's ideal. This is a strong assumption, because individuals i and j may carry out their ranking task completely differently. For example, i may be essentially indifferent to all items, whereas j likes all items very much so that it becomes difficult to decide which one he or she likes best. In unconditional ordinal unfolding, *all* 1s must be mapped into distances smaller than those representing 2s, and so on, but the row-conditional case requires only that a 1 in a given row is mapped into a distance smaller than the distance representing a 2 of the *same* row, and so on, *for all rows separately.*

For the breakfast item preferences in Table 14.1, the configuration in Figure 14.2 was obtained by ordinal row-conditional unfolding (with the program SSAR-2). The alienation coefficient of this solution is $K = .047$, so the order of the proximities in each row of data seems to match the order

[4]This restriction is called *split-by-rows* by Kruskal and Carmone (1969), which suggests that the data matrix is treated as if we had cut it into horizontal strips: the elements can be compared within a strip, but not between strips.

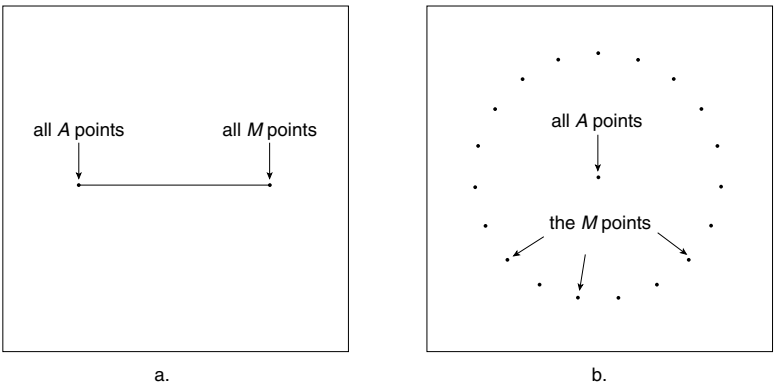


FIGURE 14.5. Trivial unconditional ordinal unfolding solutions for the AM data when using Stress.

of the corresponding distances very well. For individual 11, for example, we find that $d(11, G)$ is indeed the smallest distance, whereas $d(11, D)$ is the greatest distance, corresponding to the ranks 1 for item G and 15 for D. Moreover, the distances from point 11 to those points representing items of intermediate preference are also approximately in agreement with the data. The configuration suggests further that the individuals seem to divide the items into four groups. Yet, we notice that the object points are essentially all located on a circle. Such peculiar regularities often indicate degeneracies in the MDS solution. We turn to this question in the next section.

14.4 Trivial Unfolding Solutions and σ_2

The minimization of Stress for conditional or unconditional unfolding leads easily to trivial or even degenerate solutions, apart from the degeneracies that can occur in ordinary MDS.

The Equal Distance Solution

In unconditional ordinal unfolding there exist two trivial or degenerate solutions if Stress is used as a minimization criterion. That is, Stress can be reduced arbitrarily close to 0, irrespective of the order relations in the data. Two such degenerate solutions are presented in Figure 14.5.

For our AM problem, one trivial solution consists of only two point clusters: all points of the A are condensed into one point and all points of the M into another; the A and the M clusters are clearly separated from each other. The other trivial solution consists of all M points on a circle (not necessarily equally spaced) and the A points in the center, or vice

versa. In higher dimensions, the M points could appear on the surface of a (hyper)sphere. These two solutions share the fact that all distances from the ideal points to the object points are the same.

Why these configurations represent solutions to the scaling problem follows from the Stress-1 function, that is, from

$$\sigma_1(\mathbf{X}) = \left(\frac{\sum_{i<j} w_{ij}(d_{ij}(\mathbf{X}) - \widehat{d}_{ij})^2}{\sum_{i<j} w_{ij} d_{ij}^2(\mathbf{X})} \right)^{1/2}, \text{ for all defined } p_{ij}, \qquad (14.3)$$

where $\mathbf{X}$ is the matrix with the coordinates of the A- and the M-points. In the configurations of Figure 14.5, all between-sets distances are equal. Thus, $d_{ij}(\mathbf{X}) - \widehat{d}_{ij} = 0$, for all defined p_{ij}, but $\sum_{i<j} w_{ij} d_{ij}^2(\mathbf{X}) > 0$. This condition means that $\sigma_1(\mathbf{X})$ is zero, irrespective of the proximities. This degenerate solution is not limited to ordinal transformations. Even interval unfolding (with intercept one and slope zero) may lead to constant disparities yielding the trivial solutions above. The trivial solution of Figure 14.5a is a special case of the one discussed in Section 13.1. For ordinal or interval unfolding, it always exists, because the within-sets proximities are missing and the between-sets disparities all can be made equal.

Although these equal disparity solutions seem without any information about the data, Van Deun, Groenen, Heiser, Busing, and Delbeke (2005) showed that still a meaningful interpretation of such a solution is possible. The important idea is that one needs to zoom in on the points that are clustered together. Then it turns out that these points have different positions that depend on the data. The interpretation is done by projection using the so-called signed-compensatory distance model. For more information, we refer to Van Deun et al. (2005). Of course, without zooming, no useful information of the equal disparity solution can be derived.

To avoid these solutions, Kruskal (1968) and Kruskal and Carroll (1969) proposed the use of a variant of the stress measure called *Stress2* or *Stressform2*,

$$\sigma_2(\mathbf{X}) = \left(\frac{\sum_{i<j} w_{ij}(d_{ij}(\mathbf{X}) - \widehat{d}_{ij})^2}{\sum_{i<j} w_{ij}(d_{ij}(\mathbf{X}) - \bar{d})^2} \right)^{1/2}, \text{ for all defined } p_{ij}, \qquad (14.4)$$

where $\bar{d}$ denotes the mean of all distances over which the summation extends (see Section 11.2). For the above solutions where all between-sets distances are strictly equal, we find that $\sigma_2(\mathbf{X})$ is not defined because $\sum_{i<j} w_{ij}(d_{ij}(\mathbf{X}) - \bar{d})^2 = 0$, which leads to 0/0. However, if an MDS program is started from a configuration slightly different from the trivial solution, the program iterates away from the trivial solution. The reason is that close to the trivial solution σ_2 is large, because the denominator of (14.4) is close to zero. Several computer programs for ordinal MDS offer an option for minimizing σ_2 rather than σ_1. The criterion σ_2 always (except

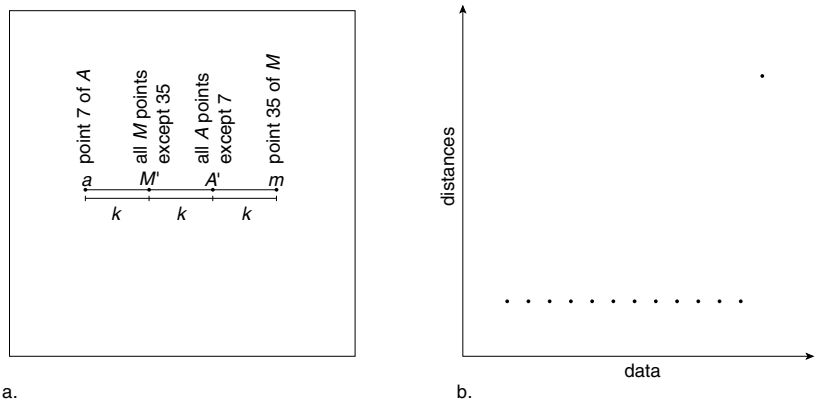


FIGURE 14.6. Trivial solution for ordinal unfolding under σ_2 (after Carroll, 1980).

at 0) yields values higher (typically twice as large) than σ_1, because it has the same numerator but a smaller denominator. Thus, using σ_2 avoids the equal-distance trivial solution.[5]

The Four-Point Solution

Using σ_2 does not free unfolding from degeneracies totally. If we compute the distances for the AM configuration in Figure 14.3 and use the between-sets distances as data for a 1D unfolding representation under σ_2, then the four-point configuration in Figure 14.6 is a perfect but trivial solution (Kruskal & Carroll, 1969; Carroll, 1980). It represents all A-points of Figure 14.3 by A', except for point 7, which corresponds to a. Similarly, all M-points of Figure 14.3 are mapped into M', except for point 35, which is carried into m. Because only the distances between A and M define the solution, σ_2 involves only two distance values, k and $3k$. $3k$ represents the greatest distance of the AM configuration, and k represents all other distances. Hence, the Shepard diagram essentially exhibits a horizontal array of points, except that the last point to the right is shifted upwards so that its value on the ordinate is three times that of the other points. This step function is perfectly monotonic, which makes the numerator of σ_2 equal to zero. At the same time, the norming factor $(d_{ij} - \bar{d})^2$ is not equal to zero. Therefore, $\sigma_2 = 0$.

This degeneracy is somewhat contrived and not likely to occur often, if at all, in real applications. It shows, however, that the norming factor used

[5]Although σ_2 tends to keep the *variance* of the distances large, this does not prevent degeneracies in "regular" ordinal MDS (see Section 13.1).

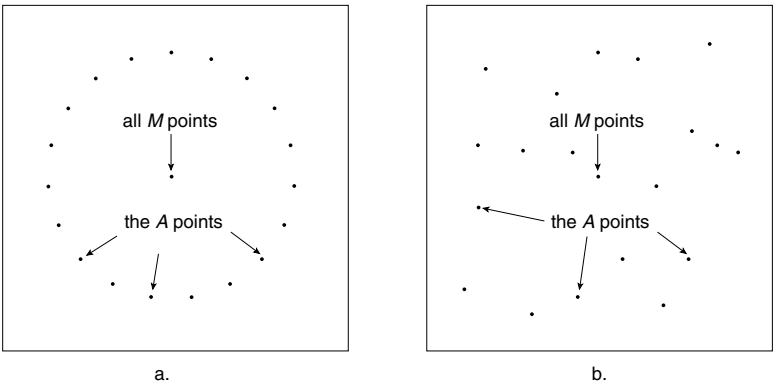


FIGURE 14.7. Trivial row-conditional ordinal unfolding solutions for the AM data (with the points of A in the rows) using Stress, panel a., and using (14.4), panel b.

in σ_2 has alleviated the degeneracy problem only to a degree. More specific degeneracies are discussed in Heiser (1989a).

Trivial Solutions for Row-Conditional Unfolding

For preference rank-orders, it is quite natural to have independent ordinal transformations for each of the individuals. If the individuals are represented by the rows, then it means that the data are treated row-conditionally. Again, minimizing Stress using a row-conditional transformation of at least interval level may lead to a zero Stress solution with equal distances as in Figure 14.5.

However, treating the transformations row-conditionally, also introduces additional trivial unfolding solutions. Consider the AM data, where the A points are the rows. Then, the equal distance solution in panel a. of Figure 14.7 looks similar to panel b. of Figure 14.5. The difference lies in the role of the points in the center which are the rows in Figure 14.5b and the column points (M) in Figure 14.7a.

A second trivial solution may occur when minimizing (14.4) with row-conditional transformations. In Figure 14.7b, all column points (M) are again represented in the center, but the row points scatter through the space. The row-conditional transformation has allowed different distances between row points to the cluster of column points in the center while keeping the distances within a row equal. In (14.4), the numerator is zero because all distances are the equal to the d-hats within each row. The denominator is nonzero because the distances from a row point to the cluster differ per row. This solution can be considered degenerate because it is independent of the data.

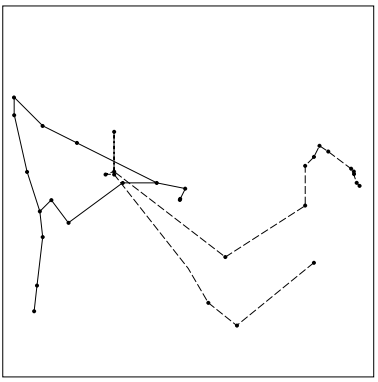


FIGURE 14.8. Row-conditional unfolding representation based on distances between points in A and M in Fig. 14.3.

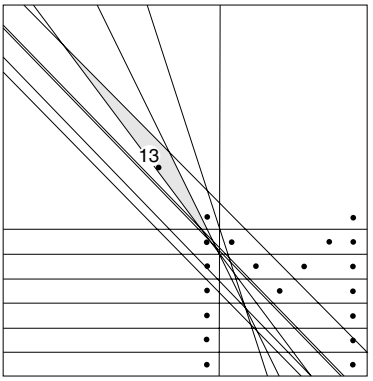


FIGURE 14.9. Isotonic region (shaded) for point 13 in configuration of Fig. 14.3; boundaries defined by order of distances of 13 to points in M.

14.5 Isotonic Regions and Indeterminacies

To get a feeling for the uniqueness or, expressed conversely, the indeterminacies of a conditional, ordinal unfolding representation we return again to our AM configuration in Figure 14.3 and use its distances. In conditional unfolding, there are two possible analyses: we may use the 16×19 proximity matrix in which A's points form the rows and M's points the columns, or the 19×16 transposed matrix in which the roles of A and M are reversed. We choose the first approach, which implies that only the distances from each point in A to every point in M are constrained by the data, but not the distances from each point in M to every point in A. (You may think of A as the set of ideal points and of M as the set of points representing choice objects.) The SSAR-2 program then leads to Figure 14.8, with the low alienation $K = 0.002$ [see (11.6)]). We note that there is a substantial deformation of the letters, in fact, a much stronger one than for the unconditional case. The M, in particular, can hardly be recognized. As could be expected, the row-conditional unfolding does not recover the underlying configuration nearly as well as the unconditional version.

In Figure 14.8, we can move the points around quite a bit without making the alienation worse. One example of what is possible is the underlying configuration itself (Figure 14.3), for which $K = 0$. Hence, the SSAR-2 solution is only weakly determined, that is, many more configurations exist with equal or even better fit to the data. This implies that it may be risky to embark on substantive interpretations of such representations, so we should study when we may do so.

A natural first question is whether the poor recovery of the AM configuration is a consequence of certain properties that are not likely to hold in

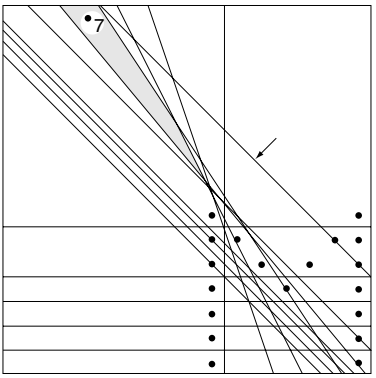


FIGURE 14.10. Isotonic region (shaded) for point 7, defined as in Fig. 14.9.

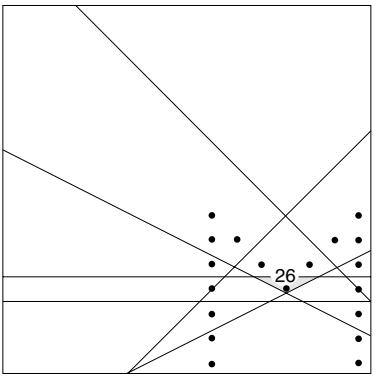


FIGURE 14.11. Isotonic region (shaded) for point 26, defined by its distances to all other points.

general. To answer this question, let us check the invariance of some of the points. Assume that M is fixed and that A's points have to be located under the constraints of conditional unfolding. For point 13, which is closest to M, we obtain as its solution space or *isotonic region* (i.e., the region in which the distances of every point to M's points are ordered equivalently) the grey area shown in Figure 14.9. Note that all boundaries are straight lines in the conditional case, in contrast to the unconditional MDS considered in Chapter 2. The indeterminacy of point 13 is considerable but not unlimited.

Determining the isotonic region for point 7 in a similar fashion leads to Figure 14.10. We notice immediately that this point's solution space is much greater and is closed to the outside only by the boundary line marked with the arrow. Thus, point 7 could be positioned much farther to the outside of this region without affecting the fit of the conditional unfolding solution at all. But why is this point's solution space so much greater than the one for point 13? One conjecture is that the boundary lines for those points that are closer to the M differ more in their directions, which leads to a network with tighter meshes. To test this conjecture, we look at the isotonic region of point 26 relative to all other points in M. Figure 14.11 shows that the boundary lines indeed run in many very different directions, which generates a comparatively small isotonic region, even though many fewer order relations are involved than in the above. The number of constraints as such does not imply anything about the metric determinacy of a point. What is important is how ideal and object points are distributed throughout the space relative to each other.

The best relative distribution of ideal points and object points is one where they are thoroughly mixed, that is, where both are evenly spread

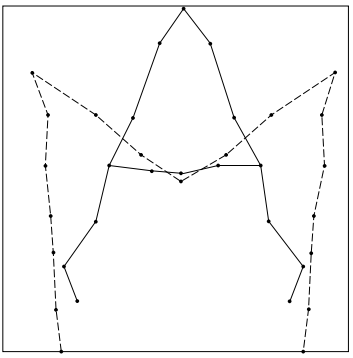


FIGURE 14.12. Row-conditional unfolding representation of distances from superimposed A and M point sets.

throughout the space. Substantively, this implies that we have individuals with many different preference patterns, so each object is someone's first choice. With our AM configuration, a situation like this can be approximated by superimposing A on M, which is done here by shifting A and M so that their respective centroids coincide with the origin. With the distances of this configuration, SSAR-2 leads to Figure 14.12, with $K = 0.002$. The metric recovery of the underlying configuration is virtually perfect, as expected. Thus, conditional unfolding does work—under favorable circumstances!

Part of the favorable circumstances of the situation leading to Figure 14.12 was also that the number of ideal and object points was high for a 2D solution. Coombs (1964) has shown that if there are n object points in an $(n-1)$-dimensional MDS space, then *all* isotonic regions for the ideal points are open to the outside. Why this is so is easy to see for the special case of three object points in the plane. We connect the points A, B, and C by straight-line segments, and draw straight lines running perpendicularly through the midpoints of the line segments. These lines will then intersect at just one point, which is the center of the circle on which A, B, and C fall. Moreover, the three lines will partition the plane into six regions, which are all open to the outside. Any ideal point falls into one of these regions, depending on the empirical preference order for the individual it represents. With three objects, there are exactly six different rank-orders, corresponding to the six regions. But, because all regions are open, the location of the ideal points is very weakly determined indeed. If the number of object points grows relative to the dimensionality of the representation space, then more and more closed regions result. These regions are located primarily where the object points are, as we concluded above.

TABLE 14.2. Similarity data for breweries A, ..., I and attributes 1, ..., 26.

	A	B	C	D	E	F	G	H	I
1	3.51	4.43	4.76	3.68	4.77	4.74	3.43	5.05	4.20
2	3.41	4.05	3.42	3.78	1.04	3.37	3.47	3.25	3.79
3	3.20	3.66	4.22	3.07	3.86	4.50	3.19	4.62	3.75
4	2.73	5.25	2.44	2.75	5.28	2.11	2.68	2.07	3.63
5	2.35	3.88	4.18	2.78	3.86	4.37	2.38	4.21	4.63
6	3.03	4.23	2.47	3.12	4.24	2.47	2.90	2.36	3.53
7	2.21	3.27	3.67	2.49	3.40	4.10	2.53	4.03	3.33
8	3.91	2.71	4.59	3.91	4.23	4.72	3.81	4.88	3.96
9	3.07	4.08	4.74	3.34	4.23	4.88	3.20	5.20	3.95
10	3.21	3.57	4.20	3.24	3.85	4.28	3.16	4.30	3.75
11	3.15	3.80	4.34	3.33	3.88	4.49	3.17	4.70	3.67
12	2.84	3.41	4.01	2.89	3.64	4.15	2.95	4.25	3.65
13	2.75	3.24	4.07	2.68	3.55	4.18	2.84	4.56	3.22
14	2.35	3.44	4.13	3.16	3.55	4.55	2.82	4.49	3.29
15	3.07	3.82	4.17	3.21	3.94	4.42	3.21	4.41	3.67
16	3.45	4.29	4.44	3.74	4.47	4.68	3.61	4.76	4.04
17	2.53	4.71	4.53	2.83	4.83	4.71	2.70	4.83	4.72
18	3.12	3.58	4.10	3.14	3.82	4.28	3.10	4.53	3.50
19	2.93	3.27	4.13	2.80	3.46	4.10	2.84	5.12	3.13
20	2.24	3.11	4.12	2.39	3.39	4.17	2.54	4.33	3.19
21	2.41	3.14	3.43	2.40	3.22	3.45	2.43	3.22	3.93
22	3.32	3.74	4.32	3.32	4.01	4.64	3.26	4.88	3.72
23	3.39	4.04	4.51	3.48	4.23	4.63	3.43	4.95	3.86
24	2.88	3.39	3.85	2.90	3.61	4.18	2.79	3.94	3.96
25	2.74	3.57	2.37	2.77	3.96	2.49	2.71	2.44	3.26
26	2.70	3.10	3.85	2.82	3.58	4.13	2.79	4.17	3.20

14.6 Unfolding Degeneracies in Practice and Metric Unfolding

We now demonstrate some of the degeneration problems with the data in Table 14.2. Beer drinkers were asked to rate nine breweries on 26 attributes (Borg & Bergermaier, 1982). The attributes were, for example, "Brewery has rich tradition" or "Brewery makes very good Pils beer". Relative to each attribute, the informant had to assign each brewery a score on a 6-point scale ranging from 1 = not true at all to 6 = very true. The resulting scores are therefore taken as similarity values.

Minimizing Stress (σ_1) in unconditional ordinal unfolding, KYST yields a computer printout similar to Figure 14.13a. We find that all of the brewery points are tightly clustered, whereas all of the attribute points lie on a J-shaped curve. The Shepard diagram for this configuration is given in Figure 14.13b. At first sight, these results do not look degenerate, even though the extremely low Stress of $\sigma_1 = .0005$ would at least suggest this possibility. Indeed, a second look at the Shepard diagram reveals that the distances scatter over only a small range. Thus, they are very similar, in spite of the considerable scatter in the diagram. The horizontal step function in Figure 14.13b is the monotone regression line. So, the sum of the squared (vertical) distances of each point from this line defines the numerator of Stress, which is definitely much smaller than the sum of the squared distance coordinates

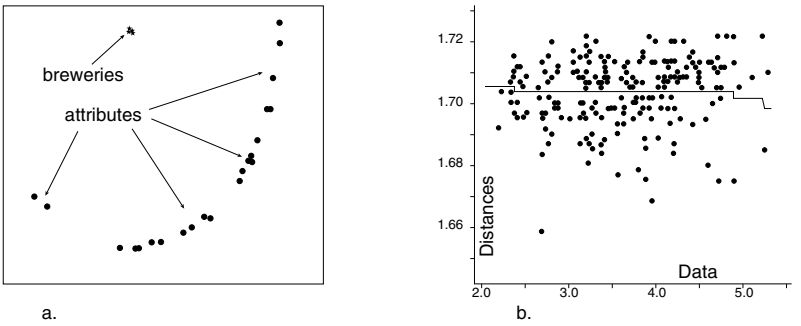


FIGURE 14.13. Ordinal unfolding representation (a) of data in Table 14.2, using Stress, σ_1, and (b) its Shepard diagram.

of the points in the Shepard diagram, the denominator of Stress. The J-shaped curve in Figure 14.13a thus turns out to be a segment of a circle with its origin at the brewery points. Thus, this example is a degenerate solution of the equal distance type shown in Figure 14.5.

Instead of ordinal unfolding, stronger assumptions (or hypotheses) about the data can be imposed, because metric MDS is often more robust than ordinal MDS. If it seems justifiable to assume that the proximities are at least roughly interval scaled, using metric MDS is no problem. But even if this is not the case, one could replace the original data with appropriate ranking numbers and then use interval MDS, because the rank-linear model is very robust vis-à-vis nonlinearities in the relations of data and distances, as we saw in Chapter 3. For metric conditional unfolding, we have

$$p_{ij} \mapsto a_i + b_i \cdot p_{ij} \approx d_{ij}, \tag{14.5}$$

where i denotes an individual, j is an object, and $\approx$ means as nearly equal as possible. In the unconditional case, the intercept a and slope b are equal for every individual i; that is,

$$p_{ij} \mapsto a + b \cdot p_{ij} \approx d_{ij}. \tag{14.6}$$

Using (unconditional) interval unfolding, however, has little effect for the data in Table 14.2 and leads to virtually the same configuration as in Figure 14.13a. Moreover, it has the additional drawback that now the regression line in the Shepard diagram has the "wrong" slope: given that the data are similarities, the regression line should run from the upper left-hand corner to the lower right-hand corner of the diagram in order to preserve the interpretation of the individuals' points as ideal points or, in the present case, the direct correspondence of geometrical and psychological closeness.

We see that using Stress as a minimization criterion can lead to wrong solutions. This is easy to see because the configuration in Figure 14.13a suggests that all breweries are evaluated in the same way with respect to

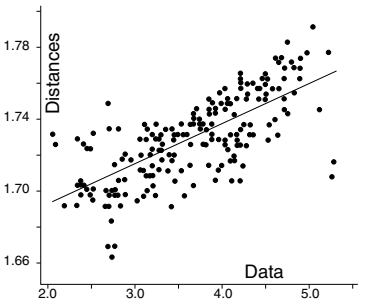


FIGURE 14.14. Shepard diagram of linear unfolding of data in Table 14.2 using Stress, σ_1.

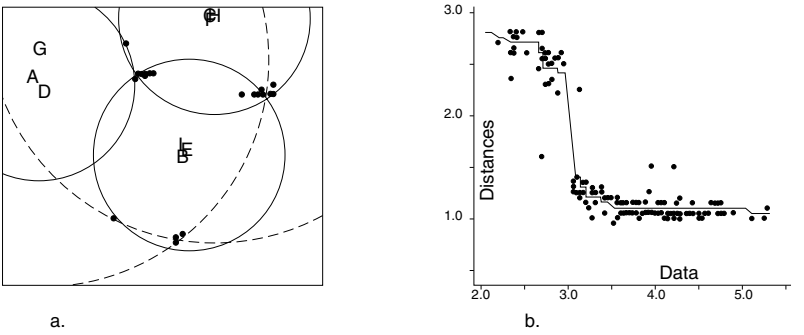


FIGURE 14.15. Ordinal unfolding representation (a) of data in Table 14.2, using σ_2, and (b) its Shepard diagram.

all attributes. From the empirical data in Table 14.2, this cannot be true. When we use σ_2, it becomes far more difficult to diagnose, from looking at the configuration, that something went wrong. The ordinal unfolding solution (under σ_2) is shown in Figure 14.15a. The letters A, ..., I stand for the nine breweries, the solid points for the 26 attributes. The figure suggests that the breweries form three groups, and the attributes also seem to cluster to some extent. But the Shepard diagram for the unfolding solution (Figure 14.15b) shows immediately that we have a degeneracy of the two-distance-classes type. Although the data scatter quite evenly over the range 2.0 to 5.5, there are practically only two distances. All of the small proximities up to about 3.0 are mapped into distances of about 2.5, whereas all other proximities are represented by distances about equal to 1.2. Almost all points lie very close to the regression line; thus, σ_2 is very low.

After learning from the Shepard diagram that there are essentially only two different distances in the scaling solution, we can identify them. Because we are only concerned with between-sets distances, we have to show that each distance from a brewery point to an attribute point is equal to

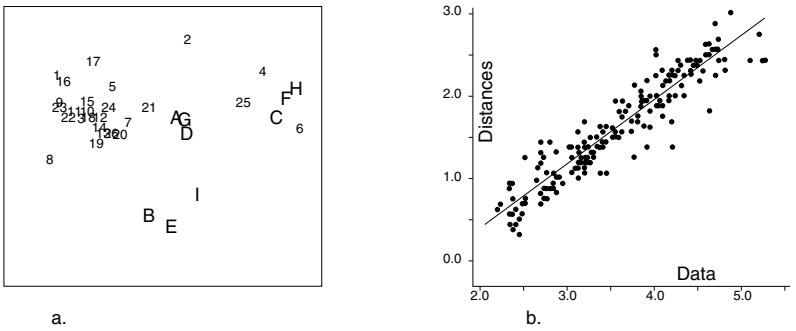


FIGURE 14.16. Linear unfolding representation (a) of data in Table 14.2, using σ_2, and its Shepard diagram (b).

either a or b, where $a < b$. Moreover, because the unfolding was done unconditionally, the same would be true in the reverse direction, that is, from each attribute point to all brewery points. In Figure 14.15a, the two distance types are indicated (for the perspective from the brewery points to the attribute points) by either solid circles (for a-type distances) or broken circles (for b-type distances). Similar circles, with radius equal to either a or b, could be drawn about the attribute points in such a way that the brewery points would fall onto or close to them.

As we did for Stress, we now unfold the data with an interval regression approach. The solution is given in Figure 14.16a, where the brewery points are labeled A, ..., I , as above, and the attribute points as 1, ..., 26. The brewery points tend to arrange themselves in the same groups as in the degenerate solution in Figure 14.15a for empirical reasons, as the Shepard diagram in Figure 14.16b shows. The distances and the proximities of the unfolding solution vary over a wide range. There are no gaps in the distribution, and the linear regression line fits very well. The problem with this solution is that the slope of the regression line is not as we would like it to be. If this is not noticed by the user, serious interpretational mistakes are bound to result. The configuration in Figure 14.16a puts a brewery closer to an attribute the less (!) this brewery was judged to possess this attribute. Thus, for example, brewery A is not really close to attribute 21 as the configuration suggests; rather, the contrary is true. This certainly leads to an awkward and unnatural meaning for the configuration, where two points are close when the objects they represent are psychologically different.

We conclude that using σ_2 instead of σ_1 does not eliminate the problems of unfolding. In the ordinal case, we again get a degenerate solution (even though it is somewhat less degenerate than for σ_1). For the metric approach, we obtain an undesirable inverse representation that is hard to interpret.

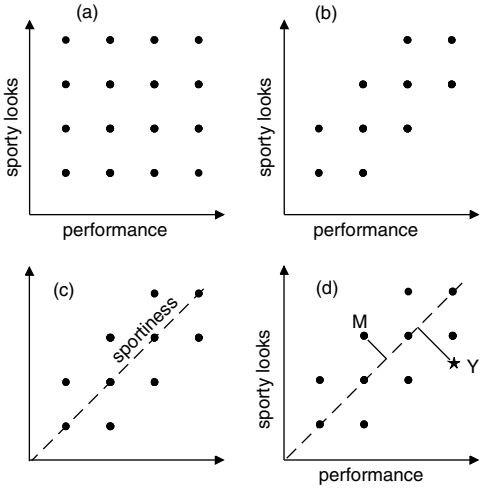


FIGURE 14.17. Hypothetical example to demonstrate problems of dimensional interpretations in unfolding.

14.7 Dimensions in Multidimensional Unfolding

Apart from degeneracies and indeterminacies, there are further problems in unfolding that one should be aware of when interpreting an unfolding solution. Consider an example. Assume that we want to know how an individual selects a car from a set of different automobiles. Assume further that the preference judgments were made in a 2D unfolding space with dimensions "performance" and "sporty looks". Figure 14.17a shows 16 hypothetical cars in a space spanned by these dimensions. A market researcher wants to infer this space from the person's similarity data. This is a difficult task if, as Figure 14.17b illustrates, there are no cars in the upper left- and the lower right-hand corners. The reason for the empty corners in this example is that cars with a very high performance must look sporty to some extent, for engineering reasons. The converse is usually also true empirically; that is, cars with extremely poor performance do not look like racing machines. But with the remaining 10 cars it is likely that the researcher would conclude that essentially only one dimension explains the similarity data, especially because the resulting dimension ("sportiness") seems to make sense psychologically (Figure 14.17c).

Figure 14.17d shows the consequences of this false interpretation. Let Y be the ideal point of some individual. This individual wants a car with very high performance and moderately sporty looks. A market researcher, therefore, should recommend making car M in Figure 14.17d less sporty in looks and more powerful in its performance. However, on the basis of the accepted unfolding solution, the market researcher would come to a

different, incorrect conclusion: with "sportiness" as the assumed decision criterion, the advice would be to increase M's sportiness so that M would move closer to Y on this dimension. Concretely, this movement could be achieved in two ways: increase performance and/or sporty looks. Because the latter is cheaper and easier to implement, this would be the likely immediate action. But this would be just the wrong thing to do, because the person wanted a reduction, not an increase, in the sporty looks of M.

The problems encountered here are a consequence of the fact that some corners of the similarity space remain empty. Coombs and Avrunin (1977, p. 617) therefore argue that "deliberate efforts" should be made to avoid collapsing the preference space due to correlated dimensions. This means, in practice, that an unfolding analysis should be based on a set of objects that are carefully selected from the product space of the presumed choice criteria, not on a haphazard collection of objects.

14.8 Multiple Versus Multidimensional Unfolding

When aggregated data are analyzed in MDS, there is always a danger that the multidimensionality is an aggregation artifact. This danger is particularly acute in multidimensional unfolding because here the data are usually from different individuals.

Unfolding assumes that all individuals perceive the world in essentially the same way. There is just one configuration of objects. Differences among individuals are restricted to different ideal points. If this assumption is not correct, unfolding preference data will be misleading. Consider an example.

Norpoth (1979a) reports two data sets, where German voters were asked to rank-order five political parties in terms of preference. The parties ranged from Nationalists to Communists, and so one could expect that the respondents should have agreed, more or less, on the position of each party on a left-to-right continuum.

Running an unfolding analysis on these data, Norpoth (1979a) concluded that he needed a 2D solution for an adequate representation of the data. The solution shows one dimension where the Communists are on one end and the Nationalists are on the other. This is interpreted as the familiar left-to-right spectrum. The second dimension shows the (then) ruling coalition parties on one end and the major opposition party on the other. This interpretation also seemed to make sense.

One can question, however, whether all voters really perceived the parties in the same way. One hypothesis is that the voters do indeed all order the parties on a left-to-right dimension, but that they do not always agree on where these parties are located relative to each other. Indeed, Van Schuur (1989) and Borg and Staufenbiel (1993) independently showed for Norpoth's data that by splitting the set of respondents into two groups

(in each sample) by simply placing the Liberals to the right of the Conservatives in one case, and to the left of the Conservatives in the other, while leaving all other parties ordered in the same way, two subsamples are obtained that each yield one-dimensional unfolding solutions.

Substantively, such multiple solutions are much more convincing: they preserve a simple dimensional model of how political parties are perceived; they explain different preferences by a simple ideal-point model; and, finally, they account for group differences by a simple shift of the position of the Liberals, an ambiguous party in any case.

There exist computer programs for multiple one-dimensional unfolding (e.g., Lingoes, 1989; Van Schuur & Post, 1990). They offer the easiest way to test for the existence of multiple 1D unfolding scales.

14.9 Concluding Remarks

Unfolding is a natural extension of MDS for two-way dissimilarity data. When no transformation is allowed on the data (or a ratio transformation), unfolding can be safely used. However, if transformations are required, for example, for preference rank-orders, then special caution is needed because the usual approaches yield a degenerate solution with all disparities being equal. Chapter 15 discusses several of such solutions.

14.10 Exercises

Exercise 14.1 Consider the unfolding solution for the breakfast items in Figure 14.2. Attempt an interpretation. In particular, find "labels" for the four groups of breakfast items, and interpret their positions relative to each other. (What lies opposite each other, and why?)

Exercise 14.2 Consider the (contrived) color preferences of six persons (A..F) in the table below (Davison, 1983). The data are ranks, where 1 = most preferred.

	Person					
Color	A	B	C	D	E	F
Orange	1	2	3	4	3	2
Red	2	1	2	3	4	3
Violet	3	2	1	2	3	4
Blue	4	3	2	1	2	3
Green	3	4	3	2	1	2
Yellow	2	3	4	3	2	1

(a) Unfold these data without any transformations.

(b) Discuss the solution(s) substantively, relating them to Figure 4.1 and to unfolding theory. In what sense are the six persons similar, in what sense do they differ?

(c) Discuss technical reasons why the unfolding analysis works for these data.

(d) Construct a set of plausible color preference data that do not satisfy the ideal point model.

(e) Discuss some data sets that satisfy the ideal point model but that would most likely lead to degenerate or other nondesirable MDS solutions. (Hint: Consider the distribution of ideal points in the perceptual space.)

Exercise 14.3 The following table shows empirical color preferences of 15 persons (Wilkinson, 1996). The data are ranks, where 1 = most preferred.

	Person														
Color	A	B	C	D	E	F	G	H	I	J	L	M	N	O	P
Red	3	1	3	1	5	3	3	2	4	2	1	1	1	2	1
Orange	5	4	5	3	3	2	4	4	5	5	5	5	4	5	2
Yellow	4	3	1	5	2	5	5	3	3	4	2	4	5	3	3
Green	1	5	4	4	4	1	2	5	1	3	4	2	2	4	4
Blue	2	2	2	2	1	4	1	1	2	1	3	3	3	1	5

(a) Unfold these data.

(b) Discuss the solution(s) substantively, connecting the color points in the order of the electromagnetic wavelengths of the respective colors.

(c) Use an external starting configuration where the color points are positioned on a rough color circle similar to the one in Figure 4.1. (Hint: Place the person points close to their most preferred color points in the starting configuration.)

(d) Compare the unfolding solutions with and without external starting configurations, both technically in terms of Stress and substantively in terms of a reasonable theory.

Exercise 14.4 The following table shows the dominant preference profiles (columns) for German political parties in 1969. A score of 1 indicates "most preferred". The row "freq" shows the frequency of the respective preference order in a representative survey of 907 persons (Norpoth, 1979b).

	Preference Type										
Political Party	1	2	3	4	5	6	7	8	9	10	11
SPD (Social Democrats)	1	1	1	1	3	3	2	2	2	2	3
FDP (Liberals)	2	2	3	3	2	2	3	3	4	1	1
CDU (Conservatives)	3	3	2	2	1	1	1	1	1	3	2
NPD (Nationalists)	4	5	4	5	5	4	5	4	3	5	4
DKP (Communists)	5	4	5	4	4	5	4	5	5	4	5
Freq	29	85	122	141	56	66	135	138	11	16	19

(a) Unfold these data in one to three dimensions and discuss the solutions. Use both ordinal and linear MDS, and both unweighted and weighted (by "freq") unfolding.

(b) Norpoth (1979a) claims that these data require a 2D unfolding space. Yet, most Germans would probably order these parties from left to right as DKP-SPD-FDP-CDU-NPD or as DKP-SPD-CDU-FDP-NPD. Sketch diagrams for these two orders, where the Y-axis represents preference ranking—the highest rank 1 getting the highest Y-score—and the X-axis the left-to-right order. What do these diagrams show you with respect to single-peakedness of the preference functions? Can you accommodate most preference profiles in the scales? Can you accommodate them in one single scale too?

(c) Compute two (or more) 1D unfoldings for subsets of the voter profiles as an alternative to one common unfolding solution for all persons combined. Discuss the substantive implications.

15

Avoiding Trivial Solutions in Unfolding

The occurrence of trivial solutions in unfolding was recognized soon after the introduction of MDS. It was one of the reasons for introducing Stress-2. However, as indicated in the previous chapter, Stress-2 does not solve the degeneracy problem totally. In this section, we discuss several methods that have been proposed in the literature to avoid trivial unfolding representations. They all adapt the unfolding procedure in such a way that the ideal point interpretation is retained. The solutions can be categorized into three classes: (a) adapting the unfolding data, (b) adjusting the transformation, and (c) modifying the loss function.

15.1 Adjusting the Unfolding Data

One way to avoid a trivial solution in unfolding is to make sure that the transformation cannot contain a nonzero intercept and a slope of zero by adapting the data. Here, we discuss two of these options. The first one, is an *ordinal-ratio* approach to unfolding.

Ordinal-Ratio Approach

The idea behind this approach is to use an ordinal and a ratio transformation simultaneously on the same data. Thus, the data are duplicated, one data set is transformed by a ratio transformation, the other one by an

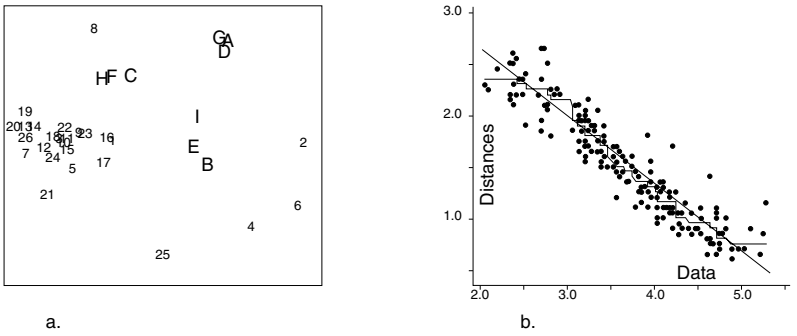


FIGURE 15.1. Unfolding representation (a) of data in Table 14.2, using mixed ordinal-linear σ_2 loss function, and its Shepard diagram (b).

ordinal transformation, and both sets of disparities are approximated by a single matrix of distances.

Let $L(o)$ be the loss function that defines an ordinal approach and $L(r)$ the corresponding loss function for ratio unfolding. For example, $L(o)$ may be σ_2 with disparities as target distances under, say, the primary approach to ties, and $L(r)$ is σ_2 with target distances computed by a ratio transformation. Then, we simply define the total loss as

$$L = a \cdot L(o) + b \cdot L(r), \tag{15.1}$$

where a and b are weights such that $a, b > 0$ and $a + b = 1$. L is equal to 0 only if both $L(o)$ and $L(r)$ are equal to 0. Note that if $L(r)$ is zero, $L(o)$ will also be zero because the ratio transformation is an admissible ordinal transformation. L will be small if both $L(o)$ and $L(r)$ are small, or if one is very small and the other is not very large. The ordinal transformation tries to model the data as usual in an ordinal manner. However, as a ratio transformation does not allow for an intercept, the trivial transformation with a nonzero intercept and zero slope cannot occur.

One drawback of this approach is that one needs to have dissimilarities in order to do a ratio transformation. For similarities, a ratio transformation with a negative slope is required to map larger similarities into smaller distances. Yet, such a transformation leads to negative disparities, which can never be properly modeled by nonnegative distances. Therefore, if we have similarities, we either have to convert similarities into dissimilarities before the unfolding analysis or revert to an *ordinal-interval* approach with $L = a \cdot L(o) + b \cdot L(i)$ and $L(i)$ the loss for unfolding of data that are interval-scaled. This ordinal-interval approach is not guaranteed to always avoid the trivial solution but the example discussed below shows a successful application.

Let us apply this approach to the brewery data, using KYST with weights $a = b = 0.5$ and loss function $L = a \cdot L(o) + b \cdot L(i)$. This yields a solution

with the Shepard diagram in Figure 15.1b. There are two regression curves now: a monotonic one, related to $L(o)$, and a linear one, related to $L(i)$. The (vertical) scatter of the points about the monotonic curve makes up one component of L, and the scatter of these same points about the linear regression line makes up the other. Hence, minimizing L tends to avoid a solution with a crude step function in the Shepard diagram, because this would make $L(i)$ large. On the other hand, the regression slope must have the desired sense to make $L(o)$ small.

The configuration resulting from this mixed ordinal-linear unfolding is presented in Figure 15.1a. It allows the usual ideal-point interpretation, but differs radically from the previous interval representation in Figure 14.16a. We now observe, for example, that brewery A is very far from the attribute point 21, which, as can be seen from studying the proximities, has the usual meaning that A possesses relatively little of this property. On the other hand, we again find that the breweries form three groups, because this closeness relation remains unaffected by the slope of the regression line.

It should be noted that, even though the loss criteria $L = a \cdot L(o) + b \cdot L(i)$ and $L = b \cdot L(i) + a \cdot L(o)$ are algebraically equivalent, they may lead to different results in an iterative optimization procedure. If the KYST program is used, for example, we find that if $L(o)$ appears as the first criterion in the weighted sum, then a solution like the one reported above is obtained; if $L(i)$ is the first criterion, then the approach does not work as desired. In other words, a solution with a Shepard diagram like Figure 14.14 results, where the monotone regression curve is a horizontal straight line. In general, such differences can result from various features of the optimization method.

Augmenting the Within-Objects Blocks

A second way to avoid a trivial solution in unfolding by "changing the data," builds on the idea that unfolding is equivalent to MDS with missing data as visualized in Figure 14.1. The main idea here is to augment the data matrix with one or both of the missing "within"-sets data. Steverink, Van der Kloot, and Heiser (2002) proposed to insert Kemeny distances for the within-individuals data. In addition, they allow for different transformations within the blocks of the data matrix. The choice of transformation is critical: it must exclude the possibility of zero within-sets disparities and constant between-sets disparities to avoid the trivial equal-distances solution. For example, ordinal transformations for the between-sets proximities should be combined with the absolute transformation for the within-persons proximities to guarantee avoiding the trivial solution.

Kemeny distances for the within-persons data appear particularly suitable for unfolding preferential choice data. They are derived from preference rankings as follows. First, each person i gets a score for each pair of items

TABLE 15.1. Illustration of computing the Kemeny distance of two persons.

Pair k,l	$z_1(k,l)$	$z_2(k,l)$	$\lvert z_i(k,l) - z_j(k,l)\rvert$
AB	1	1	0
AC	1	-1	2
BC	1	-1	2
		Sum	4

TABLE 15.2. Illustrative example of an unfolding data matrix where the within-persons data are Kemeny distances. Note that the between-sets data values are the preference orders of the persons for the objects.

	A	B	C	1	2
A	–	–	–	3	2
B	–	–	–	2	3
C	–	–	–	1	1
1	3	2	1	0	4
2	2	1	3	4	0

k and l on the function $z_i(k,l)$:

$$\begin{array}{rl} 1 & \text{if } A > B \quad (\text{person } i \text{ prefers } A \text{ over } B), \\ 0 & \text{if } A = B \quad (\text{person } i \text{ is indifferent to } A \text{ and } B), \\ -1 & \text{if } A < B \quad (\text{person } i \text{ prefers } B \text{ over } A). \end{array}$$

Second, these z_i-scores are aggregated over all pairs to yield the Kemeny distance between persons i and j:

$$d_{\text{Kem}}(i,j) = \sum_{k<l} |z_i(k,l) - z_j(k,l)|. \tag{15.2}$$

As an illustration, consider a situation where three objects A, B, and C are judged by two persons: the preference rank-order of person 1 is $A > B > C$, and $C > A > B$ of person 2. Then, there are only three possible pairs of objects, that is, AB, AC, and BC. Table 15.1 shows the steps taken to compute their Kemeny distance, which equals 4 in this case. For this mini example, the data matrix augmented by within-persons distances is presented in Table 15.2.

The augmentation approach described above was applied to the brewery data, where the between-sets similarities were transformed ordinally (and unconditionally) and Kemeny distances were computed among the 26 attributes. The results are presented in Figure 15.2. As predicted, the trivial solution with equal distances does not occur. The breweries are located in three clusters in the center, a solution that is similar to that of the ordinal-interval approach in Figure 15.1. The right panel of Figure 15.2 shows the

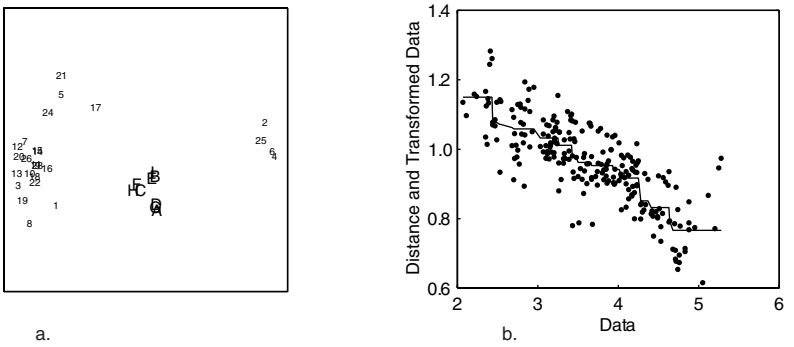


FIGURE 15.2. Unfolding representation (a) of data in Table 14.2, using the augmentation approach, and its Shepard diagram (b) for the between-sets data.

Shepard diagram of the between-sets data. The ordinal transformation is quite reasonable and does not have big jumps. However, the scatter of the distances about the regression curve is somewhat high, indicating that not all points are fitted perfectly. We also see that the disparities range from about .75 to 1.15. This means that even a brewery with the highest score on an attribute will be located at a moderate distance from the attribute. This aspect is shown in the left panel of Figure 15.2 by the fact that all attributes are distant from the center where the breweries are located.

A problem arises when the between-blocks data are transformed row-conditionally, which is a natural option for preference rank-order data. Applying the augmentation approach will yield a proper scatter of the attributes and a cluster of brewery points on top of each other. The within-block data for the attributes are properly represented, but the between-sets data (the original preference rank-orders) are trivially represented in the same way as the degeneracy in Figure 14.7b. Steverink et al. (2002) proposed to solve this problem by augmenting the data matrix with a within-columns data block as well.

Here, we propose a different type of augmentation by a within-columns data block. As the preference rank-orders are known for each subject, one can compute city-block distances between the columns using the rank-orders as coordinates. Thus, for each row, a unidimensional distance matrix is computed between the columns. Then, taking the sum of all those distance matrices over the rows gives a city-block distance matrix between the column objects. We take two additional steps. First, Steverink et al. (2002) indicate that the Kemeny distance can also be seen as a city-block distance matrix. Because we are fitting these data by Euclidean distances, we transform both within-blocks to Euclidean distances matrices by simply taking the square root of all elements (for a rationale, see Gower & Legendre, 1986). The second step involves making the range of the values in the two within-blocks equal. This adaptation is important because the

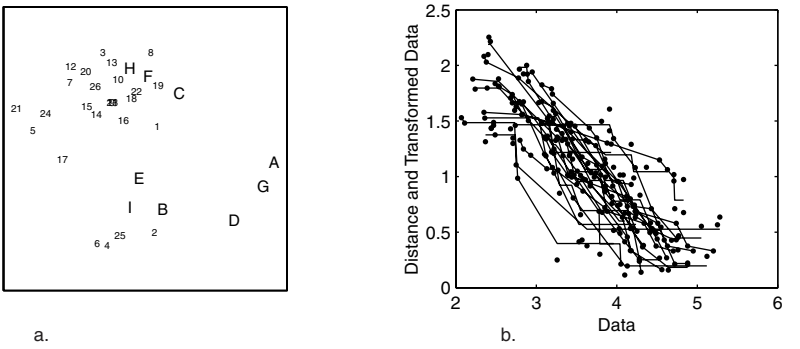


FIGURE 15.3. Ordinal row-conditional unfolding representation (a) of data in Table 14.2, augmentation both within blocks, and its Shepard diagram (b) for the between-sets data.

within-blocks are not transformed and it makes sure that the ranges of distances for the two sets of points are equivalent. Therefore, we divide each within-block by its maximum value.

The proposed procedure of augmenting both within-blocks is applied to the brewery data in Figure 15.3. In the analysis, the within-blocks data were not transformed but the between-sets data were obtained by an ordinal row-conditional transformation. The Shepard diagram in the right panel of Figure 15.3 shows that the transformations are far from constant. A similar pattern as before emerges for the configuration (Figure 15.3a), with three clusters of breweries. Note that in this analysis we may explicitly interpret the distances between all points and not only the between-sets distances because we have (generated) data for all dissimilarities.

A disadvantage of the augmentation approach proposed above is that it may be seen as doing two separate metric MDS analyses on the within-blocks data. The between-sets data are of minor importance and merely determine the translation of one of the sets with respect to the other. On the other hand, all three blocks of the data use the same rank-order information of the between-sets data. More experience with this approach is needed to see how well it performs in practice.

Other suggestions to fill the within-blocks data have been proposed by Rabinowitz (1976), Heiser and De Leeuw (1979), and Van Deun, Heiser, and Delbeke (2004).

15.2 Adjusting the Transformation

A different way to avoid the trivial equal-distances unfolding solution is to restrict the transformation so that the nonzero intercept and zero slope transformation is excluded. This goal could be either achieved by a bound

on the intercept or some restriction on the slope that excludes a slope of zero. One obvious transformation satisfying this restriction is the ratio transformation. Clearly, no intercept is estimated and the slope is equal to one, so that the zero slope and nonzero intercept cannot occur. It is a simple manner to avoid the trivial solution in unfolding, but it may not recognize the ordinal nature of data that are often used in unfolding, such as preference rank-orders.

A variant of this idea was proposed by Kim, Rangaswamy, and DeSarbo (1999), who use a two-step procedure. In their first step, they preprocess the original dissimilarities by a transformation (λ_{ij}) of the original data. These λ_{ij}s satisfy several properties. First, λ_{ij} should be strictly monotone with the dissimilarities such as, for example, a linear or a strictly ordinal transformation. Second, λ_{ij} for the most preferred item in each row is set to zero. Third, the transformations are the same for all rows. The form of the transformation is left to the user, as long as it satisfies the three conditions stated above. In the second step, after this preprocessing of the data, the λ_{ij}s are used as input data in Stress, allowing for row-conditional ratio transformations.

The reason why this approach avoids the trivial solution is that the d-hats of each row cannot become the same constant, as the d-hat corresponding to the most preferred stimulus per row is equal to zero and the remaining d-hats per row necessarily are nonzero because of the ratio transformation. It should be said, though, that this method has some arbitrariness in the way the user specifies the λ_{ij}s. Different specifications of the λ_{ij}s for the same data will lead to different solutions.

For preference rank-orders, it is more preferable to apply a transformation that has more freedom than the ratio transformation but is still able to avoid the nonzero constant and zero slope transformation. The smoothed monotone regression approach of Heiser (1985, 1989a) can do this (see also Section 9.2). The basic idea is that the absolute difference of the differences $\hat{d}_k - \hat{d}_{k-1}$ and $\hat{d}_{k-1} - \hat{d}_{k-2}$ in the transformation should be smaller than the average d-hat. Note that for $k = 1$ and $k = 2$, there will be references to nonexisting elements $\hat{d}_0$ and $\hat{d}_{-1}$ that are substituted by zero. The important consequence of this substitution is that the smallest d-hat cannot be larger than the average d-hat. Thus, this approach has an internal upper bound on the smallest d-hat, while restricting the size of the steps that can be made in the transformation. These restrictions combined with the requirement that the sum of squared d-hats are equal to some nonzero constant assure that the constant d-hat solution is excluded so that the trivial solution cannot occur.

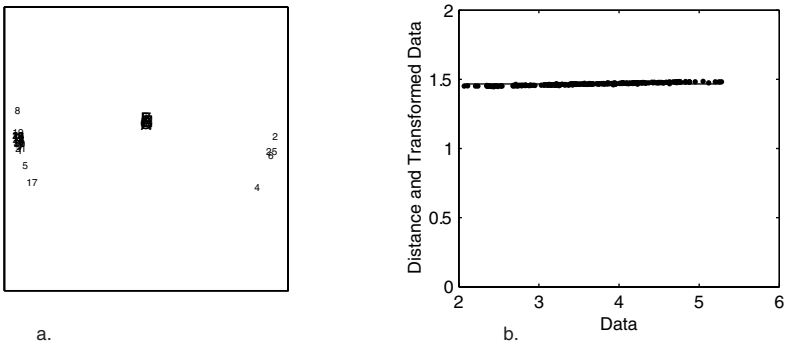


FIGURE 15.4. A degenerated linear unfolding solution obtained by KYST (Panel a) of the brewery data in Table 14.2. Panel b displays the Shepard diagram).

15.3 Adjustments to the Loss Function

Several authors have tried to avoid the trivial unfolding solution by adjusting the loss function. The first proposal was to use Stress-2; that is,

$$\sigma_2(\mathbf{X}) = \left(\frac{\sum_{i<j}[\delta_{ij} - d_{ij}(\mathbf{X})]^2}{\sum_{i<j}[d_{ij}(\mathbf{X}) - \bar{d}]^2} \right)^{1/2}.$$

The denominator of $\sigma_2(\mathbf{X})$ measures the variance of the distances about the mean distance. Therefore, the denominator will be close to zero if all distances are almost the same. This implies that if the distances become similar during the iterations, $\sigma_2(\mathbf{X})$ becomes larger and larger. Hence, equal-distances solutions should be avoided.

Ordinal unfolding by the KYST program using Stress-2 resulted in a configuration with three clusters of breweries and attributes located at two different distances (see Figure 14.13). Although the ordinal solution may not be totally satisfactory, it certainly does not display the equal-distances solution. However, a linear transformation with KYST (with strict convergence settings) does yield a constant distance solution (see Figure 15.4). To understand why this happens, we need to consider both the numerator and denominator of Stress-2 as the distances become almost equal. In that case, both the denominator and the numerator approach zero, so that no immediate conclusions can be drawn about the behavior of Stress-2. Mathematical analysis should bring more insight into this situation. We get back to this issue in the next chapter. For now it suffices to remark that apparently linear unfolding using Stress-2 does not avoid equal distances. Stress-2 may stay away from the trivial equal distance solution but it is not guaranteed to do so.

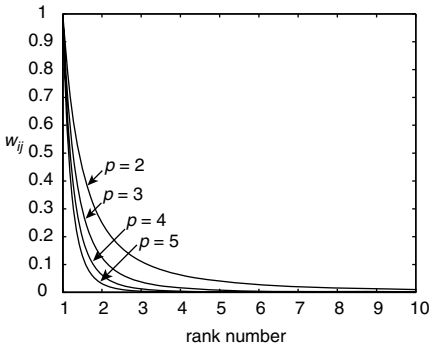


FIGURE 15.5. Influence of a weight w_{ij} as a function of the rank-order as proposed by DeSarbo and Rao (1984).

Weighting Strategies

DeSarbo and Rao (1984) proposed to use specialized weighting schemes in raw Stress to avoid the trivial solution: $w_{ij} = \delta_{ij}^{-p}$, where δ_{ij} is a ranking number (1 = most preferred) and $p > 0$ determines the influence of the object to the Stress function. Figure 15.5 shows how much an object contributes to Stress as a function of its ranking number for $p = 2, \ldots, 5$. In Figure 15.5, we see that even for $p = 2$, the residuals of the second most preferred object are weighted by only 25% compared to the most preferred object. An object with ranking number 3 is weighted by about 11%, and so on. Thus, even with a small p of 2, only the three most preferred objects of each individual (row) determine the solution. In the case of $p = 5$, the second most preferred stimulus contributes only 3%. This means that for each row there is essentially only a single stimulus that contributes to Stress. As a consequence, there will be only very few effective constraints between the points of both sets so that the points can be quite freely located in space. For this approach, the quality of the solution mainly depends on the quality of the starting configuration. A similar condition is true for the transformation. Only the transformations of the first few most preferred stimuli can be interpreted; the others hardly contribute to Stress.

Contrary to its claim, the weighting method does not exclude the trivial solution. The reason is that for any transformation that allows for a zero slope and constant intercept, distances can be obtained that are equal to the intercept. For a formal proof, we refer to Busing, Groenen, and Heiser (2005).

Penalizing the Intercept

For linear unfolding, Busing (2005) proposed a simple idea to avoid a transformation with a nonzero intercept and zero slope. His idea is to add a penalty to the Stress function to avoid a large value of the intercept. This

idea can be formalized by the following loss function.

$$\sigma_i(a, b, \mathbf{X}) = \sum_{(i,j)} [a + b\delta_{ij} - d_{ij}(\mathbf{X})]^2 + \omega a^2, \tag{15.3}$$

where ω is a nonnegative value indicating the strength of the penalty. Clearly, for $\omega = 0$ the old Stress function is retained. In the limiting case of $\omega = \infty$, the intercept a will become zero and minimizing $\sigma_i(a, b, \mathbf{X})$ reduces to ratio unfolding.

Penalizing the intercept only makes sense for dissimilarity data. If the data are similarities, we expect a transformation with a large intercept and a negative slope, so that large similarities correspond to small nonnegative d-hats and small distances, whereas small similarities correspond to large nonnegative d-hats and large distances. This means that the intercept is expected to be large, which contradicts the idea of penalizing the intercept. To overcome this problem, the similarities have to be transformed into dissimilarities before applying the current approach. As a consequence, the Shepard diagram will be increasing because dissimilarities are used in (15.3).

Penalizing the intercept is not applicable to just any transformation. For example, the approach is not effective for ordinal unfolding, because the transformation is free to find a constant transformation for all but the smallest dissimilarity. However, penalizing the intercept can be effective for spline transformations (see Section 9.6), provided that the spline is quite restricted.

We applied this approach to the brewery data of Table 14.2. Because the data (p_{ij}) in this case are similarity ratings from 1 = not true to 6 = very true, they had to be transformed into dissimilarities first. This was done by setting $\delta_{ij} = 7 - p_{ij}$ so that the dissimilarities were again in the range from 1 to 6, where 1 now indicates "very true" and 6 "not true". In this application, ω was set to 5 after some experimentation. The results are presented in Figure 15.6. Again we see the split of the breweries into the three different clusters that turn up in the other solutions as well. The Shepard diagram in panel b of Figure 15.6 is increasing indeed and has an intercept that is reasonably small compared to the slope. We may conclude that for linear unfolding, the simple approach of penalizing the intercept is effective to avoid the trivial solution.

PREFSCAL: *Penalizing Equal d-hats*

Another penalty approach was taken by Busing et al. (2005). As trivial unfolding solutions are characterized by constant d-hats, one obvious way to avoid a trivial solution is penalizing the Stress function for equal d-hats. An advantage of this approach is that all standard transformations (see Chapter 9) can be applied. Also, the resulting unfolding configuration can be interpreted in terms of the ideal point model.

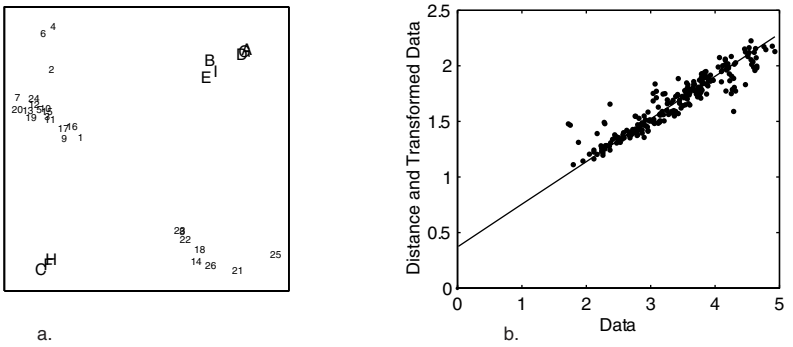


FIGURE 15.6. Linear unfolding representation (a) of data in Table 14.2, by penalizing the size of the intercept, and its Shepard diagram (b).

To identify constant d-hats, Busing et al. (2005) suggests using the variation coefficient of Pearson (1896), which is defined by

$$\nu(\hat{\mathbf{d}}) = \frac{\text{standard deviation}(\hat{\mathbf{d}})}{\text{mean}(\hat{\mathbf{d}})} = \frac{\left(K^{-1}\sum_k (\hat{d}_k - \bar{\hat{d}})^2\right)^{1/2}}{K^{-1}\sum_k \hat{d}_k}, \qquad (15.4)$$

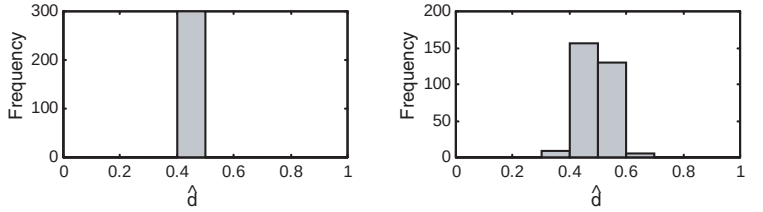
where $\bar{\hat{d}} = K^{-1}\sum_k \hat{d}_k$ and k is an index that runs over all d-hats. The coefficient of variation is a measure that indicates the spread with respect to the mean. It can be derived that $\nu(\hat{\mathbf{d}})$ is independent of the scale of $\hat{\mathbf{d}}$, so that $\nu(\hat{\mathbf{d}}) = \nu(a\hat{\mathbf{d}})$ for any $a > 0$.

To see what the variation coefficient does, we simulated four different distributions of 300 d-hats, varying the mean, the standard deviation, and the modality. Both from Figure 15.7a and from (15.4) it can be seen that a zero standard deviation yields a zero variation coefficient. If the spread around the mean is small relative to the mean, then $\nu(\hat{\mathbf{d}})$ is also small (panel b. of Figure 15.7). As the spread around the mean gets larger relative to the mean, then $\nu(\hat{\mathbf{d}})$ also increases (Figures 15.7c and 15.7d). A maximum value of $\nu(\hat{\mathbf{d}}) = (K-1)^{1/2}$ is attained if all but one of the d-hats are zero.

The variation coefficient can be used as a diagnostic for identifying solutions with constant d-hats. The PREFSCAL model proposed by Busing et al. (2005) exploits this diagnostic by using it as a penalty. To be more precise, their PREFSCAL model minimizes penalized Stress that is defined as

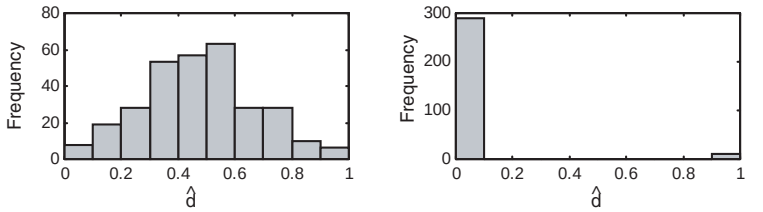
$$\sigma_p(\hat{\mathbf{d}}, \mathbf{X}) = \sigma_n^{\lambda}(\hat{\mathbf{d}}, \mathbf{X}) \left(1 + \frac{\omega}{\nu^2(\hat{\mathbf{d}})}\right), \qquad (15.5)$$

where $\sigma_n(\hat{\mathbf{d}}, \mathbf{X})$ is normalized Stress defined by (11.1) and λ and ω are two penalty parameters to be specified under the restrictions $0 < \lambda < 1$ and $\omega > 0$. The parameter λ is called a lack-of-penalty parameter that



a. std $\ll$ mean
std = 0, mean = .50, $\nu = 0$

b. std $<$ mean
std = .05, mean = .49, $\nu = .10$



c. std $\approx$ mean
std = .20, mean = .48, $\nu = .41$

d. std $\gg$ mean
std = .18, mean = .03, $\nu = 5.39$

FIGURE 15.7. Value of the variation coefficient ν as a function of the mean and the standard deviation of four hypothetical distributions of 300 $\hat{d}$s.

influences the balance between the penalty $1 + \omega\nu^{-2}(\hat{\mathbf{d}})$ and $\sigma_n(\hat{\mathbf{d}}, \mathbf{X})$: the closer λ gets to zero, the stronger the penalty. The parameter ω determines when the penalty gets active: for small ω, say, $\omega = .1$, the $\sigma_p(\hat{\mathbf{d}}, \mathbf{X})$ will hardly be influenced by the penalty for d-hats as in Figure 15.7b, whereas a large ω, say, $\omega = 5$, ensures strong influence of the penalty for the same d-hats. Based on extensive simulations, Busing et al. (2005) recommend choosing $\lambda = .5$ and a value of $\omega = .5$, although ω may need some fine tuning depending on the data.

The penalty term in (15.5) obtains high values whenever almost equal d-hats occur (thus when $\nu^2(\hat{\mathbf{d}})$ is close to zero), because the inverse of the squared variation coefficient, $\nu^{-2}(\hat{\mathbf{d}})$, will become large. Thus, when minimizing $\sigma_p(\hat{\mathbf{d}}, \mathbf{X})$, the algorithm will stay away from constant d-hats, because $\sigma_p(\hat{\mathbf{d}}, \mathbf{X})$ has high values for those d-hats. Penalized Stress has the additional advantage that as we move away from the trivial solution, the penalty term becomes less influential and $\sigma_n(\hat{\mathbf{d}}, \mathbf{X})$ will dominate the minimization. The cause of this property lies in the sum of one plus $\omega\nu^{-2}(\hat{\mathbf{d}})$. Thus, whenever $\nu(\hat{\mathbf{d}})$ is large, $\nu^{-2}(\hat{\mathbf{d}})$ gets close to zero, so that the entire penalty term is close to one. Then, the minimization of $\sigma_n(\hat{\mathbf{d}}, \mathbf{X})$ is the most important part and the penalty term will hardly influence the minimization. An additional advantage of the definition of penalized Stress is that $\sigma_p(\hat{\mathbf{d}}, \mathbf{X}) = 0$ for perfect nontrivial solutions (i.e., solutions with zero

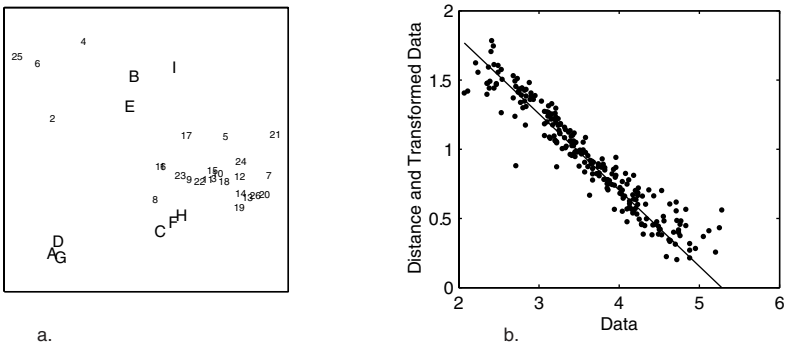


FIGURE 15.8. Linear unfolding representation (a) of data in Table 14.2 obtained by PREFSCAL, and its Shepard diagram (b).

normalized Stress). Thus, if a perfect nontrivial solution exists, penalized Stress should be able to find it. Another property of $\sigma_p(\hat{\mathbf{d}}, \mathbf{X})$ is that the minimum of $\sigma_p(\hat{\mathbf{d}}, \mathbf{X})$ is independent of the scale of $\mathbf{X}$ or $\hat{\mathbf{d}}$, by which we mean that multiplying both $\mathbf{X}$ and $\hat{\mathbf{d}}$ by a positive constant a does not change the value of σ_p. Without the property of scale independence, penalized Stress would be sensitive to the size of the unfolding problem. Thus, the PREFSCAL penalty parameters λ and ω are independent of the number of row and column objects and of the normalization of the d-hats.

Figure 15.8 displays the results of a PREFSCAL analysis on the brewery data. The PREFSCAL solution is quite similar to Figure 15.1 obtained by the ordinal-interval approach. Again, the three clusters with three breweries each emerge. However, there are some differences in the positioning of the attributes. For example, in Figure 15.1 attribute 8 is located outside the triangle spanned by the three clusters, whereas PREFSCAL locates it inside the triangle.

For row conditional transformations, constant d-hats should be avoided for each row. Therefore, the penalty should be large whenever the d-hats of a single row become constant. PREFSCAL achieves this objective by defining row conditional penalized Stress as

$$\sigma_{p.rc}(\hat{\mathbf{d}}, \mathbf{X}) = \sigma_n^{\lambda}(\hat{\mathbf{d}}, \mathbf{X}) n_2^{-1} \sum_{i=1}^{n_2} \left(1 + \frac{\omega}{\nu^2(\hat{\mathbf{d}}_i)} \right), \tag{15.6}$$

where n_2 is the number of rows in the unfolding problem and $\hat{\mathbf{d}}_i$ contains the d-hats for row i (Busing et al., 2005). Here, too, $\nu^{-2}(\hat{\mathbf{d}}_i)$ becomes large as the d-hats of row i become constant. Therefore, if any row tends to a constant, then the penalty term $n_2^{-1} \sum_{i=1}^{n_2} [1 + \omega \nu^{-2}(\hat{\mathbf{d}}_i)]$ becomes large.

To see how well PREFSCAL performs in conditional unfolding, we allowed separate transformations for each attribute of the brewery data (the rows in Table 14.2). We specified a monotone spline transformation of the sec-

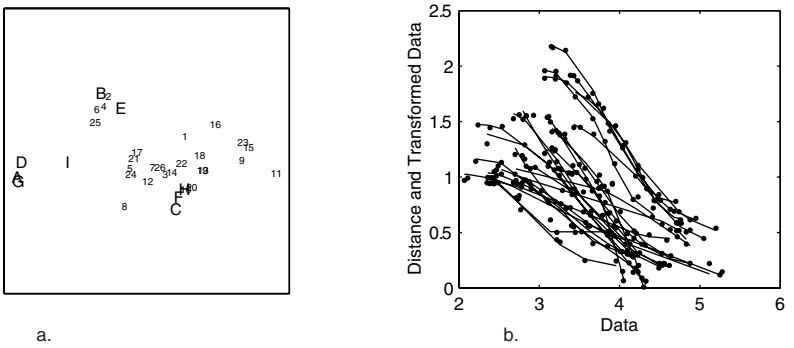


FIGURE 15.9. Representation of row conditional unfolding of data in Table 14.2 obtained by PREFSCAL(panel a), and its Shepard diagram (panel b).

ond degree with one interior knot, which is less restrictive than a linear or quadratic transformation, but more restrictive than an ordinal transformation. A striking feature of the solution (left panel of Figure 15.9) is that the breweries are located among the attributes, whereas in the other solutions discussed so far there is a clear separation of the breweries and the attributes. The right panel of Figure 15.9 contains the combined Shepard diagram of all attributes. It can be seen that the fit is high (Kruskal's Stress-1 is .0001) because most of the points are closely located to the a curve indicating that the difference between distance and d-hat will be small for these brewery and attribute pairs. The transformation curves generally are smooth and have variation coefficients markedly different from zero. Therefore, they are obviously not horizontal and not degenerated.

At the time of writing, PREFSCAL is scheduled to appear in SPSS in 2005. However, in the PREFSCAL program in SPSS, the row-conditional penalized Stress is defined slightly different from (15.6). In the program, $\sigma_n^{\lambda}(\hat{\mathbf{d}}, \mathbf{X})$ in (15.6) is replaced by an implicitly normalized form of Stress for each of the rows (Busing, 2004); that is, the PREFSCAL program in SPSS minimizes

$$\left(n^{-1} \sum_i \frac{\|\hat{\mathbf{d}}_i - \mathbf{d}_i\|^2}{\|\hat{\mathbf{d}}_i\|^2} \right)^{\lambda} n_2^{-1} \sum_{i=1}^{n_2} \left(1 + \frac{\omega}{\nu^2(\hat{\mathbf{d}}_i)} \right). \tag{15.7}$$

The reason for this difference is that it is computationally more convenient and can handle additional constraints on the configuration more easily. Both (15.6) and (15.7) are otherwise the same.

15.4 Summary

To give an overview of the quality and main properties of the methods discussed in this chapter, we have constructed Table 15.3. Most of the

TABLE 15.3. Comparison of approaches aimed at avoiding trivial solutions.

	Un-conditional	Row-conditional	Transformation	Trivial Solution Excluded	Quality
Adjusting Data					
Ratio-ordinal	+	+	Ordinal	Yes	+
Interval-ordinal	+	+	Ordinal	No	+/–
Augmenting within-persons block	+	–	All for between-sets ratio for within-sets	No	+/–
Augmenting both within-sets	+	+	All for between-sets ratio for within-sets	Yes	+
Adjusting the Transformation					
Ratio transformation	+	+	Ratio	Yes	+
Approach of Kim et al. (1999)	+	+	Ratio	Yes	+
Smoothed monotone regression	+	+	Restricted ordinal	Yes	+
Adjusting the Loss Function					
Stress-2	+	+	All	No	–
Weighting approach by DeSarbo	+	+	All	No	–
Penalizing the intercept	+	+	Interval	Yes	+
Penalized Stress by PREFSCAL	+	+	All	Yes	+

methods either have limited applicability and may depend highly on the software that is available. For example, the ratio-ordinal or the augmentation method both require that different types of transformations can be specified for the data. KYST can do that, but other programs cannot. Some of the methods discussed use forms of ratio transformations that may not be suited for preference rank-order data. The most promising approach to fit the ideal point model for unfolding seems to be the PREFSCAL model that gives good quality solutions for all standard transformations used in MDS.

When applying one of the methods for unfolding described in this chapter, one caution is needed. It is our experience that convergence criteria of the unfolding algorithms have to be set much more strictly than for ordinary MDS programs. Failing to do so may lead to a premature halt of the algorithm. The obtained solution may look nontrivial at a first glance, but continuing the algorithm with stricter convergence criteria may well lead to the trivial solution. Therefore, it is wise to make the algorithm run for many iterations so that one is sure to have avoided the trivial solution.

15.5 Exercises

Exercise 15.1 Table 15.4 on p. 333 shows the average ratings of 90 students from 15 different countries for 21 nations (columns) on 18 attributes (rows). The data were collected and reported by Wish, Deutsch, and Biener (1972). The rating scales are bipolar 9-point scales such as "collectivistic vs. individualistic" (scale 2). For most scales, only one label is shown: the

other end of the scale is obvious (as in "rich", where the other scale end is "poor").

(a) To analyze these data, first use ordinal unconditional unfolding. Represent these data in a plane. The solution is most likely degenerated into points-on-circles and/or into clusters of attributes and countries, respectively.

(b) Check whether linear unfolding helps to avoid the degeneracies. Discuss how the linear unfolding solution differs from the one for ordinal unfolding.

(c) Try out some of the methods discussed in this chapter to avoid the degeneracies. For example, compute within-country proximities and within-attribute proximities. Augment the above data matrix with these coefficients, and then run unfolding on this matrix.

Exercise 15.2 Use the data from Table 14.2 to compute coefficients for the similarity of breweries and of attributes, respectively. Then run an unfolding analysis of the data matrix in Table 14.2 after "completing" it with within-breweries and with within-attributes similarities (as suggested in Figure 14.1). Do you succeed in avoiding the degeneracies observed in Figures 14.13 and 14.15, respectively?

Exercise 15.3 Consider the contingency table below that is reported by Garmize and Rychlak (1964). Its entries show the frequencies with which different persons gave particular interpretations (rows) to Rorschach inkblot pictures when induced (by role play) into one of the moods shown in the columns.

Interpretation	Fear	Anger	Depression	Love	Ambition	Security
Bat	33	10	18	1	2	6
Bear	0	0	2	0	0	0
Blood	10	5	2	1	0	0
Boot(s)	0	1	2	0	0	0
Bridge	1	0	0	0	0	0
Butterfly	0	2	1	26	5	18
Cave	7	0	13	1	4	2
Cloud(s)	2	9	30	4	1	6
Fire	5	9	1	2	1	1
Fur	0	3	4	5	5	21
Hair	0	1	1	2	0	0
Island	0	0	0	1	0	0
Mask	3	2	6	2	2	3
Mountains	2	1	4	1	18	2
Rock(s)	0	4	2	1	2	2
Smoke	1	6	1	0	1	0

(a) Unfold these data with ordinal and metric models and test out different ways to avoid degeneracies.

TABLE 15.4. Average ratings of 90 students from 15 different countries for 21 nations on 18 attributes (Wish et al., 1972).

Country	Aligned with U.S.A.	Collect.-Individualistic	Peaceful	Individual Rights	I Like	Good	Similar to Ideal	Full of Opportunity	Stable	People Satisfied	Internally United	Influential Culture	Educated People	Rich	Industrialized	Powerful	On Way Up	Large
U.S.A.	9.0	7.4	4.5	7.5	7.5	6.6	5.5	7.5	7.2	6.1	4.8	7.4	4.4	8.8	8.8	8.9	6.6	8.8
U.K.	8.5	5.8	6.7	7.9	7.7	7.2	5.8	5.8	7.7	6.6	6.9	6.8	8.1	7.2	8.3	6.5	5.3	4.0
W.Germany	8.1	5.9	5.7	6.4	6.5	6.3	4.7	6.2	7.0	6.7	6.3	5.4	7.9	7.8	8.3	7.1	7.6	5.6
France	7.2	5.6	5.8	6.1	6.4	6.1	5.4	5.7	5.4	5.6	5.1	6.4	7.9	6.4	6.6	6.1	6.0	5.6
Israel	7.4	3.1	3.3	6.4	6.1	5.7	4.4	6.2	6.5	6.6	7.6	5.4	6.8	6.2	6.1	5.9	7.6	1.9
Japan	7.2	4.9	6.4	6.6	6.8	6.9	5.2	6.7	7.4	6.5	7.2	5.9	6.6	7.4	8.3	7.0	8.0	5.0
South Africa	6.0	6.6	5.1	2.9	3.6	3.4	2.2	2.8	4.6	3.4	2.8	3.2	6.9	6.3	5.6	4.7	4.7	5.7
Greece	6.9	5.9	6.2	3.2	6.4	4.8	3.6	3.7	3.5	3.8	4.2	6.2	3.4	3.6	3.7	3.0	4.5	3.0
Spain	6.6	6.1	6.4	3.4	5.5	4.2	2.9	3.5	5.5	4.4	4.5	5.0	4.6	3.5	3.9	3.1	4.3	4.6
Brazil	6.4	5.6	6.8	4.4	6.5	5.2	3.1	3.9	3.6	3.6	4.3	3.6	3.0	4.2	3.8	3.8	5.9	7.2
Mexico	6.8	5.2	7.0	4.6	6.4	5.7	3.5	4.1	5.2	4.6	5.7	4.7	3.3	3.8	4.0	3.4	5.7	5.4
Ethiopia	5.6	5.2	6.8	4.1	6.2	5.4	2.7	3.6	5.4	4.8	5.9	3.1	2.8	3.0	2.5	2.8	5.4	4.1
India	6.0	4.6	6.8	4.6	6.2	5.5	2.9	3.4	5.0	3.5	3.4	5.6	2.5	2.1	3.0	3.6	5.6	8.1
Indonesia	5.1	5.5	4.8	3.3	5.2	4.3	2.5	3.6	3.4	3.9	3.7	3.2	2.8	3.4	3.0	3.4	5.2	5.3
Congo	5.0	5.7	4.8	3.0	4.8	3.8	1.7	3.0	2.4	3.5	2.6	3.0	2.2	3.1	2.2	2.6	4.5	5.4
Egypt	3.6	4.1	3.1	3.8	5.1	4.2	2.5	3.5	4.1	4.5	5.6	5.0	3.0	3.4	3.5	3.7	4.7	5.1
China	1.1	2.0	2.4	2.1	4.2	3.9	2.7	3.2	4.5	3.8	4.2	5.2	3.4	3.1	4.7	7.0	6.4	8.7
Cuba	2.1	2.6	3.7	2.9	5.0	4.4	2.9	3.7	4.5	4.3	6.1	4.0	3.7	3.6	3.8	3.5	5.7	2.0
Yugoslavia	3.9	2.6	6.6	4.1	6.5	5.5	3.9	4.4	6.2	5.6	6.0	3.7	5.4	4.0	5.0	3.8	6.4	4.1
Poland	3.2	2.4	5.7	3.1	5.6	4.9	3.3	4.0	6.6	4.9	6.3	3.6	6.1	4.6	5.9	3.9	5.8	4.5
USSR	2.7	1.5	3.7	2.6	5.3	5.0	4.0	4.3	7.3	5.6	6.8	6.6	7.1	7.1	7.9	8.5	7.8	8.8

(b) Discuss in what ways the data could be preprocessed or weighted, noting, for example, that there are many zeros and also many very low frequencies.

(c) Check out what applying weights on the data does to your unfolding solutions.

16
Special Unfolding Models

In this chapter, some special unfolding models are discussed. First, we distinguish internal and external unfolding. In the latter case, one first derives an MDS configuration of the choice objects from proximity data and afterwards inserts ideal points to represent preference data. Then, the vector model for unfolding is introduced as a special case of the ideal-point model. In the vector model, individuals are represented by vectors and choice objects as points such that the projections of the objects on an individual's vector correspond to his or her preference scores. Then, in weighted unfolding, dimensional weights are chosen freely for each individual. A closer investigation reveals that these weights must be positive to yield a sensible model. A variant of metric unfolding is discussed that builds on the Bradley–Terry–Luce (BTL) choice theory.

16.1 External Unfolding

We now turn to *external unfolding* models. These models assume that a similarity configuration of the choice objects is given, possibly obtained from a previous MDS analysis. If we have preference data on these objects for one or more individuals, then external unfolding puts a point (*ideal point*) for each individual in this space so that the closer this point lies to a point that represents a choice object, the more this object is preferred by this individual. In an *internal unfolding* problem, by contrast, only the preference data are given, from which both the object configuration and the

ideal points have to be derived. Thus, external unfolding for the breakfast objects, say, would require a coordinate matrix on the objects A, ..., O and, in addition, preference data as in Table 14.1. The coordinate matrix could be obtained from an MDS analysis of an additional matrix of proximities for the 15 objects. Afterwards, an ideal point S would have to be embedded into this MDS configuration for each person in turn such that the distances from S to the points A, ..., O have an optimal monotonic correspondence to the preference ranks in Table 14.1.

Finding the optimal location for individual i's ideal point is straightforward. Consider the majorization algorithm for internal unfolding in Section 14.2. The coordinates for the set of objects, $\mathbf{X}_1$, are given and hence are fixed. Thus, we only have to compute iteratively the update for $\mathbf{X}_2$, the coordinates of the individuals, given by (14.2). Instead of δ_{ij} we may use $\widehat{d}_{ij}$ to allow for admissibly transformed preference values of individual i with respect to object j. In this case, we do not have to be concerned about degenerate solutions, because the coordinates of the objects are fixed. Because the distances among the individuals do not represent any data (the within-individuals proximities are missing in external unfolding), the individuals' points can be computed one at a time or simultaneously without giving different solutions. However, if the coordinates of the individuals are fixed and we use external unfolding to determine the coordinates of the objects, then the trivial solution in Figure 14.7b can occur in which all objects collapse in one point.

In Figure 14.1, we saw that unfolding can be viewed as MDS of two sets of points (represented by the coordinates in $\mathbf{X}_1$ for the individuals and $\mathbf{X}_2$ for the objects), where the within-sets proximities are missing. Additionally, in external unfolding, $\mathbf{X}_1$ (or $\mathbf{X}_2$) is fixed. Groenen (1993) elaborates on this idea to identify special cases for MDS on two sets of objects. Table 16.1 shows some relations of the MDS models. For example, if the proximity weights w_{ij} of the within-individuals and within-objects proximities are nonmissing ($\mathbf{W}_{11} \neq \mathbf{0}$, $\mathbf{W}_{22} \neq \mathbf{0}$), and all coordinates of $\mathbf{X}_1$ and $\mathbf{X}_2$ are free, then the model is full MDS. But if $\mathbf{W}_{11} = \mathbf{0}$ and $\mathbf{W}_{22} = \mathbf{0}$, we have (internal) unfolding. For *almost complete* MDS we have one of the within-blocks portion of the data matrix missing. (Therefore, almost complete MDS appears twice in Table 16.1.) It is *semi-complete* MDS if additionally $\mathbf{X}_1$ is fixed, so that the between-blocks and within-objects proximities are fitted by $\mathbf{X}_2$ for given $\mathbf{X}_1$. Note that for fixed $\mathbf{X}_1$, $\mathbf{W}_{11}$ is immaterial and $\mathbf{W}_{22}$ determines the model.

16.2 The Vector Model of Unfolding

The ideal-point model for unfolding has a popular companion, the *vector model* of unfolding, which goes back to Tucker (1960). It differs from the

TABLE 16.1. Relation of unfolding, external unfolding, and MDS using the partitioning in two sets, $\mathbf{X}_1$ (objects) $\mathbf{X}_2$ (individuals), as in Figure 14.1. We assume that $\mathbf{X}_2$ is always free and $\mathbf{W}_{12} \neq \mathbf{0}$.

			Model
$\mathbf{X}_1$ Free	$\mathbf{W}_{11} = \mathbf{0}$	$\mathbf{W}_{22} = \mathbf{0}$	Unfolding
$\mathbf{X}_1$ Free	$\mathbf{W}_{11} = \mathbf{0}$	$\mathbf{W}_{22} \neq \mathbf{0}$	Almost complete MDS
$\mathbf{X}_1$ Free	$\mathbf{W}_{11} \neq \mathbf{0}$	$\mathbf{W}_{22} = \mathbf{0}$	Almost complete MDS
$\mathbf{X}_1$ Free	$\mathbf{W}_{11} \neq \mathbf{0}$	$\mathbf{W}_{22} \neq \mathbf{0}$	Complete MDS
$\mathbf{X}_1$ Fixed	$\mathbf{W}_{11} = \mathbf{0}$	$\mathbf{W}_{22} = \mathbf{0}$	External unfolding
$\mathbf{X}_1$ Fixed	$\mathbf{W}_{11} = \mathbf{0}$	$\mathbf{W}_{22} \neq \mathbf{0}$	Semi-complete MDS
$\mathbf{X}_1$ Fixed	$\mathbf{W}_{11} \neq \mathbf{0}$	$\mathbf{W}_{22} = \mathbf{0}$	External unfolding
$\mathbf{X}_1$ Fixed	$\mathbf{W}_{11} \neq \mathbf{0}$	$\mathbf{W}_{22} \neq \mathbf{0}$	Semi-complete MDS

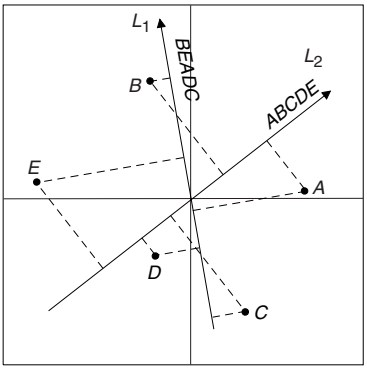


FIGURE 16.1. Illustration of the vector model of unfolding; L_1 and L_2 represent two individuals.

ideal-point model in representing each individual i not by a point but by a directed line (segment), a vector. From now on, we switch to the notation $\mathbf{X}$ for the objects and $\mathbf{Y}$ for the individuals.

Representing Individuals by Preference Vectors

For each individual i, a linear combination of the coordinate vectors of $\mathbf{X}$ is to be found so that it corresponds as much as possible to the preference data $\mathbf{p}_i$ of this individual. Figure 16.1 should clarify the situation. The diagram shows a configuration of five choice objects (points $A, \ldots, E$) and, in addition, two preference lines, L_1 and L_2.

Assume that individual i had ordered the objects as $A > B > C > D > E$ in terms of preference. Then, L_2 is a perfect (ordinal) representation of i's preferences, because the projections of the object points onto this line perfectly match i's preference rank-order.

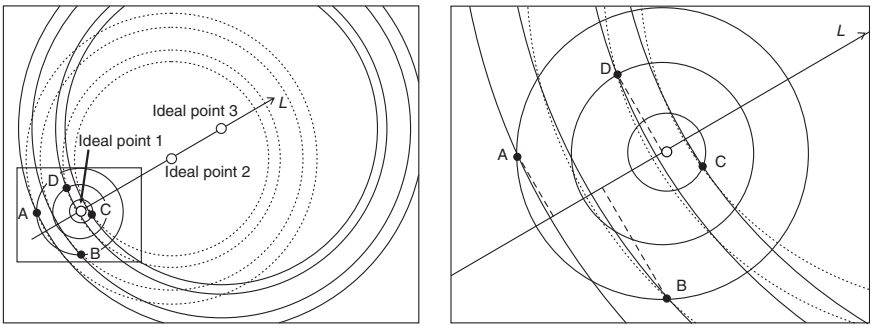


FIGURE 16.2. Illustrating the relation of the ideal-point and the vector model for preferential choice. Panel (a) shows that the more the ideal points move away from the object points (solid points) on line L, the more the ideal-point model approximates the vector model. Panel (b) zooms in on the box in panel (a). The straight dashed lines show the projections according to the vector model.

Of course, with so few points, many other lines around L_2 would be equally perfect representations for the empirical preference order $A > \cdots > E$. On the other hand, any arbitrary direction would not do, and for some preference orders (such as $D > C > E > A > B$) no perfect representation exists at all in Figure 16.1.

The vector model is, in a way, but a special case of the ideal-point model. To see this, consider Figure 16.2 and assume that person i's ideal point I is moved from ideal point 1 to 3 along the direction of vector L. As I moves away from the object points $A, \ldots, D$ in the direction of line L, the iso-preference circles grow and grow in diameter, so that the circle segments from the object points to L will become increasingly less curved. When I's distance from the centroid of the object points approaches ∞, the circle segments approximate the straight projection lines of $A, \ldots, D$ onto L (Carroll, 1972; Coombs, 1975). Expressed differently, the distances from I to the various object points approximate the distances of I to the projections of the object points onto the line L. Hence, in terms of fitting the models to data, the ideal and the vector models become very similar. However, this does not imply that the psychological models also become equivalent (Van Deun, Groenen, & Delbeke, 2005). The main difference is that in the vector model, preferences are confined to a subspace of the unfolding space (i.e., the preference vector) and any variation in the surrounding space is ignored. In a dimensional interpretation of the unfolding models, we note that in the vector model the attributes (dimensions) contribute with fixed weights to the preference function of an individual, however close or distant the object points are from line L, whereas in the ideal-point model a low score on one attribute can be compensated by a very high score on other dimensions to lead to the same projection onto L (see also Section 14.7).

Hence, the vector model and the ideal-point model imply similar decision functions only for points that are close to the vector.

Apart from this difference, the vector model also represents a particular preference notion that can be described as "the more, the better" on all dimensions. Obviously, this property does not hold in general. For example, suppose that respondents have to rate how much they like teas of various temperatures. It is certainly not true that the hotter the tea the better. The opposite (the colder, the better) is not plausible either, not even for iced tea.

Metric and Ordinal Vector Models

In a metric model, the indeterminacy of locating L_i is eliminated or at least reduced, because the distances of the projection points on L_i are also meaningful in some quantitative sense. For example, if we require that $d(B, E) = d(E, D)$ on L_i, then only a line corresponding closely to the vertical coordinate axis may be selected as a representation. But then we could conclude that this individual based his or her preference judgments on the vertical dimension only, whereas some other person whose preference line is the bisector from the lower left-hand to the upper right-hand corner used both dimensions with equal weight. Note that if we put the arrowhead at the other end of the line, the person represented by this line would still weight both dimensions equally, but now the negative, not the positive, ends of each dimension are most attractive.

Fitting the Vector Model Metrically

In the vector model, one has to find an m-dimensional space that contains two sets of elements: (a) a configuration $\mathbf{X}$ of n points that represent the objects and (b) an m-dimensional configuration $\mathbf{Y}$ of N vectors that represent the individuals. The projections of all object points onto each vector of $\mathbf{Y}$ should correspond to the given preference data in the N columns of $\mathbf{P}_{n\times N}$. The model attempts to explain individual differences of preference by different weightings of the objects' dimensions.

Formally, we have the loss function

$$L(\mathbf{X};\mathbf{Y}) \quad = \quad ||\mathbf{X}_{n\times m}\mathbf{Y}'_{m\times N} - \mathbf{P}_{n\times N}||^2. \tag{16.1}$$

Note that $\mathbf{P}$ corresponds to the upper corner matrix in Figure 14.1. The vector model is fitted by minimizing (16.1) over $\mathbf{X}$ and $\mathbf{Y}$.

The loss function can be minimized by a singular value decomposition. Let $\mathbf{P} = \mathbf{K}\mathbf{\Lambda}\mathbf{L}'$ be the SVD of $\mathbf{P}$. Then, the first m columns of $\mathbf{K}\mathbf{\Lambda}$ and of

$\mathbf{L}$ define optimal solutions for $\mathbf{X}$ and for $\mathbf{Y}$, respectively. Setting $\mathbf{X} = \mathbf{K}$ and $\mathbf{Y} = \mathbf{L}\boldsymbol{\Lambda}$ would do equally well.[1]

However, there are many more than just these two solutions. Minimizing $L(\mathbf{X};\mathbf{Y})$ by choice of $\mathbf{X}$ and $\mathbf{Y}$ does not uniquely determine particular matrices $\mathbf{X}$ and $\mathbf{Y}$. Rather, if $\mathbf{X}$ is transformed into $\mathbf{X}^* = \mathbf{X}\mathbf{M}$ by a nonsingular matrix $\mathbf{M}$, then we simply have to transform $\mathbf{Y}$ into $\mathbf{Y}^* = \mathbf{Y}(\mathbf{M}^{-1})'$ to obtain the same matrix product. Such transformations can be conceived of as rotations and stretchings along the dimensions, because $\mathbf{M}$ can be decomposed by SVD into $\mathbf{P}\boldsymbol{\Phi}\mathbf{Q}'$, where $\mathbf{P}$ and $\mathbf{Q}$ are orthonormal and $\boldsymbol{\Phi}$ is a diagonal matrix of dimension weights [see (7.14)]. Geometrically, this means, for example, that one can stretch out a planar $\mathbf{X}$ along the Y-axis (like a rubber sheet), provided $\mathbf{Y}$ is stretched out along by the same amount along the X-axis. This destroys relations of incidence, for example, and thus makes interpretation difficult.

By restricting the vectors of $\mathbf{Y}$ to the same length (1, say), the model becomes more meaningful:

$$\begin{aligned} L(\mathbf{X};\mathbf{Y}) &= ||\mathbf{X}\mathbf{Y}' - \mathbf{P}||^2, \\ \operatorname{diag}(\mathbf{Y}\mathbf{Y}') &= \operatorname{diag}(\mathbf{I}). \end{aligned} \tag{16.2}$$

The indeterminacy now reduces to a rotation; that is, $\mathbf{M}$ must satisfy $\mathbf{M}\mathbf{M}' = \mathbf{I}$, because only then does $\mathbf{Y}^* = \mathbf{Y}(\mathbf{M}^{-1})'$ satisfy the additional side constraint in formula (16.2). This rotation is unproblematic for interpretations because it affects both $\mathbf{X}$ and $\mathbf{Y}$ in the same way because $\mathbf{Y}(\mathbf{M}^{-1})' = \mathbf{Y}\mathbf{M}$ if $\mathbf{M}$ is orthonormal.

Chang and Carroll (1969) developed a popular program, MDPREF, for solving the length-restricted vector model in (16.2). It first finds an SVD of $\mathbf{P}$ and then imposes the side constraint of unit length onto $\mathbf{Y}$'s vectors. Schönemann and Borg (1983) showed that this sequential approach may be misleading. The argument is based on first deriving a direct solution for (16.2). It exists only if the data satisfy certain conditions implied by the side condition $\operatorname{diag}(\mathbf{Y}\mathbf{Y}') = \operatorname{diag}(\mathbf{I})$. Hence, (16.2) is a testable model that may or may not hold, whereas MDPREF always provides a solution.

If $\mathbf{X}$ is given, then things become very simple. The vector model for external unfolding only has to minimize (16.1) over the weights $\mathbf{Y}$. This problem is formally equivalent to one considered in Chapter 4, where we wanted to fit an external scale into an MDS configuration. If the preferences are rank-orders, then an optimal transformation also has to be computed.[2]

---

[1] In contrast to ordinary PCA, $\mathbf{P}$ has individuals as column entries and the objects as row entries. Hence, the vector model for unfolding is sometimes referred to as a "transposed PCA."

[2] This model can be fitted by the PREFMAP program (for computational details, see Carroll, 1972).

Fitting the Vector Model Ordinally

Now, suppose that $\mathbf{P}$ contains preference rank-orders. Gifi (1990) proposes to minimize the closely related problem $\sum_{i=1}^{N} ||\mathbf{X} - \widehat{\mathbf{p}}_i \mathbf{y}_i'||^2$, where $\mathbf{y}_i$ is row i of $\mathbf{Y}$ and $\widehat{\mathbf{p}}_i$ has the same rank-order as $\mathbf{p}_i$ but is optimally transformed. This resembles the strategy for conditional unfolding for the ideal-point model, except that in this case the data are treated as column conditional. To avoid the degenerate solution of $\mathbf{Y} = \mathbf{0}$, $\mathbf{X} = \mathbf{0}$, and $\widehat{\mathbf{p}}_i = \mathbf{0}$, Gifi (1990) imposes the normalization constraint $\widehat{\mathbf{p}}_i' \widehat{\mathbf{p}}_i = n$ and $\mathbf{X}'\mathbf{X} = n\mathbf{I}$. This model can be computed by the program CATPCA (categorical principal components analysis) formerly known under the name PRINCALS (nonlinear principal components analysis), both available in the SPSS package. Note that CATPCA has to be applied to the objects $\times$ individuals matrix, because the ordinal transformations are computed columnwise. More details about this and related approaches can be found in Gifi (1990).

Van Deun et al. (2005) discuss the VIPSCAL model for unfolding. This model allows some subjects to be presented by an ideal point and others by the vector model. The model also allows some length and orthogonality constraints on $\mathbf{X}$ and $\mathbf{Y}$. Special cases within VIPSCAL are the ordinary ideal point model and an (ordinal) vector model.

An Illustrative Application of the Vector Model

Consider the breakfast data in Table 14.1 again. Figure 16.3 shows the result of the vector model for unfolding obtained by CATPCA, using the preference rank-orders only. The preference vectors for every individual are scaled to have equal length, because it is the direction that matters, not the actual length. Note that high values in Table 14.1 indicate least preferred breakfast items; hence the correlations of $\widehat{\mathbf{p}}_i$ with $\mathbf{X}$ (called component loadings in CATPCA) have to be multiplied by minus one to obtain the preference vectors in Figure 16.3. The CATPCA solution indicates that there are three groups of individuals. The first group of 15 respondents is represented by the preference vectors directed away from A. This group has a strong dislike for A (toast pop-up) and does not care much about the other breakfast items either. The other groups are orthogonally related to the first group, indicating that they are indifferent to breakfast A, because A projects onto the origin. The second group is directed to the lower left-hand corner. This group prefers the breakfast items K, D, L, M, and N, and dislikes breakfast items with toast, that is, B, G, and J. The third group has the opposite preference of the second group. The interpretation of this solution is not very different from the ideal-point solution in Figure 14.2.

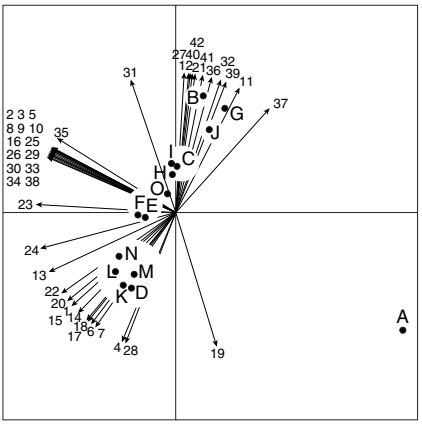


FIGURE 16.3. The vector model of unfolding of the breakfast items in Table 14.1 computed by CATPCA.

16.3 Weighted Unfolding

We now consider a generalization of external unfolding, that is, weighted unfolding (Carroll, 1980). Assume that the coordinate axes could be chosen to correspond to the dimensions that determined person i's preference judgment. It is then possible to conjecture that person i weights these dimensions in some particular way depending on how important he or she feels each dimension to be. Consider, for example, an investment problem and assume that various portfolios are distinguished with respect to risk and expected profit. All individuals agree, say, that portfolio x is riskier than y, and that y has a higher expected yield than z; that is, all individuals perceive the portfolios in the same way. But person i may be more cautious than j, so in making a preference judgment the subject weights the risk dimension more heavily than j. In other words, in making preference judgments on the basis of a common similarity space, person i stretches this space along the risk dimension, but j compresses it, and this will, of course, affect the the distances differentially. We can express such weightings of dimensions as follows.

$$\begin{aligned} d_{ij}(\mathbf{X};\mathbf{Y};\mathbf{W}) &= \left[\sum_{a=1}^{m}(w_{ia}y_{ia} - w_{ia}x_{ja})^2\right]^{1/2} \\ &= \left[\sum_{a=1}^{m} w_{ia}^2(y_{ia} - x_{ja})^2\right]^{1/2}, \end{aligned} \tag{16.3}$$

where x_{ja} is the coordinate of object j on dimension a, y_{ia} is the coordinate of the ideal points for individual i on dimension a, and w_{ia} is the weight that this individual assigns to dimension a.

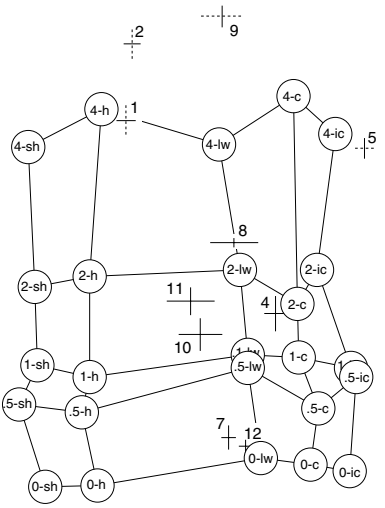


FIGURE 16.4. MDS configuration for tea proximities (circles); numbers indicate teaspoons of sugar, letters temperature of tea (sh="steaming hot", h="hot", lw="lukewarm", c="cold", ic="ice cold"). Crosses show ideal points for ten subjects; length of bars proportional to dimensional weights; dashed/solid bars indicate negative/positive weights, respectively (after Carroll, 1972).

Private Preference Spaces and Common Similarity Space

This seemingly minor modification of the distance formula has important consequences. The most obvious one is that the weighted model generally does not permit the construction of a joint space of objects and individuals in which the differences among the various individuals are represented by the different locations of the respective ideal points. Rather, each individual has his or her own *private preference space*, independent of the preference spaces for other individuals, even though they are all related to a *common similarity space* by *dimensional stretchings*. Further implications of the weighted unfolding model can be seen from the following example.

In an experiment by Wish (see Carroll, 1972), 12 subjects evaluated 25 stimuli with respect to (a) their dissimilarities and (b) their subjective values. The dissimilarity data were collected by rating each of the stimulus pairs on a scale from 0 (= identical) to 9 (= extremely different). The stimuli were verbal descriptions of tea, varying in temperature and sweetness. The proximities are represented by the MDS configuration in Figure 16.4, where the different teas are shown by circles. The configuration $\mathbf{X}$ reflects the 5×5 design of the stimuli very clearly: the horizontal axis corresponds to the temperature factor and the vertical one to the sweetness scale.

Additionally, the individuals indicated their preferences for each type of tea. These data and the fixed coordinates $\mathbf{X}$ of the stimuli are used to find the dimension weights and the ideal points for each individual i. To do this,

Carroll (1972) minimized the loss function

$$L(\mathbf{Y};\mathbf{W}) = \sum_{i=1}^{n}\sum_{j=1}^{n}\left(d_{ij}^2(\mathbf{Y};\mathbf{X};\mathbf{W}) - \delta_{ij}^2\right)^2 \tag{16.4}$$

over ideal points $\mathbf{Y}$ and dimension weights $\mathbf{W}$. $L(\mathbf{Y};\mathbf{W})$ differs from a Stress-based criterion in that it uses squared distances d_{ij}^2 for computational convenience instead of the distance d_{ij} (just as in S-Stress; see Section 11.2). $L(\mathbf{Y};\mathbf{W})$ is minimized in an alternating least-squares fashion, where the update of $\mathbf{Y}$ with $\mathbf{W}$ fixed is alternated by the update of $\mathbf{W}$ for fixed $\mathbf{Y}$, until convergence is reached.

Figure 16.4 also represents the resulting 12 private preference spaces through weight crosses on the ideal points. The scatter of the ideal points shows that the individuals differ considerably with respect to the preferred sweetness of the tea. There is much less variation on the temperature dimension, but, strangely, most individuals seem to prefer lukewarm tea, because the ideal points are concentrated mostly in the lukewarm range of the temperature dimension. On the other hand, it is not very surprising that no individual preferred steaming hot tea, and the inclusion of these choice objects might have obscured the situation in Figure 16.4, according to Carroll (1972). He therefore eliminated the steaming hot stimuli from the analysis. This led to an unfolding solution very similar to Figure 16.4 but, of course, without the "sh" points. Its ideal points were still in the lukewarm range, but now the (squared) dimension weights on the temperature dimension were negative for all individuals.

Negative Dimension Weights and Anti-Ideal Points

How are we to interpret negative dimension weights w_{ia}^2? Assume that a given object is considered "ideal" on all dimensions except for dimension a. Then, all dimensional differences are zero in (16.3), except for the one on a. If $w_{ia}^2 < 0$, the term under the square root will be negative. Hence, d_{ij}^2 is negative, and d_{ij} is an imaginary number. But then d_{ij} is not a distance, because distances are nonnegative real numbers, by definition. Thus, without any restrictions on the dimension weights, the weighted unfolding model is not a distance model.

Is such a model needed? Assume that we have a 2D configuration, with person i's ideal point at the origin, and dimension weights $w_{i1}^2 = 1$ and $w_{i2}^2 = -1$. Then, according to (16.3), all points on the bisector between dimensions 1 and 2 have distance zero to the ideal point y_i and, thus, are also ideal points. For *all* points x on, below, and above the bisector, we get $d^2(x, y_i) = 0$, $d^2(x, y_i) > 0$, and $d^2(x, y_i) < 0$, respectively. The plane thus becomes discontinuous and thereby incompatible with the ideal-point model that underlies unfolding. In such a situation, it remains unclear

what purpose further generalizations of this model might serve (Srinivasan & Shocker, 1973; Roskam, 1979b; Carroll, 1980).

Should one preserve the idea of negative dimension weights? Carroll (1972) writes: "This possibility of negative weights might be a serious problem except that a reasonable interpretation attaches to negative w's ... This interpretation is simply that if w_{it} [corresponding to our w_{ia}^2] is negative, then, with respect to dimension t, the ideal point for individual i indicates the *least preferred* rather than the most preferred value, and the farther a stimulus is *along that dimension* from the ideal point, the more highly preferred the stimulus" (p. 133). Coombs and Avrunin (1977) argue, however, that *anti-ideal points* are artifacts caused by confounding two qualitatively different sets of stimuli. For tea, they argue that one should expect single-peaked preference functions over the temperature dimension for each iced tea and for hot tea, respectively. For iced tea, each individual has some preferred coldness, and the individual's preference drops when the tea becomes warmer or colder. The same is true for hot tea, except that the ideal temperature for hot tea lies somewhere in the "hot" region of the temperature scale. Thus, iced tea and hot tea both yield single-peaked preference functions over the temperature dimension. Superimposing these functions—and thus generating a meaningless value distribution for "tea"—leads to a two-peaked function with a minimum at lukewarm.

If one restricts the dimension weights to be nonnegative, then there are two models. If zero weights are admitted, d_{ij} in (16.3) is not a distance, because it can be zero for different points. This characteristic means that one cannot interpret the formula as a psychological model saying that person i generates his or her preferences by computing weighted distances in a common similarity space. Rather, the model implies a two-step process, where the individual first weights the dimensions of the similarity space and then computes distances from the ideal point in this ("private") transformed space.

In summary, sensible dimensional weighting allows for better accounting of individual differences, but it also means giving up the joint-space property of simple unfolding. In most applications so far, it turned out that the weighted unfolding model fitted the data only marginally better, and so "relatively little appears to be gained by going beyond the simple (equal-axis weighting) ideal-point model" (Green & Rao, 1972, p.113).

16.4 Value Scales and Distances in Unfolding

We now return to internal unfolding and the simple unfolding model. So far, not much attention has been paid to the exact relationship of the distances between ideal points and object points and the subjective value of the represented objects. We simply claimed that preference strength is

linearly or monotonically related to the unfolding distances. The particular shape of the preference function was not derived from further theory. We now consider a model that does just that.

Relating Unfolding Distances to Preference Strength Data by the BTL Model

There are many proposals for modeling preference behavior and subjective value (see, e.g., Luce & Suppes, 1963). One prominent proposal is the Bradley–Terry–Luce (BTL) model (Luce, 1959). This model predicts that person i chooses an object o_j over an object o_k with a probability $p_{jk|i}$ that depends only on the pair (o_j, o_k), not on what other choice objects there are. Restricting the set of choice objects to those that are neither always chosen nor never chosen, a subjective-value scale v can be constructed for i by first selecting some object o_a as an "anchor" of the scale, and then setting

$$v_i(o_j) = \frac{p_{ja|i}}{p_{aj|i}}. \tag{16.5}$$

Conversely, pairwise choice probabilities can be derived from the ratio scale values by using

$$p_{jk|i} = \frac{v_i(o_j)}{v_i(o_j) + v_i(o_k)}. \tag{16.6}$$

Given a set of preference frequencies, it is possible to first find v-values for the choice objects and then map these values into the distances of an unfolding representation. This permits one to test a choice theory (here, the BTL theory) as a first step of data analysis. If the test rejects the choice theory, then it makes little sense to go on to unfolding, because the choice process has not been understood adequately and must be modeled differently. If, on the other hand, the test comes out positive, the distance representation has a better justification.

Luce (1961) and Krantz (1967) discuss two functions that connect the scale v with corresponding distances in the unfolding space. One would want such a function to be monotonically decreasing so that greater v-scale values are related to smaller distances. One reasonable function in this family is

$$d(x_j, y_i) = -\ln[v_i(o_j)], \tag{16.7}$$

or, expressed differently,

$$v_i(o_j) = \exp[-d(x_j, y_i)], \tag{16.8}$$

where $d(x_j, y_i)$ denotes the distance between the points x_j and y_i representing object o_j and individual i, respectively, in the unfolding space. Thus, $v_i(o_j) = \max = 1$ if $d(x_j, y_i) = 0$ (i.e., at the ideal point) and $0 < v_i(o_j) < 1$

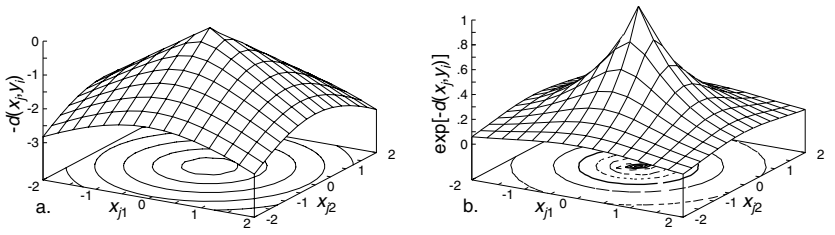


FIGURE 16.5. (a.) Distance (vertical) from ideal point y_i with coordinates (0,0) in 2D to stimulus x_j, and (b) the corresponding scale value $v_i(o_j) = \exp[-d(x_j, y_i)]$ according to (16.8).

for all other objects. Figure 16.5b shows the the preference function for individual i with ideal point (0, 0) in 2D. The circles in the horizontal plane indicate the positions of the object points with equal preference. The corresponding preference strength is shown on the vertical axis. The model defines an inverted bowl over the plane, and this bowl never touches the plane, even though it comes very close to it when we move far away from the ideal point.

A similar function is discussed by Schönemann and Wang (1972) and Wang, Schönemann, and Rusk (1975):

$$v_i(o_j) = \exp[-c \cdot d^2(x_j, y_i)], \tag{16.9}$$

where $c > 0$ is some arbitrary multiplier. Setting $c = 1$, the only difference[3] between (16.9) and (16.8) is that the distances are squared in the former case. For squared distances, the value surface over the object space is normal for each ideal point y_j. Thus, in a 2D case like the one in Figure 16.5b, the inverted bowl has the familiar bell shape. Equation (16.9) is then connected to individual i's pairwise preference probabilities $p_{jk|i}$ by using the BTL choice model. Inserting the $v_i(o_j)$ values into (16.9) yields

$$p_{jk|i} = \frac{1}{1 + \exp[d^2(x_j, y_i) - d^2(x_k, y_i)]^2}. \tag{16.10}$$

Thus, preference probabilities and (squared) distances of the unfolding space are related, according to this model, by a logistic function[4] operating on differences of (squared) distances.

---

[3]That difference, however, is critical, because it renders the model mathematically tractable so that the exact case can be solved algebraically, without iteration. The algebraic solution given in Schönemann (1970) generalizes the Young–Householder theorem to the asymmetric case.

[4]The exact form of the probability distribution is not of critical importance for fitting the model to the data. This follows from many detailed investigations on generalized Fechner scales (see, e.g., Baird & Noma, 1978), which include the logistic function as just one special case. An alternative is the normal curve, but almost any other approximately symmetrical function would do as well.

TABLE 16.2. Politicians and interviewee groups of Wang et al. (1975) study.

1. Wallace (Wal)	5. Humphrey (Hum)	9. Nixon (Nix)
2. McCarthy (McC)	6. Reagan (Rea)	10. Rockefeller (Roc)
3. Johnson (Joh)	7. Romney (Rom)	11. R. Kennedy (Ken)
4. Muskie (Mus)	8. Agnew (Agn)	12. LeMay (LeM)

	Interviewee Group	Code	N_i
1.	Black, South	BS	88
2.	Black, Non-South	BN	77
3.	White, strong Democrat, South, high ed.	SDSH	17
4.	White, strong Democrat, South, low education	SDSL	43
5.	White, weak Democrat, South, high education	WDSH	27
6.	White, weak Democrat, South, low education	WDSL	79
7.	White, strong Democrat, Non-South, high ed.	SDNH	21
8.	White, strong Democrat, Non-South, low ed.	SDNL	85
9.	White, weak Democrat, Non-South, high ed.	WDNH	65
10.	White, weak Democrat, Non-South, low ed.	WDNL	180
11.	White, Independent, South, high education	ISH	8
12.	White, Independent, South, low education	ISL	27
13.	White, Independent, Non-South, high ed.	INH	25
14.	White, Independent, Non-South, low ed.	INL	46
15.	White, strong Republican, South, low ed.	SRSL	13
16.	White, strong Republican, Non-South, high ed.	SRNH	40
17.	White, strong Republican, Non-South, low ed.	SRNL	60
18.	White, weak Republican, South, high ed.	WRSH	34
19.	White, weak Republican, South, low ed.	WRSL	36
20.	White, weak Republican, Non-South, high ed.	WRNH	90
21.	White, weak Republican, Non-South, low ed.	WRNL	117

An Application of Schönemann and Wang's BTL Model

Consider an application. Wang et al. (1975) analyzed data collected in 1968 on 1178 persons who were asked to evaluate 12 candidates for the presidency on a rating scale from 0 (= very cold or unfavorable feeling for the candidate) to 100 (= very warm or favorable feeling toward the candidate) (Rabinowitz, 1975). The respondents were classified into 21 groups according to their race, party preference, geographical region, and education. The 21 groups and the 12 candidates are listed in Table 16.2. Twenty-one 12×12 preference matrices were derived from the rating values of the respondents in each group. The $p_{jk|i}$ values (where i indicates the group $i = 1, \ldots, 21$) were computed as the relative frequencies with which candidate o_j's rating score was higher than the score for candidate o_k.

The least-squares BTL scale values for the 12 candidates and the 21 groups are shown in Table 16.3. It turned out that these scale values accounted for the probabilities sufficiently well; that is, it is possible to approximately reconstruct the $\binom{12}{2}$ probability data from the 12 scale values for each group. By taking the logarithm of both sides of (16.9), the v-values can be transformed into squared distances, which in turn are the dissimilarities for our unfolding analysis. Wang et al. (1975) then employed an iterative optimization method for finding an unfolding configuration. (Of course, the internal unfolding solution could also be computed by the majorization algorithm in Section 14.2.) The final fit to the $i = 1, \ldots, 21$ em-

TABLE 16.3. BTL scale values for interviewee groups and politicians from Table 16.2.

	Wal	Hum	Nix	McC	Rea	Roc	Joh	Rom	Ken	Mus	Agn	LeM
BS	.11	9.00	.95	.75	.28	.66	9.10	.51	21.70	1.52	.35	.14
BN	.03	12.09	.95	1.27	.29	1.78	7.54	.58	16.02	2.15	.26	.11
SDSH	.49	3.82	1.00	1.17	.42	.94	1.48	.45	1.89	3.43	.62	.43
SDSL	.86	2.64	1.09	.58	.46	.70	2.54	.53	1.89	1.70	.74	.67
WDSH	.72	1.08	2.82	1.01	.84	1.21	1.27	.56	1.37	1.52	.69	.44
WDSL	1.24	1.20	2.30	.76	.68	.66	1.12	.64	1.45	.93	1.03	.86
SDNH	.09	4.72	1.11	1.67	.40	1.07	2.64	1.23	5.92	4.44	.38	.09
SDNL	.26	3.80	.95	.86	.43	.81	3.12	.64	5.99	2.38	.49	.25
WDNH	.12	2.99	1.46	1.68	.42	1.61	1.54	.92	5.13	2.46	.49	.19
WDNL	.37	1.99	1.57	.98	.59	.82	1.58	.68	3.69	1.71	.70	.40
ISH	.43	1.24	4.07	.76	.89	.97	.88	.60	2.88	1.10	1.34	.31
ISL	6.68	.90	3.40	.87	.83	.86	.95	.51	2.53	.87	1.03	.71
INH	.43	1.77	2.54	1.49	.66	.84	1.01	.77	1.69	1.80	.88	.30
INL	6.37	1.48	2.30	1.13	.74	.95	1.12	.71	2.66	1.82	.83	.31
SRSL	.11	.66	20.37	.55	1.43	.82	.86	.78	1.93	.77	3.20	.34
SRNH	.16	.55	14.29	1.05	1.98	1.44	.54	1.29	1.26	.98	1.19	.26
SRNL	.28	.62	8.49	1.02	1.39	1.02	.61	.95	1.28	.87	1.78	.41
WRSH	.76	.45	7.53	.64	1.78	.99	.82	.65	.86	.82	1.15	.80
WRSL	.85	.56	5.23	1.07	1.03	1.10	.78	.62	1.55	.55	1.23	.66
WRNH	.28	.89	4.78	1.56	1.12	1.46	.68	.75	1.38	1.34	1.01	.34
WRNL	.33	.86	5.84	1.06	1.08	1.06	.76	.73	1.75	.99	1.17	.43

pirical preference probabilities can be checked by substituting the $d^2(x_j, y_i)$ terms in (16.10) with the reconstructed distances in the unfolding solution. Wang et al. (1975) concluded from statistical tests that a 3D representation was sufficiently precise.

The 3D unfolding representation, however, possesses a peculiar property: the ideal points are not distributed throughout the whole space, but lie almost completely in a plane. This implies that the solution has a considerable indeterminacy with respect to the point locations.[5] Figure 16.6 illustrates the problem with a 2D example. All ideal points $y_1, \ldots, y_4$ lie on a straight line, whereas the object points $x_1, \ldots, x_5$ scatter throughout the space. In internal unfolding, the only information available for determining the location of the points is the closeness of object and ideal points. But then each x_j can be reflected on the line running through y_is, because this does not change any between-sets distance. Thus, for example, instead of the solid point x_2 in Figure 16.6, we could also choose its counterpoint shown as an open circle. Such choices have a tremendous effect on the appearance of the unfolding solution and, by way of that, on its interpretation.

How can one diagnose this subspace condition in practice? One can do a principal axes rotation of the $\mathbf{X}$ configuration and of the $\mathbf{Y}$ configuration, respectively, and then check, on the basis of the eigenvalues, whether either one can be said to essentially lie in a subspace of the joint space. Table 16.4

[5]This indeterminacy is not restricted to the Schönemann and Wang model, but it is a property of all Euclidean ideal-point unfolding models.

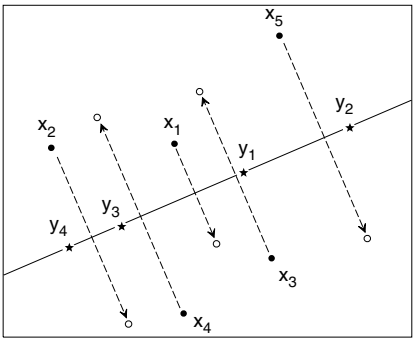


FIGURE 16.6. An indeterminacy in an unfolding space; the x_j points can be reflected on the line of the y_is without affecting the distance $d(x_j, y_i)$.

TABLE 16.4. Coordinates for candidates and subgroup ideal points in 3D: unrotated (left-hand side, $\mathbf{X}$ and $\mathbf{Y}$) and after subspace rotation (right-hand side, $\mathbf{X}^*$ and $\mathbf{Y}^*$).

X	1	2	3	**Y**	1	2	3	**X***	1	2	3	**Y***	1	2	3
Wal	-1.17	-1.40	-.31	NS	.66	.21	-.39	Wal	.99	-1.38	-.97	NS	-.10	.63	-.14
Hum	1.41	-.31	.24	NN	.74	.51	-.42	Hum	.78	1.24	.14	NN	-.40	.75	-.00
Nix	-1.21	.15	-.09	SDSH	.21	-.53	.29	Nix	-.20	-1.25	.05	SDSH	.80	.03	.04
McC	.07	.53	1.42	SDSL	.14	-.59	.20	McC	.49	-.09	1.56	SDSL	.79	-.04	-.08
Rea	-.93	.56	1.19	WDSH	-.06	-.22	.10	Rea	.18	-1.05	1.35	WDSH	.40	-.18	.03
Roc	-.03	.63	1.39	WDSL	-.11	-.43	.12	Roc	.37	-.18	1.58	WDSL	.57	-.26	-.07
Joh	1.02	-.89	-.80	SDNH	.48	.22	-.08	Joh	.66	.89	-1.06	SDNH	.02	.42	.13
Rom	-.12	.71	1.47	SDNL	.37	-.16	-.02	Rom	.32	-.26	1.69	SDNL	.35	.26	-.02
Ken	1.16	-.46	-.54	WDNH	.32	.20	-.10	Ken	.46	1.06	-.61	WDNH	.00	.26	.09
Mus	.53	.29	1.28	WDNL	.15	-.12	-.03	Mus	.69	.34	1.31	WDNL	.28	.05	-.01
Agn	-.96	-.89	-1.05	ISH	-.05	.22	-.26	Agn	.21	-1.04	-1.31	ISH	-.16	-.08	-.04
LeM	-.96	-1.34	-.91	ISL	-.11	-.06	-.12	LeM	.66	-1.11	-1.44	ISL	.14	-.19	-.07
				INH	.03	-.07	.03					INH	.25	-.07	.05
				INL	.06	.01	-.05					INL	.15	-.02	.03
				SRSL	-.25	1.14	-.92					SRSL	-1.31	-.09	-.10
				SRNH	-.27	.85	-.50					SRNH	-.85	-.19	.09
				SRNL	-.26	.53	-.39					SRNL	-.52	-.23	.01
				WRSH	-.34	.09	-.15					WRSH	-.04	-.40	-.02
				WRSL	-.25	.03	-.14					WRSL	.02	-.32	-.04
				WRNH	-.13	.28	-.15					WRNH	-.16	-.16	.09
				WRNL	-.15	.32	-.27					WRNL	-.26	-.17	.01

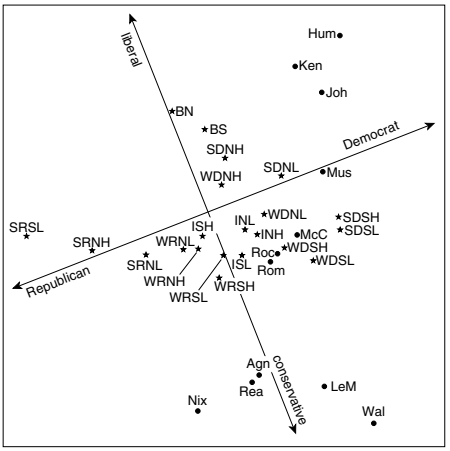


FIGURE 16.7. Unfolding representation of BTL values in Table 16.3; for labels, see Table 16.2 (after Wang et al., 1975).

illustrates this approach for the Wang et al. case. The panels on the left-hand side show the coordinates for the candidates and the ideal points in some 3D joint space. The panels on the right-hand side exhibit the coordinates of both configurations in a rotated 3D joint space whose dimensions correspond to the principal axes of the ideal-point configuration. The table (rightmost column) shows that the ideal points lie essentially in a plane of the 3D joint space, because their coordinates on the third principal axis are all very similar (here: close to zero). If one projects the candidates into this plane, one obtains an unfolding representation that is free from reflection indeterminacies.

Figure 16.7 represents the person groups as stars and the candidates as points. The dimensions correspond to the interpretation given by Wang et al. (1975) on the basis of considering the projections of the candidates onto various straight lines. But one could also proceed by studying the ideal-point labels. For example, a Republican vs. Democrat dimension is suggested by studying the party affiliations of the various groups of white voters. If we draw lines around the groups with party preference SR, WR, I, WD, and SD, respectively, regions of ideal points result that can be partitioned almost perfectly by parallel straight lines. These lines are, however, not quite orthogonal to the direction of the Republican–Democrat dimension chosen by Wang et al. Rather, they partition the axis through the points Nixon and Humphrey. The two other group facets, education and region, do not allow simple partitionings. Wang et al.'s liberal–conservative dimension essentially distinguishes blacks from whites.

Interpreting dimensions is not affected by the reflection indeterminacy discussed above. In contrast, the usual ideal-point interpretation is only partially possible in Figure 16.7. A naive approach can lead to gross mis-

takes. We know from Table 16.4 that a number of candidate points such as Rockefeller, for example, are positioned far above or below the subspace plane shown in Figure 16.7. But Figure 16.7 shows "Roc" close to most ideal points, so that one might expect, incorrectly, that Rockefeller is among the top choices for most groups. The usual ideal-point unfolding interpretation leads to correct preference predictions only for those candidates close to the subspace plane, such as Nixon and Humphrey.

How should one interpret such extra dimensions? There is no simple answer, and additional information beyond the data on which the unfolding is based is required in any case. In the given example, one might speculate that the extra dimension for the candidates reflects additional features of the candidates, unrelated to the preferential choice criteria that distinguish the different groups, such as, for example, the extent to which the candidates are known or unknown. In any case, such interpretations remain complicated because each point can be reflected on this dimension.

Although the meaning of the joint space and its ideal-point subspace remains somewhat unclear in the given example, it is easy to derive some testable implications. The BTL model states a function between v-values of objects and the probability for choosing one object o_j out of any set of choice objects. This function is simply the v-value of object o_i divided by the sum of the v-values of all choice objects. The v-values, in turn, can be estimated from the unfolding distances using (16.9). For the three candidates Nixon, Humphrey, and Wallace (i.e., those that actually remained as candidates in the general presidential election) we can thus estimate, for each group, the probability for choosing each candidate out of the three remaining ones. The prediction for a candidate's chances in the general election is then the (weighted) average of all 21 group-specific probabilities. The predicted preference probabilities of voting for Wallace, Humphrey, or Nixon, computed in this way, are 0.0797, 0.3891, and 0.5311, respectively. These values are quite close to the relative frequencies of direct votes given in the interviews, which are 0.1091, 0.4122, and 0.4788, respectively.

16.5 Exercises

Exercise 16.1 Consider the vector model for unfolding.

(a) First, set up a configuration $\mathbf{X}$ such as the one shown in the table below. Then, define preference vectors for a number of persons, $\mathbf{p}_i$ $(i = 1, \ldots)$, as lines that run through the origin $E = (0, 0)$ and through one other point (x_{1i}, x_{2i}) of $\mathbf{X}$. Finally, construct the preference scale for each person i by projecting the points of $\mathbf{X}$ onto the ideal vectors.

Object	Dim. 1	Dim. 2
A	-1	1
B	0	1
C	1	1
D	-1	0
E	0	0
F	1	0
G	-1	-1
H	0	-1
I	1	-1

(b) Discuss, in terms of psychology, the meaning of the coordinates y_{i1} and y_{i2} of each person i. What do these "weights" express? (Hint: How much do the dimensions of $\mathbf{X}$ contribute to an ideal line's direction?)

(c) How should y_{i1} and y_{i2} be restricted in model (16.2)? (Hint: Note the constraint on $\text{diag}(\mathbf{YY}')$. How can you interpret the thus-constrained coordinates?

(d) Unfold the preference data thus constructed and compare the solution to the $\mathbf{X}$ and the $\mathbf{Y}$ from which you started.

(e) Add random error to $\mathbf{X}$ and $\mathbf{Y}$ and repeat the above investigations for different levels of error. Discuss the robustness of the scaling procedure.

(f) Construct a preference vector that does *not* fit into the space of the objects, $\mathbf{X}$. What could you do to represent it in the preference vector model anyway? (Hint: Consider augmenting the dimensionality of the unfolding space.)

Exercise 16.2 Consider the country-by-attributes data in Exercise 15.1.

(a) Discuss the ideal-point unfolding model for these data. How does it differ from scaling the proximities for the countries (as in Section 1.3) and then fitting external property scales (as in Section 4.3)?

(b) Discuss the difference between an ideal-point model and a vector model in unfolding preferential data and what this difference means in the context of the attribute-by-country data.

(c) Scale the country-by-attributes data into a vector unfolding model, with countries as points and attributes as vectors. Then, scale the same data into an ideal-point model. Compare the solutions in terms of what they suggest about how the student-subjects perceived these countries.

(d) Would it make sense to also scale the countries into vectors, and the attributes into points? How would you interpret such a solution?

Exercise 16.3 The following data set is a data set reported by SAS (1999). It contains the ratings by 25 judges of their preference for each of 17 automobiles. The ratings are made on a 0 to 9 scale, with 0 meaning very weak preference and 9 meaning very strong preference for the automobile.

	Manufacturer	Type	Rating per Judge
1	Cadillac	Eldorado	8 0 0 7 9 9 0 4 9 1 2 4 0 5 0 8 9 7 1 0 9 3 8 0 9
2	Chevrolet	Chevette	0 0 5 1 2 0 0 4 2 3 4 5 1 0 4 3 0 0 3 5 1 5 6 9 8
3	Chevrolet	Citation	4 0 5 3 3 0 5 8 1 4 1 6 1 6 4 3 5 4 4 7 4 7 7 9 5
4	Chevrolet	Malibu	6 0 2 7 4 0 0 7 2 3 1 2 1 3 4 5 5 4 5 6 6 8 6 5 8
5	Ford	Fairmont	2 0 2 4 0 0 6 7 1 5 0 2 1 4 4 3 5 3 0 6 4 8 6 5 5
6	Ford	Mustang	5 0 0 7 1 9 7 7 0 5 0 2 1 1 0 1 8 5 0 6 5 7 5 5 5
7	Ford	Pinto	0 0 2 1 0 0 0 3 0 3 0 3 0 2 0 1 5 0 0 5 1 4 0 7 8
8	Honda	Accord	5 9 5 6 8 9 7 6 0 9 6 9 9 9 5 2 9 9 8 9 7 5 0 7 8
9	Honda	Civic	4 8 3 6 7 0 9 5 0 7 4 8 8 8 5 2 5 6 7 7 6 5 0 7 5
10	Lincoln	Continental	7 0 0 8 9 9 0 5 9 2 2 3 0 4 0 9 9 6 2 0 9 1 9 0 9
11	Plymouth	Gran Fury	7 0 0 6 0 0 0 4 3 4 1 0 1 1 0 7 3 3 3 4 5 8 7 0 8
12	Plymouth	Horizon	3 0 0 5 0 0 5 6 3 5 4 6 1 3 0 2 4 4 4 6 7 5 6 5 5
13	Plymouth	Volare	4 0 0 5 0 0 3 6 1 4 0 2 1 6 0 2 7 5 4 4 7 6 5 5 5
14	Pontiac	Firebird	0 1 0 7 8 9 5 6 1 3 2 0 1 2 0 6 9 5 8 2 6 5 9 0 7
15	Volkswagen	Dasher	4 8 5 8 6 9 6 5 0 8 8 7 7 7 9 5 3 7 7 8 9 5 0 0 0
16	Volkswagen	Rabbit	4 8 5 8 5 0 9 7 0 9 6 9 5 7 9 5 4 8 7 8 8 5 0 0 0
17	Volvo	DL	9 9 8 9 9 9 8 9 0 9 9 9 9 9 8 7 9 8 9 9 1 9 0 0 0

(a) Unfold these preference data into the vector model, with cars as points and vectors as persons. Discuss the solution in terms of what it says about the different automobiles, and what it suggests about groups of potential buyers of automobiles and their preferences.

(b) It was previously observed from unfolding these data that the solution "suggests that there is a market for luxury Japanese and European cars" (http://rocs.acomp.usf.edu/sas/sashtml/stat/chap53/sect25.htm). How did the market researchers arrive at this insight? On what assumptions does this interpretation hinge? Would you be willing to bet your money on this interpretation?

Exercise 16.4 Use the data in Table 16.3 on p. 349to construct a vector-model unfolding representation. Compare your solution to the configuration in Figure 16.7. Discuss where the models suggest similar substantive conclusions (despite possibly different "looks" of the plots), and where they differ.

Exercise 16.5 The table below shows the (contrived) preferences of six different persons for the composition of an ideal family in terms of how many children a person wants, and whether these children should be girls or boys. For example, person 1 wants no children at all, and his or her second choice is one boy. Person 2, on the other hand, ideally wants 2 girls and 2 boys.

Number of		Person					
Girls	Boys	1	2	3	4	5	6
0	0	1	9	5	6	6	7
0	1	2	8	3	7	2	3
0	2	5	5	1	9	7	8
1	0	3	7	8	2	3	4
1	1	4	4	4	4	1	1
1	2	8	2	2	8	4	5
2	0	6	6	9	1	8	9
2	1	7	3	7	3	5	2
2	2	9	1	6	5	9	6

(a) Use ordinal unfolding to study the structure of these preference data. Some programs and some model specifications are likely to yield degenerate solutions. Is your solution degenerate? If so, can you prevent this degeneracy?

(b) The space of choice objects and its dimensions can be thought of as a "boys by girls" space. Does your unfolding yield this space?

(c) Experiment with constraints on the unfolding model so that the boys-by-girls configuration in its solution space approximates a rectangular grid pattern.

(d) Although such family composition preference data have been analyzed before within an unfolding framework, the unfolding model is not really adequate for them. Why? (Hint: Can you have a preference for 1.3 boys and 2.8 girls, for example? Take a close look at the ideal-point isopreference-contours model.)

Part IV

MDS Geometry as a Substantive Model

17
MDS as a Psychological Model

MDS has been used not only as a tool for data analysis but also as a framework for modeling psychological phenomena. This is made clear by equating an MDS space with the notion of psychological space. A metric geometry is interpreted as a model that explains perceptions of similarity. Most attention has been devoted to investigations where the distance function was taken as a composition rule for generating similarity judgments from dimensional differences. Minkowski distances are one family of such composition rules. Guided by such modeling hypotheses, psychophysical studies on well-designed simple stimuli such as rectangles uncovered interesting regularities of human similarity judgments. This model also allows one to study how responses conditioned to particular stimuli are generalized to other stimuli.

17.1 Physical and Psychological Space

In most applications of MDS today, little attention is devoted to the Shepard diagram. It may therefore surprise the reader that ordinal MDS was originally invented to study the shape of the regression curve in this diagram, not the MDS configuration. This also makes clear how closely MDS used to be related to efforts for modeling psychological phenomena, where the MDS geometry served as a model of psychological space and the distance function as a model of mental arithmetic.

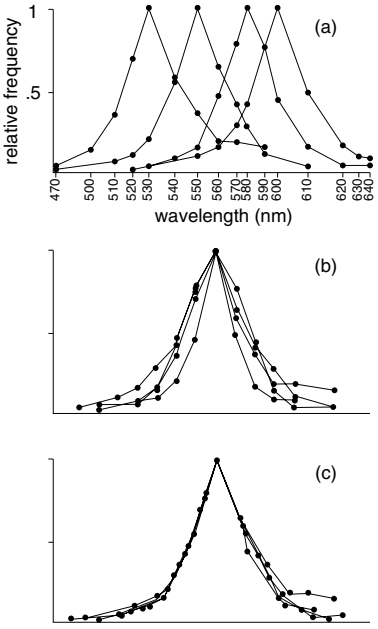


FIGURE 17.1. (a) Four generalization gradients over the electromagnetic spectrum, with intervals adjusted to make gradients similar; panel (b) shows superimposed gradients constructed over the nonadjusted scale; panel (c) shows superimposed gradients from panel (a).

The Shape of Generalization Gradients in Learning

The scientific context for the interest in Shepard diagrams becomes clear from the following experiment. Guttman and Kalish (1956) trained four groups of pigeons to peck at a translucent plastic key when illuminated from behind by monochromatic light with wavelengths 530, 550, 580, and 600 nm, respectively. After the learning sessions, they assessed the frequency with which the pigeons in each group pecked at the key when illuminated with different colors. Figure 17.1a shows that the probability of pecking at the key is highest for the original conditioned color and decreases monotonically as a function of the difference between the original and the test color.

One can ask whether such generalization gradients always have the same shape. Are they, say, always exponential decay functions over the stimulus dimensions? This is difficult to decide, because it is the *psychological*, not the *physical*, stimulus dimensions that are relevant. In a simple case such as the Guttman–Kalish experiment, the stimuli vary on just one dimension. The physical (here: wavelength) and the psychological (here: hue) dimensions are related to each other by a psychophysical mapping. The shape of generalization gradients depends on this mapping. This can be seen from

Figure 17.1, taken from Shepard (1965). The X-axis in panel (a) shows the physical wavelengths of the stimuli, but its units have been somewhat compressed and stretched locally to make the four gradients as similar as possible. Without these adjustments, that is, over the physical wavelength scale, the gradients are less similar (Figure 17.1b). After the adjustment, the gradients are almost equal in shape (Figure 17.1c).

Thus, knowledge of the psychological space of the stimuli, or at least the "psychological" distances between any two stimuli, S_i and S_k, is necessary for meaningful statements on the shape of generalization gradients. Older approaches often tried to arrive at psychological distances directly by summing just noticeable differences (JNDs) between S_i and S_k. The idea that this sum explains the subjective dissimilarity of S_i and S_k goes back to Fechner (1860). There are many problems associated with this model (Krantz, 1972), but one is particularly important for MDS: "Unfortunately, in order to sum JNDs between two stimuli, this summation must be carried out along some path between these stimuli. But the resulting sum will be invariant ... only if this path is a least path, that is, yields a shortest distance (in psychological space) between the two stimuli. We cannot presume, in arbitrarily holding certain physical parameters constant ..., that the summation is constrained thereby to a shortest path ... in psychological space, even though it is, of course, confined to a shortest path ... in physical space. ... These considerations lead us to look for some way of estimating the psychological distance between two stimuli without depending either upon physical scales or upon any arbitrary path of integration" (Shepard, 1957, p. 334).

Relating Physical Space to Psychological Space

An *external* approach for the problem of estimating psychological distances first assumes a particular correspondence of physical space to psychological space and then explains how the response probabilities are distributed over this space. An *internal* approach, in contrast, builds directly and exclusively on the response probabilities and formulates how these arise as a function of unknown psychological distances. Let us consider Shepard's original derivations (Shepard, 1957). Let p_{ik} be the probability of giving the S_i response to stimulus S_k. If $i = k$, then p_{ik} is the probability of giving the correct response. It is postulated that there exists a function f such that p_{ik} is proportional to $f(d_{ik})$, where d_{ik} is the psychological distance between S_i and S_k,

$$p_{ik} = c_i \cdot f(d_{ik}), \tag{17.1}$$

with c_i a proportionality constant associated with S_i. Summing over all k, we obtain $\sum_k p_{ik} = 1$ and $c_i \cdot \sum_k f(d_{ik})$ for the two sides of (17.1), so that

$c_i = 1/\sum_k f(d_{ik})$. Inserting this term for c_i in (17.1) yields

$$p_{ik} = f(d_{ik})/\sum_j f(d_{ij}). \tag{17.2}$$

With the p_{ik}-values given as data, we now search for a function f that satisfies (17.2). The important point here is that the d-values on the right-hand side are not just any values that satisfy (at least approximately) all equations of type (17.2), but they must also possess the properties of distances and even of Euclidean distances in a space of given dimensionality. Moreover, we would not accept any function f, but only those that are *smooth* (continuous) and monotone increasing or decreasing. Then f is invertible, so that response probabilities can in turn be derived from the psychological distances. If we assume that the psychological space is related to the physical space by a smooth transformation, then straight lines in physical space are transformed into lines in psychological space that may not be straight but smoothly curved. Hence, given any three stimuli on a straight line in physical space, their psychological images should also be approximately on a straight line if the stimuli are physically similar. From this assumption and some additional simple postulates on decay and diffusion of memory traces, Shepard (1958a) derives that f is a negative exponential function. Elsewhere, without any assumptions, Shepard (1957) simply defines f to be a negative exponential function. This function turns (17.2) into

$$p_{ik} = \exp(-d_{ik})/\sum_j \exp(-d_{ij}). \tag{17.3}$$

Because $d_{ii} = 0$, $\exp(-d_{ii}) = \exp(0) = 1$ and so

$$p_{ik}/p_{ii} = \exp(-d_{ik}). \tag{17.4}$$

Dividing p_{ik} by p_{ii} means that the probability of giving the i response to stimulus k is expressed relative to the probability of responding properly to S_i. Thus, norming all response probabilities in this way, and specifying that d_{ik} is a Euclidean distance in a space with dimensionality m, we end up with a metric MDS problem that requires finding a point space such that its distances satisfy (17.4) as closely as possible. A reasonable choice for m should be the dimensionality of the physical space.

Determining the Shape of Generalization Gradients via MDS

The discussion above led to a confirmatory MDS problem: the data (i.e., the ratios p_{ik}/p_{ii}) are to be optimally mapped into a particular model. The fit of the model to the data is then evaluated. Shepard (1958b) concluded that the negative exponential function allows one to explain the data sufficiently well, but other functions, such as a simple linear one, may also be

in good or even better agreement with the data. Shepard tried to solve this problem and allow the data to "reveal themselves" by requiring only that f in (17.1) be monotonically decreasing rather than some specific parametric function. In other words, expressed in terms of the generalization gradients, he required that they should decrease from the correct stimulus S_r monotonically into all directions of the stimulus space.

To see how a psychological scale (e.g., the X-axis in Figure 17.1a) is derived, fold Figure 17.1b at the points where the gradients peak. What will then be obtained is nothing other than a Shepard diagram, where the data appear on the Y-axis and the "psychological" distances on the X-axis. Hence, finding the psychological scale amounts to using ordinal MDS with $m = 1$ in the present case. Of course, the Shepard diagram will show a scatter of points only, and the various gradients have to be found by unfolding the Shepard diagram and connecting the respective points. The unfolding is done simply by arraying the points in the order of their physical stimulus coordinates (here: wavelengths) and with distances among them as computed by the MDS procedure.

17.2 Minkowski Distances

Over a 2D stimulus space, the generalization gradients are surfaces such as the cones and pyramids shown schematically in Figure 17.2. Assume that the directions labeled as D1 and D2 are psychologically meaningful dimensions such as hue and saturation for color stimuli. Assume further that the correct stimulus S_r corresponds to the point where D1 and D2 intersect. Cross (1965a) then distinguishes the following three models: (1) the *excitation model*, which assumes that the generalization gradient decreases evenly around S_r into all directions of the psychological space; (2) the *discrimination model*, which says that the strength of reacting to a stimulus different from S_r on both dimensions corresponds to the sum of the generalization of S_r on both dimensions; and (3) the *dominance model*, where the strength of reacting to $S_i \neq S_r$ is determined by only that dimension on which S_i and S_r differ most. These models are illustrated in Figure 17.2. The gradients are shown as linear functions to simplify the pictures. Note that the gradients for the discrimination model and the dominance model have the same shape (for a two-dimensional psychological space) but differ in their orientation relative to the dimensions.

The Family of Minkowski Distances

The generalization models in Figure 17.2 illustrate three special cases of the *Minkowski metric* or, equivalently, the *Minkowski distance*. The general

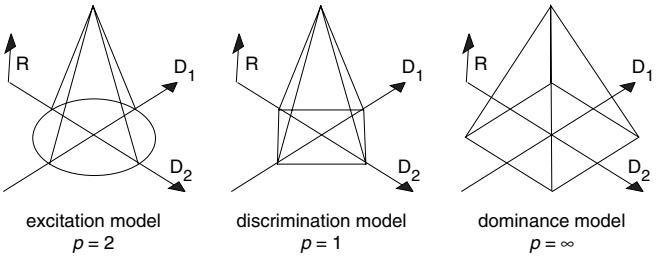


FIGURE 17.2. Three models of generalization over a 2D stimulus continuum; S_r corresponds to intersection of D1 and D2 (after Cross, 1965b).

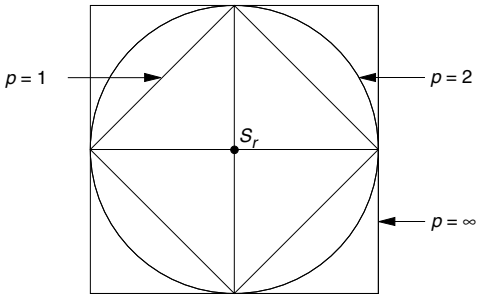


FIGURE 17.3. Three circles with same radius around S_r in 2D for different p-values in the Minkowski distance formula.

formula for this metric is

$$d_{ij}(\mathbf{X}) = \left(\sum_{a=1}^{m} |x_{ia} - x_{ja}|^p \right)^{1/p}, \ p \geq 1. \tag{17.5}$$

For $p = 2$, equation (17.5) yields the usual *Euclidean* distance formula. For $p = 1$, we obtain the *city-block* metric, and for $p \to \infty$, the *dominance* metric.

The implications of choosing different p-values can be seen from the following. If we look, from above, at the three gradient models in Figure 17.2, a circle, a diamond, and a square, respectively, appear in the psychological space. Superimposing these three figures leads to the diagram in Figure 17.3. Assume, for simplicity, that the point S_r has coordinates $(0, 0)$. Then, (17.5) reduces to

$$d_{rj} = (|x_{j1}|^p + |x_{j2}|^p)^{1/p}. \tag{17.6}$$

For $p = 1$ we obtain d_{rj} as just the sum of the absolute coordinates of S_j. Thus, all stimuli located on the diamond in Figure 17.3 have the same city-block distance to S_r. The diamond is therefore called the *isosimilarity curve* of the city-block metric. It is the set of all points with the same distance to

S_r. But this is just the definition of a circle in analytical geometry, so the diamond is nothing but *a circle in the city-block plane*, even though it does *not look* like a circle at all. Our Euclidean notion of a circle corresponds exactly to the isosimilarity curve for $p = 2$. Finally, the circle for $p \to \infty$ looks like a square with sides parallel to the dimensions.

It is important to realize that the distance d_{rj} for two given points S_r and S_j remains the same under rotations of the coordinate system only if $p = 2$. For $p = 1$, d_{rj} is smallest when both stimulus points lie on one of the coordinate axes. If the coordinate system is rotated about S_r, then d_{rj} grows (even though the points remain fixed), reaches its maximum at a 45° rotation, and then shrinks again to the original value at 90°. This behavior of a distance function may appear strange at first, but "... under a good many situations, [this distance] describes a reasonable state of affairs. For example, suppose one were in a city which is laid out in square blocks. A point three blocks away in one direction and four blocks away in the other would quite reasonably be described as seven blocks away. Few people, if asked, would describe the point as five blocks distant. Further, if new streets were put in at an angle to the old, the 'distance' between the two points would change" (Torgerson, 1958, p. 254).

Minkowski Distances and Intradimensional Differences

Further properties of different Minkowski distances follow directly from (17.5). Cross (1965b, 1965a) rearranges its terms in a way that we show here for the special case of (17.6):

$$\begin{array}{rcl} d_{rj}^p & = & |x_{j1}|^p + |x_{j2}|^p, \\ d_{rj} d_{rj}^{p-1} & = & |x_{j1}|^{p-1} \cdot |x_{j1}| + |x_{j2}|^{p-1} \cdot |x_{j2}|, \\ d_{rj} & = & \underbrace{(|x_{j1}|^{p-1}/d_{rj}^{p-1})}_{w_1} \cdot|x_{j1}| + \underbrace{(|x_{j2}|^{p-1}/d_{rj}^{p-1})}_{w_2} \cdot|x_{j2}|, \\ d_{rj} & = & w_1 \cdot|x_{j1}| + w_2 \cdot|x_{j2}|. \end{array} \tag{17.7}$$

It follows that for $p = 1$, d_{rj} is just the sum of the coordinate values of stimulus S_j, because $w_1 = w_2 = 1$. If $p > 1$, then the coordinates are weighted by w_1 and w_2 in proportion to their size. If $p \to \infty$, d_{rj} approximates its largest coordinate value. This can be seen most easily from a numerical example. Table 17.1 shows such an example for S_r = (0,0) and S_j = (1,2), for which $|x_{j1}| = 1$ and $|x_{j2}| = 2$. For $p = 1$, we obtain $d_{rj} = (1/3)^0 \cdot 1 + (2/3)^2 \cdot 2 = 1 \cdot 1 + 1 \cdot 2 = 3$. For $p = 2$, we get $d_{rj} = (1/\sqrt{5})^1 \cdot 1 + (2/\sqrt{5})^1 \cdot 2 = 0.44721360 + 1.78885438 = 2.23606798$.

Generally, if $p \to \infty$, then $d_{rj} \to 2$; that is, as p grows, the larger of the two coordinates of S_j (i.e., the larger of the two-dimensional differences between S_r and S_j) tends to *dominate* the global distance value. Indeed, d_{rj} approximates the limiting value 2 quite rapidly as p grows: for $p = 20$, d_{rj} differs from 2 only in the seventh position after the decimal point.

TABLE 17.1. Demonstration of how dimensional differences (x_{ja}) enter the distance of two points r and j under different Minkowski p parameters, with $x_{r1} = 0, x_{r2} = 0, x_{j1} = 1, x_{j2} = 2$.

p	$w_1 \cdot x_{j1}$	$w_2 \cdot x_{j2}$	w_2/w_1	d_{rj}
1.0	1.00000000	2.00000000	1.00	3.00000000
1.5	0.63923401	1.80802681	1.41	2.44726081
2.0	0.44721360	1.78885438	2.00	2.23606798
3.0	0.23112042	1.84896340	4.00	2.08008382
4.0	0.11944372	1.91109947	8.00	2.03054318
5.0	0.06098020	1.95136642	16.00	2.01234662
10.0	0.00195141	1.99824382	512.00	2.00019523
20.0	0.00000191	1.99999819	524288.00	2.00000010

In terms of Figure 17.3, increasing p from 1 to 2 means that the diamond bulges outwards and approximates the Euclidean circle. For Minkowski parameters greater than 2, the circle then moves towards the square for $p \to \infty$. Hence, the three generalization models in Figure 17.2 correspond to different ways of *composing* a distance from given *intradimensional* differences between pairs of stimuli. For example, given two tones that differ in frequency and sound pressure, one possible composition rule yielding their subjective global dissimilarity would be simply to add their frequency and pressure differences in the corresponding psychological space, that is, add their differences in pitch and loudness. This corresponds to computing a city-block distance. The Euclidean distance formula, on the other hand, implies a composition rule that is much harder to understand. What is clear, though, is that, for all $p > 1$, the differences first are weighted and then added, with the larger differences receiving a larger weight. In the extreme case ($p \to \infty$), the largest difference completely dominates the dissimilarity judgment.[1]

Torgerson (1958), Garner (1962), and others argue that if the stimuli are such that their dimensions are obvious and natural (*analyzable stimuli*), then the city-block distance should be the best model to explain dissimilarity judgments. If, on the other hand, the stimuli are *integral*, then the Euclidean metric should be more appropriate.[2]

[1] Interpreting the Minkowski distance as a composition rule is just one possibility. Micko and Fischer (1970) and Fischer and Micko (1972), for example, present an alternative conceptualization in which the composition rule is not a summation of intradimensional differences. Rather, an attention distribution is postulated to exist over all directions in space, so that the effect of an increment in p in the Minkowski model corresponds to a concentration of attention in certain spatial directions.

[2] An example of an analyzable stimulus is the one-spoked wheel shown in Figure 1.6. Its "obvious and compelling dimensions" (Torgerson, 1958, p. 254) are its size and the inclination angle of its spoke. A color patch, on the other hand, is an integral stimulus whose dimensions hue, saturation, and brightness can be extracted only with effort.

Wender (1971) and Ahrens (1972) propose that as similarity judgments become more difficult—because of, say, time constraints or increasing complexity of the stimuli—subjects tend to simplify by concentrating on the largest stimulus differences only. Hence, we should expect that such similarity data could be explained best with large Minkowski p parameters.

Maximum Dimensionality for Minkowski Distances

Suppose that $\mathbf{D}$ is a matrix of Minkowski distances. If $\mathbf{D}$ is Euclidean, then there are at most $m = n - 1$ dimensions. But what about other cases of Minkowski distances? Fichet (1994) shows that for city-block distances the dimensionality can be at most $[n(n-1)/2]-1$. For the dominance distance, the maximum dimensionality is $n - 1$ (Critchley & Fichet, 1994), a result that goes back to Fréchet (1910). Note though that these theoretical results are not based on analyses that would allow us to identify the dimensionality of the underlying configuration $\mathbf{X}$ of a given $\mathbf{D}$, except for Euclidean distances (see Section 19.3).

In addition, Critchley and Fichet (1994) show that certain Minkowski distance matrices are *exchangeable*. To be more precise, for every Euclidean distance matrix, there exists a city-block and dominance distance matrix having the same values (most likely in a different dimensionality and with a different configuration). Also, for every city-block distance matrix there exists a dominance distance matrix having the same values. And, of course, all unidimensional Minkowski distance matrices are equal irrespective of the Minkowski parameter p. These results imply that a solution found by MDS using the Euclidean distance can be exchanged by a solution using the city-block distance (or the dominance distance) *without* changing the Stress value, although the dimensionality of the three solutions is most likely not the same.

17.3 Identifying the True Minkowski Distance

How can the true Minkowski distance be identified? There are two approaches, one based on scaling proximities in MDS with different metrics, and one based on analyzing the proximities and assuming certain properties of the psychological space.

Take two points in psychological space. The Euclidean distance between these points is not affected by rotations of the dimension system. The city-

---

Indeed, "if dimensions are integral, they are not really perceived as dimensions at all. Dimensions exist for the experimenter... But these are constructs... and do not reflect the immediate perceptual experience of the subject in such experiments..."(Garner, 1974, p. 119).

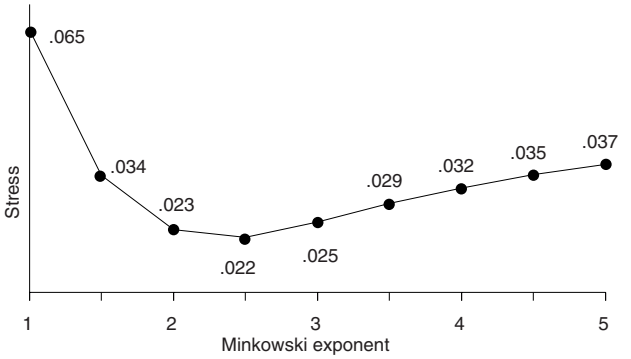


FIGURE 17.4. Stress values for representing data in Table 4.1 in 2D MDS spaces using Minkowski distances with different exponents (after Kruskal, 1964a).

block distance, however, is smallest if these points lie on a line parallel to a dimension and greatest if this line forms an angle of 45° to the dimensions.

This suggests that one should test whether two points with a given Euclidean distance are perceived as more dissimilar if they differ on just one dimension rather than on several dimensions. If this matters, then the Euclidean distance cannot be the true metric. Shepard (1964) attempted to check this condition by constructing one-spoked wheels with dimensions "size" and "angle of spoke" (as in Figure 1.6) and assuming that the psychological space is essentially equivalent to this 2D physical space. He observed that stimuli that differed on one dimension only were perceived as relatively similar as compared to those that differed on two dimensions, although their Euclidean distances in physical space were equal. He took this finding as supporting evidence for the city-block metric, which was predicted to be appropriate for such analyzable stimuli.

Determining the True Minkowski Distance by MDS

A second approach for determining the true Minkowski distance is to test how well given proximities can be represented in a space with a given metric. Such scaling tests are easy to compute but difficult to evaluate. If the dimensionality question can be settled beforehand in some way, Kruskal (1964a) suggests computing MDS representations for a large range of different p-values and then selecting as the true metric the one that leads to the lowest Stress. This is shown in Figure 17.4 for Ekman's color data from Table 4.1. The lowest Stress (.0215) occurs at $p = 2.5$. Kruskal (1964a) comments on this finding: "We do not feel that this demonstrates any sig-

nificant fact about color vision, though there is the hint that subjective distance between colors may be slightly non-Euclidean" (p. 24).[3]

Ahrens (1974) proposes varying both p and m. In this way, a curve like the one in Figure 17.4 is obtained for each m. If these curves all dip at the same p-value, then we can decide the metric question independently of the dimensionality question.

Yet, proposals for deciding on the true metric empirically and not by theoretical considerations assume that the Stress values arrived at under different specifications for p and m are comparable. This requires that all solutions must be global minima, because otherwise it would not make sense to conclude that $p = 1$, say, yields a better solution than $p = 2$. The global minimum condition can be checked by using many—Hubert, Arabie, and Hesson-McInnis (1992) used 100!—different starting configurations for each fixed pair of p and m.[4]

We must, moreover, decide whether any small difference between two Stress values is significant. In Figure 17.4, the Stress values around $p = 2.5$ are quite similar. Should we really conclude that the subjects use $p = 2.5$ and not, say, $p = 2$, because the Stress is slightly smaller for the p parameter than for the latter? Probably not. It seems more reasonable to decide that the subjects used a p parameter close[5] to 2.

Distinguishing among MDS Solutions with Different Minkowski Distances

There are p-values that lead to the same Stress for a given 2D configuration, for example, the extreme cases $p = 1$ and $p \to \infty$. Figure 17.3 shows why this is so. If the dimension system is rotated by 45°, the isosimilarity contour for $p = 1$ is transformed into the isosimilarity contour for $p \to \infty$, except for its overall size. This means that city-block distances computed from a given MDS configuration and a given coordinate system are, except

---

[3] There are several ways to minimize Stress for Minkowski distances. A general gradient approach is taken in KYST, SYSTAT, and MINISSA. Groenen et al. (1995) and Groenen et al. (1999) give a majorization algorithm of which the SMACOF algorithm of Section 8.6 is a special case. The majorizing algorithm turns out to have a quadratic majorizing function for $1 \leq p \leq \infty$, so that each update can be found in one step. For p outside this range, the update has to be found by an iterative procedure.

[4] For the special (but important) case of city-block distances, Groenen and Heiser (1996) found many local minima. To find the global minimum, they applied the tunneling method (see Section 13.7). Different approaches were pursued by Heiser (1989b) and Hubert et al. (1992), who used combinatorial strategies, and Pliner (1996), who proposed to apply the smoothing strategy (see Section 13.5).

[5] Indeed, by scaling the data with more modern MDS programs, one finds that the minimum Stress is at $p = 2$. Arabie (1991) conjectured, moreover, that "to the extent that our theory predicts a circle ..., the curve in Figure [17.4] should be flat unless disturbed by either (a) numerical artifacts in computation or (b) noise in the data."

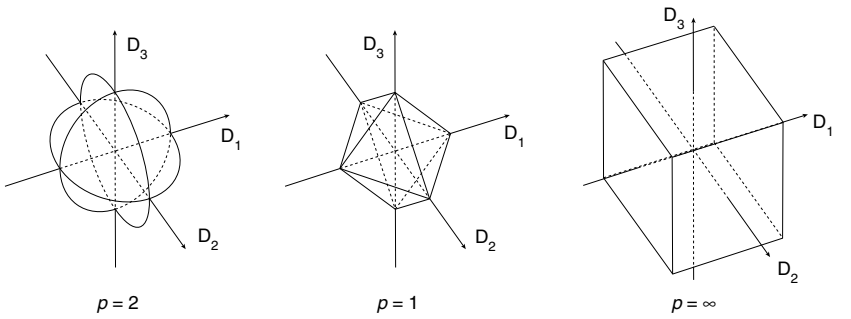


FIGURE 17.5. Unit balls in 3D for the Euclidean, the city-block, and the dominance metric, respectively.

for an overall multiplicative constant, identical to dominance distances, provided the dimension system is rotated by 45°. The converse is also true. Hence, given some MDS configuration that is perfect for $p \to \infty$, it must also be perfect for $p = 1$, and vice versa, because the Stress is the same for two sets of distances that differ by a multiplicative constant only.

The close relationship between city-block distances and dominance distances holds, however, only for 2D. In 3D, the unit circles become unit balls, and Figure 17.5 shows that these balls look quite different for $p = 1$ and $p = \infty$. The city-block ball has, for example, six corners, and the dominance ball has eight corners. The two types of distances therefore cannot be related to each other by a simple transformation and a stretch, as is true for the 2D case.

For given 2D configurations, Stress is, moreover, almost equal for distances with p-exponents of p_1 and $p_2 = p_1/(p_1 - 1)$ (Wender, 1969; Bortz, 1974). For example, for $p = 1.5$ and $p = (1.5)/(1.5-1) = 3$, the Stress values should be nearly equal. The geometrical reasons for this *quasi-equivalency* have been studied in detail by Wolfrum (1976a).

Furthermore, Stress may also be somewhat misleading. Consider the following case (Borg & Staufenbiel, 1984). For a given configuration, the distances are greatest for $p = 1$. When p grows, all distances that relate to line segments not parallel to one of the dimensions drop sharply in size. They continue to drop monotonically, but reach asymptotic values for larger ps ($p > 10$, say). As long as these size functions over p do not intersect, one obtains intervals of rank-equivalent distances over p (Wolfrum, 1976b). Yet, one should not expect that Stress (for nonperfect solutions) is equal for each p within such an interval, because the variance of the distances generally shrinks substantially if p grows. This makes it easier to fit a monotone regression function, and, hence, Stress tends to become smaller with greater p. Nevertheless, the existence of rank-equivalent intervals means that there is no unique optimal p-value but rather intervals of ps that are all equally

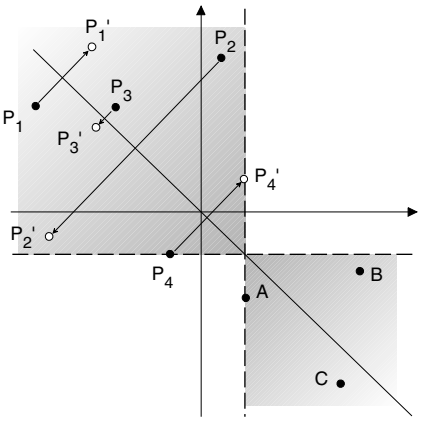


FIGURE 17.6. Demonstration of an indeterminacy in a city-block plane (after Bortz, 1974).

good, even though Stress would, in general, not allow one to diagnose this situation correctly.

On the other hand, there is also an opposite trend that makes low-Stress MDS solutions more likely when $p = 1$ than when $p = 2$, for example. To see this, consider the four corner points of the diamond curve in Figure 17.3. One can readily verify that the city-block distances between these points are all equal, whereas the Euclidean distances form two different classes. Thus, if $p = 1$, four points can be represented in a plane so that all possible distances among them are equal; but if $p = 2$, this is only true for the three corners of an equilateral triangle. Because making distances equal would reduce Stress, such solutions are systematically approximated over the iterations (Shepard, 1974). These effects become more and more pronounced as p approaches the extremes 1 and ∞. Shepard (1974, p. 404) concludes, therefore, that "while finding that the lowest Stress is attainable for $p = 2$ may be evidence that the underlying metric is Euclidean, the finding that a lower Stress is attainable for a value of p that is much smaller or larger may be artifactual."

Interpreting Non-Euclidean MDS Spaces

It has been suggested that the problem of finding the true p-value empirically is easier to solve if other criteria, especially the solution's interpretability, are also taken into account. However, interpreting non-Euclidean Minkowski spaces requires much care. Things are not always what they seem to be, for example, a circle in a city-block space looks like a square. In addition, it can happen that for $p = 1$ and $p \to \infty$ the configurations are indeterminate in peculiar ways. Bortz (1974) reports some examples of *partial isometries*, that is, transformations that preserve

the distances within a point configuration while substantially changing the configuration itself. Consider Figure 17.6. If we reflect all points labeled by capital Ps on the diagonal line, we find that the city-block distances of their images (primed Ps) to any point in the shaded region in the lower right-hand corner are exactly the same as before. Hence, either configuration is an equally good data representation, although they may suggest different substantive interpretations. For $p = 2$, no such partial isometries exist in general.

Robustness of the Euclidean Metric

Is the Euclidean metric robust if incorrect? That is, is it likely that MDS closely approximates a true configuration defined by non-Euclidean distances if the scaling is done with $p = 2$? Shepard (1969) concluded from simulation studies using as proximities non-Euclidean distances and even *semi-metrics* (measures that satisfy only nonnegativity and symmetry, but not the triangle inequality) that the true underlying configuration could be recovered almost perfectly with $p = 2$.

This successful recovery of the original configuration using $p = 2$, however, may be partially attributed to the large number (=50) of points in 2D so that the points' locations were highly restricted. The circular isosimilarity contour of the Euclidean distance then is a good approximation to the isosimilarity contours of other Minkowski metrics (see Figure 17.3).

There are no systematic studies that allow one to predict under what conditions the Euclidean metric is robust and when it is not. However, using the Euclidean metric if, say, the city-block metric is true may lead to erroneous conclusions. Consider the following case. Lüer and Fillbrandt (1970), Lüer, Osterloh, and Ruge (1970), and Torgerson (1965) report empirical evidence that similarity judgments for simple two-dimensional stimuli (such as one-spoked wheels) seem to be perceived in an "over-determined" (3D) psychological space. That is, the psychological space seemed to contain additional and redundant dimensions. However, when scaling the data with $p = 1$ rather than with $p = 2$, the underlying physical space is clearly recovered (Borg, Schönemann, & Leutner, 1982). Taking a closer look reveals that using $p = 2$ warps the city-block plane by pulling two of its "corners" upwards and pushing the two other corners downwards along the third dimension.

17.4 The Psychology of Rectangles

We now consider a classic case using MDS as a model of judgmental behavior. In this model, the Minkowski distance formula is taken as a theory of how a dissimilarity judgment on two stimuli is generated. The choice of the

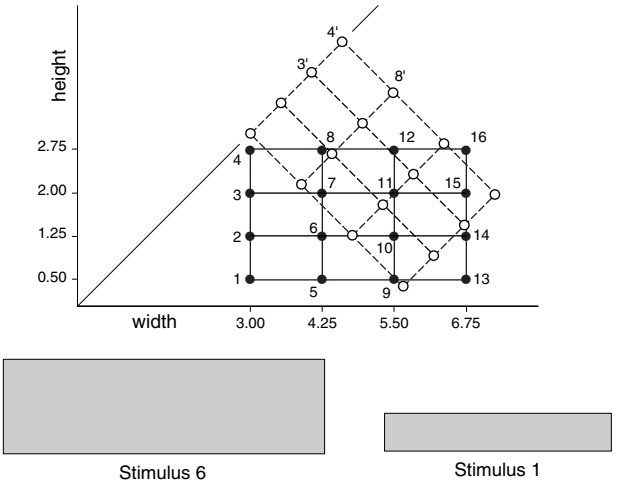


FIGURE 17.7. Design for two sets of rectangles by varying width and height (upper panel). As an example, stimuli 6 and 1 are shown (lower panel).

particular p-values is decided a priori on theoretical grounds. Two different dimension systems appear natural, so that we have to decide empirically which one is the more appropriate.

Two-Dimensional Models for Rectangle Perception

The stimuli here are rectangles. A particular design for rectangles is given in Figure 17.7. It defines two sets of rectangles, characterized by the grid of 16 solid points connected by solid lines and the rotated set of 16 open points connected by dashed lines. The first set is called the width × height (WH) design, because it is orthogonal to the width and height dimensions. In other words, for each level of width, there are rectangles of all height levels. Note that for all rectangles it holds that their width exceeds their heights.

The dashed grid is orthogonal to the WH system rotated by 45°. The point coordinates on this system can be computed from the width × height system as width + height and width − height (multiplied by a constant). Psychologically, these dimensions represent something like size and shape (SS). (If width and height are rescaled logarithmically, then size becomes area.) The SS system represents an alternative model for the perception of rectangles.

Borg and Leutner (1983) randomly assigned 42 subjects to two groups of 21 persons each, one group judging the SS rectangles and the other the WH stimuli. Each subject rated all possible 120 stimulus pairs twice on a scale with end categories 0=equal, identical, and 9=extremely different.

TABLE 17.2. Dissimilarity ratings for rectangle pairs; row and column numbers correspond to rectangle numbers in Figure 17.7; ratings averaged over all subjects and replications in WH group (lower half) and in SS group (upper half).

No.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1		2.05	2.64	3.31	4.93	4.31	4.60	5.79	6.50	6.55	6.19	5.52	8.00	6.98	6.79	7.14
2	4.33		2.12	2.71	4.71	4.69	4.43	4.98	6.40	5.98	5.81	5.71	8.14	6.95	6.76	6.79
3	6.12	4.07		1.79	5.40	5.07	4.36	4.24	6.93	6.29	5.98	5.71	8.17	7.40	6.76	6.71
4	7.21	5.62	3.24		6.36	5.83	4.88	4.31	7.14	6.52	5.71	5.79	8.67	7.69	7.17	6.40
5	2.38	5.76	7.12	7.57		3.17	4.19	4.57	3.52	3.79	3.69	4.95	6.33	5.67	5.29	4.69
6	4.52	2.52	5.48	6.86	4.10		3.43	3.93	4.12	3.57	3.74	3.60	6.62	5.76	5.31	4.90
7	6.00	4.52	3.38	5.21	6.10	4.31		3.43	5.64	4.07	3.48	2.98	7.26	5.83	5.64	5.26
8	7.76	6.21	4.40	3.12	6.83	5.45	4.00		5.55	4.45	3.71	3.64	6.95	5.98	5.24	5.00
9	3.36	6.14	7.14	8.10	2.00	4.71	6.52	7.71		2.86	4.45	5.79	4.14	3.02	3.00	4.57
10	5.93	4.24	6.07	6.93	5.00	2.81	5.43	5.67	4.38		2.86	4.17	4.50	3.48	3.05	3.17
11	6.71	5.60	4.29	5.90	6.86	4.50	2.64	5.21	6.26	3.60		3.31	5.52	3.83	3.40	2.50
12	7.88	6.31	5.48	5.00	7.83	5.55	4.43	2.69	7.21	5.83	3.60		5.95	5.17	3.88	3.55
13	3.69	6.98	7.98	8.45	2.60	5.95	7.69	7.86	1.60	4.31	6.95	7.43		2.38	4.29	5.43
14	5.86	4.55	6.64	7.17	4.86	2.88	5.40	6.50	4.14	1.19	3.79	5.88	4.17		2.64	3.81
15	7.36	5.88	4.55	6.79	6.93	4.50	3.50	5.55	5.95	3.95	1.48	4.60	6.07	4.02		2.74
16	8.36	7.02	5.86	5.40	7.57	5.86	4.52	3.50	6.86	5.17	3.71	1.62	7.07	5.26	3.45	

The resulting proximities, averaged over all 21 subjects in each group, are shown in Table 17.2.

An ordinal MDS representation of the WH data is given by the solid points in Figure 17.8. Because the city-block metric was used, the coordinate axes cannot be rotated without adversely affecting Stress. The MDS result thus suggests that the solid grid of the physical space (Figure 17.7) was transformed into the MDS representation by simple rescalings of the width and height dimensions. These rescalings are such that the physical units decrease more on each dimension the more one moves away from the origin. Thus, perceptually, physically constant increments of an attribute affect the overall impression of similarity increasingly less the more the rectangle already possesses this attribute. This suggests that the psychophysical rescalings might follow the Weber–Fechner law, which postulates a logarithmic correspondence of psychological and physical units. Indeed, the design configuration (grid of solid points in Figure 17.7) can be rescaled in this way to closely fit the MDS representation (grid of open squares in Figure 17.8). Thus, it seems that the subjects in the WH group judged the dissimilarities of the rectangles by first logarithmically rescaling the width and height dimensions, and then simply adding intradimensional differences over the dimensions. But if this were so, what should be expected for the MDS configuration of the SS data?

If width and height are the dimensions that the subjects attend to, and not size and shape, then the SS design grid in Figure 17.7 should be psychophysically rescaled along the width and height axes. A nonlinear rescaling such as the logarithm would lead to some bending of the design lattice, destroying all right angles. The solid points in Figure 17.9 show the MDS representation for the SS data, together with the logarithmically rescaled SS

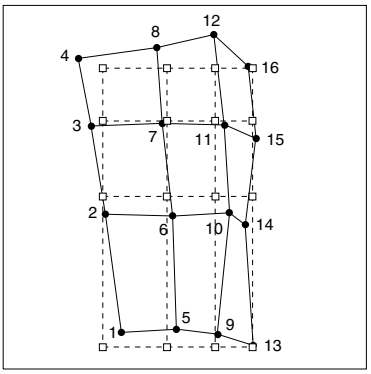


FIGURE 17.8. City-block MDS configuration (solid points) of data of WH group in Table 17.2 with fitted physical WH configuration (squares).

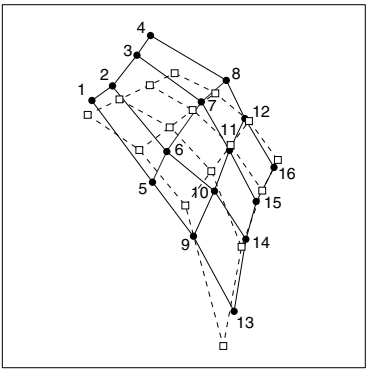


FIGURE 17.9. City-block MDS configuration (solid points) of data of SS group in Table 17.2 with fitted physical SS configuration (squares).

design grid (open squares). One notes that the predictions are not strictly satisfied. In particular, the rectangles in the upper left-hand corner (which look like squares!) seem to involve some further effects. The result suggests, however, that explaining similarity judgments for rectangles seems impossible with size and shape dimensions, because there is no way to explain the bending effects by any rescalings of these dimensions. Rather, a size–shape theory requires additional components such as "dimensional interaction" [for such a theory, see Krantz and Tversky (1975)].

Some Open Questions on Rectangle Perception

These findings are, unfortunately, less simple to interpret than it may appear at first sight. In the following, we give a brief listing of some problem points.

(1) Hubert et al. (1992) reanalyzed the above rectangle data to test the performance of their combinatorial algorithm for MDS. Using 100 random configurations as initial configurations, as well as the physical stimulus space, they found that most solutions for the WH data were similar to Figure 17.8. However, for the SS data, there exist a number of quite different solutions with almost the same Stress. Indeed, based on results from SYSTAT's MDS module, Hubert et al. (1992) concluded generally for gradient-based algorithms that "where one begins is close to where one ends" (p. 234). In particular, it turns out that the SS data can also be explained in a grid that is roughly similar to the SS design configuration.

(2) Staufenbiel and Borg (1987) found similar results for ellipses constructed in WH and SS designs. Using the city-block metric and the KYST program with an ordinal as well as an interval MDS model, the proximities of both types of ellipses could be explained with low Stress by different

configurations. These configurations were related to the WH or the SS design configuration by monotonic adjustments of the width and height dimension or by the size and shape dimension, respectively. The particular configuration computed by the KYST program was a function of the initial configuration, as observed above. Solutions that were not either roughly WH or SS consistent, however, did not result when random starts were chosen. Using confirmatory MDS to enforce solutions that were perfectly consistent with either a WH or an SS model confirmed that either dimension system allows one to explain the data well.

(3) Schönemann, Dorcey, and Kienapple (1985), Schönemann and Lazarte (1987), and Lazarte and Schönemann (1991) studied whether it makes sense to aggregate individual proximities in the first place. They concluded that such aggregation "was unjustified because distinct strategy groups were found. Some subjects used mainly height and width, others mainly area and shape, and still others mainly shape alone to form their dissimilarity ratings" (Schönemann, 1994, p. 156).

(4) A closer look at the raw data at the subject level also showed that most ratings were *subadditive* in the sense that $\delta(x,y) + \delta(y,z) > \delta(x,z)$. This relation is interesting if x, y, and z differ on one dimension only, because for triples where y lies between x and z one should expect that $\delta(x,y) + \delta(y,z) \approx \delta(x,z)$, provided one takes the data seriously as they come and does not allow for transformations such as adding some constant.[6] Subadditivity is also evident in Table 17.2 for the unidimensional triple $(1,5,9)$, for example, where one finds $\delta(1,5) + \delta(5,9) = 2.38 + 2.00 > \delta(1,9) = 3.36$. This inequality suggests "a ceiling effect. Once the ceiling was removed (by transforming the data with Fisher's z-transformation), most distortions, such as curvature and non-parallelism of lines, markedly diminished" (Schönemann, 1994, p. 156).

(5) δs that satisfy the triangle inequality "can always be modeled as distances. However, because the observed direct dissimilarities are consistently segmentally subadditive along any possible judgment dimension [of the hypothesized systems; our addition], they cannot be modeled as Minkowski metrics because these metrics assume intradimensional additivity" (Lazarte & Schönemann, 1991, p. 144). (This is shown in the section below.)

(6) One may even question the whole notion of a psychological space—in the sense of a metric geometrical space—where all stimuli are represented at the same time and whose distances, after a possible additional transforma-

[6] If one admits an arbitrary additive constant (interval scale), then subadditivity becomes less meaningful, because one can at least reduce systematic subadditivity for one-dimensional triples by subtracting a sufficiently large constant from all δs (Attneave, 1950). On an interval scale, all such constants are considered admissible and substantively meaningless. One may question, however, whether it is scientifically wise to eliminate an apparent empirical lawfulness—subadditivity of one-dimensional triples—by such transformations.

tion, define the observed dissimilarities. Lazarte and Schönemann (1991), for example, used simple linear models ("psychophysical maps") to relate the observed dissimilarity to physical dimensions of the observed stimulus pair and describe a strategy that is a function of pair-by-pair comparisons. Restle (1959) and Tversky and Gati (1982), among others, proposed alternative (set-theoretical) models that explain similarity judgments on the basis of the common and the distinctive features of the stimuli.

In summary, one notes that building psychological models via MDS is a difficult and complex undertaking. Early MDS applications tended to be over-optimistic, relying almost exclusively on the global loss, Stress, for answering a whole series of questions—such as the appropriateness of a particular mapping of the data into distances, the dimensionality of the psychological space, the true metric of this space, or the validity of the metric space model as such—all at the same time. This clearly was asking too much from one measure.

17.5 Axiomatic Foundations of Minkowski Spaces

Under certain circumstances, one can study the appropriateness of a multidimensional scaling representation in a way that does not rely on computing this representation and therefore does not depend on minimizing a loss function such as Stress. The approach requires that a theory be given that explains the observed proximities as resulting from an additive combination of dimensional differences. For example, one may hypothesize that similarity judgments on pairs of rectangles can be explained by city-block distances of these rectangles with respect to the physical dimensions width and height. A somewhat less demanding theory might allow for a reasonable psychophysical scaling of the width and height dimensions and for a monotonic function that relates the computed distances to dissimilarity ratings ("response function").

Outside psychophysics, such theories may appear too difficult to formulate. Yet, it is nevertheless worthwhile to study what they imply for MDS, because they provide interesting insights into some of the mathematical properties of MDS representations that are not revealed by mere data fitting. Moreover, to view distances as the image of some underlying composition rule for the basic dimensions of the objects corresponds to a common way of interpreting MDS spaces.

Asking for the conditions that must be satisfied by a set of observations (such as dissimilarity judgments) so that they can be mapped (by an ordinal transformation, say) onto some elements of a particular mathematical system (such as distances of a Euclidean space) is the domain of measurement theory (see, e.g., Krantz, Luce, Suppes, & Tversky, 1971; Schönemann & Borg, 1983). Measurement theorists attempt to specify, first of all, condi-

p	ap	bp	cp	dp
q	aq	bq	cq	dq
r	ar	br	cr	dr
s	as	bs	cs	ds
	a	b	c	d

FIGURE 17.10. An $A \times P$ array.

tions (*axioms*) that must be satisfied by the observations or else the desired representation does not exist (with Loss = 0). Such *necessary* conditions may not be sufficient, that is, they may not guarantee the existence of the model representation, and so one typically asks for conditions that are not only necessary but also *sufficient*. The art of measurement theory is to formulate conditions that are not only necessary and sufficient, but that also can be tested on a *finite* set of data assumed to have a relatively *weak* scale level (such as an ordinal one). It is generally easier to axiomatize an assumed infinite set of data for which no transformation is allowed.

The way one sets up such axiomatic systems is to start with the desired representation and check what properties it implies for observations that can be mapped into this model. So, what are the properties that Minkowski spaces imply for its data? For simplicity, we consider the 2D case only. It represents the most interesting case for psychological modeling and can be easily generalized to higher dimensionality.

Let $A = \{a, b, c, \ldots\}$ and $P = \{p, q, r, \ldots\}$ denote the levels of two design factors, A and P, and let $A \times P$ be the set of all combinations $ap, bp, bq, \ldots$ in the factorial design (Figure 17.10). Assume that dissimilarities are collected for pairs of objects characterized by the cells of this design structure. Under what conditions can such dissimilarities (δs) be interpreted as Minkowski distances computed on dimensions that are some monotonic functions of A and P? This is possible only if the δs possess some general properties.

If the δs are ordinal measures, then any monotone transformation is admissible. Yet, even under such transformations, some properties must hold. For example, distances are always *symmetric* and, thus, δs must be symmetric, because there is no admissible transformation (on any scale level) that would turn nonsymmetric δs into symmetric values. Furthermore, the distance of any point to itself is always 0, and any distance between two different points in greater than zero (*minimality*). For ordinal dissimilarities, symmetry and minimality require that $\delta(x, y) = \delta(y, x) > \delta(x, x) = \delta(y, y)$, for all objects x and y. If the δs do not satisfy this condition, they cannot be represented by Minkowski distances or, indeed, by any other distance.

In the following, we discuss further *qualitative* requirements (i.e., conditions involving only notions of order and equality on the δs) and also some properties that can only be partially tested with ordinal data.

Dimensional Axioms

According to Gati and Tversky (1982), a two-way proximity structure is called *monotone* if the following three conditions are satisfied.

The first condition is called *dominance*:

$$\delta(ap, bq) > \delta(ap, aq), \delta(aq, bq); \tag{17.8}$$

that is, any two-way difference always exceeds its one-way components. The second condition is called *consistency*:

$$\begin{gathered}\delta(ap, bp) > \delta(cp, dp) \quad \text{if and only if} \quad \delta(aq, bq) > \delta(cq, dq), \\ \text{and} \\ \delta(ap, aq) > \delta(ar, as) \quad \text{if and only if} \quad \delta(bp, bq) > \delta(br, bs);\end{gathered} \tag{17.9}$$

that is, the ordering of differences on one dimension is independent of the other dimension. The third condition is called *transitivity*:

$$\begin{array}{lrcl} \text{if} & \delta(ap, cq) & > & \delta(ap, bp), \delta(bp, cp), \\ \text{and} & \delta(bp, dp) & > & \delta(bp, cp), \delta(cp, dp), \\ \text{then} & \delta(ap, dp) & > & \delta(ap, cp), \delta(bp, dp). \end{array} \tag{17.10}$$

Condition (17.10) is required to hold also for the second dimension. Transitivity on the δs is equivalent to transitivity of betweenness for the points: $a|b|c$ and $b|c|d$ imply $a|b|d$ and $a|c|d$, where $a|b|c$ means that b lies between a and c (Gati & Tversky, 1982).

The conditions of dominance (17.8), consistency (17.9), and transitivity (17.10) are called *monotonicity* axioms (for a two-way monotone proximity structure) because they specify requirements on the order among the δs.

A more particular property of Minkowski distances is *decomposability*:

$$\delta(ap, bq) = F[g(a, b), h(p, q)], \tag{17.11}$$

where F is a strictly increasing function in two arguments, and g and h are real-valued functions defined on $A \times A$ and $P \times P$, respectively. The arguments g and h are the contributions of the two dimensions to the dissimilarity. If δ is symmetric, g and h satisfy $g(a, b) = g(b, a)$ and $h(p, q) = h(q, p)$. If δ is also minimal, one can set $g(a, a) = 0$ and $h(p, p) = 0$, for all a and p. If g and h can be assumed to be absolute-value functions, then (17.11) can be expressed as *intradimensional subtractivity*:

$$\delta(ap, bq) = F(|X_a - X_b|, |Y_p - Y_q|), \tag{17.12}$$

where X_a and Y_p represent the coordinates of a and p on dimensions X and Y, respectively.

If one assumes that the two dimensions contribute additively to δ, then (17.11) becomes

$$\delta(ap, bq) = F[g(a, b) + h(p, q)], \tag{17.13}$$

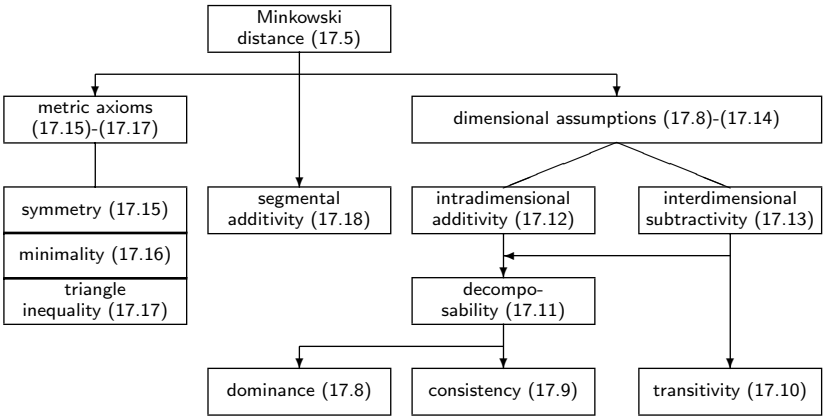


FIGURE 17.11. A hierarchy of conditions necessary for Minkowski distances; entailment is denoted by an arrow.

which is called *interdimensional additivity*. If (17.13) holds, then the actual dissimilarities, not merely their order, are independent of the second dimension, so that:

$$\delta(ap, bp) = \delta(aq, bq) \quad \text{and} \quad \delta(ap, aq) = \delta(bp, bq). \tag{17.14}$$

The conditions (17.8)–(17.14) (sometimes collectively called *dimensional assumptions*) are organized in a hierarchy (Figure 17.11). The diagram shows, for example, that (17.11) implies both (17.8) and (17.9). All Minkowski metrics imply (17.13) and (17.12).

If axiom (17.8), say, is not satisfied by δs that are, by construction or by hypothesis, related to an $A \times P$ design, then these δs cannot be modeled by any Minkowski metric that operates on A and P. This does *not* rule out that the δs can be represented by a Minkowski metric computed on dimensions other than A and P in the same space and/or in a higher-dimensional space. (Indeed, *any* $\delta(i, j)$s, $i < j$, can be represented by Euclidean distances in $n - 2$ dimensions, where n is the number of objects. See Chapter 19.)

Distance Axioms

In addition to the dimensional assumptions, it must also be possible to map the δs into distances d. Distances satisfy three conditions, the *metric axioms*. Two of them, symmetry and minimality, were already discussed above, but are repeated here for completeness. For any points x, y, and z,

$$d(x, y) = d(y, x) \qquad (\textit{symmetry}), \tag{17.15}$$

$$d(x, y) > d(x, x) = d(y, y) = 0 \qquad (\textit{minimality}), \tag{17.16}$$

$$d(x, z) \geq d(x, y) + d(y, z) \qquad (\textit{triangle inequality}). \tag{17.17}$$

Axioms (17.15)–(17.16), in practice, are almost never testable, simply because one rarely collects a complete matrix of δs. Axiom (17.17) can be trivially satisfied in all MDS models that allow at least an interval transformation of the data: one simply determines the triangle inequality that is violated most and then finds a constant c that, when added to every δ in this inequality, turns the inequality into an equality; the same c is then added to every δ, an admissible transformation for interval-scaled δs.

Segmental Additivity Axiom

Minkowski distances assume a dimensional structure that restricts the choice of such additive constants c, because the triangle inequality becomes an equality for points that lie on a straight line in psychological space. That is, for any three points x, y, and z that are ordered as $x|y|z$ on a straight line (such as a dimension), *segmental additivity* is satisfied:

$$d(x, z) = d(x, y) + d(y, z). \tag{17.18}$$

Minkowski Space Axioms in Practice

Tversky and Krantz (1970) have shown that segmental additivity in conjunction with the dimensional assumptions and the metric axioms imply the Minkowski distance. If one wants to test the dimensional conditions (17.8)–(17.14) on real (2D) data, one has to specify the $A \times P$ structure that supposedly underlies the δs (see, e.g., Krantz & Tversky, 1975; Tversky & Gati, 1982; Schönemann & Borg, 1981b).

Staufenbiel and Borg (1987) tested some of these conditions for ellipses constructed in designs analogous to the above WH and SS designs for rectangles. Their data are interesting because they also collected similarity judgments on pairs of identical stimuli, which allow one to test the minimality requirement. It was found that minimality was satisfied for data aggregated over subjects in the sense that $\delta(i, i) < \delta(i, j)$, for all $i \neq j$. Tests of the triangle inequality showed marked subadditivity. Subadditivity correlated highly with violations of minimality on the subject level: these subjects seemed to avoid using the category "0 = equal, identical" on the rating scale, thus, in effect, always adding a positive constant to each δ. Tests of the equality requirements (17.14) showed that they were satisfied in only 20% of the cases. However, the violations revealed no particular systematic pattern and, thus, could be explained as largely due to error.

17.6 Subadditivity and the MBR Metric

Subadditivity of dissimilarities is a frequently observed phenomenon. If the δs are judgments on a rating scale, there are various ways to explain

why $\delta(x,y) + \delta(y,z) > \delta(x,z)$ might occur even for triples (x,y,z) that differ on one dimension only. One possibility was offered by Staufenbiel and Borg (1987), who argue that respondents tend to stay away from the lower bound of the scale, thus in effect adding a positive constant to all distance estimates (see item (2) in Section 17.4). Another, or possibly additional, explanation concentrates more on the upper bound, which makes it impossible for the respondent to generate huge dissimilarities. Thus, if $\delta(x,y)$ is rated as quite different, and $\delta(y,z)$ is also rated as quite different, then the respondent tends to run out of possibilities to properly express the extent of the difference of x and z. Because of upper response bounds, "the subject therefore has to contract his response in a continuous fashion, more so for larger than for smaller arguments" (Schönemann, 1982, p. 318). Even with unbounded response scales, subjects typically underestimate large differences (Borg & Tremmel, 1988). The MBR metric (*metric for monotone-bounded response scales*) proposes a hypothesis on how numerical dissimilarities—not just some monotone transformation of them—might be generated under such upper-bound conditions. Let us consider the 2D case and assume that u is the upper bound. The MBR metric of Schönemann (1982) predicts that, given two stimuli, x and y, and given their differences on the *physical* dimensions, Δ_1^* and Δ_2^* (measured in the metric of the observations), it holds that

$$\delta(x,y) = d_M^*(x,y) = \frac{\Delta_1^* + \Delta_2^*}{1 + \Delta_1^*\Delta_2^*/u^2}, \quad 0 \le d_M^* \le u. \tag{17.19}$$

The numerator of the composition rule on the right-hand side of this formula is the city-block metric. The denominator is a contraction factor that ensures that the distance of x and y does not exceed the upper bound u when either Δ_1^* or Δ_2^*, or both, are close to it. This upper bound may be experimenter-imposed ("Please tell me the dissimilarity on a scale from 0 to 9."), but it may also be self-imposed by the subjects (e.g., as a consequence of their laziness to generate best-possible answers) or imposed by nature (e.g., in form of limitations of the subjects' cognitive capacities). The proper value for u is therefore open to some experimentation. Simple specifications for u in practice are to set it equal to the greatest category of the response scale or to the greatest observed dissimilarity. However, Lazarte and Schönemann (1991) found that "within subjects, the MBR with a slightly reduced upper bound was optimal in restoring additivity among collinear points" (p. 144). Formally, one notes that "permitting [u] to vary across subjects, one obtains a one-parameter family of subject-specific MBR's" (Schönemann et al., 1985, p. 6).

To apply the MBR in practice, one first expresses all observations relative the upper bound u (which need not be the same for all subjects). Dividing the dissimilarities by u, formula (17.19) simplifies to a standardized version,

$$\delta(x,y) \quad = \quad d_M(x,y) = \frac{\Delta_1 + \Delta_2}{1 + \Delta_1\Delta_2}, \quad 0 \le a, b, d_M \le 1. \tag{17.20}$$

This formula can be further simplified by applying the hyperbolic tangent transformation (Schönemann, 1983). This yields

$$\begin{aligned} d_M &= (\Delta_1 + \Delta_2)/(1 + \Delta_1\Delta_2) \\ &= [\tanh(u) + \tanh(v)]/[1 + \tanh(u)\tanh(v)] \\ &= \tanh(u + v), \end{aligned} \tag{17.21}$$

where $\Delta_1 = \tanh(u)$ and $\Delta_2 = \tanh(v)$. Hence,

$$\tanh^{-1}(d_M) = u + v. \tag{17.22}$$

This offers a way for testing the model: preprocessing the given dissimilarities by applying the inverse hyperbolic tangent should "linearize" the data (expressed as proportions to some upper bound such such as the greatest category on the response scale[7]) so that they can be explained by a simple city-block distance

The MBR metric may strike one as a rather odd composition rule. Should one understand it as a model for how dissimilarity judgments are actually generated? Schönemann (1990) suggests that subjects first compute a city-block metric and then do some contraction to fit it into the bounded rating scale. However, he adds: "We do not expect subjects to do this literally, but we know they must make some contracting adjustment if they want to use the simple city-block addition rule" (p. 154).

One may want to think of alternatives to the MBR metric that seem more plausible as composition rules. One example is the rule

$$f(x, y) = \Delta_1 + \Delta_2 - \Delta_1\Delta_2, \quad 0 \leq \Delta_1, \Delta_2, f(x, y) \leq 1. \tag{17.23}$$

This function yields values that are very similar to MBR distances but always somewhat smaller. But what are the formal properties of these composition rules? One property that can be proved is that

$$\max(\Delta_1, \Delta_2) \leq f(x, y) \leq d_M(x, y) \leq \Delta_1 + \Delta_2, \tag{17.24}$$

so that the two composition rules lead to values that lie between the two extreme metrics of the Minkowski family, the dominance distance and the city-block distance.

Formally, though, the MBR distance has some nice additional properties. Circles in the MBR plane have a peculiar resemblance to circles in different Minkowski planes. Namely, circles with small radius closely resemble city-block circles (see Figure 17.3), and the larger the radius, the more they

[7]Note that the value for the upper bound b must be chosen such that all distance estimates fall into the half-open interval $[0, 1)$. This is required to make sure that the tangent function exists everywhere. Hence, one proper choice for b is $\max(\delta) + \varepsilon$, where ε is "a small constant" (Schönemann et al., 1985).

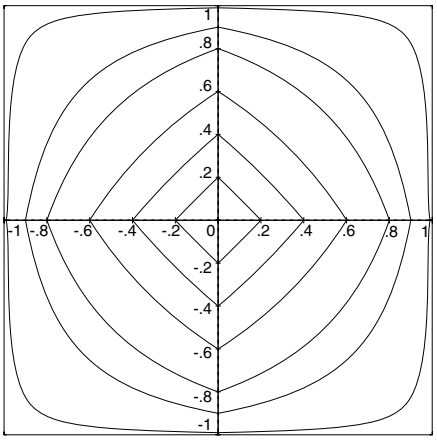


FIGURE 17.12. Circles in the MBR plane, with radii 0.20, ..., 0.90, 0.99.

approximate Euclidean circles, before they asymptotically tend towards dominance circles. This is shown in Figure 17.12. Thus, the shape of a circle in the MBR plane depends on its radius. MBR distances, in other words, emulate the various Minkowski distances depending on the size of the distance: relatively small MBR distances are like city-block distances, large MBR distances are like dominance distances, and intermediate MBR distances are like Euclidean distances.

But does the MBR distance solve the subadditivity problem? Tversky and Gati (1982) report evidence that shows that subadditivity may not affect all triples in a set of objects to the same extent. Dissimilarities were collected for a series of simple 2D stimuli, using different methods of assessment. Three types of stimulus triples were distinguished: corner triples [such as (ap, aq, bq) in Figure 17.10], unidimensional triples [such as (ap, aq, ac)], and two-dimensional triples [such as (ap, aq, ar)]. In uni- and two-dimensional triples, all stimuli differ on the same number of dimensions. Geometrically, such triples lie on a straight line in the design space (collinear points). In corner triples, two pairs differ on one dimension only, and one pair on both dimensions. City-block distances are additive in all cases, but Euclidean distances are subadditive for corner triples and additive for collinear triples. (Under nonlinear transformations of the dimensions, unidimensional triples, in any case, remain collinear.) The observed dissimilarities, then, were almost additive for corner triples, but clearly subadditive for both uni- and two-dimensional triples.

It could be argued that the data are only interval-scaled so that they can be admissibly transformed by adding a constant, for example. Indeed, by subtracting an appropriate constant from all dissimilarities, one could produce values that are, more or less, segmentally additive for the collinear triples. This transformation would, however, make the corner triples *super-*

additive, so that the direct dissimilarity between two points such as *ap* and *bq* in Figure 17.10 becomes *larger* than the sum of its intradimensional differences, which is impossible to model by any distance function. MDS analyses with the program KYST thus showed the least Stress for $p < 1$ for all data sets (except for "color"). With $p < 1$, however, the Minkowski formula (17.5) does not yield a distance. Tversky and Gati (1982) took this finding as supportive for a nongeometric ("feature-matching") model of psychological similarity.

17.7 Minkowski Spaces, Metric Spaces, and Psychological Models

In summary, one may question the ultimate validity of Minkowski spaces for modeling psychological similarity. Indeed, even the much wider class of metric spaces (i.e., sets with distance functions that relate their elements) may be inappropriate, because dissimilarities may systematically violate the symmetry requirement, for example. In this situation, one has four alternatives: (a) give up distance models altogether, as Tversky and Gati (1982) and Gati and Tversky (1982) recommend; (b) modify the distance models by additional notions to make them more flexible (see, e.g., Krumhansl, 1978); (c) possibly drop the restriction to Minkowski spaces and also consider other geometries such as curved spaces (see, e.g., Lindman & Caelli, 1978; Drösler, 1979); and (d) study the conditions under which Minkowski models are likely to be bad or good models of similarity.

The last route is, in fact, necessary for any modeling attempts, because no model is valid without bounds. In this sense, research by Tversky (1977) is relevant. He reports some examples, conditions, and set-theoretical models that allow one to predict when the general distance axioms can be expected to be violated in dissimilarity judgments. For example, symmetry should not hold if one object is a prototype and the other one a variant of this prototype, just as an ellipse is an "imperfect" circle. In that case, the variant should be judged as more similar to the prototype than vice versa. The triangle inequality should be violated if the similarity judgments are based on different criteria. For example, although Jamaica may be judged similar to Cuba, and Cuba is seen as similar to Russia, Jamaica is not seen as similar to Russia at all. (The criteria of similarity in this example could be geographic closeness in the first case and political alignment in the second.) In spite of such counterexamples, the distance axioms are often satisfied in practice. The counterexamples suggest conditions when this should not be the case.

More generally, such fine-grained studies into the foundations of MDS as a psychological model show how one could proceed in cumulative theory building, beginning with exploratory studies on convenient stimuli such as

nations (see Chapter 1), proceeding to efforts where one explicitly models judgments for well-designed stimuli such as rectangles, and finally turning to the axiomatic foundations of a particular model.

Studying well-designed stimuli does not have to limit itself to simple contrived stimuli such as rectangles, for example. Steyvers and Busey (2000) study similarity ratings on extremely complex stimuli, namely faces. They comment on the method to collect global ratings of similarity on pairs of faces and then analyzing these data as follows: "The resulting MDS solutions ... can give valuable insights about the way faces are perceived, and sometimes form a useful basis for modeling performance in recognition and/or categorization tasks" (p. 116). However, "this approach explicitly ignores the physical representation of the features comprising the faces. In this purely top-down approach, the multidimensional representations are sometime difficult to relate back to the physical stimulus" (p.116). To remedy this problem, they suggest a complementary *bottom-up approach*, which offers a way to predict the usual similarity ratings for faces on the basis of studying, via MDS, the structure of proximities derived from a large number of physical measurements on these faces (e.g., eye width, eye separation, or nose length), possibly even the vectors containing the light intensities of all the pixels of an image of each face. Using this methodology, they conclude, for example, that facial adiposity (from narrow and skinny to wide and pudgy) and age (from young to old) are major dimensions of the perceived similarity of faces.

17.8 Exercises

Exercise 17.1 Consider the data in Table 1.4 on p. 12.

(a) Repeat the two-dimensional MDS analysis that led to Figure 1.7 using an ordinal MDS approach and city-block distances.

(b) Repeat the MDS analysis using an explicit starting configuration with coordinates as shown in Figure 1.6. Compare the solutions with and without an external starting configuration. Discuss using such an external starting configuration. Is it justified?

(c) Repeat the MDS analysis with $p = 2$. Compare the $p = 2$ solution to the one computed with the city-block metric both in terms of the configuration and in terms of the Stress value.

(d) Specify the set of admissible transformations for the city-block and the Euclidean solutions.

Exercise 17.2 Consider Table 17.2 on p. 374.

(a) Check the dissimilarity ratings in the lower-half matrix for subadditivity and find the intradimensional triple and the corner triple that violate subadditivity most.

(b) Apply the MBR theory to these data. For this you first have to transform the data so that they lie in the half-open interval [0,1). One reasonable way of doing this in this particular case is to divide all values by the maximal value of the rating scale (i.e., by 9). Then, use the inverse hyperbolic tangent function. Finally, check whether the transformed data can be represented in a 2D city-block plane with lower Stress than without this transformation, using linear MDS in both cases.

(c) Plot the original dissimilarity ratings from Table 17.2 against the transformed data. Describe the effect of the hyperbolic tangent transformation on the values.

(d) Discuss the transformation that maps the dissimilarities into the half-open interval [0,1). This mapping expresses the original dissimilarities as proportions relative to an upper bound b. Dividing the dissimilarities by the greatest observed dissimilarity value does not strictly achieve a mapping into the half-open interval $[0, 1)$. The upper bound value b must at least be "slightly" greater than the greatest dissimilarity. Why? (Hint: Note the "open" in half-open!)

(e) Discuss the consequences of choosing a relatively small upper-bound value b or a huge value for b, where "small" and "huge" means "relative to the size of the dissimilarities." How do such choices of b affect the following hyperbolic tangent transformation?

(f) Experiment with a few different choices for upper bounds b that are slightly greater (say, 0.1 to 0.000001) than the greatest observed dissimilarity. Test out how such different choices of b affect the MDS solutions of the rescaled data (see Borg & Staufenbiel, 1986).

(g) Check whether the dissimilarities in Table 17.2 provide evidence that subadditivity affects corner triples, unidimensional triples, and two-dimensional triples in the sense of Tversky & Gati to a different extent.

Exercise 17.3 Consider the data matrix below (Schönemann et al., 1985). It shows relative dissimilarity ratings (averaged over 20 subjects) for nine different rectangles. The physical width–height design characteristics (in cm) of the rectangles are shown in the first two columns.

Width	Height	No.	1	2	3	4	5	6	7	8	9
2.7	3.1	1	0	0.388	0.491	0.405	0.613	0.771	0.649	0.769	0.865
5.4	3.1	2	0.388	0	0.305	0.660	0.466	0.527	0.749	0.630	0.752
8.1	3.1	3	0.491	0.305	0	0.802	0.655	0.369	0.849	0.777	0.585
2.7	5.4	4	0.405	0.660	0.802	0	0.508	0.669	0.358	0.583	0.757
5.4	5.4	5	0.613	0.466	0.655	0.508	0	0.397	0.594	0.447	0.530
8.1	5.4	6	0.771	0.527	0.369	0.669	0.397	0	0.777	0.608	0.369
2.7	8.1	7	0.649	0.749	0.849	0.358	0.594	0.777	0	0.474	0.660
5.4	8.1	8	0.769	0.630	0.777	0.583	0.447	0.608	0.474	0	0.377
8.1	8.1	9	0.865	0.752	0.585	0.757	0.530	0.369	0.660	0.377	0

(a) Plot the design space of the rectangles. Sketch the nine rectangles.

(b) Scale the dissimilarities with and without a rational starting configuration. What evidence do you find that the respondents generated their dissimilarities from a width–height dimension system?

(c) Check the dissimilarities for subadditivities.

(d) Preprocess the data by the MBR logic and then repeat the MDS scalings. Do you find theoretically interesting differences?

Exercise 17.4 Consider the data in Table 1.4 on p. 12. Theoretical considerations suggest that they were generated by city-block composition of two intradimensional differences. Observe what happens when you scale these data with the "incorrect" Euclidean distance in 2D and in 3D, using the design configuration in Figure 1.6 as a starting configuration.

Exercise 17.5 Construct a grid of points in the plane (as in Figure 19.3, e.g.) and measure their city-block distances. Then scale these distances in Euclidean 3D space, using

(a) ordinal MDS,

(b) interval MDS, and

(c) classical scaling.

Carefully study the resulting configurations in the planes spanned by the principal components. Discuss the effects of using the improper Euclidean distance function with MDS models that allow for arbitrary monotone transformations, linear transformations, and ratio transformations of the data, respectively.

18

Scalar Products and Euclidean Distances

Scalar products are functions that are closely related to Euclidean distances. They are often used as an index for the similarity of a pair of vectors. A particularly well-known variant is the product-moment correlation for (deviation) scores. Scalar products have convenient mathematical properties and, thus, it seems natural to ask whether they can serve not only as indices but as models for judgments of similarity. Although there is no direct way to collect scalar product judgments, it seems possible to derive scalar products from "containment" questions such as "How much of A is contained in B?" Because distance judgments can be collected directly, but scalar products are easier to handle numerically, it is also interesting to study whether distances can be converted into scalar products.

18.1 The Scalar Product Function

The earliest papers on MDS paid more attention to scalar products than to distances. The reason was simply computational. Given a matrix of scalar products, it is easy to find a representing MDS configuration for them. In fact, this MDS problem can be solved analytically (see Chapter 7).

In the usual geometry, the scalar product b_{ij} of the points i and j is defined as the sum of the products of the coordinates of i and j:

$$b_{ij} = \sum_{a=1}^{m} x_{ia} x_{ja}. \tag{18.1}$$

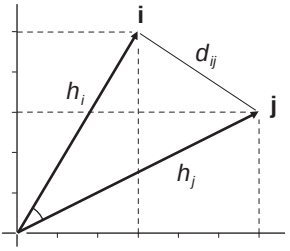


FIGURE 18.1. Illustration of two vectors (arrows) and of the Euclidean distance between their endpoints.

To illustrate, Figure 18.1 shows three distinct points in the $X-Y$-plane: the origin O with coordinates $(0,0)$, the point i with coordinates $(x_{i1}, x_{i2}) = (3,5)$, and the point $j = (6,3)$. The points i and j are depicted as endpoints of vectors ("arrows") emanating from O.

Once a particular point in a plane is chosen as the origin, then all points can be conceived as vectors bound to this origin, and vice versa. So one can alternate between the notions of point and vector whenever it seems useful to do so. Notationally, the origin to which the vectors are bound is not explicitly shown, so one simply writes a bold **j** for the vector from O to j.

For the scalar product of i and j in Figure 18.1, we find $b_{ij} = 3 \cdot 6 + 5 \cdot 3 = 33$. But formula (18.1) can also be used on each vector alone. For example, for **j**, one finds $b_{jj} = 6 \cdot 6 + 3 \cdot 3 = 45$. This corresponds to the length of **j**. For the length of j, one often writes h_j. So, $h_j = \sqrt{b_{jj}} = d_{Oj}$.[1]

Some of the relations between a scalar product, the lengths of its vectors, and the angle subtended by them may be seen by considering the triangle formed by the points O, i, and j in Figure 18.1. The lengths of its sides are easily found by using the Pythagorean theorem, which yields $h_i^2 = 6^2 + 3^2 = 45, h_j^2 = 5^2 + 3^2 = 34$, and $d_{ij}^2 = (6-3)^2 + (5-3)^2 = 13$. The angle α in Figure 18.1 is computed by using the cosine law for a triangle with sides a, b, and c, which says that $a^2 = b^2 + c^2 - 2bc\cos(\alpha)$, where α is the angle between the sides b and c. Thus, $d_{ij}^2 = h_i^2 + h_j^2 - 2h_i h_j \cos(\alpha)$, and solving this equation for $\cos(\alpha)$ yields

$$\cos(\alpha) = \frac{h_i^2 + h_j^2 - d_{ij}^2}{2h_i h_j}, \tag{18.2}$$

[1]The term h_j is the image of a scalar function on the vector argument **j**. The function is the Euclidean norm $\|\mathbf{j}\|$ (see Chapter 7). Norm functions on vectors have general properties that are similar to those of distances between points, but, unlike distances, they cannot be defined on just any set. Rather, they require sets that possess the properties of *vector spaces* so that operations such as + in (7.2) have a particular well-defined meaning (see Chapter 19).

TABLE 18.1. Scalar product matrix for three variables.

Variable	x	y	z
x	25	15	20
y	15	25	12
z	20	12	25

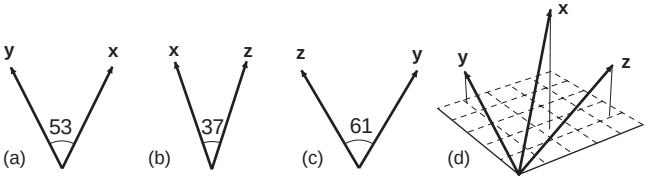


FIGURE 18.2. Vector configurations for scalar products in Table 18.1; panels (a), (b), and (c) show combinations of pairs of vectors; panel (d) results from combining panels (a), (b), and (c).

or, in more detail,

$$\cos(\alpha) = \frac{\sum_a x_{ia}^2 + \sum_a x_{ja}^2 - \sum_a (x_{ia} - x_{ja})^2}{2(\sum_a x_{ia}^2)^{1/2}(\sum_a x_{ja}^2)^{1/2}}, \tag{18.3}$$

which simplifies to

$$\cos(\alpha) = \frac{\sum_a x_{ia}x_{ja}}{(\sum_a x_{ia}^2 \sum_a x_{ja}^2)^{1/2}} = \frac{b_{ij}}{h_i h_j}. \tag{18.4}$$

(This is the formula for the product-moment correlation coefficient for deviation scores, which can therefore be interpreted as an angle function of the data vectors **i** and **j**.) The scalar product b_{ij} is thus

$$b_{ij} = h_i h_j \cos(\alpha). \tag{18.5}$$

One notes that the value of b_{ij} depends on three arguments: the length of the vector **i**; the length of **j**; and the angle α subtended by **i** and **j**. If $\alpha = 0$, then $\cos(\alpha) = 1$, and the scalar product is equivalent to the squared Euclidean distance between the origin O and the endpoint of vector **i**.

If all scalar products are given for a set of vectors, then it is possible to construct the corresponding vector configuration from these values. This was shown algebraically in Section 7.9, but it is also easy to understand geometrically. Consider an example. Assume that Table 18.1 is a matrix whose entries are scalar products.[2] Then, the values in the main diagonal

[2]The matrix does not obviously violate this assumption. It is symmetric and its main diagonal elements are nonnegative. Symmetry must be satisfied by all scalar product

tell us that the vectors $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$ all have the same length, $h_x = h_y = h_z = \sqrt{25} = 5$. To construct the vector configuration, we need to know the angles between each pair of vectors. The angle between $\mathbf{x}$ and $\mathbf{y}$, say, is found from $b_{xy} = 15$. By formula (18.5), $b_{xy} = 5 \cdot 5 \cdot \cos(\alpha) = 15$ or $\cos(\alpha) = 3/5$, and this yields $\alpha = 53.13°$. Figure 18.2a shows the resulting configuration of the two vectors $\mathbf{x}$ and $\mathbf{y}$. Proceeding in the same way for the other vector pairs, we arrive at Figures 18.2b and 18.2c. If everything is put together, we find the configuration of all three vectors, which requires a 3D space (Figure 18.2d).

18.2 Collecting Scalar Products Empirically

A scalar product is a more complex measure than a distance. In terms of points in a space, it not only involves the endpoints i and j but also a third point that serves as the origin. Does such a complicated function serve any other purpose than a purely mathematical one? Is it possible to translate all or some of the properties of scalar products into real questions on the similarity of two objects, i and j?

Building Scalar Products from Empirical Judgments

It would be futile, of course, to ask a subject to directly rate the "scalar product" of two stimuli i and j, although we could certainly ask him or her to rate their "distance". A scalar product is a notion that has no intuitive meaning. Ekman (1963), therefore, suggested an indirect approach, asking for two particular judgments, which are then combined to form a scalar product. Consider Figure 18.3. Let vectors $\mathbf{i}$ and $\mathbf{j}$ in panel (b) represent two stimuli such as the colors blue and red. We could ask the subject to assess the ratio c_{ij}/h_j, that is, judge the length of the projection of $\mathbf{i}$ onto $\mathbf{j}$ relative to the length of $\mathbf{j}$. Concretely, this could be operationalized in a question like, "How much of this blue is contained in this red?" to which the subject may answer by giving a percentage judgment (such as "80%"). The question is then inverted to, "How much of this red is contained in this blue?" and the subject's answer is taken as an assessment of the ratio c_{ji}/h_i.

---

matrices, because it follows from (18.1) that $b_{ij} = b_{ji}$. Moreover, all elements in the main diagonal must be nonnegative, because, for $i = j$, formula (18.1) is but a sum of squared numbers. These conditions do not guarantee, however, that the matrix is a scalar product matrix. A matrix $\mathbf{B}$ is a (Euclidean) scalar product matrix only if it can be decomposed into the matrix product $\mathbf{XX'}$, with real $\mathbf{X}$ (see Chapter 19). If this is possible, then each element of $\mathbf{B}$ satisfies formula (18.1).

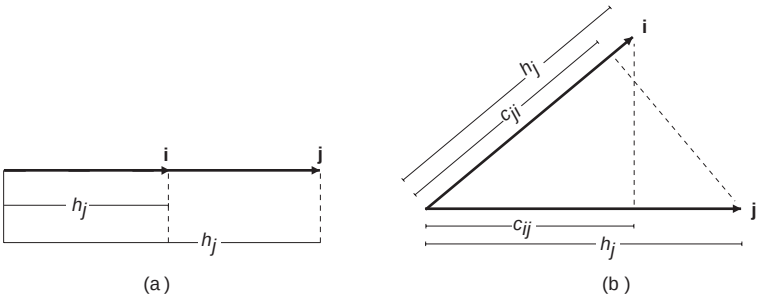


FIGURE 18.3. Vector representations of two stimuli, i and j.

We show what can be done with such data. To simplify the notation, let

$$v_{ij} = c_{ij}/h_j, \tag{18.6}$$
$$v_{ji} = c_{ji}/h_i. \tag{18.7}$$

Note that what one observes are the v-values; the expressions on the right-hand side of the equations are how these data are explained by the vector model. Note also that one can assume that the v-values are nonnegative because a score of zero is the least possible containment. In terms of the model, the c-terms are projections,

$$c_{ij} = h_i \cdot \cos(\alpha), \tag{18.8}$$
$$c_{ji} = h_j \cdot \cos(\alpha). \tag{18.9}$$

Thus,

$$\cos(\alpha) = (v_{ij} \cdot v_{ji})^{1/2}, \tag{18.10}$$
$$h_i/h_j = (v_{ij}/v_{ji})^{1/2}. \tag{18.11}$$

If v_{ij} and v_{ji} can be estimated empirically, we can derive from them (a) the angle α between the vectors **i** and **j** via (18.10), and (b) the ratio of the lengths of these vectors via (18.11). If one of the vectors is fixed arbitrarily (say, by setting $h_i = 1$), then a unit for scaling is given, and the vector configuration can be constructed.

Consider some data (Ekman, 1963). Six monochromatic lights of equal brightness served as stimuli. Their wavelengths were 593, 600, 610, 628, 651, and 674 nm; that is, the lights were in the red-yellow range. All possible pairs of lights were projected onto a screen, and 10 subjects were asked for "contained-in" judgments on each pair. The averages of the observed values are presented in Table 18.2. This data matrix is denoted as $\mathbf{V}$. From $\mathbf{V}$ we can derive a matrix $\mathbf{H}^{(2)}$ that contains the quotients v_{ij}/v_{ji} as its elements (Table 18.3). These values are, by formula (18.11), the quotients of the squared lengths of our six vectors. For example, the second element in the first row is $v_{12}/v_{21} = h_1^2/h_2^2 = .94/.95 = .99$. Summing over all elements

TABLE 18.2. Averaged v-data for colors with wavelengths 593, ... , 674 nm (Ekman, 1963).

nm	593	600	610	628	651	674
593	1.00	.94	.67	.22	.08	.04
600	.95	1.00	.80	.31	.16	.06
610	.63	.75	1.00	.78	.56	.38
628	.21	.37	.78	1.00	.81	.72
651	.14	.23	.61	.85	1.00	.86
674	.07	.13	.40	.80	.90	1.00

TABLE 18.3. $\mathbf{H}^{(2)}$ matrix, based on v-values in Table 18.2; $\mathbf{H}^{(2)} = (v_{ij}/v_{ji})$.

nm	593	600	610	628	651	674	First Est. h_i^2	Sec. Est. h_i^2
593	1.00	.99	1.07	1.05	.56	.62	5.29	4.11
600	1.01	1.00	1.07	.85	.70	.47	5.10	3.93
610	.94	.93	1.00	1.00	.91	.94	5.72	3.87
628	.95	1.18	1.00	1.00	.95	.89	5.97	4.13
651	1.79	1.44	1.09	1.05	1.00	.96	7.33	4.33
674	1.61	2.14	1.06	1.11	1.05	1.00	7.97	4.58

of row i of $\mathbf{H}^{(2)}$ yields, symbolically, $\sum_{j=1}^{n} h_i^2/h_j^2 = h_i^2 \sum_{j=1}^{n}(1/h_j^2)$. Hence, this sum always involves a constant term, $\sum_{j=1}^{n}(1/h_j^2)$. Ekman (1963) suggested simply setting this term equal to 1, thus introducing a scaling norm for the vectors. With $\sum_{j=1}^{n}(1/h_i^2) = 1$, we get $h_1^2 = 5.29$, $h_2^2 = 5.10$, and so on, as shown in Table 18.3.

Further Considerations on Constructing Scalar Products

Selecting a scaling norm for the vectors is a trivial matter in the case of error-free data. The simplest choice would be to arbitrarily select one vector and take its length as the norming length for all vectors. With real data, however, Ekman's suggestion for norming by setting $\sum_i(1/h_i^2) = 1$ seems better, because the vector lengths are derived from all data, not just a subset. This should lead to more robust vector-length estimates for fallible data.

Computational accuracy remains a problem in any case, because our estimates rely very much on divisions and multiplications. Such operations are numerically unstable. Consider, for example, the v-values for 593 and 674. Their quotient forms two entries in the $\mathbf{H}^{(2)}$ matrix. For example, we should find the value for the element in row 593 and column 674 of $\mathbf{H}^{(2)}$ from 0.04/0.07. But 0.04/0.07 = 0.57, and not 0.62, as we find in the table. The discrepancy is a consequence of the fact that Table 18.3 reports only

two decimal places. Other small changes in the v-values also render quite different h^2-values.

The reliability of the v-data could also be taken into account. One may argue that very small v-data should be less reliable because they express that the subject felt that the two objects had essentially nothing in common. A subject should, therefore, be relatively indifferent whether, say, one object is said to contain 4% or 3% of the other. Forming ratios of such small contained-in percentages is, however, very much affected by such small changes in the magnitude of the contained-in data. Ekman (1963) therefore suggested skipping such small v-values in estimating the vector lengths. In Table 18.2, he decided to ignore the values in the upper right-hand and the lower left-hand corners of the matrix, because these corners contain relatively many small v-values. The vector lengths were estimated only from the values in the upper 4×4 left-hand and the lower 3×3 right-hand submatrices of $\mathbf{H}^{(2)}$ in Table 18.3. That means, for example, that h^2_{593} results from adding the first four elements in row 593. For h^2_{651}, the last three elements of row 651 are added, and, because this sum involves one element less than before, the sum is rescaled by the adjustment factor 1.45 to yield $h^2_{651} = 4.33$. The factor 1.45 is computed from row 628, where the first four elements add up to 4.13 and the last three to 2.84, giving the ratio $4.13/2.84 = 1.45$. If one compares the results of this estimation (see column "Second Estimates" in Table 18.3) with those obtained before, one notes (apart from the irrelevant differences in the magnitudes of the h^2_i-values) that their proportions are quite different. Whether these values are indeed better estimates is, of course, impossible to say without any further evidence. We note, however, that the estimation approach is obviously not very robust.

The angles for each vector pair could be found in a similar way using (18.10). This would then yield scalar products by (18.5). However, there is a more direct estimation approach that also provides a test. It follows from (18.8) that $b_{ij} = h^2_j \cdot v_{ij}$. Because we also obtain b_{ji} from $h^2_i \cdot v_{ji}$, nothing guarantees that $\mathbf{B}$ is symmetric so that $b_{ij} = b_{ji}$, for all i, j. This provides a test for assessing to what extent the properties derived from the vector model are consistent with the data. We find that there are indeed some asymmetries, for example, $b(593,600) = (3.93)(0.94) = 3.69$, but $b(600,593) = (4.12)(0.95) = 3.91$ (Table 18.4). However, these asymmetries are quite small, as Figure 18.4 makes clear, and can be assumed to lie within the error range. Thus, finally, a scalar product matrix is obtained by averaging these b-values.

Different Vector Lengths and v-Data

One notes that the $\mathbf{V}$-matrix in Table 18.2 is not symmetric; that is, $v_{ij} \neq v_{ji}$ for most i, j. The asymmetries are, however, minor in their magnitudes. Empirically, one also finds cases where asymmetries are substantial. One

TABLE 18.4. Preliminary (nonsymmetrized) **B** matrix for values in Table 18.2.

nm	593	600	610	628	651	674
593	4.12	3.68	2.59	.92	.35	.20
600	3.89	3.93	3.09	1.28	.71	.27
610	2.58	2.92	3.87	3.21	2.42	1.71
628	.87	1.44	3.01	4.13	3.51	3.27
651	.59	.92	2.37	3.51	4.33	3.89
674	.29	.50	1.54	3.30	3.88	4.55

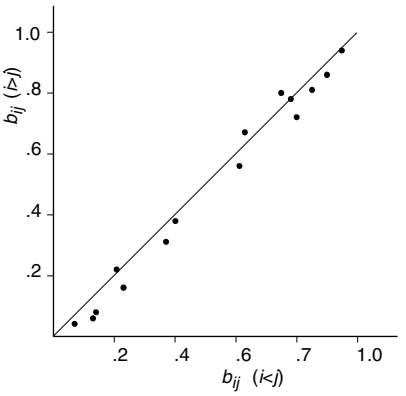


FIGURE 18.4. Scatter plot of scalar products in Table 18.4; points' coordinates on X-axis (Y-axis) are values in upper (lower) half of Table 18.4.

example is a study by Sixtl (1967) on the similarity of different emotional experiences or feelings. He reports that his subjects felt that "wrath" has 83% in common with "aggressiveness", but "aggressiveness" overlaps with "wrath" only to 60%. Within the vector model, such asymmetries of the containment judgments imply different lengths for the representing vectors. This can be seen from Figure 18.3b, where **i** does have more in common with **j** than vice versa. The reason is that **j** is longer and, thus, its perpendicular projection is also longer.

This example also shows that, in the model, symmetric contained-in judgments are not necessary for symmetric scalar products: $b_{ij} = b_{ji}$ implies that $h_j^2 v_{ij} = h_i^2 v_{ji}$. Hence, asymmetries of v_{ij} and v_{ji} judgments can be compensated by the different lengths of **i** and **j**.

Vastly different vector lengths may, on the other hand, lead to a serious problem for the contained-in judgments. Consider Figure 18.3a. Here, $\mathbf{j} = 2 \cdot \mathbf{i}$ and, so, $v_{ji} = 2$. This yields consistent equations: if one sets $h_j = 1$ (units), then $b_{ij} = h_j^2 v_{ij} = (1)^2(1/2) = 0.5$ and $b_{ji} = h_i^2 v_{ji} = (1/2)^2(2) = 0.5$. However, the operationalization, "How much of j is contained in i?" does not work anymore for this case, because it seems impossible that a containment judgment is smaller than 0% or larger than 100%.

But are such cases really impossible? The projection of **j** onto **i** should be longer than **i** itself if **j** contains more of **i** than **i** itself. This case is not quite as paradoxical as it may appear at first sight. A conceivable instance of this situation involves the colors **j** = bright red and **i** = pale reddish, where the latter is but a "pale" instance of the prototypical color. If this situation seems likely, conventional contained-in judgments do not appear to be sufficient to measure scalar products. It would be desirable, for example, to somehow assess the vector lengths independently of any notions of similarity or containment. A set-theoretic approach where objects are equated with feature sets might be a possibility (see, e.g., Restle, 1959).

18.3 Scalar Products and Euclidean Distances: Formal Relations

In an exploratory context, the mapping of v-values into a vector configuration does not have to pass major tests that would allow one to conclude that the model is inappropriate. The only such test is the required rough symmetry of the preliminary **B**-matrix. If this matrix is grossly asymmetric, the model should be dropped as inappropriate.

One could devise further tests though, for example, constraints on the dimensionality of the vector configuration or predictions as to how the vectors should be positioned relative to each other. The more such tests there are, the more can be learned about the data. Ekman, Engen, Künnapas, and Lindman (1964) suggested collecting further measures besides the

contained-in judgments (v-data), and then checking whether everything fits together. Given two stimuli i and j as in Figure 18.1, we would expect that the dissimilarity judgments on i and j could be mapped into the distance d_{ij}, and the v-data would mirror the discussed projection-to-length ratios.

Converting Scalar Products into Euclidean Distances, and Vice Versa

Scalar products and Euclidean distances are closely related; for example, for vectors of constant length, they stand in an inverse monotonic relation to each other, so that if d_{ij} grows, b_{ij} gets smaller, and vice versa. But there is a major difference between scalar products and distances: if the origin of the coordinate system is shifted in space, then the scalar products will also change, whereas the distances remain the same. Expressed in terms of the formulas, we have $b_{ij} = \sum_a x_{ia}x_{ja}$ and $d_{ij}^2 = \sum_a (x_{ia} - x_{ja})^2$ for the old coordinate system. Shifting the coordinate system by the translation vector $(t_1, \ldots, t_m)$, one obtains $b_{ij(t)} = \sum_a (x_{ia}+t_a)(x_{ja}+t_a) \neq b_{ij}$, unless $t_1 = 0, \ldots, t_m = 0$. For distances, on the other hand, one gets $d_{ij(t)}^2 = \sum_a [(x_{ia}+t_a)-(x_{ja}+t_a)]^2 = d_{ij}^2$. However, once some point has been chosen to serve as the origin, we can compute scalar products from distances and vice versa (see also Chapter 12). To see this, consider Figure 18.5. Let point k be the origin. Then, by the cosine theorem,

$$d_{ij}^2 = d_{kj}^2 + d_{ki}^2 - 2d_{kj}d_{ki}\cos(\alpha), \tag{18.12}$$

where α is the angle between the vectors from point k to j and from k to i, respectively. Rearranging (18.12), we find

$$d_{kj}d_{ki}\cos(\alpha) = \tfrac{1}{2}(d_{kj}^2 + d_{ki}^2 - d_{ij}^2), \tag{18.13}$$

which is, by (18.5),

$$b_{ij} = \tfrac{1}{2}(d_{kj}^2 + d_{ki}^2 - d_{ij}^2), \tag{18.14}$$

because d_{kj} and d_{ki} are just the lengths of the vectors $\mathbf{j}$ and $\mathbf{i}$. Thus, we find the scalar product b_{ij} from three distances. Conversely, we find the distance d_{ij} from three scalar products: observing that $d_{kj}^2 = b_{jj}$ and $d_{ki}^2 = b_{ii}$, we have $d_{ij}^2 = b_{ii} + b_{jj} - 2b_{ij}$. Note that the origin k always enters into these conversions.

Typically, one chooses the centroid as the origin, because this point is supposedly more reliable than any point representing a single variable (Torgerson, 1958). This choice should therefore lead to more robust scalar-product estimates. The centroid is the point z with coordinates

$$(z_1, \ldots, z_m) = \left(\frac{1}{n}\sum_{i=1}^{n} x_{i1}, \ldots, \frac{1}{n}\sum_{i=1}^{n} x_{im} \right). \tag{18.15}$$

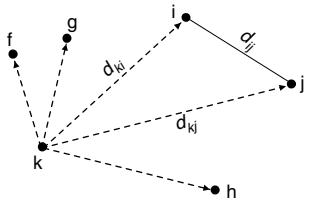


FIGURE 18.5. Defining a vector configuration on the points $f, \ldots, k$ by choosing one point, k, as an origin.

TABLE 18.5. Numerical example for the relation (18.17).

Point	Coordinates		Centered Coordinates		Squared Distances				Scalar Products			
	1	2	1	2	1	2	3	4	1	2	3	4
1	1	2	-1.5	0.75	0	1	8	10	2.81	1.31	-1.69	-2.44
2	2	2	-0.5	0.75	1	0	5	5	1.31	0.81	-1.19	-0.94
3	3	0	0.5	-1.25	8	5	0	2	-1.69	-1.19	1.81	1.06
4	4	1	1.5	-0.25	10	5	2	0	-2.44	-0.94	1.06	2.31
Sum	2.5	1.25	0.0	0.0	19	11	15	17				

With the centroid of all points as the origin, we obtain the scalar product

$$b_{ij} = \sum_a (x_{ia} - z_a)(x_{ja} - z_a), \tag{18.16}$$

because each coordinate is now expressed as a deviation score from the origin z. This expression is transformed into a formula with only distances appearing on the right-hand side, as in (18.14). Such a formula allows one to convert distances—for which empirical estimates are assumed to be given—into scalar products relative to the centroid. Inserting z values into (18.16), one obtains, after some rearrangements of terms,

$$b_{ij} = -\tfrac{1}{2}\left(d_{ij}^2 - \frac{1}{n}\sum_i d_{ij}^2 - \frac{1}{n}\sum_j d_{ij}^2 + \frac{1}{n^2}\sum_i\sum_j d_{ij}^2\right). \tag{18.17}$$

Table 18.5 shows an example for this conversion. Given the squared distances, b_{ij} values are found by first subtracting from each d_{ij}^2 value the mean of row i and column j, then adding to it the mean of all squared distances, and finally multiplying all values by $-\frac{1}{2}$. For example, for points 1 and 2 we get $d_{12}^2 = (1-2)^2 + (2-2)^2 = 1$ from the coordinates. The scalar product relative to the centroid is $b_{12} = -\frac{1}{2}(1 - 19/4 - 11/4 + 62/16) = 1.31$. But this should be the same as computing the scalar product directly by formula (18.5) from the centered coordinates. Indeed, we find $b_{12} = (-1.5)(-0.5) + (0.75)(0.75) = 1.31$.

Table 18.5 also demonstrates that the scalar product between **i** and **j** depends on the origin, whereas the distance of **i** and **j** does not. For example, the scalar product for points 1 and 2 is 5.00 for the raw coordinates but, as we saw, 1.31 for the centered coordinates. On the other hand, the distance $d_{12} = 1.00$ for any origin. Thus, scalar products are in a sense stronger or richer in information than distances, because they depend on the n points of a configuration and, in addition, on an origin. This issue is unrelated to the scale level of the data. Even for absolute proximities, any point of the MDS configuration can be chosen to serve as an origin. Hence, in distance scaling, the origin is, by itself, meaningless, although meaning may be brought in from elsewhere, as, for example, in the radexes in Chapter 5. For scalar-product data, in contrast, the origin necessarily has an empirical meaning. In Figure 18.7, it represents the color gray, and the fact that all points have the same distance from it reflects the equal saturation of the six colors used in the experiment. Consequently, Ekman (1963) interprets the different directions of the color vectors as due to their *qualitative* differences, whereas different vector lengths represent their *quantitative* differences.

18.4 Scalar Products and Euclidean Distances: Empirical Relations

The formal relations between scalar products and distances may be used in empirical research. If one collects both contained-in data (v-data) and also asks the subjects to directly assess the global similarity (s-data) of the objects of interest, it becomes possible to test whether the subjects' proximity judgments can be accounted for by their scalar products. If so, our confidence in the empirical validity of the geometrical models should be increased. The converse, however, is not possible, because one cannot uniquely derive scalar products from given distances due to the arbitrary choice of origin.

A number of researchers have studied whether there exist *empirical* relationships between distance and scalar-product data that allow such two-way conversions. Let us first consider the special case where $h_i = h_j$, for all i, j. Under this equal-length condition, Ekman proposed that the relation $s_{ij} = \cos(\alpha)/\cos(\alpha/2)$ could be shown to hold very well empirically, where $\cos(\alpha) = \sqrt{v_{ij}v_{ji}}$ from equations (18.6)–(18.10). (Both the s- and the v-data were collected on percentage scales, in which 100 meant "identity" for proximity judgments, and "completely contained in" for contained-in judgments.) If such a relation would indeed hold, then we could arrive at a natural origin by converting proximity data into scalar products. This has the advantage that all we need are proximity data, which are much easier to collect. Of course, this should work only if $h_i = h_j$ holds for all i and j, a condition that supposedly is guaranteed by proper instruction

of the subjects. Therefore, in an experiment on the similarity of different emotions, Ekman et al. (1964) asked their subjects "to consider emotions of equal intensity" (p. 532) and "to disregard possible quantitative differences in the intensity of the (emotions) and base their judgments on qualitative characteristics" (p. 533).

It seems hard to evaluate what the results of such experiments mean. Because they involve complicated formal relations and equally complicated instructions, it is impossible to see where things break down. Nevertheless, it is interesting to consider the more general principles from which Ekman derived such relations. He started by studying models for the subjective similarity of stimuli differing on one attribute only. For example, Ekman, Goude, and Waern (1961) report an experiment in which subjects had to assess all possible pairs of different grays (a) with respect to their global similarity on a 10-point scale, and (b) relative to their darkness ratios. The resulting proximity values were divided by 10, and a simple function was found closely describing the relation between ratio and similarity data:

$$s_{ij} = \frac{2h_i}{h_i + h_j}, \quad h_i \leq h_j, \tag{18.18}$$

where h_i and h_j are the values of stimuli i and j on the darkness scale, and s_{ij} is the (rescaled) distance estimate for i and j. In terms of the actual data collection procedure, (18.18) can be written as $s_{ij} = 2/(1 + h_j/h_i)$, with h_j/h_i being the empirical ratio judgment. Because the different gray stimuli on which the relation (18.18) is based do not differ qualitatively, the situation can be best understood by considering Figure 18.3a, where **i** is completely contained in **j**. Also, **i** is, of course, completely contained in itself. Hence, one can interpret the term $2h_i$ in formula (18.18) as an expression for what i and j have in common (e.g., in the sense of their stimulation). The term $h_i + h_j$, on the other hand, expresses what i and j comprise together. What equation (18.18) says, thus, is that the subjective dissimilarity of i and j is given as the ratio[3] of the *communality* and the *totality* of i and j,

$$s_{ij} = \frac{\text{communality of } i \text{ and } j}{\text{totality of } i \text{ and } j} = \frac{K_{ij}}{T_{ij}}. \tag{18.19}$$

We now want to drop the model constraint that **i** and **j** are both collinear (as in Figure 18.3a) and generalize the notions of communality and totality to the higher-dimensional case. One possibility is to set $K_{ij} = c_{ij} + c_{ji}$, because c_{ij} is just that component that **i** shares with **j**, and the converse is

[3]This is similar to feature-set models of stimuli, where communality is equated with the intersection of the object's feature sets, and totality with the union of these sets. The 2 in the numerator of (18.18) could be interpreted as a scaling factor on the similarity judgments.

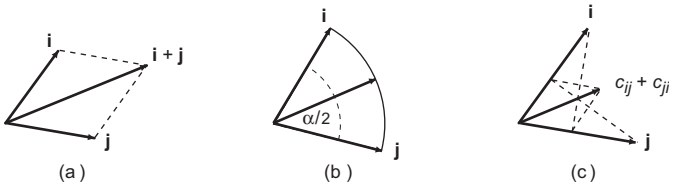


FIGURE 18.6. Illustrations of some notions of communality and totality.

true for c_{ji}. With $T_{ij} = h_i + h_j$ as before, this gives

$$s_{ij} = \frac{c_{ij} + c_{ji}}{h_i + h_j}. \tag{18.20}$$

But $c_{ij} = h_i \cos(\alpha)$ and $c_{ji} = h_j \cos(\alpha)$, so (18.20) is equal to

$$s_{ij} = \cos(\alpha), \tag{18.21}$$

assuming that $c_{ij} \leq h_j$ and $c_{ji} \leq h_i$, that is, that the projections of any vector onto another vector should not be longer than the vector itself (see Section 18.2). This equation means, in terms of the observations, that $s_{ij} = \sqrt{v_{ij} v_{ji}}$.

Unfortunately, this simple hypothesis on the relation between s- and v-values was found to describe empirical correspondences rather poorly. Hence, other proposals were made. Ekman et al. (1964) tested a version of (18.19) in which the totality of i and j was modeled by the vector sum of **i** and **j**, as shown in Figure 18.6a. If $h_i = h_j$, we obtain

$$s_{ij} = \frac{\cos(\alpha)}{\cos(\alpha/2)}. \tag{18.22}$$

But (18.22) can also be interpreted in a different way. The projections of **i** and **j** onto the stimulus vector that lies ("qualitatively") halfway between **i** and **j** (see Figure 18.6b) are $h_i \cos(\alpha/2)$ and $h_j \cos(\alpha/2)$, respectively. If $h_i = h_j$, then the these projections sum to $2h_i \cos(\alpha)$. If this term is used for T_{ij}, one also obtains (18.22). One could also reason that $K_{ij} = [h_i \cos(\alpha) + h_j \cos(\alpha)]/2 = h_i \cos(\alpha)$ and $T_{ij} = [h_i \cos(\alpha/2) + h_j \cos(\alpha/2)]/2 = h_i \cos(\alpha/2)$, which again implies (18.22).

Of course, many more possibilities for K_{ij} and T_{ij} offer themselves if the special constraint $h_i = h_j$ is dropped. Sjöberg (1975) presents the partial overview shown in Table 18.6. It is not surprising that none of these hypotheses has been found to be universally superior. But this leads us back to the question raised at the beginning of this section, and we can now conclude that there is no empirical correspondence between proximities and scalar-product estimates that allows one to derive the latter from the former.

TABLE 18.6. Some formulations for communality K_{ij} and totality T_{ij} of two stimuli. $\bar{K}_{ij}$ in model 4 (Goude, 1972) denotes what is not common to i and j, so that $s_{ij} = 1 - \bar{K}_{ij}/T_{ij}$. The function $\min(a, b)$ selects the smaller of a and b. If $h_i = h_j$, model 2 (Ekman et al., 1964) is equal to formula (18.22). K_{ij} in model 3 (Ekehammar, 1972) is the vector sum of c_{ij} and c_{ji} in Figure 18.6c. Model 1 is by Ekman and Lindman (1961), and model 5 (the *content model*) is by Eisler and Roskam (1977) and Eisler and Lindman (1990).

Model	Communality K_{ij}	Totality T_{ij}
1	$(h_i + h_j)\cos(\alpha)$	$h_i + h_j$
2	$\min[h_j, h_i\cos(\alpha)] + \min[h_i, h_j\cos(\alpha)]$	$[h_i^2 + h_j^2 + 2h_ih_j\cos(\alpha)]^{1/2}$
3	$\cos(\alpha)[h_i^2 + h_j^2 + 2h_ih_j\cos(\alpha)]^{1/2}$	$h_i + h_j$
4	$[h_i^2 + h_j^2 - 2h_ih_j\cos(\alpha)]^{1/2} = \bar{K}_{ij} = d_{ij}$	$[h_i^2 + h_j^2 + 2h_ih_j\cos(\alpha)]^{1/2}$
5	$2 \cdot \min(h_i, h_j)\cos(\alpha)$	$h_i + h_j$

18.5 MDS of Scalar Products

Given a matrix of scalar products, we can compute—by solving $\mathbf{B} = \mathbf{X}\mathbf{X}'$ for $\mathbf{X}$—a configuration $\mathbf{X}$ that represents or approximates the scalar products (see Chapter 7). Because $\mathbf{B} = \mathbf{X}\mathbf{X}' = (\mathbf{X}\mathbf{T})(\mathbf{X}\mathbf{T})' = \mathbf{X}\mathbf{T}\mathbf{T}'\mathbf{X}' = \mathbf{X}\mathbf{X}'$ for $\mathbf{T}\mathbf{T}' = \mathbf{I}$, $\mathbf{X}$ is unique up to an orthogonal transformation $\mathbf{T}$. That is, $\mathbf{X}$ can be rotated and/or reflected freely without affecting the quality of the solution.

An Application on the Color Data

For the symmetrized matrix of Table 18.4, $\mathbf{B} = (\mathbf{B} + \mathbf{B}')/2$, Ekman (1963) reports the point coordinates in Table 18.7. The column $\widehat{h}_i^2$ shows the squared length of vector $\mathbf{i}$ in the MDS space; the hat denotes that this length is a reconstruction of the vector length computed directly from the data. For example, using (18.5) with $i = j$, we find for $i = 3$: $(1.29)(1.29) + (-1.47)(-1.47) + (0.15)(0.15) = 3.85$. In Table 18.3, we had concluded that this color's vector should have a length of 3.87, so the 3D MDS configuration comes very close to representing this value accurately. The five other vectors also represent their colors well.

Table 18.7 shows that the vectors are distributed primarily around the first principal axis: the sum of the squared projections onto this dimension is 14.87, and only 7.72 and 1.54 for the second and third principal axes, respectively. Thus, the vector configuration is essentially two-dimensional. The dimensions of this space are principal axes. Therefore, we know that the plane spanned by the first two axes is the best possible approximation to the 3D vector configuration $\mathbf{X}$.

Any coordinate system can be picked to coordinate this plane. Ekman (1963), for example, rotated the principal axes to a *simple structure ori-*

TABLE 18.7. Coordinates of vector configuration for symmetrized scalar products of Table 18.4; PAs are principal axes of 3D representation; D1 and D2 are dimensions of the 2D plane spanned by PA_1 and PA_2 after rotation to simple structure; SS is the sum-of-squares of the column elements; $\widehat{h}_i^2$ is the squared length of the vector in space.

nm	PA_1	PA_2	PA_3	$\widehat{h}_i^2$	D_1	D_2	$\widehat{h}_i^2$
593	1.14	-1.58	0.51	4.04	0.00	1.95	3.79
600	1.29	-1.47	0.15	3.84	0.18	1.95	3.81
610	1.75	-0.59	-0.64	3.81	1.07	1.50	3.40
628	1.82	0.60	-0.55	3.97	1.82	0.58	3.67
651	1.74	0.99	0.14	4.03	1.99	0.22	4.01
674	1.59	1.20	0.70	4.45	1.99	-0.05	3.97
SS	14.86	7.79	1.50	24.14	12.43	10.22	22.65

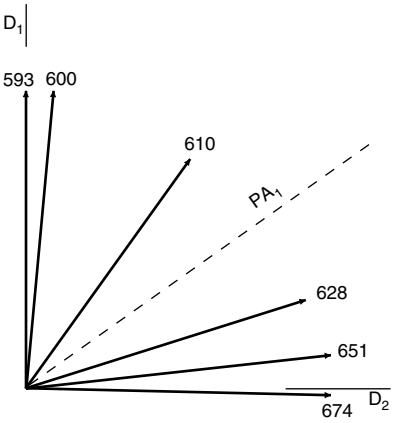


FIGURE 18.7. Vector representation of point coordinates in Table 18.7.

entation (Thurstone, 1947). Simple structure requests that the point coordinates are either very large or very small, and intermediate values are avoided. Table 18.7 shows such simple-structure coordinates resulting from rotating the first two principal axes. For the rotated dimensions, D_1 and D_2, the vector for color 593 has a coordinate value of 1.95 on the second axis, while its projection onto the first axis has zero length. Thus, this vector is collinear with the second axis. For 674, the converse is almost true.

Because the third dimension accounts for so little, we may simply ignore it, and concentrate on the plane spanned by the first two PAs. This plane is presented in Figure 18.7, together with the principal axes and the simple-structure coordinate system that give rise to the values in Tables 18.7. The endpoints of our six color vectors fall almost onto a circle about the origin in the order of their wavelengths. Thus, the perceived dissimilarities in the colors are represented by the different orientations of the vectors. The fact that all vectors have roughly the same length is, according to Ekman (1963), a consequence of the fact that the colors were all matched in brightness and saturation.

How well does Figure 18.7 represent the data? A global answer is provided by comparing the data with the scalar products implied by the given vector configuration. The latter are computed from the first two principal axes in Table 18.7 or, equivalently, from D_1 and D_2 in Table 18.7. One finds, for example, that the reconstructed scalar product is $\widehat{b}(593, 600) = (1.14)(1.29) + (-1.58)(-1.47) = 3.793$, and this is almost the same as the data value $b(593, 600) = (3.68 + 3.89)/2 = 3.785$. Given all $\widehat{b}$ and b values, we can combine them into a global fit measure. One possibility is to use the correlation coefficient of the $\widehat{b}$ and the b values. It yields $r = 0.9805$. Another measure is obtained by adding the squared differences of all $\widehat{b}$ and b values and dividing this sum by the sum-of-squares of the b values. There are no standards for evaluating such a loss function, but it suggests other representation criteria; for example, the $\widehat{b}$ values could be replaced by the rank-image values of the data, defining a loss function for a procedure that maps the data *ordinally* into scalar products. This criterion was used in SSA-III, a program for *nonmetric factor analysis* (Lingoes & Guttman, 1967).

Successive Extractions of Dimensions from Scalar Products

Rather than computing an MDS solution for scalar products in one fixed dimensionality, one can extract this solution dimension by dimension. This allows further tests.

Consider the preliminary **B**-matrix in Table 18.4. One may interpret its asymmetries as essentially due to random noise and thus generate a "better" **B**-matrix by averaging the corresponding b_{ij}- and b_{ji}-values. For the

TABLE 18.8. Upper half with diagonal shows error values $b_{ij} - \widehat{b}_{ij}$ for 3D vector configuration; lower half shows asymmetries, $(b_{ij} - b_{ji})/2$ of values in Table 18.4.

nm	593	600	610	628	651	674
593	.08	-.08	-.01	.05	-.02	-.01
600	.11	.09	-.01	-.02	.01	.01
610	-.01	-.09	.06	-.07	.03	.01
628	-.03	.08	-.10	.16	-.17	.07
651	.12	.11	-.03	.00	.30	-.16
674	.05	.12	-.09	.02	.00	.10

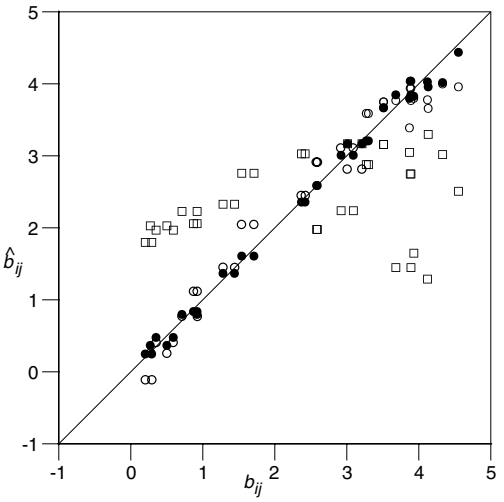


FIGURE 18.8. Plot of residuals of empirical scalar products of Table 18.4 estimated by vector configurations in 1D (squares), 2D (open circles), and 3D (filled circles).

MDS representation of **B**, we now proceed stepwise, extracting one principal component after the other,[4] until the representation seems sufficiently precise. Precision may be defined by requiring that the b_{ij}s do not differ from the scalar products computed on the MDS coordinates, $\widehat{b}_{ij}$, by a magnitude that lies in the range of the asymmetries. Because these asymmetries were assumed to be due to error, further principal components would only represent structure that cannot be distinguished from error. Table 18.8 shows that the criterion is satisfied by a 3D solution. Figure 18.8 shows the residuals of the estimation of the scalar products in 1D, 2D, and 3D. The sum of squared errors is in 1D 63.17, in 2D 2.53, and in 3D .29. According to these criteria, a representation in at most 3D seems adequate, because the error sum-of-squares in 3D is less than the sum-of-squares of the asymmetries in Table 18.8 (= .336).

MDS Representations of v- and s-Data

We have seen that scalar products determine not only n stimulus points but also a unique origin. However, scalar products are often employed in a purely ancillary fashion, because they allow direct computation of a vector configuration by algebraic means. If we begin with distance estimates, we can convert them into scalar products by picking some point to serve as an origin. In that case, the origin has no direct empirical meaning. If the data are scalar products that are not just indices such as correlations computed over persons, say, but measurements constructed from contained-in judgments, then the origin has a meaning, as we have seen for the color data.

But will there be other differences between the MDS solutions derived from scalar-product and distance data? Yes, because of restrictions built into v-judgments. Figure 18.3b shows that the contained-in judgments have a lower bound when the two respective stimulus vectors **i** and **j** are perpendicular, so that $c_{ij} = 0$. In the color circle in Figure 4.1, this is the case, for example, for the colors with wavelengths 674 nm and 584 nm. Indeed, Table 18.2 shows for the very similar stimulus pair (674 nm and 593 nm) that the contained-in rating is almost equal to 0. But what can the subject say when asked to evaluate to what extent the color with wavelength 555 nm is contained in the color 674 nm? For these colors, the respective vectors in the color circle form an obtuse angle. Even more extreme are the complementary colors red and green, which are opposite each other in the color circle: what portion of red is contained in green? Because the subject cannot respond with v-values of less than 0 (unless the procedure is gen-

[4]This extraction process amounts to a spectral decomposition of **B**, $\mathbf{B} = \lambda_1 \mathbf{q}_1 \mathbf{q}_1' + \ldots + \lambda_m \mathbf{q}_m \mathbf{q}_m'$, where λ_i is the ith eigenvalue and $\mathbf{q}_i$ the corresponding eigenvector. See formula (7.12).

eralized in some way), it seems plausible that we end up with $b(674,584) = 0$, $b(674,490) = 0$, and also $b(584, 490) = 0$. But this means that the scaling problem corresponds to a situation like that in Figure 18.2, where all three angles are equal to 90°. To fit the three quarter-circles together necessitates a 3D space. So, for the complete color circle, we should expect a 4D MDS representation if it is based on v-data.

18.6 Exercises

Exercise 18.1 Given a matrix $\mathbf{B}$, how can one check, by matrix computation, whether $\mathbf{B}$ is a scalar-product matrix?

Exercise 18.2 Consider the following geometric problems.

(a) What is the angle between $\mathbf{x} = (2, -2, 1)$ and $\mathbf{y} = (1, 2, 2)$?

(b) What is the projection of $\mathbf{x}$ onto $\mathbf{y}$?

(c) What is the projection of $\mathbf{x}$ onto the plane spanned by (1, 0, 0) and (1, 1, 0)?

Exercise 18.3 What multiple of $\mathbf{a} = (1, 1)$ should be subtracted from $\mathbf{b} = (4, 0)$ to make the result orthogonal to $\mathbf{a}$? Sketch a figure.

Exercise 18.4 Draw two vectors $\mathbf{a}$ and $\mathbf{b}$ in the plane, both emanating from the same origin, such that $\mathbf{a}+\mathbf{b}$ is perpendicular to $\mathbf{a}-\mathbf{b}$. What properties have to hold for $\mathbf{a}$ and $\mathbf{b}$ to make this possible?

Exercise 18.5 Consider the experiment by Sixtl (1967) reported on p. 397. He asked subjects to assess how much of emotion x is contained in emotion y. The exact question posed to the subjects was: "How much does x have of y?" Whether this instruction was further explained is not reported. The emotions were shyness, compassion, desire, love, humbleness, tenderness, anxiety, aggressiveness, wrath, and disgust.

(a) Discuss the task to which these subjects had to respond. Devise additional or alternative instructions that would make it very clear to them what they were expected to deliver.

(b) What type of questions concerning this task do you expect the subjects to raise in this context?

(c) Compare the above experimental method to one where proximities are collected. What type of data collection would you prefer? Which one is more likely to yield better data?

(d) Assume that the contained-in judgments generate the type of data that proponents of this method are hoping to get. What are the additional insights that these data would then allow over and beyond direct similarity ratings, say?

Exercise 18.6 Data collection by way of contained-in judgments has been restricted to a range of 0% to 100%. This implies that the respondents cannot distinguish stimuli that are "orthogonal" to each other from those that are opposite to each other, for example. They would both be rated as 0%. Devise a method that does away with this restriction. Work out the instructions that you would use to instruct the respondents about their task, and discuss your approach in the context of both the color similarities and the similarity of emotional experiences discussed in this chapter.

19
Euclidean Embeddings

Distances are functions that can be defined on *any* set of objects. Euclidean distances, in contrast, are functions that can only be defined on sets that possess a particular structure. Given a set of dissimilarities, one can test whether these values are distances and, moreover, whether they can even be interpreted as Euclidean distances. More generally, one can ask the same questions allowing for particular transformations of the given dissimilarities such as adding a constant to each value. For ordinal transformations, the hypothesis that dissimilarities are Euclidean distances is trivially true. Hence, in ordinal MDS, we learn nothing from the fact that the dissimilarities can be represented in a Euclidean space. In interval MDS, in contrast, Euclidean embedding is not trivial. If the data can be mapped into Euclidean distances, one can ask how many dimensions at most are necessary for a perfect representation. A further question, related to classical MDS, is how to find an interval transformation that leads to approximate Euclidean distances, while keeping the dimensionality of the MDS space as low as possible.

19.1 Distances and Euclidean Distances

Given a matrix of distances, one can ask whether these distances can be interpreted as Euclidean distances. This is true only if they can be embedded into a Euclidean space. The answer is positive if the scalar product matrix $\mathbf{B}$ derived from these distances (see Section 7.9 or 18.4) can be de-

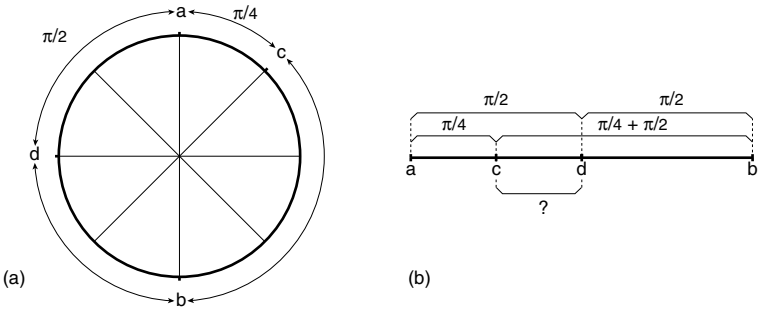


FIGURE 19.1. (a) Radian distances among points $a, \ldots, d$, and (b) their interpretation as Euclidean distances.

TABLE 19.1. Distances between points $a, \ldots, d$ on the circle in Figure 19.1a measured along the circle (radius=1).

Point	a	b	c	d
a	0.0000	3.1416	0.7854	1.5708
b	3.1416	0.0000	2.3562	1.5708
c	0.7854	2.3562	0.0000	2.3562
d	1.5708	1.5708	2.3562	0.0000

composed into $\mathbf{B} = \mathbf{X}\mathbf{X}'$, with real $\mathbf{X}$, or, equivalently, if $\mathbf{B}$'s eigenvalues are nonnegative (see Chapter 7). Conversely, if $\mathbf{B}$ has negative eigenvalues, the dissimilarities on which it is based can still be distances, albeit non-Euclidean distances. Consider an example.

Distances on a Circle

Figure 19.1a shows a configuration of four points on a circle. To determine their distances, we usually employ a straight ruler. This yields Euclidean distances. But here we measure the length of the shortest path ("geodesic") between points i and j on the circle. The circumference of a circle with radius 1 is equal to 2π. Thus, $d_{ab} = \pi$, $d_{ac} = \pi/4$, and so on, leading to the values in Table 19.1. These values are definitely distances: they are symmetric, they are nonnegative and exactly equal to 0 in the main diagonal, and the triangle inequality holds for all triples.

In fact, all triangle inequalities turn out to be equalities; for example, $d_{ab} = d_{ac} + d_{cb}$. In Euclidean geometry, this implies that a, b, and c lie on a straight line. Moreover, $d_{ab} = d_{ad} + d_{db}$ and, thus, the points a, b, and d must also lie on a straight line if the dissimilarities are interpreted as Euclidean distances. But in Euclidean geometry, there is just one line through the points a and b; hence, a, b, c, and d must all lie on it.

Figure 19.1b shows this line. The points c and d are positioned on it so that their distances satisfy the two triangle equalities above, and this implies that the distance between c and d should be $\pi/4$, which, however, is not in agreement with the value in Table 19.1.

Similarly, the scalar-product matrix derived from the distances in Table 19.1 using formula (18.17) yields the eigenvalues 5.61, 2.22, 0.00, and −1.21. Hence, this matrix is not positive semidefinite and so we are led to the same conclusion as before: the distances in Table 19.1 cannot be embedded into a Euclidean space.

Properties of Euclidean Distances

Because we did not arrive at the values in Table 19.1 by using a straight ruler, they cannot be Euclidean distances. Indeed, checking through them, we are led to contradictions if we assume that they were. Euclidean distances, therefore, have properties above and beyond those of general distances. The contradiction to which we were led in Figure 19.1b rests on the fact that for Euclidean distances there is just one geodesic path between any two points; that is, all points x that satisfy $d_{ab} = d_{ax} + d_{xb}$ must lie between a and b on the line through a and b.

This is not always true for other Minkowski distances. If points a and b lie on a line not parallel to the coordinate axes, then the city-block metric, for example, allows for infinitely many geodesics between a and b, so that the above triangle equality for x does not mean that x will be crossed if we move from a to b on a path of length d_{ab}. Hence, other Minkowski distances have special properties that require investigation.

Investigations of a mathematical structure typically begin by considering particular cases (such as, e.g., a plane with a straight-ruler distance measurement). One then attempts to describe the "essential" properties of these cases and to write them up in a simple list of axioms from which all of the theorems one has in mind may be proved. The axioms should be abstract in the sense that they do not rely on ad hoc features of the cases such as the dimensionality of the chosen geometry or a particular coordination for its points.

Euclidean distances are defined abstractly (coordinate-free and dimension-free) as the square root of the scalar product $b(\mathbf{i}-\mathbf{j}, \mathbf{i}-\mathbf{j})$, where $\mathbf{i}-\mathbf{j}$ is the difference vector of the vectors $\mathbf{i}$ and $\mathbf{j}$. Thus, Euclidean distances have properties related to those of scalar products. Two of these properties correspond to the axioms of (general) distances, namely, *symmetry* and *nonnegativity*. The remaining property, *linearity*, brings in the special properties of Euclidean distances: $b(s \circ \mathbf{u} + t \circ \mathbf{v}, \mathbf{w}) = s \cdot b(\mathbf{u}, \mathbf{w}) + t \cdot b(\mathbf{v}, \mathbf{w})$, for any vectors $\mathbf{u}$,$\mathbf{v}$,$\mathbf{w}$ and scalars s, t. The operation $\cdot$ denotes the usual multiplication of real numbers, whereas $\circ$ is different. It denotes that a vector is multiplied by a number (scalar). Also, + denotes addition of vectors,

not the usual addition of numbers.[1] As long as the rules that govern scalar multiplication and vector addition are not specified, the linearity axiom remains meaningless. But what are these rules?

The rules are collected in a system of axioms known as *Abelian vector spaces*. It comprises two structures, a *field* and a *group*. The field is usually the set of real numbers, with its two operations of addition and multiplication. A group is a set of elements with one operation that satisfies the following axioms.

g1 for any three of its elements, x, y, and z, $(x + y) + z = x + (y + z)$ ($+$ is *associative*);

g2 there exists a *zero element*, z, so that $x + z = x$, for any x;

g3 there exists an *inverse element* $x^{(i)}$ for any x, so that $x + x^{(i)} = z$;

g4 (for Abelian groups only) for any elements x, y, $x + y = y + x$ ($+$ is *commutative*).

A vector space ties together the field and the group (whose elements are now called vectors and written in this book in bold fonts) by an operation $\circ$ so that:

v1 $k \circ (\mathbf{x} + \mathbf{y}) = k \circ \mathbf{x} + c \circ \mathbf{y}$;

v2 $(s + t) \circ \mathbf{x} = s \circ \mathbf{x} + t \circ \mathbf{x}$;

v3 $s \circ (t \circ \mathbf{x}) = (s \cdot t) \circ \mathbf{x}$;

v4 $e \circ \mathbf{x} = \mathbf{x}$,

where s, t, e are scalars, e is the neutral element of the field, and $\mathbf{x}, \mathbf{y}$ are any elements of the group.

What does that tell us? It means that when we talk about Euclidean distances we are necessarily talking (at least by implication) about a rich mathematical structure. The notion of Euclidean distance is defined only in this system. It can be defined on a set of points u, v, w only if these points are first linked to corresponding elements $\mathbf{u}$,$\mathbf{v}$,$\mathbf{w}$ of a vector space ("embedding"). Distances in general need no such structural embeddings. The trivial distance, for example, defined as $d_{ij} = 1$ and $d_{ii} = 0$ for all i, j, exists on any set of elements i, j, whether they can be interpreted as vectors or not.

This also means that the properties of vector spaces cannot be tested for any finite set of vectors, because they must hold, for example, for any

[1] A different symbol (such as $\oplus$) might be better to denote vector addition. We do not use such particular notation here because we are almost always dealing with vectors that are n-tuples of real numbers in this book. In this case, addition of vectors is defined as the familiar addition of corresponding elements.

TABLE 19.2. (a) Dissimilarities for five objects; - denotes a missing value; (b) completing the proximity matrix by setting $\delta_{ii} = 0$ and $\delta_{ij} = \delta_{ji}$, for all i, j; (c) matrix after adding 4.8 to each element.

(a)	1	2	3	4	5
1	-	-	-	-	-
2	0.2	-	-	-	-
3	1.2	0.2	-	-	-
4	0.2	3.2	0.2	-	-
5	-1.8	-0.8	-1.8	-0.8	-

(b)	1	2	3	4	5
1	0.0	0.2	1.2	0.2	-1.8
2	0.2	0.0	0.2	3.2	-0.8
3	1.2	0.2	0.0	0.2	-1.8
4	0.2	3.2	0.2	0.0	-0.8
5	-1.8	-0.8	-1.8	-0.8	0.0

(c)	1	2	3	4	5
1	0	5	6	5	3
2	5	0	5	8	4
3	6	5	0	5	3
4	5	8	5	0	4
5	3	4	3	4	0

scalars s and t, and, therefore, involve *all* vectors of the space. This is why testing whether a given set of numbers are Euclidean distances is often called, more correctly, testing whether these numbers can be embedded into distances of a Euclidean space.

19.2 Mapping Dissimilarities into Distances

MDS models almost never assume that the given dissimilarities are distances. Rather, all models (except absolute MDS) admit some transformation on the dissimilarities such as, for example, a free choice of additive and multiplicative constants on the dissimilarities in interval MDS. We now study to what extent one can claim that some given dissimilarities can be embedded into a Euclidean space, given that some such transformation can be picked in an optimal way.

Allowing for a Multiplier on the Dissimilarities

Consider the dissimilarity matrix in Table 19.2a. This table is typical insofar as often only the δ_{ij}s for $i < j$ are collected. This immediately makes it impossible to test whether these values satisfy two of the properties of distances: $\delta_{ij} = \delta_{ji}$ and $\delta_{ii} = 0$, for all i, j. With no data to the contrary, we assume that these conditions are satisfied and complete the matrix as usual (Table 19.2b).

The resulting values violate the nonnegativity condition for distances. However, ratio MDS does not claim that the dissimilarities are distances but only that $k \cdot \delta_{ij} = d_{ij}, k \neq 0$. Hence, one can ask whether there exists a multiplier k such that the $k \cdot \delta_{ij}$ values satisfy all three distance axioms. For Table 19.2a, the answer is easily found: there is no such constant k for these data, because a negative k would make the positive values negative, and a positive one would not reverse the sign of the negative values. Hence, the hypothesis that the values in Table 19.2a are distances except for a

multiplicative constant k is wrong. Because they are not distances, they are not, a fortiori, Euclidean distances.

Generally, we note that the relation $k \cdot \delta_{ij} = d_{ij}$ (for some appropriately chosen k) is a hypothesis that may prove to be empirically wrong. Such hypotheses are called (empirically) *falsifiable*.

Allowing for an Interval Transformation on the Dissimilarities

More important than ratio MDS is interval MDS. Interval MDS also allows for an additive constant and, hence, claims that $k \cdot \delta_{ij} + c = d_{ij}$, for some $k \neq 0$ and c. Under this condition, we can transform all of the values in Table 19.2a into positive numbers. We simply add a number $c > 1.8$ to each δ_{ij} ($c = 1.9$, say), which transforms, for example, $\delta_{35} = -1.8$ into the new value $\delta^*_{35} = \delta_{35} + 1.9 = 0.1$.

This then leaves only the triangle inequality as a distance criterion. We find that it is violated for the δ^*_{ij}-values, because $\delta^*_{45} + \delta^*_{52} < \delta^*_{42}$. However, this inequality can be reversed by adding a larger constant c to all δ_{ij}s, because c appears twice on the left-hand side $\delta^*_{45} + \delta^*_{52} = \delta_{45} + c + \delta_{52} + c$ and only once in $\delta^*_{24} = \delta_{24} + c$. To find the smallest possible c that gives all triangle inequalities the desired sense, we check through all inequalities and find that $\delta_{45} + \delta_{52} = -1.6 \geq 3.2 = \delta_{42}$ is most violated; adding c to the dissimilarities, we should obtain $-1.6 + 2c > 3.2 + c$ or, at least, $-1.6+2c = 3.2+c$; hence, the minimal c is $c = 4.8$. If we turn this inequality around in the desired way by adding some $c \geq 4.8$ to all dissimilarities, then all of the other inequalities will also have the proper sense, because in each case c is added twice to the side that should be greater and only once to the other side. Taking $c = 4.8$ and setting all $\delta_{ii} = 0$, we arrive at Table 19.2c, which satisfies all distance axioms. We can conclude that the proposition that given dissimilarities are distances apart from an appropriate interval transformation is always true (tautological) if δ_{ij}s are given for only $i < j$.

Adding a positive additive constant will, in any case, transform any set of dissimilarities δ_{ij}, $i < j$, into distances, provided the constant is large enough. Yet, in the extreme case where $c \to \infty$, the distances thus generated approximate trivial distances.

If, on the other hand, a complete data matrix is given, it cannot be guaranteed that such constants exist. In fact, if just the δ_{ii}s are given, then the constants k and c must be chosen such that $k \cdot \delta_{ii} + c = d_{ii} = 0$. This restricts them so much that it is impossible to transform the dissimilarities into distances if $n \geq 3$.

Interval Transformed Dissimilarities and Euclidean Distances

We now go on and ask whether it is always possible to transform dissimilarities δ_{ij}, $i < j$, not only into distances, but into Euclidean distances by picking appropriate additive and multiplicative constants. The answer is

yes. Assume that some constant has already been added to the dissimilarities to make them all positive and that $\delta_{ii} = 0$, for all i, by definition. The factor k is irrelevant in the following and is set to $k = 1$. Substituting $\delta_{ij} + c$ for d_{ij} in (18.17) should yield a matrix of b_{ij}s that is positive semidefinite if an appropriate c is chosen. Setting $\delta_{ij} + c$ for d_{ij} (for $i \neq j$) and $d_{ii} = 0$ (for all i) in (18.17), or, more compactly, $d_{ij} = \delta_{ij} + (1 - \theta_{ij})c$, where $\theta_{ij} = 1$ (for $i = j$) and $\theta_{ij} = 0$ (for $i \neq j$), we obtain

$$\begin{aligned} b^*_{ij} &= \left[\tfrac{1}{2}(\delta^2_{i.} + \delta^2_{.j} - \delta^2_{..} - \delta^2_{ij})\right] \\ &\quad +2c\left[\tfrac{1}{2}(\delta_{i.} + \delta_{.j} - \delta_{..} - \delta_{ij})\right] + \frac{c^2}{2}\left[\theta_{ij} - \frac{1}{n}\right], \end{aligned} \tag{19.1}$$

where the point subscripts mean that the δs are averaged over the respective indices.

If $c = 0$, then (19.1) is equal to (18.17). Otherwise, there are two additional terms. If we store the bracketed terms in (19.1) in the ij cells of the matrices $\mathbf{B}$, $\mathbf{B}_r$, and $\mathbf{J}$, respectively, then (19.1) reads in matrix notation

$$\mathbf{B}^* = \mathbf{B} + 2c\mathbf{B}_r + \frac{c^2}{2}\mathbf{J}. \tag{19.2}$$

Note that $\mathbf{B}$ is the usual scalar-product matrix associated with the δ_{ij}s, and $\mathbf{B}_r$ is the scalar-product matrix associated with the square roots of the dissimilarities. $\mathbf{J}$, finally, is the centering matrix used in (12.2). Our task is to choose c such that $\mathbf{B}^*$ is positive semidefinite. There are several equivalent ways to state this condition. So far, we have seen two closely related tests: $\mathbf{B}^*$ has nonnegative eigenvalues; $\mathbf{B}^*$ can be factored into $\mathbf{X}\mathbf{X}'$, with real $\mathbf{X}$. A third way to state positive semidefiniteness is that $\mathbf{x}'\mathbf{B}^*\mathbf{x} \geq 0$, for all $\mathbf{x}$. That is, the number resulting from premultiplying $\mathbf{B}^*$ by any (real) vector $\mathbf{x}'$ and then postmultiplying $\mathbf{x}'\mathbf{B}^*$ by $\mathbf{x}$ must be nonnegative (see Chapter 7).

The condition $\mathbf{x}'\mathbf{B}^*\mathbf{x} \geq 0$ is trivially true if $\mathbf{x}$ is the zero-vector: then we have $\mathbf{x}'\mathbf{B}^*\mathbf{x} = 0$. If $\mathbf{x}$ is any other vector, this product should also be nonnegative. This condition is generally not as convenient as the eigenvalue test, but sometimes it leads to insights. The condition requires that

$$\begin{aligned} \mathbf{x}'\mathbf{B}^*\mathbf{x} &= \mathbf{x}'\left[\mathbf{B} + 2c\mathbf{B}_r + \frac{c^2}{2}\mathbf{J}\right]\mathbf{x} \\ &= \mathbf{x}'\mathbf{B}\mathbf{x} + 2c\mathbf{x}'\mathbf{B}_r\mathbf{x} + \frac{c^2}{2}\mathbf{x}'\mathbf{J}\mathbf{x} \\ &= k_1 + c \cdot k_2 + c^2 \cdot k_3 \geq 0. \end{aligned} \tag{19.3}$$

We find that $k_3 > 0$, because $\mathbf{x}'\mathbf{J}\mathbf{x}$ is positive for any $\mathbf{x} \neq \mathbf{0}$. ($\mathbf{x}'\mathbf{J}\mathbf{x}$ simply says $\sum_i (x_i - \bar{x})^2$ in summation notation.) The term k_3 is multiplied by c^2, but k_2 is multiplied by c only, and k_1 does not change as a function of c at all. Thus, if c is chosen ever larger, then $c^2 \cdot k_3$ will eventually dominate

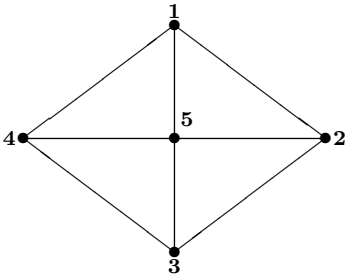


FIGURE 19.2. Five-point configuration, with distances among points as in Table 19.2c.

the sum of the other two terms and make $\mathbf{x}'\mathbf{B}^*\mathbf{x}$ positive semidefinite. It is therefore always possible to find an additive constant c that turns dissimilarities δ_{ij} $(i < j)$ into Euclidean distances.

19.3 Maximal Dimensionality for Perfect Interval MDS

We now know that dissimilarities $\delta_{ij}, i < j$, can always be mapped into Euclidean distances by an interval transformation and by setting $\delta_{ij} = \delta_{ji}$ and $\delta_{ii} = 0$, for all i, j. With respect to the additive constant c, any sufficiently large value will do. There are reasons, however, to choose the smallest possible value for c. For the values in Table 19.2a, we saw that they can be transformed into distances by adding $c_1 = 4.8$. This value turns the triangle inequality that was most violated into an equality. The resulting distances in Table 19.2c are Euclidean distances, because, by applying straight-ruler measurements, we obtain the configuration in Figure 19.2. Adding some $c_2 > c_1 = 4.8$ also leads to values that satisfy the triangle inequalities, but wherever we had a triangle equality for c_1 we will have a triangle inequality for c_2. Geometrically, adding some segment of length $c_2 - c_1$ to each line segment in Figure 19.2 will force point 5 out of the plane of the paper, so that our 5-point configuration will form a pyramid, and a space of three dimensions will be required to represent the data.

Because this makes the representation unnecessarily inaccessible for interpretation, it should be avoided. Of course, there is nothing in the data that would allow us to decide whether the pyramid or the square-with-midpoint configuration from Figure 19.2 is the true configuration, but, in the absence of any further knowledge or hypotheses, there is no reason not to assume that point 5 lies in the middle of the shortest path from 1 to 3.

We show how many dimensions are needed at most for a geometric embedding of an $n \times n$ matrix of Euclidean distances. In equation (12.2), $\mathbf{D}^{(2)}$ is double-centered by $\mathbf{J}$. This makes the rows/columns of $\mathbf{B}$ linearly depen-

TABLE 19.3. Matrix for finding the minimal additive constant c for data in Table 19.1 using formula (19.4); $c = 1.291$, the largest real eigenvalue of this matrix.

$$\left[\begin{array}{cccc|cccc} 0 & 0 & 0 & 0 & 3.16 & -5.48 & 2.24 & 0.08 \\ 0 & 0 & 0 & 0 & -5.48 & 5.63 & -1.47 & 1.31 \\ 0 & 0 & 0 & 0 & 2.24 & -1.47 & 2.55 & -3.32 \\ 0 & 0 & 0 & 0 & 0.08 & 1.31 & -3.32 & 1.93 \\ \hline -1 & 0 & 0 & 0 & -2.55 & 2.95 & -0.98 & 0.59 \\ 0 & -1 & 0 & 0 & 2.95 & -4.12 & 1.37 & -0.20 \\ 0 & 0 & -1 & 0 & -0.98 & 1.37 & -2.55 & 2.16 \\ 0 & 0 & 0 & -1 & 0.59 & -0.20 & 2.16 & -2.55 \end{array}\right]$$

dent so that $\text{rank}(\mathbf{B}) < n$: the centering matrix $\mathbf{J}$ generates deviation scores in the matrix it operates on, and, thus, the rows or columns, respectively, of the matrix product sum to the null vector $\mathbf{0}$. Hence, $\text{rank}(\mathbf{B}) \leq n-1$, so that the maximum dimensionality of a Euclidean distance matrix is $n-1$. But, as we saw above in Figure 19.2, there may be a c that reduces the dimensionality further. Cailliez (1983) presents a solution for c which guarantees distances that can be represented in at most $n-2$ dimensions. The minimal additive constant c is given by

$$c = \text{largest (real) eigenvalue of} \quad = \quad \left[\begin{array}{cc} \mathbf{0} & 2\mathbf{B} \\ -\mathbf{I} & -4\mathbf{B}_r \end{array}\right]. \qquad (19.4)$$

The matrix in (19.4) is set up by collecting the matrices $2\mathbf{B}$, $4\mathbf{B}_r$, the null matrix $\mathbf{0}$, and the identity matrix $\mathbf{I}$ into one supermatrix. All four matrices have the order $n \times n$; hence, the supermatrix has the order $2n \times 2n$. For the values in Table 19.1, we find by formula (19.4) that $c \approx 1.29$. Adding 1.29 to all numbers in Table 19.1 leads (almost precisely) to a positive semidefinite $\mathbf{B}^*$ with two zero eigenvalues or $\text{rank}(\mathbf{B}^*) = n - 2 = 2$.

If we deal with an ordinal MDS problem, we are not restricted to interval transformations for mapping dissimilarities into Euclidean distances. However, it seems that this does not allow one to reduce the maximal dimensionality of the MDS space below $n-2$. Lingoes (1971), in an earlier paper, describes a simple monotonic transformation on the dissimilarities that guarantees Euclidean distances but does not reduce the dimensionality below $n-2$.

19.4 Mapping Fallible Dissimilarities into Euclidean Distances

In the preceding sections, we ignored the issue of measurement error. But now that we understand how error-free dissimilarities are related to dis-

tances and Euclidean distances under various transformations, some statistical considerations should be made. For fallible data, the transformation problem becomes $k \cdot p_{ij} + c = d_{ij} + e_{ij}$, where d_{ij} is the true distance and e_{ij} is an error component. The task, then, is to find an additive constant c such that the transformed dissimilarities are distances except for a random component. In other words, the shifted data values may violate the critical triangle inequality condition only to such an extent that the violations can be attributed to error. This requires an error theory and results in a much more complicated problem than those considered above. We may require, in addition, that the d_{ij}s be Euclidean distances and that their representation space be as small as possible. This represents a difficult problem, which is subject to different interpretations. We consider the formulation of Messick and Abelson (1956), which, in combination with the double-centering conversion in formula (12.3), is known as *classical MDS*.

The Minimum Statistical Additive Constant

For error-free Euclidean distances, the eigenvalues of the associated scalar-product matrix $\mathbf{B}$ are all positive or zero. The number of positive eigenvalues is equal to the rank of $\mathbf{B}$. Thus, an additive constant c should be chosen such that (a) $\mathbf{B}$ becomes positive semidefinite and (b) the number of zero eigenvalues is maximal.

For error-affected Euclidean distances, this c would be too large. Because of error, the distance estimates cannot be expected to be Euclidean distances so that $\mathbf{B}$ has, in general, some negative eigenvalues. But the distribution of the eigenvalues should have a peculiar form. If the error component is small, there should be some large eigenvalues and some small ones. The large eigenvalues represent the true structure, and the small ones are due to the random over- and under-estimation of the distances. Moreover, "... with fallible data ... the small roots will probably not equal zero but will vary positively and negatively around zero" (Messick & Abelson, 1956, p. 7). If this assumption is made, the sum of the small eigenvalues should be equal to zero, and c should be chosen accordingly.

We start with equation (12.2) and see what can be derived from this assumption about the eigenvalue distribution. Messick and Abelson (1956) use a theorem from matrix algebra which says that the trace of a symmetric matrix $\mathbf{B}$ is equal to the sum of its eigenvalues. That is, if $\mathbf{Q}\mathbf{\Lambda}\mathbf{Q}'$ is the eigendecomposition of $\mathbf{B}$, then

$$\text{tr } \mathbf{B} = \text{tr } \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}' = \text{tr } \mathbf{\Lambda}\mathbf{Q}'\mathbf{Q} = \text{tr } \mathbf{\Lambda},$$

which uses $\mathbf{Q}'\mathbf{Q} = \mathbf{I}$ and the invariance of the trace function under cyclic permutation (property 3 of Table 7.4). Assume that the eigendecomposition of $\mathbf{B}^*$—which, of course, cannot be computed before c is defined—yields the eigenvalues $\lambda_1, \ldots, \lambda_n$ and the corresponding eigenvectors $\mathbf{q}_1, \ldots, \mathbf{q}_n$.

Thus, $\mathbf{B}^*\mathbf{q}_i = \lambda_i\mathbf{q}_i$ or $\mathbf{q}_i'\mathbf{B}^*\mathbf{q}_i = \mathbf{q}_i'\lambda_i\mathbf{q}_i = \lambda_i\mathbf{q}_i'\mathbf{q}_i = \lambda_i$, because $\mathbf{q}_i'\mathbf{q}_i = 1$, by convention. Now, let the first r eigenvalues be large and the remaining $n - r$ small, as discussed above. Then, r is the dimensionality of the true distances and their scalar products. The sum of the first r eigenvalues is $\sum_{i=1}^{r}\lambda_i = \sum_{i=1}^{r}\mathbf{q}_i'\mathbf{B}^*\mathbf{q}_i$. Hence, by the trace-eigenvalue theorem (see also Section 7.4), we find

$$\sum_{i=1}^{n} b_{ii}^* = \sum_{i=1}^{r}\lambda_i, \tag{19.5}$$

or

$$\text{tr } \mathbf{B}^* = \sum_{i=1}^{r}\mathbf{q}_i'\mathbf{B}^*\mathbf{q}_i. \tag{19.6}$$

Substituting $\mathbf{B} + 2c\mathbf{B}_r + (c^2/2)\mathbf{J}$ for $\mathbf{B}^*$ leads to

$$\text{tr } \left[\mathbf{B} + 2c\mathbf{B}_r + \frac{c^2}{2}\mathbf{J}\right] = \sum_{i=1}^{r}\mathbf{q}_i'\left[\mathbf{B} + 2c\mathbf{B}_r + \frac{c^2}{2}\mathbf{J}\right]\mathbf{q}_i, \tag{19.7}$$

a quadratic equation with the unknown c. The derivation hinges on (19.5): the sum of the first r eigenvalues of $\mathbf{B}^*$ is equal to the trace of $\mathbf{B}^*$ only if the sum of the remaining $n - r$ eigenvalues is equal to zero. This means that the $n - r$ smallest eigenvalues are either all equal to zero or they are distributed symmetrically about zero, as assumed.

Equation (19.7) involves two unknowns, r and c. However, even if we assume for a moment that r has been estimated in some way, we note that it still is not possible to solve the equation for c, because the eigenvectors $\mathbf{q}_i$ are computed from $\mathbf{B}^*$ and thus also depend on c. Solving the problem may therefore be attempted in the usual iterative fashion. First, choose some value for $c^{[0]}$, compute the eigenvalues for $\mathbf{B}^*$, and solve (19.7) for a new c, $c^{[1]}$. This $c^{[1]}$ leads to a new $\mathbf{B}^*$, new eigenvalues, and a new c, $c^{[2]}$, and so on. We show that it is better to choose $c^{[0]}$ too large than too small. A good choice for $c^{[0]}$ would be the additive constant that strictly satisfies all triangle inequalities.

An Illustration for Finding the Statistical Additive Constant

It is peculiar that Messick and Abelson (1956) illustrate their method by an example in which there is no error at all in the distances, that is, a case where we do not really have to estimate the additive constant c but can simply compute it. We nevertheless present this example here because it is transparent and instructive. We start by defining the configuration in Figure 19.3, which yields the true Euclidean distances. As before, only the values in one-half of the distance matrix are considered. Assume that subtracting 1 from these distances generates the dissimilarities that we observe; for example, $d_{AB} = 1$ and hence $\delta_{AB} = d_{AB} - 1 = 1 - 1 = 0$.

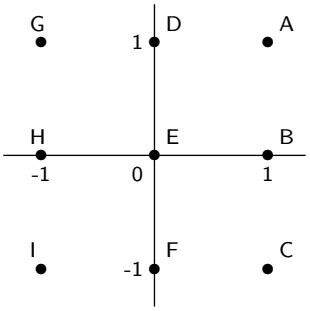


FIGURE 19.3. Configuration used in Messick and Abelson (1956) study.

Because $\delta_{AC} = 1$ and $\delta_{CB} = 0$, the triangle inequality $\delta_{AC} \leq \delta_{AB} + \delta_{BC}$ is violated for the dissimilarities.

To find the true additive constant c in the sense of Messick and Abelson (1956) (which here is $c = 1$ because there is no error in the dissimilarities) a starting value $c^{[0]}$ has to be chosen so that $\mathbf{B}^*$ is defined and its eigenvectors can be computed. Table 19.4 shows the effect of different $c^{[0]}$-values on the eigenvalues and eigenvectors of $\mathbf{B}^*$. All values equal to or greater than 1 transform the dissimilarities into Euclidean distances. For $c^{[0]} = 1$, the true additive constant, only two nonzero eigenvalues result. (One eigenvalue is equal to 0 in all cases due to the centering of $\mathbf{B}^*$.) For $c^{[0]} < 1$, negative eigenvalues arise, because the triangle inequalities remain violated under this condition. Moreover, for $c^{[0]} = 0$, the first two eigenvectors define a configuration very similar to the one in Figure 19.3, but this is not the case for $c^{[0]} = -1$ and $= -2$. Messick and Abelson (1956) claim that, in these latter cases, it is the eighth and ninth eigenvectors whose coordinates define a configuration similar to the one in Figure 19.3. However, such similarities are more apparent than real, because negative eigenvalues correspond to negative distances, and it is quite unclear what this means geometrically. What is definite, in contrast, is that choosing a "small" value for $c^{[0]}$ may lead to problems, because it may result in using the "wrong" r eigenvectors in (19.7). We also note that, for larger initial c-values, two eigenvalues are definitely dominant, which enables us to make a decision on the true dimensionality r.

Assume now that $c^{[0]} = 4$ was chosen. This defines $\mathbf{B}^*$ in (19.6), which can then be factored. Studying the resulting eigenvalue distribution suggests setting $r = 2$. This defines (19.7) and yields as the solutions for its unknown $c_1 = 0.997$ and $c_2 = -0.55$. The value -0.55 is evidently not the desired additive constant, because it does not eliminate violations of the triangle inequalities. Hence, 0.997 must be the solution. We know that the true $c = 1$, so $c_1 = 0.997$ is quite close. The Messick–Abelson procedure has, thus, after just one iteration, almost recovered the true value. But why is c_1 not exactly equal to 1? The reason is that $c^{[0]} = 4$ was too large a value.

TABLE 19.4. First two eigenvectors (fitted to correspond to configuration in Fig. 19.3) and all eigenvalues for different choices of $c^{[0]}$; eigenvalues with star correspond to shown eigenvectors; after Messick and Abelson (1956).

	$c^{[0]}=4$		3		2		1		0		−1		−2	
	$\mathbf{q}_1$	$\mathbf{q}_2$	$\mathbf{q}_1$	$\mathbf{q}_2$	$\mathbf{q}_1$	$\mathbf{q}_2$	$\mathbf{q}_1$	$\mathbf{q}_2$	$\mathbf{q}_1$	$\mathbf{q}_2$	$\mathbf{q}_1$	$\mathbf{q}_2$	$\mathbf{q}_1$	$\mathbf{q}_2$
A	.97	.97	.98	.98	.99	.99	1.00	1.00	1.05	1.05	.72	.72	.90	.90
B	1.05	.00	1.04	.00	1.03	.00	1.00	.00	.88	.00	1.31	.00	1.15	.00
C	.97	−.97	.98	−.98	.99	−.99	1.00	−1.00	1.05	−1.05	.72	−.72	.90	−.90
D	.00	1.05	.00	1.04	.00	1.03	.00	1.00	.00	.88	.00	1.31	.00	1.15
E	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
F	.00	−1.05	.00	−1.04	.00	−1.03	.00	−1.00	.00	−.88	.00	−1.31	.00	−1.15
G	−.97	.97	−.98	.98	−.99	.99	−1.00	1.00	−1.05	1.05	−.72	.72	−.90	.90
H	−1.05	.00	−1.00	.00	−1.03	.00	−1.00	.00	−.88	.00	−1.31	.00	−1.15	.00
I	−.97	−.97	−.98	−.98	−.99	−.99	−1.00	−1.00	−1.05	−1.05	−.72	−.72	−.90	−.90
λ_1	23.12*		16.41*		1.70*		6.00*		2.31*		1.03		3.05	
λ_2	23.12*		16.41*		1.70*		6.00*		2.31*		1.03		3.05	
λ_3	8.02		4.34		1.67		.00		.02		.86		2.69	
λ_4	7.32		3.88		1.44		.00		.00		.11		2.01	
λ_5	6.93		3.66		1.33		.00		−.14		.00		1.68	
λ_6	6.36		3.24		1.12		.00		−.14		− .34		.99	
λ_7	6.36		3.24		1.12		.00		.33		− .34		.00	
λ_8	5.95		2.97		.98		.00		−.33		−.52*		−2.17*	
λ_9	.00		.00		.00		.00		−.44		−.52*		−2.17*	

On the other hand, we see from Table 19.3 that the first two coordinate vectors (which are the eigenvectors rotated to match the true coordinate vectors of Figure 19.3 as closely as possible) are very similar across different values for $c \geq 1$. Thus, it hardly matters which eigenvectors are used in (19.7). For this reason, c_1 is found to be so close to the true value after just one iteration. If, on the other hand, too small a value had been chosen for $c^{[0]}$, negative eigenvalues would have resulted for $\mathbf{B}^*$. In this case, one should start all over again using a larger constant.

Geometric Effects of Nonminimal Additive Constants

Table 19.4 shows that choosing any value other than the true additive constant has a distorting effect on the recovered configuration.[2] The true underlying configuration in Figure 19.3 is a pattern of squares in which the points lie on a network of straight lines. If we plot the point coordinates for $c = 4$ in Table 19.4, we find that the resulting configuration is very similar to Figure 19.3, but the grid is bent convexly outwards from the origin. For example, point B is shifted away from the origin on the Y-axis, whereas A and C stay put. The analogous situation is true for D, F, and H. Moreover, in the 3D MDS space, the plane that best represents Figure 19.3 is warped to form a peculiar saddle shape: A and I are pulled upwards, but G and

[2]Similar distorting effects can be observed when a metric is chosen in MDS that does not correspond to the metric used to generate the distance estimates in the true underlying space. See Section 17.3.

C are pushed downwards, with all other points in a horizontal plane. In contrast, if $c = 0$, the points on the coordinate axes of the 2D space are, relative to the other points, shifted towards the origin, resulting in a convex distortion of the grid. Hence, choosing an inappropriate additive constant results not merely in higher dimensionality but in a *systematic distortion* of the configuration.

Once the data are transformed into distances, statistically or strictly speaking, any further additive constant will change the distance function and thus affect the geometric representation (in a metric context). This is important because dissimilarity data may already be distances, without any transformation, and so adding a constant to them has direct effects on their geometry. In practice, one finds, for example, that ratings of dissimilarity typically require an additive constant that is negative. Such data satisfy the properties of distances so that adding a constant merely serves the purpose of transforming them into Euclidean distances of low dimensionality or into distances with particular segmental additivity properties (see Chapter 17). In that case, an alternative and possibly more fruitful way to proceed would be to consider alternative geometries in which the given distances can be embedded as they are.

19.5 Fitting Dissimilarities into a Euclidean Space

We have seen that the additive constant problem for interval-scaled dissimilarities $\delta_{ij}, i < j$, has a simple solution if it is formulated in an algebraic or error-free way. A statistical model, in which the unknown additive constant is not computed but estimated, is more demanding. The Messick–Abelson solution is complicated, however, and its underlying model is not entirely clear. It suggests, perhaps, that we should not insist on an additive constant strictly satisfying the requirement that the transformed dissimilarities be Euclidean distances. Yet, it seems that in most applications we could drop the parameter r from those that have to be estimated and simply set it to some value that appears theoretically appropriate. With a fixed r, and with the requirement that the distances should be approximately mapped into Euclidean distances, we end up with a familiar problem: interval MDS.

In this context, the transformation question gets a positive answer if the resulting value for the loss criterion is sufficiently small, so that the required conditions are more or less satisfied. What should be considered sufficiently small depends on the context. Among the earliest proposals for treating the additive constant problem in this way are those of Cooper (1972) and Roskam (1972). These authors use the algebraic solution for c as a starting value; that is, $c^{[0]} = \max[\delta_{ij} - (\delta_{ik} + \delta_{kj})]$, over all i, j, k. The resulting $\mathbf{B}^*$ is decomposed into $\mathbf{X}\mathbf{X}'$, and the first r columns of $\mathbf{X}$ are used as the starting configuration. With these starting parameters, a flip-flop procedure

for minimizing $L = \sum[d_{ij} - (k \cdot \delta_{ij} + c)]^2 / \sum d_{ij}^2$ is entered. As we have seen, however, this procedure may not produce the best possible solution for c. Nevertheless, the method works in practice, and we can always check the optimality of the solution by trying other starting configurations. In any case, it is important to distinguish the *optimization* approach conceptually from the *algebraic* and the *statistical* viewpoints taken above. In the first case, c is optimized, in the second it is computed, and in the third it is estimated.

The so-called rational starting configurations for ordinal MDS are constructed by using the optimization method of interval MDS. Often, ranking-numbers are first substituted for the given dissimilarities: if the data are dissimilarities, the smallest δ_{ij} is set equal to 1, the second-smallest to $2, \ldots,$ and the largest to $\binom{n}{2}$; for similarities, the largest δ_{ij} is set equal to 1, the second largest to 2, and so on. We can also use the δ_{ij} values as they are. In either case, there are several options for proceeding. One possibility would be to add the algebraic additive constant, find the associated $\mathbf{B}$, decompose this into $\mathbf{XX}'$, and use the first r dimensions as an initial configuration. Another possibility would be to use the data or ranking-number matrix without adding any constant c and check whether the resulting $\mathbf{X}$ has some small imaginary dimensions. If so, we keep the first r and proceed with ordinal optimization. If not, a constant c can be added to the dissimilarities repeatedly until this situation results: if there are no negative eigenvalues for $\mathbf{B}^*$, then we choose $c < 0$; otherwise, we set $c > 0$.

19.6 Exercises

Exercise 19.1 Consider the similarities in Table 4.1 on p. 65. For this exercise you need software that can do matrix algebra.

(a) Transform the similarities into dissimilarities.

(b) Then, find the smallest possible additive constant that turns these values into Euclidean distances.

(c) Use classical scaling on the transformed dissimilarities. Compare the solution to the one obtained in Exercise 12.1 and in Figure 4.1. What do you conclude?

(d) Instead of the distances being Euclidean, find the smallest possible additive constant that turns dissimilarities into distances (not necessarily Euclidean). Is this constant the same as the one for Euclidean distance?

(d) Using a large additive constant, the dissimilarities are turned into distances. Are they also turned into Euclidean distances? Try a few cases numerically.

Exercise 19.2 Consider the data matrix below (Torgerson, 1958). It shows "absolute distances" based on 84 judgments of closeness for all possible triads of nine colors. The colors were all of the same red hue (=5R in Munsell notation) but differed from each other in brightness (value) and saturation (chroma). The conversion of the triadic closeness judgments into the values shown below involved a series of conversions aimed at adding the best additive constant.

			Number of Stimulus								
No	Value	Chroma	1	2	3	4	5	6	7	8	9
1	7	4	–	1.23	3.48	2.98	3.83	5.16	4.69	5.62	5.83
2	6	6	1.23	–	2.59	1.67	2.70	4.40	3.13	4.65	4.38
3	6	10	3.48	2.59	–	4.30	2.28	2.93	4.67	4.30	6.22
4	5	4	2.98	1.67	4.30	–	2.82	4.85	1.85	3.88	2.88
5	5	8	3.83	2.70	2.28	2.82	–	2.58	2.37	1.95	4.09
6	5	12	5.16	4.40	2.93	4.85	2.58	–	4.17	2.93	5.48
7	4	6	4.69	3.13	4.67	1.85	2.37	4.17	–	2.42	2.30
8	4	10	5.62	4.65	4.30	3.88	1.95	2.93	2.42	–	4.02
9	3	4	5.83	4.38	6.22	2.88	4.09	5.48	2.30	4.02	–

(a) Check whether these data violate any distance axioms.

(b) Determine the minimum additive constant that turns these values into distances, if possible. (If this constant exists, it may be equal to zero. When?)

(c) Same as (b), but now replace "distances" by "Euclidean distances".

(d) Use classical scaling to check to which extent the data mirror their physical design. (The design is given by the Munsell values for value and chroma; hue is constant.)

(e) Enforce an MDS structure that mirrors the physical design except for possible monotonic transformations along the coordinate axes value and chroma.

Part V

MDS and Related Methods

20
Procrustes Procedures

The Procrustes problem is concerned with fitting a configuration (testee) to another (target) as closely as possible. In the simplest case, both configurations have the same dimensionality and the same number of points, which can be brought into a 1–1 correspondence by substantive considerations. Under orthogonal transformations, the testee can be rotated and reflected arbitrarily in an effort to fit it to the target. In addition to such rigid motions, one may also allow for dilations and for shifts. In the oblique case, the testee can also be distorted linearly. Further generalizations include an incompletely specified target configuration, different dimensionalities of the configurations, and different numbers of points in both configurations.

20.1 The Problem

We now consider a problem that arose repeatedly throughout the text. In Figure 2.14, using rotations, reflections, and dilations, we found it possible to match two configurations almost perfectly. Without these transformations, it would have been difficult to see that ratio and ordinal MDS led to virtually the same configurations. If the dimensionality of two configurations is higher than 2D, such comparisons become even more difficult or, indeed, impossible. Therefore, one needs procedures that eliminate *meaningless* differences as much as possible by transforming one configuration (*testee*) by a set of *admissible* transformations so that it most closely ap-

proximates a given *target* configuration. Such fitting problems are known as *Procrustes problems* (Hurley & Cattell, 1962).[1]

In geometry, two figures (configurations) are called *similar* if they can be brought to a complete match by rigid motions and dilations. These transformations are admissible for all MDS solutions up to ratio MDS, so we can freely exploit similarity transformations to facilitate comparisons of different MDS configurations. Before considering similarity transformations, however, we first consider a restricted Procrustes problem, the orthogonal Procrustes. Once this problem is solved, it can be easily extended to cover the similarity case.

20.2 Solving the Orthogonal Procrustean Problem

Let $\mathbf{A}$ be the target configuration and $\mathbf{B}$ the corresponding testee. Assume that $\mathbf{A}$ and $\mathbf{B}$ are both of order $n \times m$. We now want to fit $\mathbf{B}$ to $\mathbf{A}$ by rigid motions. That is, we want $\mathbf{A} \approx \mathbf{BT}$ by picking a best-possible matrix $\mathbf{T}$ out of the set of all orthogonal $\mathbf{T}$. Geometrically, $\mathbf{T}$ therefore is restricted to rotations and reflections.

Without the restriction $\mathbf{TT}' = \mathbf{T}'\mathbf{T} = \mathbf{I}$, $\mathbf{T}$ could be *any* matrix, which means, geometrically, that $\mathbf{T}$ is some *linear transformation*. Such transformations, however, do not, in general, preserve $\mathbf{B}$'s "shape". Rather, linear transformations can cause shears, stretch $\mathbf{B}$ differentially along some directions, or collapse its dimensionality (see, e.g., Green & Carroll, 1976). Such transformations are clearly inadmissible ones, because they generally change the ratios of the distances among $\mathbf{B}$'s points and, thus, affect the fit of these distances to the data. For the moment, we are not interested in such transformations.

As for the $\approx$ criterion, a reasonable definition would be to measure the distances between corresponding points, square these values, and add them to obtain the sum-of-squares criterion L. The transformation $\mathbf{T}$ should be chosen to minimize this L. Expressed in matrix notation, the differences of the coordinates of $\mathbf{A}$ and $\mathbf{BT}$ are given by $\mathbf{A} - \mathbf{BT}$. We want to minimize the sum of the squared error, that is,

$$L(\mathbf{T}) = \text{tr } (\mathbf{A} - \mathbf{BT})'(\mathbf{A} - \mathbf{BT}) \tag{20.1}$$

or, equivalently,

$$L(\mathbf{T}) = \text{tr } (\mathbf{A} - \mathbf{BT})(\mathbf{A} - \mathbf{BT})',$$

[1] Procrustes was an innkeeper in Greek mythology who "fitted" his guests to his beds by stretching them or by chopping off their legs. The terminology "Procrustes problem" is now standard, even though it is generally misleading, inasmuch we do *not* want to mutilate or distort the testee configuration.

as explained in Table 7.4. In other words, $L(\mathbf{T})$ measures the squared distances of the points of $\mathbf{A}$ and the corresponding points of $\mathbf{BT}$.

Expanding (20.1), we get

$$\begin{aligned} L(\mathbf{T}) &= \text{tr } (\mathbf{A} - \mathbf{BT})'(\mathbf{A} - \mathbf{BT}) \\ &= \text{tr } \mathbf{A}'\mathbf{A} + \text{tr } \mathbf{T}'\mathbf{B}'\mathbf{BT} - 2\text{tr } \mathbf{A}'\mathbf{BT} \\ &= \text{tr } \mathbf{A}'\mathbf{A} + \text{tr } \mathbf{B}'\mathbf{B} - 2\text{tr } \mathbf{A}'\mathbf{BT} \end{aligned}$$

over $\mathbf{T}$ subject to $\mathbf{T}'\mathbf{T} = \mathbf{TT}' = \mathbf{I}$. Note that the simplification tr $\mathbf{T}'\mathbf{B}'\mathbf{BT}$ = tr $\mathbf{B}'\mathbf{B}$ is obtained by using the property of invariance of the trace function under cyclic permutation (see Table 7.4, property 3), which implies tr $\mathbf{T}'\mathbf{B}'\mathbf{BT}$ = tr $\mathbf{B}'\mathbf{BTT}'$, and using $\mathbf{T}'\mathbf{T} = \mathbf{TT}' = \mathbf{I}$, so that tr $\mathbf{B}'\mathbf{BTT}'$ = tr $\mathbf{B}'\mathbf{B}$. Because tr $\mathbf{A}'\mathbf{A}$ and tr $\mathbf{B}'\mathbf{B}$ are not dependent on $\mathbf{T}$, minimizing $L(\mathbf{T})$ is equivalent to minimizing

$$L(\mathbf{T}) = c - 2\text{tr } \mathbf{A}'\mathbf{BT} \tag{20.2}$$

over $\mathbf{T}$ subject to $\mathbf{T}'\mathbf{T} = \mathbf{I}$, where c is a constant that is not dependent on $\mathbf{T}$.

Minimization of $L(\mathbf{T})$ can be accomplished by applying the concept of an attainable lower bound (Ten Berge, 1993).[2] Suppose that we can find an inequality that tells us that $L(\mathbf{T}) \geq h$ and also gives the condition under which $L(\mathbf{T}) = h$. Solving $L(\mathbf{T}) = h$ for $\mathbf{T}$ (subject to the appropriate constraints) automatically gives us the smallest possible value of $L(\mathbf{T})$ and hence the global minimum.

To apply this notion to the problem in (20.2), let us first consider a lower bound inequality derived by Kristof (1970). If $\mathbf{Y}$ is a *diagonal* matrix with nonnegative entries, and $\mathbf{R}$ is orthogonal, Kristof's inequality states that

$$-\text{tr } \mathbf{RY} \geq -\text{tr } \mathbf{Y}, \tag{20.3}$$

with equality if and only if $\mathbf{R} = \mathbf{I}$.

To prove this theorem, note that because $\mathbf{Y}$ is diagonal, we may express (20.3) as

$$-\text{tr } \mathbf{RY} = -\sum_i r_{ii} y_{ii} \geq -\sum_i y_{ii}.$$

Now, because $\mathbf{RR}' = \mathbf{R}'\mathbf{R} = \mathbf{I}$, it holds for each column j of $\mathbf{R}$ that $\mathbf{r}_j'\mathbf{r}_j = \sum_i r_{ij}^2 = 1$, so that $-1 \leq r_{ii} \leq 1$. Thus, $-r_{ii}y_{ii} \geq -y_{ii}$. Obviously, only if $r_{ii} = 1$ or, in matrix terms, only if $\mathbf{R} = \mathbf{I}$, then inequality (20.3) is an equality.

---

[2]The orthogonal Procrustes problem was first solved by Green (1952) and later simultaneously by Cliff (1966) and Schönemann (1966). Their solutions are, however, somewhat less easy to understand and to compute.

We can use this theorem to find $L(\mathbf{T})$ as follows. Let $\mathbf{P\Phi Q'}$ be the singular value decomposition of $\mathbf{A'B}$, where $\mathbf{P'P} = \mathbf{I}$, $\mathbf{Q'Q} = \mathbf{I}$, and $\mathbf{\Phi}$ is the *diagonal* matrix with the singular values. Using the invariance of the trace function under cyclic permutation (see Table 8.3)

$$\begin{aligned} L(\mathbf{T}) &= c - 2\text{tr } \mathbf{A'BT} = c - 2\text{tr } \mathbf{P\Phi Q'T} \\ &= c - 2\text{tr } \mathbf{Q'TP\Phi} \\ &\geq c - 2\text{tr } \mathbf{\Phi}. \end{aligned}$$

Because $\mathbf{T}$ is orthonormal, so is $\mathbf{Q'TP}$. Now the minimization of $L(\mathbf{T})$ is written in the form of (20.3) with $\mathbf{R} = \mathbf{Q'TP}$ and $\mathbf{Y} = \mathbf{\Phi}$. We know that $L(\mathbf{T})$ is minimal if $\mathbf{R} = \mathbf{I}$ or, equivalently, $\mathbf{Q'TP} = \mathbf{I}$. Hence, we have to choose $\mathbf{T}$ as

$$\mathbf{T} = \mathbf{QP'}, \tag{20.4}$$

because substitution of (20.4) in $\mathbf{Q'TP}$ yields $\mathbf{Q'QP'P} = \mathbf{I}$, so that $L(\mathbf{T}) = c - 2\text{tr } \mathbf{\Phi}$.

20.3 Examples for Orthogonal Procrustean Transformations

We now consider a simple artificial case where $\mathbf{T}$ can be computed by hand. In Figure 20.1, two vector configurations, $\mathbf{A}$ and $\mathbf{B}$, are shown. Their points are connected to form rectangles. If panels 1 and 2 of Figure 20.1 are superimposed (panel 3), then $L(\mathbf{T})$ is equal to the sum of the squared lengths of the dashed-line segments that connect corresponding points of $\mathbf{A}$ and $\mathbf{B}$. Computing $\mathbf{T}$ as discussed above, we find

$$\mathbf{T} = \begin{pmatrix} -.866 & -.500 \\ -.500 & .866 \end{pmatrix}.$$

What does $\mathbf{T}$ do to $\mathbf{B}$? From Figure 20.1, we see that $\mathbf{T}$ should first reflect $\mathbf{B}$ along the horizontal axis (or, reflect it on the vertical axis) and then rotate it by 30° counterclockwise. The reflection matrix is thus

$$\mathbf{U}_1 = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}$$

and the rotation matrix by 30° is

$$\mathbf{R}_1 = \begin{pmatrix} \cos 30° & \sin 30° \\ -\sin 30° & \cos 30° \end{pmatrix} = \begin{pmatrix} .866 & .500 \\ -.500 & .866 \end{pmatrix}.$$

Applying $\mathbf{U}_1$ first and $\mathbf{R}_1$ afterwards yields $\mathbf{U}_1\mathbf{R}_1 = \mathbf{T}$ and $\mathbf{BT} = \mathbf{BU}_1\mathbf{R}_1$. But the decomposition of $\mathbf{T}$ into $\mathbf{U}_1$ and $\mathbf{R}_1$ is not unique. This may be

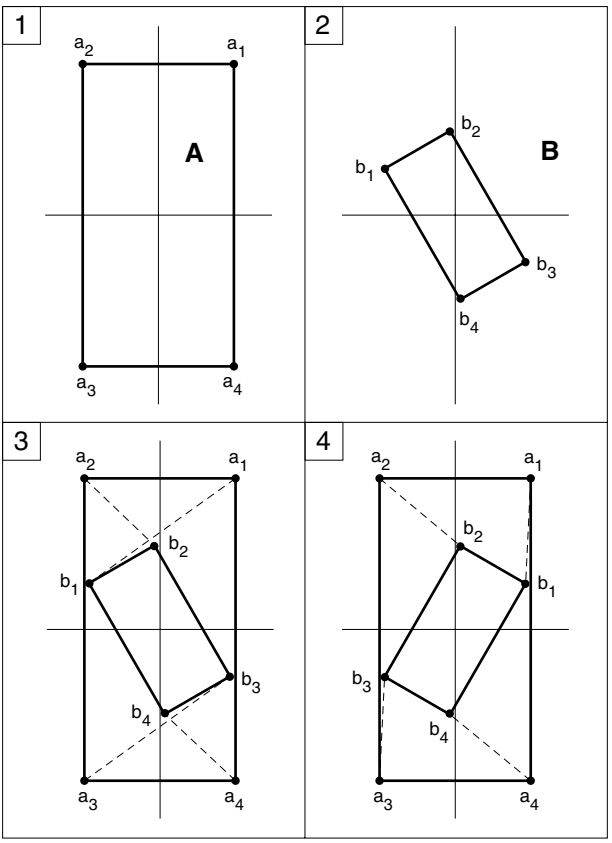


FIGURE 20.1. Illustration of some steps involved in fitting **B** to **A** by an orthogonal transformation.

more evident geometrically: in order to transform $\mathbf{B}$ into $\mathbf{BT}$, it would also be possible to first rotate $\mathbf{B}$ by $-30°$ (i.e., clockwise by 30°) and then reflect it horizontally. This reverses the order of rotation and reflection but leads to the same result. Another possibility would be to reflect $\mathbf{B}$ vertically and then turn it by 210°. To see that this produces the same effect, we simply find the corresponding reflection and rotation matrices,

$$\mathbf{U}_2 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix},$$

$$\mathbf{R}_2 = \begin{pmatrix} \cos 210° & \sin 210° \\ -\sin 210° & \cos 210° \end{pmatrix} = \begin{pmatrix} -.866 & -.500 \\ .500 & -.866 \end{pmatrix},$$

which yield $\mathbf{T} = \mathbf{U}_2\mathbf{R}_2 = \mathbf{U}_1\mathbf{R}_1$. Thus, $\mathbf{T}$ can be interpreted in different ways.

20.4 Procrustean Similarity Transformations

We now return to our original Procrustean problem of fitting one MDS configuration (testee) to another (target) MDS configuration. Because the overall size and the origin of MDS configurations are irrelevant, we now attempt to optimally exploit these additional transformations in fitting the testee matrix to the target. That is, we now extend the rotation/reflection task by finding an optimal dilation factor and an optimal translation for $\mathbf{B}$ (Schönemann & Carroll, 1970). In the context of Figure 20.1, this means that $\mathbf{BT}$ should also be scaled to the size of $\mathbf{A}$, so that the corresponding points are *incident*, i.e., lie on top of each other. The translation generalizes the fitting problem so that it can be used for distance representations where there is no fixed origin.

Consider now the example in Figure 20.2, where $\mathbf{Y}$ is derived from $\mathbf{X}$ by reflecting it horizontally, then rotating it by 30°, shrinking it by $s = 1/2$, and finally shifting it by the translation vector $\mathbf{t}' = (1.00, 2.00)$. Formally, $\mathbf{Y} = s\mathbf{XT}+\mathbf{1t}'$, where $\mathbf{T}$ is the rotation/reflection matrix and $\mathbf{1}$ is a vector of 1s. Given the coordinate matrices

$$\mathbf{X} = \begin{pmatrix} 1 & 2 \\ -1 & 2 \\ -1 & -2 \\ 1 & -2 \end{pmatrix} \text{ and } \mathbf{Y} = \begin{pmatrix} 0.07 & 2.62 \\ 0.93 & 3.12 \\ 1.93 & 1.38 \\ 1.07 & 0.88 \end{pmatrix},$$

we want to find s, $\mathbf{T}$, and $\mathbf{t}$ that transform $\mathbf{Y}$ back to $\mathbf{X}$. In this case, we know the solutions: because $\mathbf{Y} = s\mathbf{XT} + \mathbf{1t}'$, we subtract first $\mathbf{1t}'$ on both sides, which yields $\mathbf{Y} - \mathbf{1t}' = s\mathbf{XT}$; then, premultiplying by $1/s$ and postmultiplying by $\mathbf{T}^{-1} = \mathbf{T}'$ gives $(1/s)(\mathbf{Y} - \mathbf{1t}')\mathbf{T}' = \mathbf{X}$, which is $(1/s)\mathbf{YT}' - (1/s)\mathbf{1t}'\mathbf{T}' = \mathbf{X}$. In words: we first multiply $\mathbf{Y}$ by $1/s$, then

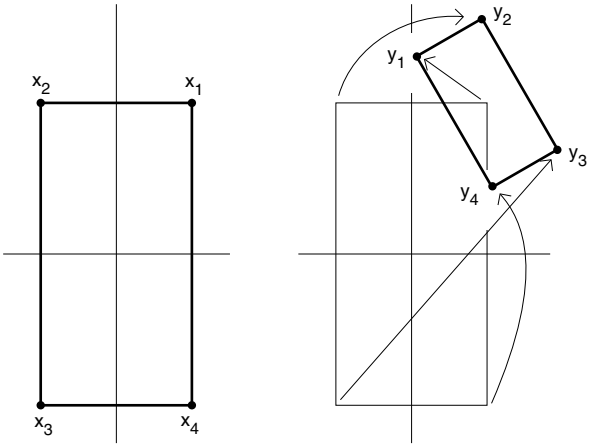


FIGURE 20.2. Illustration of fitting $\mathbf{Y}$ to $\mathbf{X}$ by a similarity transformation.

rotate it clockwise by 30° and reflect it horizontally, and then subtract the translation vector $(1/s)\mathbf{1t}'\mathbf{T}'$ from it. Because the $\mathbf{T}$ matrix is the same as the one discussed in the last section, and $1/s = 2$ and $\mathbf{t}' = (1, 2)$ are also known, the transformations involved in mapping $\mathbf{Y}$ back into $\mathbf{X}$ can be computed easily.

In general, of course, only $\mathbf{X}$ and $\mathbf{Y}$ are given, and we have to find optimal s, $\mathbf{T}$, and $\mathbf{t}$. The loss function $L(s, \mathbf{t}, \mathbf{T})$ is therefore

$$L(s, \mathbf{t}, \mathbf{T}) = \text{tr } [\mathbf{X} - (s\mathbf{YT} + \mathbf{1t}')]'[\mathbf{X} - (s\mathbf{YT} + \mathbf{1t}')], \tag{20.5}$$

subject to $\mathbf{T}'\mathbf{T} = \mathbf{I}$. An optimal translation vector $\mathbf{t}$ is obtained by setting the derivative of $L(s, \mathbf{t}, \mathbf{T})$ with respect to $\mathbf{t}$ equal to zero and solving for $\mathbf{t}$, i.e.,

$$\partial L(s, \mathbf{t}, \mathbf{T})/\partial \mathbf{t} = 2n\mathbf{t} - 2\mathbf{X}'\mathbf{1} + 2s\mathbf{T}'\mathbf{Y}'\mathbf{1} = \mathbf{0}, \tag{20.6}$$

$$\mathbf{t} = n^{-1}(\mathbf{X} - s\mathbf{YT})'\mathbf{1}. \tag{20.7}$$

Inserting the optimal $\mathbf{t}$ (20.7) into (20.5) gives

$$\begin{aligned} L(s, \mathbf{T}) &= \text{tr } [(\mathbf{X} - s\mathbf{YT}) - \frac{\mathbf{11}'}{n}(\mathbf{X} - s\mathbf{YT})]'[(\mathbf{X} - s\mathbf{YT}) - \frac{\mathbf{11}'}{n}(\mathbf{X} - s\mathbf{YT})] \\ &= \text{tr } [(\mathbf{I} - \frac{\mathbf{11}'}{n})(\mathbf{X} - s\mathbf{YT})]'[(\mathbf{I} - \frac{\mathbf{11}'}{n})(\mathbf{X} - s\mathbf{YT})] \\ &= \text{tr } [\mathbf{JX} - s\mathbf{JYT}]'[\mathbf{JX} - s\mathbf{JYT}], \end{aligned}$$

with $\mathbf{J}$ the centering matrix $\mathbf{I} - n^{-1}\mathbf{11}'$. Similarly, setting the partial derivative of $L(s, \mathbf{T})$ to s equal to zero and solving for s yields

$$\partial L(s, \mathbf{T})/\partial s = 2s(\text{tr } \mathbf{Y}'\mathbf{JY}) - 2\text{tr } \mathbf{X}'\mathbf{JYT} = 0, \tag{20.8}$$

$$s = (\text{tr } \mathbf{X}'\mathbf{JYT})/(\text{tr } \mathbf{Y}'\mathbf{JY}). \tag{20.9}$$

Inserting the optimal s into $L(s, \mathbf{T})$ gives

$$\begin{aligned} L(s,\mathbf{T}) &= \text{tr}\,[\mathbf{JX} - \frac{\text{tr}\,\mathbf{X'JYT}}{\text{tr}\,\mathbf{Y'JY}}\mathbf{JYT}]'[\mathbf{JX} - \frac{\text{tr}\,\mathbf{X'JYT}}{\text{tr}\,\mathbf{Y'JY}}\mathbf{JYT}] \\ &= \text{tr}\,\mathbf{X'JX} + \frac{(\text{tr}\,\mathbf{X'JYT})^2}{\text{tr}\,\mathbf{Y'JY}} - 2\frac{(\text{tr}\,\mathbf{X'JYT})^2}{\text{tr}\,\mathbf{Y'JY}} \\ &= \text{tr}\,\mathbf{X'JX} - \frac{(\text{tr}\,\mathbf{X'JYT})^2}{\text{tr}\,\mathbf{Y'JY}}. \end{aligned} \tag{20.10}$$

Minimizing (20.10) over $\mathbf{T}$ ($\mathbf{TT'}$=$\mathbf{I}$) is equal to minimizing $-\text{tr}\,\mathbf{X'JYT}$ over $\mathbf{T}$ because $\mathbf{T}$ may always be chosen such that $\text{tr}\,\mathbf{X'JYT}$ is nonnegative. Therefore, we can apply the results from the previous section to find the optimal $\mathbf{T}$. This also explains why maximizing the correlation $r(\mathbf{A}, \mathbf{BT})$ (see Section 20.6) or (20.1) yields the same $\mathbf{T}$ as the Procrustes problem (20.1).

The steps to compute the Procrustean similarity transformation are:

1. Compute $\mathbf{C} = \mathbf{X'JY}$.
2. Compute the SVD of $\mathbf{C}$; that is, $\mathbf{C} = \mathbf{P\Phi Q'}$.
3. The optimal rotation matrix is $\mathbf{T} = \mathbf{QP'}$.
4. The optimal dilation factor is $s = (\text{tr}\,\mathbf{X'JYT})/(\text{tr}\,\mathbf{Y'JY})$.
5. The optimal translation vector is $\mathbf{t} = n^{-1}(\mathbf{X} - s\mathbf{YT})'\mathbf{1}$.

20.5 An Example of Procrustean Similarity Transformations

We now return to Figure 20.2. To transform $\mathbf{Y}$ back to $\mathbf{X}$, the original transformations that led to $\mathbf{Y}$ have to be undone. According to our computation scheme of the previous section, what has to be found first is the orthogonal matrix $\mathbf{T}$, then the dilation factor s, and finally $\mathbf{t}$.

$\mathbf{C} = \mathbf{X'JY}$ turns out to be simply $\mathbf{C} = \mathbf{X'Y}$ in the present case, because $\mathbf{J} = \mathbf{I} - \mathbf{11'}/n$ can be seen to center the rows of $\mathbf{X'}$ or the columns of $\mathbf{Y}$. But the columns of $\mathbf{X}$ are centered already (i.e., the values in the columns of $\mathbf{X}$ sum to 0); thus $\mathbf{J}$ is not needed here. For $\mathbf{C} = \mathbf{X'Y}$ we obtain

$$\mathbf{C} = \begin{pmatrix} -1.72 & -1.00 \\ -4.00 & 6.96 \end{pmatrix}.$$

The singular value decomposition of $\mathbf{C}$, $\mathbf{C} = \mathbf{P\Phi Q'}$, is

$$\mathbf{C} = \begin{pmatrix} .00 & -1.00 \\ 1.00 & .00 \end{pmatrix} \begin{pmatrix} 8.03 & .00 \\ .00 & 1.99 \end{pmatrix} \begin{pmatrix} -.50 & .87 \\ .87 & .50 \end{pmatrix}.$$

Thus, $\mathbf{T}$ is given by

$$\mathbf{T} = \mathbf{QP}' = \begin{pmatrix} -.87 & -.50 \\ -.50 & .87 \end{pmatrix}.$$

It is easier to see what $\mathbf{T}$ does when it is decomposed into a rotation and a subsequent reflection:

$$\mathbf{T} = \mathbf{RU} = \begin{pmatrix} .87 & -.50 \\ .50 & .87 \end{pmatrix} \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}.$$

In $\mathbf{R}$, we have $\cos(\alpha) = .87$ and hence[3] $\alpha = 30°$. Also, $\sin(\alpha) = .50$; thus, $\alpha = 30°$. Hence, $\mathbf{R}$ rotates $\mathbf{Y}$ by 30° to the right or clockwise, which aligns the sides of $\mathbf{Y}$ in Figure 20.2 with the coordinate axes. $\mathbf{U}$ then undoes the previous reflection along the horizontal axis, because all of the coordinates in the first column of $\mathbf{YR}$ are reflected by -1.

The transformations s and $\mathbf{t}$ are also easy to compute. For s we compute $s = (\text{tr } \mathbf{X}'\mathbf{JYT})/(\text{tr } \mathbf{Y}'\mathbf{JY}) = 10.02/5.01 = 2$, which is just the inverse of the dilation factor from above. Finally, we find $\mathbf{t}' = (3.73, -2.47)$. It is harder to understand why such a translation is obtained, and not just $(-1, -2)$. At the beginning of the previous section, it was shown algebraically that to undo the translation $\mathbf{t}$ it is generally not correct to set $-\mathbf{t}$. This is so because other transformations are also done at the same time; thus, what has to be back-translated is not $\mathbf{Y}$, but $\mathbf{Y}$ after it has been back-rotated, -reflected, and -dilated. If we check what these transformations do to $\mathbf{Y}$ in Figure 20.2, we can see that $\mathbf{t} = (3.73, -2.47)$ must result. (Note, in particular, that $\mathbf{R}$ rotates $\mathbf{Y}$ about the origin, not about the centroid of $\mathbf{Y}$.)

20.6 Configurational Similarity and Correlation Coefficients

So far, we have considered Procrustean procedures primarily for transforming a configuration so that it becomes easier, in one sense or another, to look at. We now discuss a measure that assesses the degree of similarity between the transformed configuration and its target. One obvious choice for such a measure is the product-moment correlation coefficient computed over the corresponding coordinates of $\mathbf{X}$ and $\mathbf{YT}$.

Consider the three data matrices in Table 20.1, taken from a study by Andrews and Inglehart (1979). The matrices show the product-moment

[3]Note that $\mathbf{R}' = \mathbf{R}^{-1}$ rotates a configuration to the left or counterclockwise. See (7.31), which shows a rotation matrix for the plane that moves the points counterclockwise.

TABLE 20.1. Intercorrelations of items in three studies on well-being in the U.S.A., Italy, and Denmark, respectively. Items are: (1) housing, (2) neighborhood, (3) income, (4) standard of living, (5) job, (6) spare time activities, (7) transportation, (8) health, (9) amount of spare time, (10) treatment by others, (11) getting along with others. Decimal points omitted.

Italy

	1	2	3	4	5	6	7	8	9	10	11
1	–	33	44	52	38	37	26	23	20	27	23
2	38	–	23	19	29	23	28	18	21	29	31
3	30	21	–	77	57	43	38	25	23	17	19
4	42	30	66	–	56	52	38	28	29	22	2
5	10	18	34	23	–	49	35	23	23	32	29
6	27	28	33	36	31	–	28	28	39	25	32
7	14	19	28	29	26	26	–	18	24	18	22
8	15	11	23	21	17	15	22	–	27	17	21
9	17	15	18	26	25	29	18	08	–	20	28
10	23	30	27	30	38	26	26	20	26	–	69
11	18	18	14	24	23	21	21	29	14	36	–

U.S.A.

	1	2	3	4	5	6	7	8	9	10	11
1	–										
2	46	–									
3	38	29	–								
4	46	35	64	–							
5	29	30	42	47	–						
6	27	29	28	37	41	–					
7	16	16	22	24	20	18	–				
8	08	13	14	12	24	15	17	–			
9	22	21	19	26	24	37	15	14	–		
10	27	33	26	30	32	26	17	14	23	–	
11	31	39	29	37	34	38	22	13	28	53	-

Denmark

TABLE 20.2. Similarity coefficients of three attitude structures on well-being. Lower half: squared correlations over coordinates. Upper half: squared congruence coefficients of distances.

	U.S.A.	Italy	Denmark
U.S.A.	1.000	0.883	0.859
Italy	0.347	1.000	0.857
Denmark	0.521	0.515	1.000

correlations of 11 questions on subjective well-being asked in the U.S.A., Italy, and Denmark, respectively. The questions were always phrased according to the schema "How satisfied are you with [X]?". The interviewees responded by giving a score on a rating scale. The scores were correlated over persons. Data from representative samples in each of nine different Western societies were available. The general hypothesis was that the attitude structures on well-being in these countries would be very similar.

Andrews and Inglehart (1979) represented each of these correlation matrices in a 3D MDS space. For the matrices in Table 20.1, this leads to the Stress values .09, .08, and .04, respectively. It was then asked how similar each pair of these configurations is, and Procrustean transformations were used to "remove inconsequential differences in the original locations, orientations, and sizes of the configurations" (Andrews & Inglehart, 1979, p.78). For our three data sets, this leads to the indices in Table 20.2 (lower half). (Note that we report squared correlations here, which is in agreement with

the notion of common variance in statistics.) On the basis of such measures, Andrews and Inglehart conclude that "there seems to be a basic similarity in structures among all nine of these Western societies" (p.83).

Such an evaluation assumes that the observed similarity indices are greater than can be expected by chance alone. For two configurations, $\mathbf{X}$ and $\mathbf{Y}$, both chosen completely at random, $r(\mathbf{X}, s\mathbf{YT} + \mathbf{1t}') = r(\mathbf{X}, \mathbf{Y}^*)$ would probably not be zero but should be positive. The fewer points there are in $\mathbf{X}$ and $\mathbf{Y}$, the greater the correlation should be. The Procrustean transformations are designed to maximize $r(\mathbf{X}, \mathbf{Y}^*)$; the fewer points there are, the greater the effect of these transformations, in general. Langeheine (1980b, 1982) has studied by extensive computer simulations what r-values could be expected for different numbers of points (n) and different dimensionalities (m). He finds virtually the same results for different error models (such as sampling the points from multidimensional rectangular or normal distributions). For $n = 10$ and $m = 3$, the parameters relevant for the present 3D MDS configurations with ten points, he reports $0.072 \leq r^2(\mathbf{X}, \mathbf{Y}^*) \leq 0.522$ and $\bar{r}^2(\mathbf{X}, \mathbf{Y}^*) = 0.260$. Furthermore, only 5% of the observed coefficients were greater than 0.457. We should therefore conclude that the degree of similarity observed for these structures is hardly impressive.

20.7 Configurational Similarity and Congruence Coefficients

It is possible to skip the Procrustean transformations altogether and still arrive at a measure of similarity for each pair of configurations. This can be done by directly comparing the distances of $\mathbf{X}$ and $\mathbf{Y}$, because their ratios remain the same under any transformations where $\mathbf{T}'\mathbf{T} = \mathbf{I}$. Thus, Shepard (1966) computes the product-moment correlation coefficient over the corresponding distances of $\mathbf{X}$ and $\mathbf{Y}$, and Poor and Wherry (1976) report extensive simulations on the behavior of such correlations in randomly chosen configurations. Yet, the usual correlation is an inadmissible and misleading index when used on distances. To see why, consider the following example. Assume that $\mathbf{X}$ and $\mathbf{Y}$ consist of three points each. Let the distances in $\mathbf{X}$ be $d_{12}(\mathbf{X}) = 1, d_{23}(\mathbf{X}) = 2, d_{13}(\mathbf{X}) = 3$ and the distances in $\mathbf{Y}$, $d_{12}(\mathbf{Y}) = 2, d_{23}(\mathbf{Y}) = 3, d_{13}(\mathbf{Y}) = 4$. The correlation of these distances is $r = 1$, indicating perfect similarity of $\mathbf{X}$ and $\mathbf{Y}$. But this is false; $\mathbf{X}$ and $\mathbf{Y}$ do not have the same shape: $\mathbf{Y}$ forms a triangle, but $\mathbf{X}$'s points lie on a straight line because they satisfy the equation $d_{12}(\mathbf{X}) + d_{23}(\mathbf{X}) = d_{13}(\mathbf{X})$. If a constant s is subtracted from each distance in this equation, the inequality $d_{12}(\mathbf{X}) - k + d_{23}(\mathbf{X}) - k \neq d_{13}(\mathbf{X}) - k$ results. The translated values $v_{ij} = d_{ij}(\mathbf{X}) - k$ are therefore not distances of three collinear points. Thus, pulling out any nonzero constant from the distances implies that the

new values are either distances of a configuration different from the one we wanted to assess, or are not even distances at all, i.e., they correspond to no geometric configuration whatsoever. Hence, correlating distances does not properly assess the similarity of geometric figures (configurations).

The problem is easily resolved, however, if we do not extract the mean from the distances and compute a correlation about the origin, not the centroid. The resulting *congruence coefficient* is

$$c(\mathbf{X},\mathbf{Y}) = \frac{\sum_{i<j} w_{ij} d_{ij}(\mathbf{X}) d_{ij}(\mathbf{Y})}{[\sum_{i<j} w_{ij} d_{ij}^2(\mathbf{X})]^{1/2} [\sum_{i<j} w_{ij} d_{ij}^2(\mathbf{Y})]^{1/2}} \quad ,$$

with w_{ij} nonnegative weights. Because distances are nonnegative and, by the Cauchy–Schwarz inequality, $c(\mathbf{X},\mathbf{Y})$ ranges from 0 to 1, we have $c(\mathbf{X},\mathbf{Y}) = 1$ if $\mathbf{X}$ and $\mathbf{Y}$ are perfectly similar [i.e., if $r(\mathbf{X}, s\mathbf{Y}\mathbf{T} + \mathbf{1}\mathbf{t}') = 1$], and $c = 0$ if $r = 0$. But apart from these boundary cases, there is no easy relation of r and c and it seems impossible to convert a given r-value directly into the corresponding c-value, and vice versa.

Computing the congruence coefficients for the MDS configurations of the data in Table 20.1 yields the values in the upper half of Table 20.2. In comparison with the Procrustean correlation values in the lower half of the matrix, these measures lead to a different interpretation of the similarity pattern: the similarity of the Italian and the U.S.A. configurations is lowest in terms of r but highest in terms of c. Indeed, the order of the similarities among countries is exactly the opposite for both coefficients. Thus, using two equally admissible indices leads to different conclusions. Why this is so is not difficult to see: each coefficient must condense a great deal of information on the similarity of two configurations into a single number, and this can be done by weighting this or that piece of information more or less. Furthermore, the distinction between geometric and correlational similarity should be noted in problems of this sort.

The question that remains is whether r and c typically yield different answers in practical applications. In particular, by comparing r with its statistical norms (Langeheine, 1982) and c with analogous norms (Leutner & Borg, 1983), are we likely to conclude in one case that the configuration pair is significantly similar and in the other that it is not? In simulation studies, Borg and Leutner (1985) showed that, for randomly chosen configurations with different numbers of points and dimensionalities, r and c led to the same statistical judgment in not more than 60% of the cases. Hence, if we claim that two configurations are more similar than can reasonably be expected by chance alone, both the r and c values should be well above their respective statistical norms.

The problems associated with such similarity coefficients are ultimately due to the fact that these measures are extrinsic to substantive problems. It would be an illusion to believe that somehow a better coefficient could be constructed, because any such coefficient must condense the given complex

information into a single number. It is evident that the way this should be done depends on the substantive question being studied. Moreover, it seems that, in a case like the Andrews–Inglehart study on attitude structures, the question of how close corresponding points can be brought to each other is much too parametric. The formal reason is that with 10 points in a 3D space, the MDS configurations are not strongly determined by Stress; that is, many other configurations exist (some more similar, some less similar among themselves) that represent the data almost equally well. This was shown by Borg and Bergermaier (1981). The deeper scientific reason is that there is actually no basis for expecting such a point-by-point matching of different attitude structures in the first place. In Section 5.3, the similarity question was therefore asked quite differently: can two (or more) MDS configurations both be partitioned into regions by facets from the *same* facet design [see also Shye (1981)]. Because this could be done, it was concluded that the structures were indeed similar in the sense that they all reflected the same facets. In other contexts, the pointwise matching of two configurations may be more meaningful, but this has to be checked in each single case. For the psychophysical example discussed in Section 17.4, for example, such indices are adequate in Figure 17.8 to assess the fit of the design configuration (transformed in a theoretically meaningful way) and the respective MDS representations. It is a widespread fallacy, however, to believe that such indices are somehow "harder" and "more meaningful" than the pictures themselves. Rather, the indices play only a supplementary role, because the pictures show in detail where the configurations match and where they do not.

20.8 Artificial Target Matrices in Procrustean Analysis

Procrustean procedures were first introduced in factor analysis because it frequently deals with relatively high-dimensional vector configurations which would otherwise be hard to compare. Moreover, with very few exceptions [e.g., the radex of Guttman (1954) or the positive manifold hypothesis of Thurstone (1935)], factor analysts have been interested only in dimensions, whose similarity can be seen directly from comparing $\mathbf{X}$ and $\mathbf{YT}$. In addition, it was soon noted that the Procrustean methods can also be used in a confirmatory manner, where $\mathbf{X}$ does not relate to a configuration of empirical data but is a matrix constructed to express a substantive hypothesis; for example, $\mathbf{X}$ could contain the point coordinates for an expected configuration in a psychophysical study such as the one on rectangles in Section 17.4. Here, we might simply take the solid grid configuration in Figure 17.7 as a target for the MDS configuration in Figure 17.8 (assuming, for the sake of argument, that the MDS configuration

was generated under the Euclidean metric, because otherwise no rotations are admissible). Of course, in the plane, such rotations are more cosmetic (to obtain better aligned plots, e.g.) and we can easily do without them. However, imagine that the stimuli had been boxes rather than rectangles. A theory for the similarity judgments on such stimuli would certainly ask for a 3D representation, but the common principal-component orientation routinely used by most MDS procedures may not give us the desired orientation. Hence, even though using the design configuration without, say, any logarithmic rescalings of its dimensions may be primitive, it may lead to a more interpretable orientation of the MDS configuration.

Sometimes the target matrix $\mathbf{X}$ and testee matrix $\mathbf{Y}$ do not to have the same dimensionality. For example, in the rectangle study from above, we might have various other variables associated with the rectangles (such as different colorings and patterns). A higher-dimensional MDS space is then probably necessary to represent their similarity scores. Nevertheless, the 2D design lattice can still serve to derive a *partial* target matrix $\mathbf{X}$, which can serve as a partial hypothesis structure for the higher-dimensional MDS configuration. Technically, what needs to be done in this case to guarantee that the necessary matrix computations can be carried out is to append columns of zeros on $\mathbf{X}$ until the column orders of the augmented $\mathbf{X}$ and the $\mathbf{Y}$ matrix match. A reference for procedures on missing dimensionalities in Procrustean analysis is Peay (1988).

A further generalization of the Procrustean procedures allows partial specification of $\mathbf{X}$ by leaving some of its elements undefined (Browne, 1972a; Ten Berge, Kiers, & Commandeur, 1993). This possibility is needed when only partial hypotheses exist. A typical application is the case in which some points represent previously investigated variables and the remaining variables are "new" ones. We might then use the configuration from a previous study as a partial target for the present data in order to check how well this structure has been replicated. A different strategy was pursued by Commandeur (1991) in the MATCHALS program where entire rows can be undefined.

Another case of an incomplete formal match of $\mathbf{X}$ and $\mathbf{Y}$ is one in which the configurations contain a different number of points. Consider a study of Levy (1976) concerned with the psychology of well-being. Using a facet-theoretical approach, Levy used items based on two facets: A = {*state, resource*} and B = *life area* with eight elements. Respondents were asked how satisfied they were with the content of an item on a 9-point rating scale. For example, the respondent had to indicate how satisfied he or she was with "the city as place to live" on a scale of -4 for "very dissatisfied" to $+4$ for "very satisfied". The data were taken from two studies carried out in 1971, one in the U.S. and one in Israel. Similarity coefficients were derived from the items (correlations for the U.S. study and μ_2 for the Israel study), followed by an MDS analysis for each country. The resulting coordinate matrices for the configurations are given in Table 20.3. There

TABLE 20.3. Generating comparable matrices $\mathbf{X}_c$ and $\mathbf{Y}_c$ by averaging the coordinates of points in $\mathbf{X}$ and $\mathbf{Y}$ that have equivalent structuples, dropping rows that do not have matching structuples, and permuting the rows of the resulting matrices into a common order of structuples. Bold-face structuples are common to both studies.

	U.S. Study		Structuple	Structuple	$\mathbf{X}_c$	
1	82.878	-42.163	**23**	23	83.014	-41.638
2	88.993	-60.939	**23**	17	-4.189	-31.551
3	60.183	-46.662	**23**	14	3.004	-8.451
4	100.000	-16.787	**23**	26	-100.000	-28.496
5	-13.325	-87.959	21	22	19.631	-46.593
6	-19.009	-100.000	21	18	8.226	-15.692
7	-4.189	-31.551	**17**			
8	3.004	-8.451	**14**			
9	-100.000	-28.496	**26**			
10	27.065	-38.147	12			
11	19.631	-46.593	**22**			
12	41.695	20.110	29			
13	-7.944	40.172	25			
14	7.994	15.670	15			
15	8.226	-15.692	**18**			
	Israel Study		Structuple	Structuple	$\mathbf{Y}_c$	
1	55.109	-38.601	**22**	23	100.000	-87.625
2	100.000	-87.625	**23**	17	-20.501	45.374
3	-100.000	-59.374	**26**	14	9.139	9.563
4	-89.866	-100.000	**26**	26	-94.933	-79.687
5	-50.625	-60.229	16	22	55.109	-38.601
6	3.523	-48.208	**18**	18	-12.976	-39.149
7	-20.501	45.374	**17**			
8	-31.567	49.099	27			
9	-29.474	-30.089	**18**			
10	9.139	-9.563	14			

are 15 points in the U.S. representation, but only 10 in the Israeli solution. However, most of these points are associated with structuples that are common across the two studies. Hence, we can proceed as indicated in Table 20.3: (1) in each configuration, average the coordinates of all points that have common structuples; (2) set up matrices $\mathbf{X}_c$ and $\mathbf{Y}_c$ consisting of the average coordinate vectors in such a way that the rows of $\mathbf{X}_c$ and $\mathbf{Y}_c$ correspond substantively (i.e., in terms of their structuples); centroids without a partner in the other configuration are dropped; (3) with $\mathbf{X}_c$ and $\mathbf{Y}_c$ proceed as in a usual Procrustean problem; (4) finally, use the transformations computed in (3) to transform the original matrices (Borg, 1977b, 1978a). Provided there are enough different common structuples, this procedure does what can be done to make the configurations easier to compare.

20.9 Other Generalizations of Procrustean Analysis

Here, we consider variants of the Procrustes problem. In particular, we discuss the so-called oblique Procrustean rotation, rotation to optimal congruence, and robust Procrustean rotation.

The problem of *oblique* Procrustean rotation has been encountered previously in this book under different names. It consists of rotating each coordinate axis independently of the others in such a way that $\mathbf{BT}$ approximates $\mathbf{A}$ as closely as possible. Thus, we want to minimize (20.1) without any constraint on $\mathbf{T}$. Such a solution can be readily found by multiple regression for each dimension separately.

Oblique Procrustes rotation can be interpreted as follows. Let $\mathbf{T}$ be decomposed by a singular value decomposition; then $\mathbf{T} = \mathbf{P}\mathbf{\Phi}\mathbf{Q}'$. It follows what $\mathbf{T}$ does: first, $\mathbf{B}$ is rotated by $\mathbf{P}$; then $\mathbf{\Phi}$ multiplies the coordinate vectors of $\mathbf{BP}$ with different weights, thus geometrically stretching $\mathbf{BP}$ differentially along the axes, and finally $\mathbf{BP\Phi}$ is rotated by $\mathbf{Q}'$. Hence, only if $\mathbf{\Phi} = \mathbf{I}$ is $\mathbf{T} = \mathbf{PQ}'$ an orthonormal matrix. The transformation problem turns out to be the same as the one encountered in Section 4.3, where *external* scales had to be optimally placed into a given configuration. In factor analysis, certain additional constraints are often placed on $\mathbf{T}$, so that the oblique Procrustes problem is not always equivalent to the linear fitting. However, these additional constraints are not relevant in the MDS context (see, e.g., Browne, 1969, 1972b; Browne & Kristof, 1969; Mulaik, 1972). Applying oblique Procrustes rotation of MDS solutions has to be done with caution, because the transformed solution has different distances.

A different fit criterion for Procrustes rotation is based on the congruence between the columns of $\mathbf{A}$ and $\mathbf{BT}$. Brokken (1983) proposed a rotation method where the congruence between corresponding columns is optimized. If $\mathbf{A}$ and $\mathbf{B}$ have column means of zero, then rotation to optimal congruence can be interpreted as Procrustes rotation while assuming that each column of $\mathbf{A}$ and $\mathbf{B}$ is measured on an interval level. Kiers and Groenen (1996) developed a majorizing algorithm to optimize this criterion. This type of analysis is particularly suitable if the columns of the matrices are interval variables measured on the same scale.

In some cases, all but a few of the points of two configurations may be similar (after rotation). Verboon (1994) discusses the following artificial example. Let $\mathbf{A}$ contain the coordinates of cornerpoints of a centered square and $\mathbf{B}$ the same coordinates but rotated by 45°. An "outlier" in $\mathbf{B}$ is created by multiplying the second coordinate of point 1 by -10. This outlier has a different orientation (180°) and is located much farther from the origin than the other points. Ordinary Procrustes analysis yields a rotation matrix with an angle of $-18°$, a deviation of more than 60°.

Verboon and Heiser (1992) and Verboon (1994) propose a *robust* form of Procrustes analysis that is less sensitive to outliers. They start by decomposing the misfit into the contribution of error of each object to the total error; that is,

$$\begin{aligned} L(\mathbf{T}) &= \text{tr } (\mathbf{A}-\mathbf{BT})'(\mathbf{A}-\mathbf{BT}) \\ &= \sum_{i=1}^{n}(\mathbf{a}_i-\mathbf{T}'\mathbf{b}_i)'(\mathbf{a}_i-\mathbf{T}'\mathbf{b}_i)=\sum_{i=1}^{n} r_i^2, \end{aligned}$$

where $\mathbf{a}_i$ denotes row i of $\mathbf{A}$. The basic idea is to downweight large residuals so that outliers have less influence on the final rotation. This can be achieved by minimizing

$$L_r(\mathbf{T})=\sum_{i=1}^{n} f(r_i),$$

with $f(x)$ a robust function. Some often-used robust functions are $|x|$, the Huber function (Huber, 1964), and the biweight function (Beaton & Tukey, 1974), all of which have as a main characteristic that $f(r_i) < r_i^2$ for large values of r_i. Clearly, choosing $f(x) = x^2$ reduces $L_r(\mathbf{T})$ to $L(\mathbf{T})$. Algorithms for minimizing $L_r(\mathbf{T})$ for different robust functions $f(x)$ based on iterative majorization can be found in Verboon (1994).

20.10 Exercises

Exercise 20.1 Consider the three correlation matrices in Table 20.1 on p. 438. Scale each data matrix individually via MDS. Then use Procrustean transformations to eliminate irrelevant differences among the MDS solutions. How do the three solutions differ from one another?

Exercise 20.2 It looks as if the plane spanned by dimension 1 and dimension 2 in Figure 4.3 corresponds closely to the 2D configuration in Figure 4.1.

(a) Replicate the scalings and then fit the 3D solution to the 2D solution by Procrustean methods.

(b) Compute indices that indicate the similarity of the 2D MDS solution and the fitted plane of the 3D solution. Use two different measures of similarity.

Exercise 20.3 Use the data matrix of Table 4.1 on p. 65 and represent it in a 3D MDS space. Then use an artificial target matrix to swing the MDS solution into a plane that shows a color circle.

Exercise 20.4 The matrices below show the point coordinates of two configurations in three dimensions.

$$\mathbf{X} = \begin{bmatrix} 1 & 2 & -.5 \\ -1 & 2 & .5 \\ -1 & 0 & -.5 \\ 1 & 0 & .5 \\ -1 & -2 & .5 \\ 1 & -2 & -.5 \end{bmatrix} \text{ and } \mathbf{Y} = \begin{bmatrix} 1.2449 & -0.8589 & -1.7202 \\ 0.4572 & 1.1834 & -1.7793 \\ -0.9626 & 0.5000 & -0.5321 \\ 0.9086 & -0.4459 & 0.3364 \\ -1.1118 & 0.8645 & 1.8433 \\ -0.5361 & -1.2432 & 1.8519 \end{bmatrix}.$$

(a) Find the rotation that optimally fits $\mathbf{Y}$ to $\mathbf{X}$.

(b) Assess the fit of the fitted $\mathbf{Y}$ to the target $\mathbf{X}$.

Exercise 20.5 The matrix below shows the coordinates of four points in 4D. Transform this configuration so that it optimally fits into a 2D plane. (Hint: Procrustean transformations may not be the best method to solve this problem.)

$$\mathbf{M} = \begin{bmatrix} 1.4944 & -0.2109 & -1.5806 & -0.4718 \\ 0.2397 & 0.4019 & -1.9928 & 0.8993 \\ -1.4944 & 0.2109 & 1.5806 & 0.4718 \\ -0.2397 & -0.4019 & 1.9928 & -0.8993 \end{bmatrix}$$

Exercise 20.6 Use the coordinate matrices $\mathbf{X}$ and $\mathbf{Y}$ from Section 20.4.

(a) Augment matrix $\mathbf{Y}$ with a vector of random error so that $\mathbf{Y}$ becomes three-dimensional. Repeat the Procrustean transformations and assess the fit to the target configuration.

(b) Repeat the above with different amounts of random error. How does this error affect the fittings?

Exercise 20.7 Assume we drop the constraint that $\mathbf{TT}' = \mathbf{I}$ in Section 20.2 and admit any linear transformation $\mathbf{T}$ to solve the loss function in formula 20.1.

(a) Show that $\mathbf{T} = (\mathbf{B}'\mathbf{B})^{-1}\mathbf{B}'\mathbf{A}$ minimizes the loss function in this case. (Hint: Expand the expression and use the rules developed in Section 8.3.)

(b) Apply the result to two simple 2D configurations, $\mathbf{A}$ and $\mathbf{B}$, that are both centered.

(c) Study geometrically in which way $\mathbf{T}$ affects $\mathbf{B}$ in fitting it to $\mathbf{A}$.

(d) Analyze what $\mathbf{T}$ does to $\mathbf{B}$ in terms of its singular value decomposition. (Hint: Note the the SVD decomposes $\mathbf{T}$ into a rotation/reflection, followed by a stretching along the dimensions, followed by another rotation/reflection.)

(e) What properties of a configuration **B** are generally left unchanged when using a linear transformation **T**? (Hint: Check points that are on a straight line in **B**. Where do they end up in **BT**? Also, consider the dashed grid in Figure 17.9 and how it is related to its design grid in Figure 17.7.)

(f) Repeat fitting **B** to **A**, but now make sure that neither **A** nor **B** is centered. Compare the shape of **BT** in this case to the shape of **BT** in the centered case above.

21
Three-Way Procrustean Models

In this chapter, we look at some varieties of generalized Procrustes analysis. The simplest task is to fit several given coordinate matrices $\mathbf{X}_k (k = 1, \ldots, K)$ to each other in such a way that uninformative differences are eliminated. We also consider generalizations of the Procrustean problem that first find an optimal average configuration for all $\mathbf{X}_k$ and then attempt to explain each individual $\mathbf{X}_k$ in turn by some simple transformation of the average configuration. One important case is to admit different weights on the dimensions of the average configuration. This case defines an interesting model for individual differences scaling: if the fit is good, then the perceptual space of individual k corresponds to the group's perceptual space, except that k weights the space's dimensions in his or her own idiosyncratic way.

21.1 Generalized Procrustean Analysis

We now begin by generalizing the Procrustes problem to the case of more than two configurations. To introduce the problem, assume that we had K proximity matrices and that each matrix was generated by one of K different individuals. Assume further that we had computed an MDS solution $\mathbf{X}_k$ for each of these K individuals. What we would have, then, is a stack of $\mathbf{X}_k$s as depicted in Figure 21.1, a *three-way array* of coordinates x_{iak} $(i = 1, \ldots, n; a = 1, \ldots, m; k = 1, \ldots, K)$.

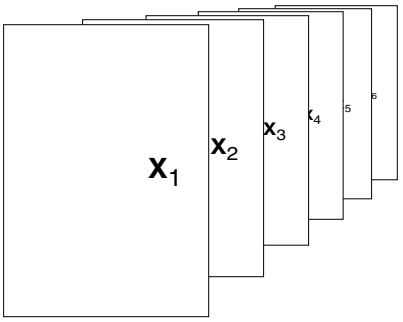


FIGURE 21.1. Schematic representation of a three-way data matrix, where the coordinate matrices $\mathbf{X}_k$ of six subjects are stacked one after the other.

We now ask to what extent the K different $\mathbf{X}_k$ "really" differ. We know from Chapter 20 that just looking at different $\mathbf{X}_k$s may be misleading, because one may notice differences that are uninformative in terms of the data. The task to visually separate uninformative from data-based differences becomes difficult or, indeed, unsolvable in the case of higher-dimensional spaces, but even in 2D it is at least helpful to first align different MDS solutions before comparing them.

Technically, given a set of K matrices $\mathbf{X}_k$, generalized Procrustean analysis is confronted with the task of optimally fitting these matrices to each other under a choice of rigid motions, dilations, and translations.

All of the above transformations are *admissible* ones, because they do not change the ratio of the distances and, hence, do not affect the way in which the various $\mathbf{X}_k$ represent the corresponding proximity data. Generalized Procrustean fitting can, however, be generalized further by admitting *nonadmissible* free parameters to the transformations. For example, after fitting the K *individual configurations* $\mathbf{X}_k$ to each other by similarity transformations, one may compute from them a *group configuration*,[1] $\mathbf{Z}$. We may then attempt to explain how the individuals differ from each other by considering certain simple transformations of $\mathbf{Z}$ that allow one to approximate each $\mathbf{X}_k$ in turn. The most important example is to compress and stretch $\mathbf{Z}$ along its dimensions so that it best explains $\mathbf{X}_k$. The dimensional weights used in these deformations of $\mathbf{Z}$ may be interpreted psychologically, for example, as expressions of the different importance that individual k attaches to the dimensions of the group space.

Gower and Dijksterhuis (2004) discuss the Procrustes problem and its three-way extensions in great mathematical depth. It is an excellent overview

[1]This choice of terminology refers to the frequent case where each $\mathbf{X}_k$ represents the proximity data from one individual k. The group configuration, then, is some kind of multidimensional average that represents the respective group of K individuals.

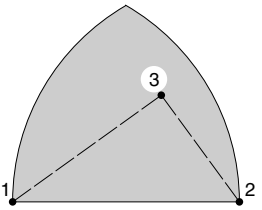


FIGURE 21.2. Data collection device used by Helm (1959).

of developments in this area. In the remainder of this chapter, we discuss a selection of three-way Procrustean models.

21.2 Helm's Color Data

We now consider an experiment by Helm (1959) that is used later on to illustrate various formal considerations. Different individuals were asked to judge the similarity of colors. The stimuli were 10 chips with different hues but constant brightness and saturation. The colors ranged over the entire spectrum from red to purple. With 10 stimuli, 120 different triples can be formed. For each triple, the subjects were asked first to identify the two colors that appeared most different. The respective chips were then placed onto points 1 and 2 in a schema like the one shown in Figure 21.2. The remaining chip of the triple was positioned somewhere in the shaded area so that the resulting distances would correspond to the perceived similarities among the colors. In this way, each subject generates more than the usual $n(n-1)/2$ distance judgments, because each stimulus pair is presented not just once but in combination with each of the remaining eight colors. The data were averaged to get more reliable estimates of the perceived distances than presenting each pair only once. The resulting values are reported in Table 21.1, where each column contains the $n(n-1)/2 = 45$ dissimilarity pairs of colors for one subject.

There were 14 different subjects, two of whom replicated the experiment after a four-week interval ($s_6^{[1]}, s_6^{[2]}$ and $s_{12}^{[1]}, s_{12}^{[2]}$) leading to a total of 16 replications. The resulting dissimilarity vectors of the two replications correlate with $r(s_6^{[1]}, s_6^{[2]}) = .96$ and $r(s_{12}^{[1]}, s_{12}^{[2]}) = .91$, which indicates high reliability for the judgments.

The subjects fall into two groups. Some of them have normal color vision, and others are deuteranopic (red-green deficient) in varying degrees. For a deuteranopic person, both red and green stimuli look gray. The subjects with deuteranopia are ordered, from least to most severe disability, as $s_{11} < s_{12} < s_{13} < s_{14}$.

Helm (1959, 1964) treats the data in Table 21.1 as direct distance estimates and applies classical scaling in maximal dimensionality, without any

TABLE 21.1. Distance estimates for color pairs (Helm, 1959) $s_6^{[1]}$ and $s_6^{[2]}$, and $s_{12}^{[1]}$ $s_{12}^{[2]}$ are replications for one subject each. Subjects s_1 to s_{10} are color-normals, and s_{11} to s_{14} are color-deficient.

	Color-Normals											Color-Deficients				
Pair	s_1	s_2	s_3	s_4	s_5	$s_6^{[1]}$	$s_6^{[2]}$	s_7	s_8	s_9	s_{10}	s_{11}	$s_{12}^{[1]}$	$s_{12}^{[2]}$	s_{13}	s_{14}
AC	6.8	5.9	7.1	7.5	6.6	5.2	5.8	6.2	7.5	6.0	9.2	11.5	9.3	9.0	10.4	9.9
AE	12.5	11.1	10.2	10.3	10.5	9.4	10.5	10.8	9.1	9.4	10.8	13.1	10.7	10.0	12.4	13.2
AG	13.8	18.8	11.1	10.7	10.2	11.4	13.4	9.9	10.2	9.5	9.7	12.6	10.7	10.4	12.8	12.3
AI	14.2	17.3	12.5	11.6	9.6	13.3	14.0	11.1	12.1	9.5	10.1	10.6	11.9	10.0	13.7	11.1
AK	12.5	16.6	11.8	10.6	10.8	12.0	13.2	10.3	12.5	9.8	10.3	10.6	11.0	9.3	11.8	8.7
AM	11.0	16.5	9.9	9.7	9.7	12.3	11.7	8.8	9.7	8.7	9.7	10.8	9.8	8.6	4.3	5.6
AO	8.6	8.3	8.6	8.4	8.5	10.6	10.2	7.6	9.8	6.7	9.0	7.3	8.9	8.8	4.0	7.4
AQ	5.5	5.7	4.3	5.8	4.9	4.9	6.4	5.8	8.3	4.9	6.6	5.4	8.9	7.5	5.5	6.4
AS	3.5	4.2	2.9	3.6	3.5	3.5	3.5	3.0	6.7	4.1	4.6	5.0	5.1	5.8	4.1	5.8
CE	5.4	4.9	5.7	6.9	5.5	6.2	4.9	7.5	4.4	7.1	5.5	6.0	6.5	6.9	8.1	7.3
CG	8.3	10.6	11.5	8.5	9.6	11.2	12.2	8.9	7.9	9.5	8.2	7.9	8.0	8.9	10.8	7.9
CI	10.4	14.3	10.7	10.7	9.3	13.5	14.8	10.7	10.4	9.5	9.4	8.4	8.2	8.4	10.4	6.9
CK	11.6	16.6	11.8	11.1	9.9	12.9	14.6	10.8	11.2	9.9	10.1	9.4	8.9	8.3	4.6	6.8
CM	13.8	17.3	11.2	12.2	11.7	12.0	14.1	10.6	12.6	10.6	10.5	10.2	9.3	9.7	9.6	9.9
CO	14.3	14.5	12.5	10.8	11.6	11.5	13.4	10.4	11.4	10.6	10.8	11.3	10.7	11.1	12.3	13.1
CQ	11.8	9.5	9.2	9.9	10.3	8.2	9.7	9.0	11.3	8.5	11.2	11.5	10.1	10.6	14.2	12.7
CS	8.9	7.3	8.2	8.0	8.0	6.3	7.9	7.5	10.4	7.9	10.5	11.5	9.6	10.3	13.0	12.1
EG	5.2	4.8	6.7	4.9	7.2	5.6	4.6	6.3	5.7	7.6	4.6	6.2	4.4	6.0	3.5	4.5
EI	7.2	8.3	8.9	6.6	8.3	8.2	8.3	8.7	8.3	8.9	6.7	8.4	7.0	6.8	4.3	5.3
EK	9.5	13.2	9.4	8.7	9.3	9.6	10.7	9.6	10.2	9.8	9.8	9.9	10.8	8.2	7.9	9.7
EM	11.3	14.6	11.3	10.6	11.3	12.7	12.8	10.1	11.3	10.5	11.3	10.3	10.4	10.9	13.0	11.5
EO	13.5	16.1	12.5	11.7	11.9	13.7	14.1	10.8	12.2	10.7	11.9	12.7	11.8	11.6	13.8	13.7
EQ	14.6	14.0	11.9	11.1	11.8	13.4	12.9	11.7	11.9	9.7	11.5	12.9	11.6	9.6	14.8	14.1
ES	14.1	13.8	10.5	12.0	11.5	11.7	10.9	9.4	10.7	10.2	10.2	10.7	10.2	10.5	13.9	13.4
GI	3.7	3.6	3.7	3.5	4.7	4.0	3.5	3.9	3.9	3.8	3.7	5.2	4.6	4.2	3.5	5.3
GK	5.9	5.3	5.9	6.3	6.2	5.8	4.7	6.8	6.5	5.3	6.6	6.5	9.6	7.3	9.0	8.6
GM	10.1	8.2	10.3	7.8	8.9	6.8	8.8	9.4	8.7	7.3	8.7	8.8	10.8	10.1	12.3	12.5
GO	11.1	14.5	11.6	10.4	10.3	9.3	11.0	9.7	10.3	7.6	10.6	11.2	11.9	10.2	12.3	13.4
GQ	12.3	17.0	10.9	11.6	11.6	10.5	11.8	10.4	10.7	9.2	10.0	11.7	11.3	10.6	12.9	14.1
GS	12.5	17.3	11.5	11.3	10.2	12.2	11.7	9.7	12.6	10.1	7.7	10.2	10.9	10.3	14.5	13.1
IK	4.2	3.5	3.6	4.1	3.3	3.8	3.6	5.0	4.6	4.8	4.0	4.1	5.8	5.2	7.0	6.9
IM	6.9	6.8	8.2	6.5	6.3	5.4	6.9	8.3	7.8	6.2	7.5	7.0	8.0	7.6	13.1	9.0
IO	10.2	11.0	9.8	8.6	9.1	7.9	9.4	9.0	9.9	8.2	9.9	10.4	10.5	9.2	13.1	12.2
IQ	12.1	15.8	11.3	10.0	11.1	9.9	12.4	10.9	11.2	9.1	10.9	10.8	10.4	10.3	13.6	12.5
IS	11.2	15.8	11.1	10.8	10.4	13.2	13.7	9.6	11.6	9.7	10.6	10.6	10.7	10.3	14.1	13.4
KM	4.3	3.8	5.1	5.0	4.2	3.6	4.1	4.3	6.3	4.7	5.4	6.4	7.7	6.4	9.9	6.7
KO	6.8	7.4	8.1	7.4	8.9	5.6	6.9	7.3	9.6	6.7	9.3	9.9	9.6	9.5	11.3	9.7
KQ	9.9	13.8	10.2	9.1	9.4	9.0	10.6	9.0	10.6	8.8	9.9	9.4	10.6	10.0	13.6	11.3
KS	10.7	15.1	10.6	10.7	10.6	10.4	12.2	8.8	11.6	9.9	9.7	10.1	10.7	9.6	12.3	9.9
MO	4.8	5.7	4.9	5.9	6.6	4.2	4.1	4.9	4.8	4.5	5.6	4.2	7.4	7.0	3.9	5.5
MQ	7.4	10.9	8.7	8.7	8.9	8.2	10.0	7.2	6.8	7.2	8.2	8.4	9.0	7.9	5.3	7.4
MS	8.7	13.9	9.7	9.6	9.2	9.8	11.1	7.6	9.1	6.8	9.7	8.1	8.7	8.7	6.4	5.4
OQ	4.5	5.0	6.3	5.6	5.8	5.1	4.1	4.7	4.6	4.0	5.3	4.5	4.5	4.8	4.7	4.2
OS	6.1	6.0	7.5	6.7	7.3	6.8	6.9	5.6	7.4	5.3	6.3	6.4	7.0	6.7	3.2	4.0
QS	3.6	3.5	3.0	3.5	2.9	3.8	3.4	3.5	5.2	3.4	3.4	3.0	4.5	4.3	2.4	4.3

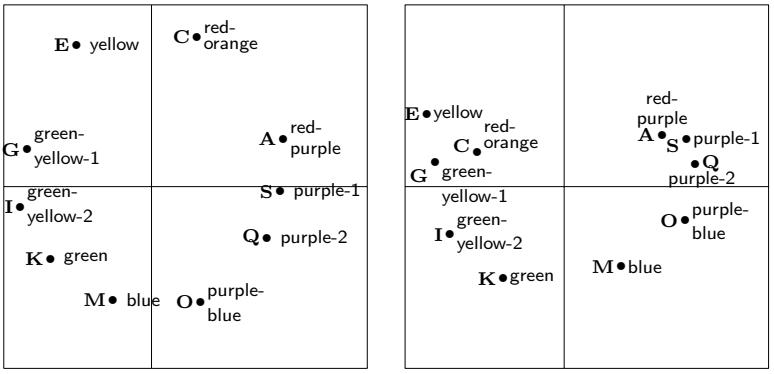
TABLE 21.2. Eigenvalues obtained by classical scaling on each of the 16 individual dissimilarity matrices of the Helm (1959) data; color-normal subjects in upper table, color-deficient subjects in lower table; "Average" shows eigenvalues for averaged data.

	s_1	s_2	s_3	s_4	s_5	$s_6^{[1]}$	$s_6^{[2]}$	s_7	s_8	s_9	s_{10}	Average
1	260.0	449.2	191.6	179.7	166.8	233.8	276.4	147.7	182.0	126.6	164.8	204.4
2	178.3	276.5	143.6	125.8	127.1	165.1	190.1	110.4	150.0	105.9	102.9	160.2
3	28.6	32.0	44.1	22.8	29.5	23.3	27.5	32.7	28.5	28.4	48.6	12.1
4	17.9	15.8	18.4	18.6	26.8	16.4	8.9	21.6	24.9	18.2	32.3	6.7
5	4.8	5.2	11.3	13.3	15.8	6.6	7.0	11.2	14.7	12.9	10.7	6.5
6	4.3	.0	5.2	8.9	7.3	1.3	2.3	7.8	10.7	7.2	6.7	4.8
7	.0	-12.6	2.8	1.0	5.6	.0	.0	6.7	7.9	3.0	6.0	2.5
8	-9.5	-17.2	.0	.0	.0	-6.2	-6.2	.0	.0	.0	.0	1.3
9	-18.2	-40.6	-5.4	-3.4	-2.1	-10.5	-17.3	-.4	-5.9	-2.4	-3.1	.0
10	-30.5	-71.8	-17.1	-8.6	-15.3	-35.8	-26.5	-8.0	-11.3	-3.0	-14.1	-.4

	s_{11}	$s_{12}^{[1]}$	$s_{12}^{[2]}$	s_{14}	s_{14}	Average
1	213.2	175.2	154.0	347.7	296.0	232.0
2	80.7	92.5	72.5	98.7	56.9	59.9
3	48.4	47.5	51.7	34.2	38.3	51.5
4	36.0	32.7	31.3	25.0	26.3	22.0
5	14.9	28.3	19.9	9.8	21.6	15.8
6	10.9	14.8	13.2	.0	13.1	11.6
7	.8	7.0	6.8	-1.8	.4	8.6
8	.0	.0	3.3	-4.0	.0	2.6
9	-3.0	-2.5	.0	-6.2	-3.1	.0
10	-13.0	-5.0	-3.0	-17.7	-13.1	-1.3

prior transformations. The eigenvalues of classical scaling for each subject are reported in Table 21.2. One notes some negative eigenvalues, because the dissimilarities are not exact distances. The negative eigenvalues are, however, relatively small and can be explained by the Messick–Abelson model (Section 19.4). On the average, the color-normal subjects have two rather large eigenvalues, with the remaining eight eigenvalues close to zero. For the deuteranopic subjects, on the other hand, we find essentially only one large eigenvalue.

If a configuration is sought that is most representative for all color-normal subjects, the simplest answer is to derive it from the scores averaged over all respective data sets. This leads to the eigenvalues shown in the column "Average" of Table 21.2. Their distribution suggests that the color-normal subjects have a true 2D MDS configuration and that further dimensions are due to errors in perception and judgment. This interpretation is buttressed by the fact that the plane spanned by the first two eigenvectors shows the expected color circle as shown in Figure 21.3a. Classical scaling on the average data of the color-deficient subjects leads to Figure 21.3b. For



(a) color normal subjects (b) color deficient subjects

FIGURE 21.3. The 2D configuration obtained by classical scaling on the average of (a) the color-normal subjects, (b) the color-deficient subjects.

these subjects, one notes that the second principal axis of the color circle is clearly less pronounced.

21.3 Generalized Procrustean Analysis

Using average data is a rather crude approach. A possible alternative is to map all 11 data sets simultaneously into one configuration. Another possibility is *generalized Procrustes* analysis (*GPA*), which transforms all K *individual* configurations, $\mathbf{X}_1, \ldots, \mathbf{X}_K$, at the same time so that each configuration matches all others as closely as possible. The admissible transformations consist of rotations, reflections, dilations, and translations, as in Procrustean similarity transformations.

Expressed in terms of a loss function, generalized Procrustes analysis amounts to minimizing

$$GPA = \sum_{k<l}^{K} \text{tr } (\widetilde{\mathbf{X}}_k - \widetilde{\mathbf{X}}_l)'(\widetilde{\mathbf{X}}_k - \widetilde{\mathbf{X}}_l), \tag{21.1}$$

where $\widetilde{\mathbf{X}}_k = s_k \mathbf{X}_k \mathbf{T}_k + \mathbf{1}\mathbf{t}_k'$ and $\mathbf{T}_k'\mathbf{T}_k = \mathbf{I}$. The function (21.1) is to be minimized through a proper choice of K scale factors s_k, K orthonormal matrices $\mathbf{T}_k$, and K translation vectors $\mathbf{t}_k$. The trivial solution where $s_k = 0$ must be avoided by imposing additional restrictions. For example, Commandeur (1991) proposes to require $\sum_k^K s_k^2 \text{tr } \widetilde{\mathbf{X}}_k' \widetilde{\mathbf{X}}_k = \sum_k^K \text{tr } \mathbf{X}_k' \mathbf{X}_k$, which we assume implicitly whenever needed.

The *GPA* loss function (21.1) has to be minimized with an iterative algorithm, because no direct analytical solution is known. We describe three

methods for minimizing *GPA*. The first method consists of cyclically updating one configuration while keeping the others fixed. Thus, each iteration consists of first updating $\widetilde{\mathbf{X}}_1$ while keeping the remaining configurations fixed, then updating $\widetilde{\mathbf{X}}_2$ while keeping the remaining configurations fixed, and so on. Writing only the terms of the *GPA* function dependent on $\widetilde{\mathbf{X}}_k$ gives

$$\begin{aligned} GPA_k(\widetilde{\mathbf{X}}_k) &= (K-1)\text{tr } \widetilde{\mathbf{X}}_k'\widetilde{\mathbf{X}}_k - 2\text{tr } \widetilde{\mathbf{X}}_k' \sum_{l\neq k} \widetilde{\mathbf{X}}_l + c \\ &= (K-1)(\text{tr } \widetilde{\mathbf{X}}_k'\widetilde{\mathbf{X}}_k - 2\text{tr } \widetilde{\mathbf{X}}_k'\mathbf{Y}) + c, \end{aligned}$$

where $\mathbf{Y} = (K-1)^{-1} \sum_{l\neq k} \widetilde{\mathbf{X}}_l$, and c contains terms that are not dependent on $\widetilde{\mathbf{X}}_k$. The minimum of $GPA_k(\widetilde{\mathbf{X}}_k)$ can be found by the Procrustean similarity transformation procedure outlined in Section 20.4. This procedure is iteratively repeated over all ks until *GPA* no longer drops. The proposed algorithm must converge, because (21.1) can never become greater as a consequence of any individual Procrustean fitting and because (21.1) has a lower bound of 0. Usually, very few iterations are required to reach convergence. The current procedure is used by Kristof and Wingersky (1971) and Ten Berge (1977).

A second procedure for solving the *GPA* problem is described by Gower (1975). Differentiating (21.1) with respect to $\mathbf{t}_k$, he first finds that all configurations must be translated so that their respective centroids are all incident with the origin. Hence, all $\mathbf{X}_k$s must be centered so that their columns sum to 0. This solves the translation problem directly. The rotation/reflection problems associated with $\mathbf{T}_k$ can then be solved in the iterative manner described above. Finally, a direct solution exists for the scale factors s_k (Ten Berge, 1977). Let $\mathbf{B}$ be the $K \times K$ matrix with elements $b_{kl} = \text{tr } \mathbf{X}_k'\mathbf{X}_l$ and $\mathbf{Q}\mathbf{\Lambda}\mathbf{Q}'$ the eigendecomposition of $\mathbf{B}$. Then, the scale factors should be chosen as $s_k = (\sum_k \mathbf{X}_k'\mathbf{X}_k / \text{tr } \mathbf{X}_k'\mathbf{X}_k)^{1/2} q_{k1}$, where q_{k1} is element k of the largest eigenvector of $\mathbf{B}$.

A third method for minimizing *GPA* uses the centroid configuration $\mathbf{Z}$ of all $\widetilde{\mathbf{X}}_k$s, $\mathbf{Z} = (1/K) \sum_k \widetilde{\mathbf{X}}_k$. The function *GPA* in (21.1) is equivalent to

$$GPA = K \sum_{k=1}^{K} \text{tr } (\widetilde{\mathbf{X}}_k - \mathbf{Z})'(\widetilde{\mathbf{X}}_k - \mathbf{Z}), \tag{21.2}$$

which is minimized by updating the $\widetilde{\mathbf{X}}_k$s and the centroid configuration $\mathbf{Z}$ one at a time while keeping the others fixed. Commandeur (1991) notes, however, that this procedure has slower convergence properties than the method described above.

To see that (21.1) is the same as (21.2), consider the following.

$$GPA = \sum_{k<l}^{K} \text{tr } (\widetilde{\mathbf{X}}_k - \widetilde{\mathbf{X}}_l)'(\widetilde{\mathbf{X}}_k - \widetilde{\mathbf{X}}_l) \tag{21.3}$$

$$= \frac{1}{2}\sum_{k=1}^{K}\sum_{l=1}^{K}\text{tr }(\widetilde{\mathbf{X}}_k - \widetilde{\mathbf{X}}_l)'(\widetilde{\mathbf{X}}_k - \widetilde{\mathbf{X}}_l) \tag{21.4}$$

$$= \frac{1}{2}\sum_{k=1}^{K}\sum_{l=1}^{K}\text{tr }\widetilde{\mathbf{X}}_k'\widetilde{\mathbf{X}}_k + \frac{1}{2}\sum_{k=1}^{K}\sum_{l=1}^{K}\text{tr }\widetilde{\mathbf{X}}_l'\widetilde{\mathbf{X}}_l - \sum_{k=1}^{K}\sum_{l=1}^{K}\text{tr }\widetilde{\mathbf{X}}_k'\widetilde{\mathbf{X}}_l. \tag{21.5}$$

Summing the first two terms of (21.5) yields $K\sum_{k=1}^{K}\text{tr }\widetilde{\mathbf{X}}_k'\widetilde{\mathbf{X}}_k$. Because $\widetilde{\mathbf{X}}_k$ in the last term of (21.5) does not depend on l, this term can be written as $K\sum_{k=1}^{K}\text{tr }\widetilde{\mathbf{X}}_k'(K^{-1}\sum_{l=1}^{K}\widetilde{\mathbf{X}}_l)$, so that

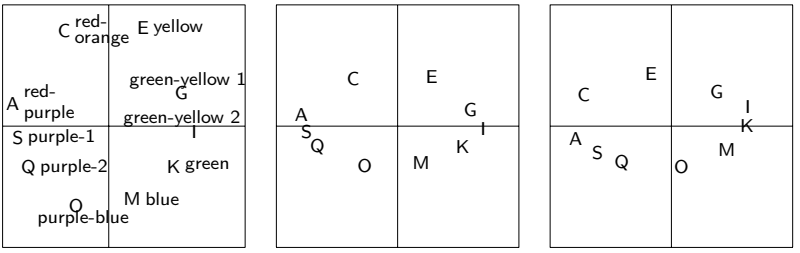
$$\begin{aligned} GPA &= K\sum_{k=1}^{K}\text{tr }\widetilde{\mathbf{X}}_k'\widetilde{\mathbf{X}}_k - K\sum_{k=1}^{K}\text{tr }\widetilde{\mathbf{X}}_k'(K^{-1}\sum_{l=1}^{K}\widetilde{\mathbf{X}}_l) \\ &= K\left(\sum_{k=1}^{K}\text{tr }\widetilde{\mathbf{X}}_k'\widetilde{\mathbf{X}}_k - \sum_{k=1}^{K}\text{tr }\widetilde{\mathbf{X}}_k'\mathbf{Z}\right). \end{aligned}$$

Using this result, the derivation continues as

$$\begin{aligned} GPA &= K\left(\sum_{k=1}^{K}\text{tr }\widetilde{\mathbf{X}}_k'\widetilde{\mathbf{X}}_k + \sum_{k=1}^{K}\text{tr }\widetilde{\mathbf{X}}_k'\mathbf{Z} - 2\sum_{k=1}^{K}\text{tr }\widetilde{\mathbf{X}}_k'\mathbf{Z}\right) \\ &= K\left(\sum_{k=1}^{K}\text{tr }\widetilde{\mathbf{X}}_k'\widetilde{\mathbf{X}}_k + K(K^{-1}\sum_{k=1}^{K}\text{tr }\widetilde{\mathbf{X}}_k)'\mathbf{Z} - 2\sum_{k=1}^{K}\text{tr }\widetilde{\mathbf{X}}_k'\mathbf{Z}\right) \\ &= K\left(\sum_{k=1}^{K}\text{tr }\widetilde{\mathbf{X}}_k'\widetilde{\mathbf{X}}_k + K(\text{tr }\mathbf{Z}'\mathbf{Z}) - 2\sum_{k=1}^{K}\text{tr }\widetilde{\mathbf{X}}_k'\mathbf{Z}\right) \\ &= K\left(\sum_{k=1}^{K}\text{tr }\widetilde{\mathbf{X}}_k'\widetilde{\mathbf{X}}_k + \sum_{k=1}^{K}\text{tr }\mathbf{Z}'\mathbf{Z} - 2\sum_{k=1}^{K}\text{tr }\widetilde{\mathbf{X}}_k'\mathbf{Z}\right) \\ &= K\sum_{k=1}^{K}\text{tr }(\widetilde{\mathbf{X}}_k - \mathbf{Z})'(\widetilde{\mathbf{X}}_k - \mathbf{Z}). \end{aligned} \tag{21.6}$$

This shows that *GPA* minimizes the squared differences of all $\widetilde{\mathbf{X}}_k$ to the centroid configuration $\mathbf{Z}$. Therefore, $\mathbf{Z}$ can be used as the configuration that summarizes all of the optimally transformed $\mathbf{X}_k$s. Dijksterhuis and Gower (1991) go one step further. They provide a much more detailed analysis-of-variance-like decomposition of the error of the *GPA* loss function, so that several sources of misfit can be attributed.

Geometrically, each of $\mathbf{Z}$'s points is the centroid of the corresponding points from the fitted individual configurations. Thus, if (21.1) is small, these centroids lie somewhere in the middle of a tight cluster of K points, where each single point belongs to a different $\widetilde{\mathbf{X}}_k$.



a. Centroid configuration

b. Weighted Euclidean distance

c. Generalized Euclidean distance

FIGURE 21.4. (a) Centroid configuration for 11 color-normal subjects in Table 21.1. (b) Example of an individual space using the weighted Euclidean distance for a hypothetical subject with dimension weights 1.0 (dim 1) and .5 (dim 2). (c) Example of an individual space using the generalized Euclidean distance for a hypothetical subject with dimension weights 1.0 (dim 1), .5 (dim 2), and idiosyncratic anticlockwise rotation of 30°.

21.4 Individual Differences Models: Dimension Weights

The generalized Procrustean transformation problem is of limited interest in practice. A more interesting question to ask is whether each individual configuration can be accounted for by stretching the centroid configuration appropriately along the dimensions. This idea for explaining individual differences was introduced by Horan (1969) and Bloxom (1968) and developed by Carroll and Chang (1970) in the INDSCAL procedure (see Section 22.1).

To illustrate the model, let us look at Figure 21.4a which shows the centroid configuration obtained by *GPA*. Every individual is allowed his or her own weights for every column of $\mathbf{Z}$. For example, a mildly color-deficient subject k could weight the first dimension by 1.0 and the second dimension by 0.5, showing that the second dimension accounts for less variance in his or her data than the first dimension. Figure 21.4a shows the centroid configuration obtained by *GPA*. The weighted centroid configuration in Figure 21.4b shows clearly that for this subject the first dimension of $\mathbf{Z}$ is more important than the second dimension.

The weighted centroid configuration for subject k can be expressed as $\mathbf{ZW}_k$, where $\mathbf{W}_k$ is an $m \times m$ diagonal matrix of nonzero dimension weights. Hence, the corresponding distance between points i and j is

$$\begin{aligned} d_{ijk}(\mathbf{ZW}_k) &= \left[\sum_{a=1}^{m}(w_{aak}z_{ia} - w_{aak}z_{ja})^2\right]^{1/2} \\ &= \left[(\mathbf{z}_i - \mathbf{z}_j)'\mathbf{W}_k^2(\mathbf{z}_i - \mathbf{z}_j)\right]^{1/2}, \end{aligned} \tag{21.7}$$

where $\mathbf{z}_i'$ is row i of $\mathbf{Z}$. Equation (21.7) is called the *weighted Euclidean distance*.[2] Note w_{aak} may be positive or not: a negative dimension weight simply reflects the corresponding axis but does not change the distances.

An extension of dimension weighting allows for *idiosyncratic rotations* as well. Before applying dimension weights, an individual would first orient $\mathbf{Z}$ in his or her particular way. The transformed centroid configuration for individual k becomes $\mathbf{Z}\mathbf{S}_k\mathbf{W}_k$, where $\mathbf{S}_k$ is a rotation matrix with $\mathbf{S}_k'\mathbf{S}_k = \mathbf{S}_k\mathbf{S}_k' = \mathbf{I}$. The *generalized Euclidean distance* is

$$d_{ijk}(\mathbf{Z}\mathbf{S}_k\mathbf{W}_k) \quad = \quad \left[(\mathbf{z}_i - \mathbf{z}_j)'\mathbf{S}_k\mathbf{W}_k^2\mathbf{S}_k'(\mathbf{z}_i - \mathbf{z}_j)\right]^{1/2}. \qquad (21.8)$$

The use of this type of distance was popularized by Carroll and Wish (1974a) in the IDIOSCAL model (see Section 22.2). In Figure 21.4c, the perceptual space of a hypothetical individual is shown. This individual first rotates the centroid configuration of Figure 21.4a by 30° anticlockwise and then weights the newly obtained axes by $w_{11k} = 1.0$ and $w_{22k} = 0.5$.

Helm's Color Data and the Subject Space

Consider our color perception example. Figure 21.4a shows the centroid configuration $\mathbf{Z}$ derived from the 11 MDS configurations of the color-normal subjects by minimizing (21.1). $\mathbf{Z}$ matches each of the 11 individual configurations exceedingly well, as can be seen from the $r^2(\widetilde{\mathbf{X}}_k, \mathbf{Z})$ values in Table 21.3. None of the fit values shows an agreement of less than 96%; hence, $\mathbf{Z}$ is truly representative for these subjects. In Figure 21.4a, the coordinate axes are rotated so that the vertical dimension intersects the color circle at the points red-purple and green-blue. But these are just the colors that the deuteranopic subjects cannot reliably discriminate, although they have no problems distinguishing yellow from blue. Thus, their color circles should be squeezed together in the red-green direction, because the point distances represent the perceived dissimilarities. Figure 21.4b shows what distance structures would be expected for a mildly deuteranopic subject. In this case, the transformation can be represented in terms of point coordinates as $\widetilde{\mathbf{X}}_k \approx \mathbf{Z}\mathbf{W}_k$, where $\mathbf{Z}\mathbf{W}_k$ is the approximated MDS configuration of individual k, and $\mathbf{W}_k$ is a 2×2 diagonal matrix consisting of the weights $w_{11} = 1.00$ and $w_{22} = 0.5$, which has the effect of shrinking all coordinates in $\mathbf{Z}$'s second column to a half of their original magnitude.

In general, this fitting problem can be expressed as

$$\text{tr } (\mathbf{Z}\mathbf{W}_k - \widetilde{\mathbf{X}}_k)'(\mathbf{Z}\mathbf{W}_k - \widetilde{\mathbf{X}}_k) = \min, \qquad (21.9)$$

[2] Actually, d_{ijk} is simply a Euclidean distance on a "weighted" MDS space, $\mathbf{X}_k = \mathbf{Z}\mathbf{W}_k$. The term "weighted Euclidean distance", therefore, characterizes formula (21.7) but does not imply that we are dealing with a special type of distance.

TABLE 21.3. Fit measures for simple and weighted Procrustean analyses of MDS configurations of Helm data, split by color-normal subjects and color-deficient subjects.

Color-Normals	$r^2(\widetilde{\mathbf{X}}_k, \mathbf{Z})$	$r^2(\widetilde{\mathbf{X}}_k, \mathbf{Z}\mathbf{W}_k)$
s_1	0.98	0.98
s_2	0.98	0.98
s_3	0.99	0.99
s_4	0.99	0.99
s_5	0.99	0.99
$s_6^{[1]}$	0.97	0.97
$s_6^{[2]}$	0.98	0.99
s_7	0.98	0.98
s_8	0.97	0.97
s_9	0.98	0.98
s_{10}	0.96	0.97
Average	0.98	0.98

Color-Deficients	$r^2(\widetilde{\mathbf{X}}_k, \mathbf{Z})$	$r^2(\widetilde{\mathbf{X}}_k, \mathbf{Z}\mathbf{W}_k)$
s_{11}	0.89	0.92
$s_{12}^{[1]}$	0.91	0.92
$s_{12}^{[2]}$	0.94	0.97
s_{13}	0.50	0.75
s_{14}	0.44	0.82
Average	0.74	0.88

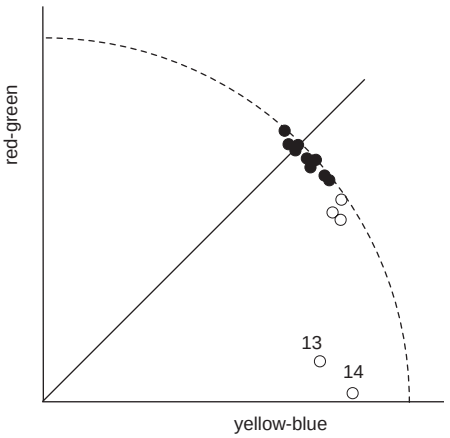


FIGURE 21.5. Subject space for centroid configuration in Fig. 21.4a and data in Table 21.1; solid points represent color-normal and open points color-deficient subjects.

where $\mathbf{W}_k$ is the unknown diagonal matrix of weights, and $\widetilde{\mathbf{X}}_k$ is the ith individual configuration optimally fitted to $\mathbf{Z}$ by similarity transformations. Because $\mathbf{W}_k$ is diagonal, finding the best $\mathbf{W}_k$ amounts to solving a set of simple regression problems. To see this, let $\mathbf{z}_a$ and $\mathbf{x}_a$ be the column vectors of of $\mathbf{Z}$ and $\widetilde{\mathbf{X}}_a$, respectively, and z_{ia} and x_{ia} the ith elements in these vectors. Then, find the weight w_a (the ath diagonal element of $\mathbf{W}_k$) for $\mathbf{z}_a$ such that $\sum_{i=1}^{n}(w_a z_{ia} - x_{ia})^2$ is minimal. Differentiating and setting the derivative equal to 0 leads to $w_a = (\sum_i x_{ia} z_{ia}) / \sum_i z_{ia}^2$, the formula for the regression coefficient. With such weights in each $\mathbf{W}_k$, the agreement of the $\mathbf{ZW}_k$s and the individual configurations of the deuteranopic subjects goes up substantially relative to the unweighted case (Table 21.3). The size of the increments mirrors the degree of deuteranopia.

If the weights are normalized appropriately (see below), they can be displayed as in Figure 21.5. The diagram, known as a *subject space* (Carroll & Chang, 1970), shows the color-normal persons represented by solid points and the color-deficient subjects by open points. The coordinates of these points correspond to the weights assigned to the dimensions by the respective individual. The color-normal persons weight the centroid configuration on both the red-green (X-axis) and the yellow-blue (Y-axis) dimensions about equally, because the points cluster tightly around the bisector. The color-deficient persons, on the other hand, weight the red-green dimension of $\mathbf{Z}$ less than the yellow-blue dimension. Moreover, the open point closest to the yellow-blue axis represents s_{14}, the individual with the most severe case of deuteranopia, and the open point next to it stands for s_{13}, who is next in color deficiency. For these individuals, the red-green dimension is practically irrelevant for their dissimilarity judgments, as predicted.

For the subject space, the dimension weights w_{aak} were normed so that their sum-of-squares is equal to $r^2(\widetilde{\mathbf{X}}_k, \mathbf{ZW}_k)$. This equality holds if the normed weights, $\bar{w}_{aak}$, satisfy $\bar{w}_{aak} = w_{aak} / (\sum_i z_{ia}^2)^{1/2}$, which follows from writing out the squared correlation $r^2(\widetilde{\mathbf{X}}_k, \mathbf{ZW}_k)$ in scalar notation (Borg, 1977a). Thus, the distance of the points in the subject space from the origin corresponds to the communality of the weighted average configuration and an individual configuration. We have $\widetilde{\mathbf{X}}_k = \mathbf{ZW}_k$ if the weight point of individual k lies on the circle with radius 1 around the origin.

Common Misinterpretations of the Subject Space

We should note here that the subject space depends on how the group space $\mathbf{Z}$ is defined. In the above example, $\mathbf{Z}$ was the centroid configuration of the color-normal persons only. Alternatively, it would also be possible to derive $\mathbf{Z}$ from, say, the configurations of all subjects. But then $\mathbf{Z}$ would have a different shape: it would be more elliptical, and this would entail that all points and stars in the subject space be rotated towards the red-green dimension, so that the color-normals would not be distributed around

the bisector. Hence, it is not possible to infer from a subject's point in the subject space that he or she weights dimension a more than dimension b, because these ratios change if we change the group space. The weights, thus, are only relative to those of other subjects.

Another misinterpretation can occur if we take the name "subject space" in the sense of a Euclidean point space. The meaning of distances computed by the Euclidean distance formula in the subject space is completely obscure, because each of its points is placed so that the distance from the origin denotes the communality of the respective configuration $\widetilde{\mathbf{X}}_k$ with $\mathbf{Z}$, and the direction of the ray on which the point lies represents the weights in $\mathbf{W}_k$. Moreover, as we mentioned above, the distances in the subject space are conditional on how $\mathbf{Z}$ is defined: if we choose different $\mathbf{Z}$s, different subject spaces result. To avoid the distance interpretation between points in the subject space, one can project the points on the dimensions and interpret the weights for each dimension separately.

Dimension Weighting with Idiosyncratic Rotations

The dimensional weighting of $\mathbf{Z}$ in Figure 21.4 was done along the given dimensions red-green and yellow-blue. Different ellipses would have resulted in Figures 21.4b and 21.4c if $\mathbf{Z}$ were squeezed together in other *directions*. For example, we could squeeze $\mathbf{Z}$ in the direction purple 1/green-yellow 2, which would bring the points S and I in close proximity. Expressed more technically, if the dimensional system in Figure 21.4a were rotated by $\mathbf{S}$, then weighting $\mathbf{ZS}$ by $\mathbf{W}_k$ would lead to a different result than weighting $\mathbf{Z}$ by $\mathbf{W}_k$, in general. Thus, for each $\mathbf{S}$, we obtain the loss function

$$\text{tr } (\mathbf{ZSW}_k - \widetilde{\mathbf{X}}_k)'(\mathbf{ZSW}_k - \widetilde{\mathbf{X}}_k), \tag{21.10}$$

where $\mathbf{S}'\mathbf{S} = \mathbf{I}$, and $\widetilde{\mathbf{X}}_k$ is an individual configuration fitted optimally to $\mathbf{ZS}$. Note that $\mathbf{S}$ does not have a subscript here, so that $\mathbf{ZS}$ is the group space for all individuals.

How do we minimize (21.10 over all $k = 1, \ldots, K$ individuals? To do this, we need to find the best $\mathbf{S}$ in

$$L = \sum_{k=1}^{K} \text{tr } (\mathbf{ZSW}_k - \mathbf{X}_k\mathbf{T}_k)'(\mathbf{ZSW}_k - \mathbf{X}_k\mathbf{T}_k), \tag{21.11}$$

where we write $\mathbf{X}_k\mathbf{T}_k$ (with $\mathbf{T}_k'\mathbf{T}_k = \mathbf{I}$) for $\widetilde{\mathbf{X}}_k$, because it turns out that the optimal translation of an individual configuration is always to center it, and because the optimal scaling factor becomes irrelevant when correlations are used as similarity measures (Lingoes & Borg, 1978). Equation (21.11) involves the unknown $\mathbf{S}$, and K unknown weight matrices $\mathbf{W}_k$ and rotations $\mathbf{T}_k$. To find the optimal matrices is a difficult problem, and it is useful to consider a simpler case first.

Let $\mathbf{S}_k$ be an *idiosyncratic* rotation, that is, a different rotation matrix $\mathbf{S}_k$ for every subject k. Find $\mathbf{S}_k$ and $\mathbf{W}_k$ in

$$\text{tr } (\mathbf{Z}\mathbf{S}_k\mathbf{W}_k - \mathbf{X}_k\mathbf{T}_k)'(\mathbf{Z}\mathbf{S}_k\mathbf{W}_k - \mathbf{X}_k\mathbf{T}_k). \tag{21.12}$$

In terms of our color perception example, we want to find a rotation $\mathbf{S}_k$ and a set of dimensional weights $\mathbf{W}_k$ that distort the color circle in Figure 21.4a such that individual k-th's configuration, rotated appropriately by $\mathbf{T}_k$, is approximated as closely as possible. A direct solution for this problem is known only for the 2D case (Lingoes & Borg, 1978; Mooijaart & Commandeur, 1990). This solution can be used for each plane of the space in turn, and then one can repeat the fittings iteratively, because every $m \times m$ rotation matrix can be expressed as the product of $m(m-1)/2$ planar rotation matrices (see also Section 7.10). The average of all $\mathbf{Z}\mathbf{S}_k$s is then used as a target matrix to solve for the $\mathbf{Z}\mathbf{S}$ of (21.11).

We obtain a group space that is uniquely rotated; that is, L in (21.11) is minimal for one particular $\mathbf{S}$ and increases for any other rotation, except in special cases. This will be so for any $\mathbf{Z}$, whether it represents empirical data or whether it has been defined completely at random. Hence, to conclude that "this method automatically gives psychologically meaningful axes, if they exist" (Indow, 1974, p.497) appears too optimistic. It may just be the case that no dimensional theory is psychologically relevant and that the dimension-weighting model, which leads to the unique dimensions, yields nothing but substantively empty formal relations. But even when the model is adequate, we should keep in mind that the rotational uniqueness is an algebraic property. For real data, which always have error components, it may just be that the resulting dimensions are a consequence of the particular error distribution. In any case, we usually find that the rotational uniqueness is statistically weak. Thus, for formal as well as substantive reasons, we recommend rotating $\mathbf{Z}$ such that the dimensions reflect some substantive theory (as in Figure 21.4) rather than leaving the finding of the coordinate axes to a blind procedure.

21.5 An Application of the Dimension-Weighting Model

Consider another example. Green and Rao (1972) asked 41 individuals to evaluate the 15 breakfast objects described in Table 14.1 with respect to their pairwise similarities. (The 41 proximity matrices are presented in Green & Rao's book.) By an ordinal MDS of each of the 41 15×15 proximity matrices, 2D representations are obtained. Using the program PINDIS (Lingoes & Borg, 1977), the centroid configuration $\mathbf{Z}$ shown in Figure 21.6 is derived. The representation shows $\mathbf{Z}$ in its optimal rotation with respect to the loss function (21.12), so that we already have $\mathbf{Z}\mathbf{S}$. The

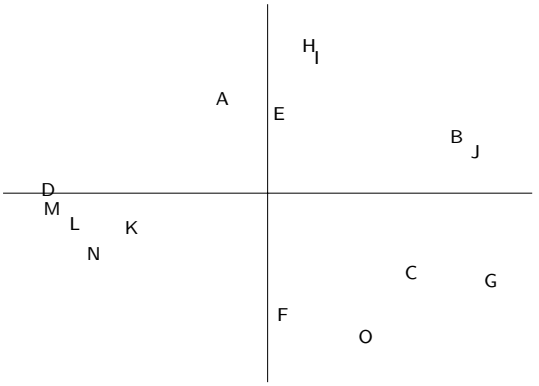


FIGURE 21.6. The PINDIS centroid configuration derived from 41 individual configurations obtained by MDS of the 15 breakfast items reported in Table 14.1 (after Borg & Lingoes, 1977).

fit measures (squared correlations, common variances, or *communalities*) between this group space and each individual configuration are given in Table 21.4 in the column $r^2(\widetilde{\mathbf{X}}_k, \mathbf{Z})$. These values vary substantially over the individuals. Although $\widetilde{\mathbf{X}}_{22}$ and $\mathbf{Z}$ share some 92% of their variance, $\widetilde{\mathbf{X}}_{38}$ and $\mathbf{Z}$ have almost nothing in common.

There is one typical misinterpretation of such results. Table 21.4 shows that, for example, $\widetilde{\mathbf{X}}_{25}$ and $\widetilde{\mathbf{X}}_{35}$ correlate with $\mathbf{Z}$ with about $r^2 = .50$. Yet we cannot infer from this value that $\widetilde{\mathbf{X}}_{25}$ and $\widetilde{\mathbf{X}}_{35}$ have anything in common: a simple, pairwise Procrustean analysis could show that $r^2(\widetilde{\mathbf{X}}_{25}, \widetilde{\mathbf{X}}_{35}) = 0$. It is easy to see why this is so. $\widetilde{\mathbf{X}}_{25}$ and $\widetilde{\mathbf{X}}_{35}$ each share some 50% variance with $\mathbf{Z}$, but these variance proportions may be complementary, so that $\mathbf{Z}$ shares with $\widetilde{\mathbf{X}}_{25}$ one-half of its variance, and with $\widetilde{\mathbf{X}}_{35}$ the remaining half. To see how similar $\widetilde{\mathbf{X}}_{25}$ and $\widetilde{\mathbf{X}}_{35}$ are, we would have to do a pairwise Procrustean analysis.

We now use dimensional weightings with and without idiosyncratic rotations. This is a purely formal exercise here because, in contrast to the color perception data considered above, there is no reason why the individuals should perceive the similarity of these breakfast items dimensionally, and also no reason why they should differ with respect to the importance of dimensions. This lack of a substantive theory is, in fact, evidenced by the very fact that we use such blindly optimizing rotations in the first place. Not surprisingly, it turns out that both dimension-weighting models do not account for much additional variance relative to the model using unit weights. Table 21.4 shows the respective fit values in columns $r^2(\widetilde{\mathbf{X}}_k, \mathbf{ZSW}_k)$ and $r^2(\widetilde{\mathbf{X}}_k, \mathbf{ZS}_k\mathbf{W}_k)$. On the average, the fit increments are just 2.6% and 4.5%, and in no individual case is there an increment of the magnitude found in Table 21.3 for the severely color-deficient persons.

TABLE 21.4. Communalities of individual configurations fitted to different transformations of centroid configuration $\mathbf{Z}$.

Subject	$r^2(\widetilde{\mathbf{X}}_k, \mathbf{Z})$	$r^2(\widetilde{\mathbf{X}}_k, \mathbf{Z}\mathbf{S}_k)$	$r^2(\widetilde{\mathbf{X}}_k, \mathbf{Z}\mathbf{S}_k\mathbf{W}_k)$	$r^2(\widetilde{\mathbf{X}}_k, \mathbf{V}_k\mathbf{Z})$
1	0.7999	0.8005	0.8008	0.8335
2	0.8520	0.8713	0.8714	0.8926
3	0.9088	0.9112	0.9159	0.9519
4	0.9194	0.9222	0.9283	0.9488
5	0.5669	0.6811	0.7352	0.8410
6	0.8939	0.9056	0.9065	0.9376
7	0.8376	0.8469	0.8480	0.8692
8	0.8365	0.9014	0.9032	0.8900
9	0.8518	0.8520	0.8849	0.8977
10	0.7369	0.7404	0.7439	0.8407
11	0.7765	0.7958	0.8803	0.8631
12	0.7044	0.7188	0.7649	0.8099
13	0.7833	0.9152	0.9161	0.9154
14	0.7772	0.7814	0.8020	0.9219
15	0.8982	0.9175	0.9175	0.9339
16	0.6199	0.6698	0.6747	0.7751
17	0.6871	0.7765	0.7805	0.8015
18	0.7881	0.8174	0.8324	0.8821
19	0.8050	0.8469	0.8493	0.8540
20	0.1118	0.1307	0.1307	0.7782
21	0.6179	0.6310	0.6631	0.6754
22	0.9222	0.9429	0.9470	0.9588
23	0.8770	0.8794	0.9005	0.9184
24	0.8721	0.8777	0.8785	0.8866
25	0.5101	0.5135	0.5419	0.8383
26	0.6827	0.6884	0.6895	0.7731
27	0.8251	0.8280	0.8385	0.8712
28	0.7198	0.7268	0.7644	0.8292
29	0.8493	0.8931	0.8936	0.9199
30	0.8593	0.8978	0.9068	0.9289
31	0.3929	0.4067	0.4695	0.8143
32	0.2728	0.2994	0.3642	0.6585
33	0.8498	0.8700	0.8776	0.9448
34	0.7973	0.8299	0.8451	0.8860
35	0.5126	0.6170	0.6623	0.6976
36	0.6076	0.6212	0.6894	0.6806
37	0.8137	0.8192	0.8322	0.8786
38	0.0192	0.0262	0.0331	0.4690
39	0.7077	0.7426	0.7478	0.8903
40	0.7824	0.8004	0.8019	0.8306
41	0.8559	0.8581	0.9178	0.9178
Mean	0.7196	0.7456	0.7647	0.8465

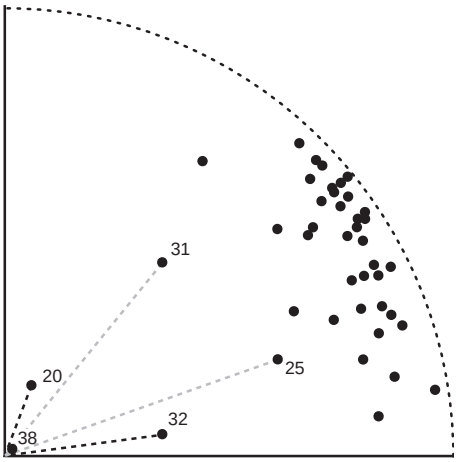


FIGURE 21.7. The subject space for the dimension-weighting model of the Green–Rao data (after Borg & Lingoes, 1977).

Most programs (e.g., INDSCAL; see Chapter 22) designed specifically to represent data in the dimension-weighting model skip the step with the unit weights, which yields the fit values in column $r^2(\widetilde{\mathbf{X}}_k, \mathbf{Z})$ of Table 21.3. In other words, they analyze the entire $41 \times 15 \times 15$ data block at once in the sense of the dimension-weighting models, which yields a group space and its related subject space of weights. This information alone is difficult to interpret, however. Consider the subject space for the dimension-weighting model of the present data (Figure 21.7). We observe that: (1) the subject points scatter considerably in their distance from the origin, which expresses the different communalities of **ZS** and the individual configurations; (2) the subject points also scatter in terms of their North-West directions, and this, as we saw, indicates that the subjects weight the dimensions differently. But, as Table 21.4 demonstrates, this second scatter really does not mean much, because if all of the points were forced onto the bisector, the model would be reduced to the unit-weighting case, whose average communality is just 2.6% lower. Thus, it would be risky to infer that because the subject points scatter so much in terms of direction, the differential weights (and with them, the particular dimensions) are meaningful or even descriptively important. The scatter simply reflects the fact that no restrictions were placed on the Procrustean procedure, so that whatever reduces the loss function most is chosen for a weight.

21.6 Vector Weightings

Because the dimension-weighting models proved ineffective in explaining the interindividual differences in the Green–Rao breakfast data, we might

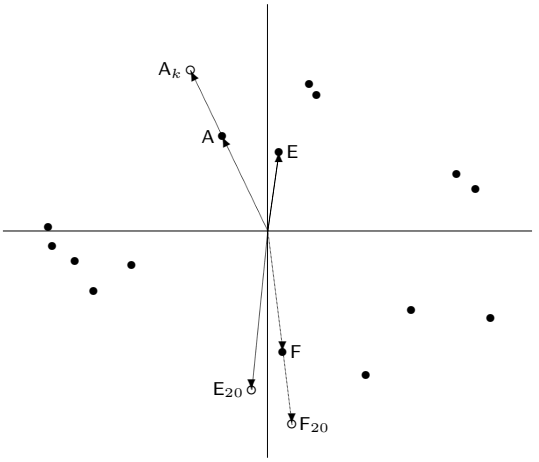


FIGURE 21.8. Illustration of some vector weights for centroid configuration and MDS configuration of subject 20 in Green–Rao study.

seek other, more successful models. Dimensional weightings are constrained by the property that the neighborhood relations of the points in $\mathbf{Z}$ are preserved in a certain way when transforming $\mathbf{Z}$ into $\mathbf{ZW}_k$. Namely, those points that are close together in $\mathbf{Z}$ are also close together in $\mathbf{ZW}_k$.

Radial Point Shifts by Vector Weightings

For the Green–Rao breakfast data, it may be possible that most individuals do indeed perceive the breakfast objects just as the *average person* $\mathbf{Z}$ does but that they see some neighborhood relations differently. We consider a particularly simple transformation, whose loss function is expressed by

$$\sum_{k=1}^{K} \text{tr } (\mathbf{V}_k\mathbf{Z} - \widetilde{\mathbf{X}}_k)'(\mathbf{V}_k\mathbf{Z} - \widetilde{\mathbf{X}}_k), \tag{21.13}$$

where $\mathbf{V}_k$ is an $n \times n$ diagonal matrix of unknown weights and $\widetilde{\mathbf{X}}_k$ is the individual configuration $\mathbf{X}_k$ optimally fitted to $\mathbf{V}_k\mathbf{Z}$. The elements in $\mathbf{V}_k$ act on the points and are called *vector weights*. Formally, (21.13) differs from (21.9) only insofar as $\mathbf{Z}$ is now weighted by a diagonal matrix from the left, not the right. The solution of (21.13) is simple in the 2D case, but to find all transformations ($\mathbf{V}_k$ and all those on $\mathbf{X}_k$) in higher-dimensional spaces simultaneously appears intractable. Hence, to minimize (21.13) we have to iterate over all planes of the space (see Lingoes & Borg, 1978). We continue with an example of the vector-weighting model, and then discuss several ways to interpret and apply vector weights.

Analyzing $\mathbf{X}_{20}$ and the $\mathbf{Z}$ configuration from Figure 21.6 with the vector-weighting model yields 15 weights, one for each point. Most of these weights

are very close to +1, except those for points A, E, and F, where the procedure finds 1.7, 1.6, and −1.4, respectively. Premultiplying $\mathbf{Z}$ by $\mathbf{V}_{20}$ has the effect shown in Figure 21.8. Point A is shifted away from the origin in the direction and sense of the vector associated with it. In other words, the vector with endpoint A is simply stretched by the factor 1.7. The analogous movement is true for point F. For point E, in contrast, the weight −1.4 not only stretches the respective vector but also flips it over, or *reverses its sense*. From Table 21.4, column $r^2(\widetilde{\mathbf{X}}_k, \mathbf{V}_k\mathbf{Z})$, we see that these movements lead to a communality increment of almost 70% relative to the unweighted Procrustean fitting. Thus, moving the points A, E, and F into different neighborhoods in the described way seems to capture an important characteristic in which person 20 differs from most others.

Evaluating the Fit in Vector Weighting

Table 21.4 shows that the vector weightings allow a much better approximation of the individual configurations than the dimension-weighting models. However, the dimension-weighting models use only 2 (the dimensional weights) or 3 (the idiosyncratic rotation angle, in addition) parameters, but the vector weightings use up to 15 parameters. Of course, the sheer number of free parameters cannot be compared directly if the models are restricted in different ways, but simulation studies (Langeheine, 1980a, 1982) show that the vector-weighting model can be expected to fit 2D random configurations considerably better than dimensional weightings. For $K = 41, m = 2$, and $n = 15$, the average fit value of 0.169 was found for the unweighted Procrustean fitting, 0.186 for the dimension-weighting model, 0.200 for the dimension-weighting model with idiosyncratic orientation, and 0.699 for the vector-weighting model. If we evaluate the observed fit values against these expectations for random configurations, the performance of the vector weighting is less impressive in this example.

As the dimensionality m goes up, the dimension-weighting model offers increasingly more fitting parameters, whereas their number remains constant in the vector-weighting model. This partly explains why, when m goes up and n remains constant, the communalities for random configurations grow substantially for unweighted Procrustean fittings and dimensional weightings but do not increase much for the vector-weighting models. Naturally, this is also a consequence of how these parameters are used in the analyses. Because there are various complicated interdependencies, it becomes difficult to say what should be expected for random configurations in general, but fortunately Langeheine (1980a) provides extensive tables.

Interpreting Vector Weights

In contrast to dimensional weighting, there is no convincing interpretation of vector weighting as a psychological model. For dimension weights,

such terms as *relative importance* or *salience* may be appropriate. No such interpretations can be given to the vector weights, except when their values are constrained to be nonnegative (see below: the perspective model). However, vector weighting may provide valuable index information. If we find that an optimal fitting of $\mathbf{Z}$ to each individual configuration can be done only with weights varying considerably around +1, then it makes little sense to consider the centroid configuration $\mathbf{Z}$ as a structure *common* to all individuals. To see this, consider the distribution of voters in France. This distribution is almost perfectly bimodal over the political left–right continuum. But it would be foolish to say that "the" French voter is politically "in the middle". In fact, no one really is. Similarly, in respect to centroid configurations, it may just be the case that this configuration does not really represent anybody. But there may be groups of individuals with very similar perceptions. Whether this is so may be seen from studying the distributions of the vector weights.

Because we can also arrive at such conclusions by directly studying the data, we return to the question of whether there is an interpretation of vector weighting as a psychological model. The answer is yes, but only under some restrictions on (21.13). One possibility would be to carry out the vector weightings with respect not to the centroid of $\mathbf{Z}$ but to a substantively meaningful origin. In a radex (see Chapter 5), for example, the centroid is extrinsic to the scientific problem under investigation, but the point chosen as the radex origin is not. If several such radexes were given, we could first translate them all such that their origins lie at these points. If we then fit each individual configuration to an average configuration derived from them all, the vector weights would express the different relative centrality of the points.

A different interpretation is given by Feger (1980) which clarifies why the vector-weighting transformations are called the *perspective model* under certain conditions. Feger asked nine subjects to rate pairs of 10 attitude objects (the six major political parties in West Germany; the trade unions; the Church; the employers' association; the subject him- or herself) with respect to the criterion "closeness". The ratings were replicated 12 times in intervals of 3 weeks. The data led to 108 2D MDS configurations. A centroid configuration was computed, and all configurations were translated such that their origins were at the points representing the object *self*. Using vector weightings on $\mathbf{Z}$, it was possible to explain most of the intra- and interindividual differences. It seemed as if the individuals perceived the attitude objects from this perspective in space, sometimes pulling some objects closer to themselves, sometimes pushing them farther away. Feger (1980) goes on to interpret the *self* point as the ideal point of the subjects (just as in the unfolding models) and the distances from the ideal point to the (possibly shifted) points of the nine other objects as indicators of the strength of preference of the respective person for these objects. In this interpretation, the point shiftings assessed by the vector weights are

due to changes of preferences over time and individuals. This complex but interesting interpretation implies a dependency between similarity judgments and preferences but becomes intractable if negative vector weights are observed.

Adding an Idiosyncratic Origin to Vector Weighting

Finally, as in the idiosyncratic rotations in the dimension-weighting model, we can generalize the vector-weighting transformations to a model with an idiosyncratic origin. In other words, rather than fixing the perspective origin externally either at the centroid or at some other more meaningful point, it is also possible to leave it to the model to find an origin that maximizes the correspondence of an individual configuration and a transformed $\mathbf{Z}$. The relevant loss function becomes rather complex:

$$\sum_{k=1}^{K} \text{tr } \mathbf{E}_k'\mathbf{E}_k, \text{ with } \mathbf{E}_k = \mathbf{V}_k(\mathbf{Z} - \mathbf{1}\mathbf{t}_k') - s_k(\mathbf{X}_k - \mathbf{1}\mathbf{u}_k')\mathbf{T}_k, \qquad (21.14)$$

where $\mathbf{t}_k$ and $\mathbf{u}_k$ are translation vectors for $\mathbf{Z}$ and $\mathbf{X}_k$, respectively, $\mathbf{T}_k'\mathbf{T}_k = \mathbf{I}$, $\mathbf{V}_k$ is diagonal, and s_k is a scalar. Differentiating (21.14) with respect to the unknowns $\mathbf{V}_k, \mathbf{t}_k, \mathbf{u}_k, s_k$, and $\mathbf{T}_k$ shows that none of these unknowns is redundant and that they are interrelated in a complicated way (Lingoes & Borg, 1978). However, minimizing (21.14) is uninteresting in itself, because what was true for idiosyncratic rotations holds here: an origin chosen by substantive considerations is always preferable to one found by blind optimization. The latter may, at best, serve an exploratory purpose as an index.

21.7 PINDIS, a Collection of Procrustean Models

The transformations of $\mathbf{Z}$ for individual k discussed so far are the dimension weights $\mathbf{W}_k$, the idiosyncratic rotations $\mathbf{S}_k$, and the vector weightings $\mathbf{V}_k$. Table 21.5 summarizes all combinations of these transformations and the resulting models. Note that the models with the idiosyncratic rotation $\mathbf{S}_k$ but without the dimension weights $\mathbf{W}_k$ are equivalent to the same model without $\mathbf{S}_k$. In this case, the idiosyncratic rotation appears both in $\mathbf{Z}$ and in $\widetilde{\mathbf{X}}_k$, so that one of them can be omitted. The vector dimension-weighting model, $\mathbf{V}_k\mathbf{Z}\mathbf{W}_k$, and the full model, $\mathbf{V}_k\mathbf{Z}\mathbf{S}_k\mathbf{W}_k$, are difficult to interpret. This explains why the other models are more popular.

Most of the Procrustean transformations discussed above are carried out by the program PINDIS[3] (Lingoes & Borg, 1977). The program needs as in-

[3] A good overview of the least-squares estimation of the PINDIS models can be found in Commandeur (1991). Moreover, his MATCHALS algorithm can handle entire rows of miss-

TABLE 21.5. Overview of transformations of $\mathbf{Z}$ in Procrustes models. For each model a '+' indicates the presence of the factor, a '–' the absence. The factors are: dimension weights $\mathbf{W}_k$, idiosyncratic rotations $\mathbf{S}_k$, vector weights $\mathbf{V}_k$. $\mathbf{E}_k$ denotes the error of the model, and tr $\mathbf{E}_k'\mathbf{E}_k$ is the loss function minimized.

$\mathbf{W}_k$	$\mathbf{S}_k$	$\mathbf{V}_k$	Model	$\mathbf{E}_k$
–	–	–	*GPA*	$\mathbf{Z} - \widetilde{\mathbf{X}}_k$
+	–	–	Dimension weighting	$\mathbf{Z}\mathbf{W}_k - \widetilde{\mathbf{X}}_k$
–	+	–	*GPA*	$\mathbf{Z}\mathbf{S}_k - \widetilde{\mathbf{X}}_k \Leftrightarrow \mathbf{Z} - \widetilde{\mathbf{X}}_k$
–	–	+	Vector weighting	$\mathbf{V}_k\mathbf{Z} - \widetilde{\mathbf{X}}_k$
+	+	–	Idiosyncratic rotations	$\mathbf{Z}\mathbf{S}_k\mathbf{W}_k - \widetilde{\mathbf{X}}_k$
+	–	+	Vector and dimension weighting	$\mathbf{V}_k\mathbf{Z}\mathbf{W}_k - \widetilde{\mathbf{X}}_k$
–	+	+	Vector weighting	$\mathbf{V}_k\mathbf{Z}\mathbf{S}_k - \widetilde{\mathbf{X}}_k \Leftrightarrow \mathbf{V}_k\mathbf{Z}_k - \widetilde{\mathbf{X}}_k$
+	+	+	Full	$\mathbf{V}_k\mathbf{Z}\mathbf{S}_k\mathbf{W}_k - \widetilde{\mathbf{X}}_k$

put K individual configurations ($\mathbf{X}_k$). From these, it computes a centroid configuration $\mathbf{Z}$ via the generalized Procrustean fitting in (21.1). Alternatively, we can input some configuration $\mathbf{Z}$ derived externally from, say, substantive considerations. $\mathbf{Z}$ can also be based on an empirical configuration that is fixed in some desirable rotation and/or translated to some meaningful origin.

The various models and transformations discussed above are summarized in Table 21.6. $\widetilde{\mathbf{X}}_k$ always denotes an $\mathbf{X}_k$ optimally rotated and translated relative to the (weighted, rotated, translated) $\mathbf{Z}$ that tries to account for it. For the Procrustean transformations involving rigid motions and dilations only, all of the parameters chosen to maximize $r^2(\widetilde{\mathbf{X}}_k, \mathbf{Z})$ are *admissible* and, thus, uninformative because they leave the distance ratios of $\mathbf{X}_k$ and $\mathbf{Z}$ invariant. *Informative* are those parameters on $\mathbf{Z}$ that change the ratios of its distances directly or in combination with other transformations that are applied at the same time. The dimension-weighting model uses, in general, different weights for each of the m dimensions of $\mathbf{Z}$, which has the effect of changing the distance ratios of $\mathbf{Z}$. Hence, these weight parameters are informative about simple ways in which $\mathbf{Z}$ relates to $\mathbf{X}_k$. In contrast to admissible fitting parameters that cannot possibly be interpreted in a substantive sense, informative parameters are potential candidates for interpretations: the dimensional weights, for example, might be viewed as *dimensional saliences*. Similarly, the rotation angles in the dimension-weighting model with idiosyncratic rotation lead to the interpretation that this subject uses dimensions differently from those chosen by the average person. Because there is one such angle for each of the $m(m-1)/2$ planes, this model has $m(m-1)/2$ additional inadmissible parameters. The perspec-

ing values in the coordinate matrices. Ten Berge et al. (1993) discuss a *GPA* algorithm in which only the missing values themselves are discarded.

TABLE 21.6. Overview of the transformations in PINDIS. $\mathbf{Z}^r$ = optimally rotated $\mathbf{Z}$ in model 2; $\mathbf{Z}_k^r$ = idiosyncratically optimal $\mathbf{Z}$ in model 3; $\mathbf{W}_k^r$ is $\mathbf{W}_k$ relative to $\mathbf{Z}_k^r$. Similarly for $\mathbf{Z}^t, \mathbf{V}^t$, and $\mathbf{Z}_k^t$, where t denotes an optimal translation.

	Model	Number of Informative Fitting Parameters	Fit Index
(1)	Similarity transformation (unit weighting)	0	$r^2(\widetilde{\mathbf{X}}_k, \mathbf{Z})$
(2)	Dimension weighting (dimensional salience)	m	$r^2(\widetilde{\mathbf{X}}_k, \mathbf{Z}^r\mathbf{W}_k)$
(3)	Dimension weighting with Idiosyncratic orientation	$m + m(m-1)/2$	$r^2(\widetilde{\mathbf{X}}_k, \mathbf{Z}^r\mathbf{W}_k^r)$
(4)	Perspective model with fixed origin (vector weighting)	n	$r^2(\widetilde{\mathbf{X}}_k, \mathbf{V}_k\mathbf{Z})$
(5)	Perspective model with idiosyncratic origin	$n + m$	$r^2(\widetilde{\mathbf{X}}_k, \mathbf{V}_k^t\mathbf{Z}_k^t)$

tive model involves n inadmissible fitting parameters, one for each point. In its generalized version with an idiosyncratic origin, there are m additional parameters corresponding to the m coordinates of the freely chosen origin of $\mathbf{Z}$.

The PINDIS transformations form two genuine hierarchies: the models denoted as 1, 2, and 3 in Table 21.6 establish one such set of nested approaches, and models 1, 4, and 5 the other. Moreover, in practical applications, n is usually much greater than m, so that $0 < m < m + m(m-1)/2 < n < n + m$ results. Thus, in terms of complexity, the models in PINDIS are linearly ordered. The order of the various communality values typically mirrors the order of complexity.

21.8 Exercises

Exercise 21.1 Consider the data matrix from Exercise 1.6 on p. 16.

(a) Compute Euclidean distances for its columns. Then scale these distances in a two-dimensional MDS space (ordinal MDS).

(b) Next, randomly eliminate 20, 30, and 40 percent of the Euclidean distances and consider these distances as missing data. With the remaining distance values run three ordinal MDS analyses in 2D.

(c) Then compare the four configurations using Procrustean methods. (Hint: Would you choose one fixed target, or would you rather do one generalized Procrustean fitting?)

(d) Discuss the similarity of the MDS solutions in terms of robustness of MDS.

Exercise 21.2 The table below contains the coordinates of four configurations, $\mathbf{X}_1$ to $\mathbf{X}_4$.

Point	$\mathbf{X}_1$		$\mathbf{X}_2$		$\mathbf{X}_3$		$\mathbf{X}_4$	
1	1	2	2	1	1	1	2.64	0.50
2	-1	2	-2	1	-1	1	-1.00	1.32
3	-1	-2	-2	-1	-1	-1	-2.64	-0.50
4	1	-2	2	-1	1	-1	1.00	-1.32

(a) Find, by geometric means, the centroid configuration $\mathbf{Z}$ of the configurations $\mathbf{X}_1$ to $\mathbf{X}_3$. (Hint: Plot the configurations in one chart, then determine $\mathbf{Z}$'s points.)

(b) Find the dimension weights that turn $\mathbf{Z}$ into $\mathbf{X}_1$, $\mathbf{X}_2$, and $\mathbf{X}_3$, respectively.

(c) Find the dimension weights that turn (a possibly rotated) $\mathbf{Z}$ into $\mathbf{X}_4$.

(d) Characterize in which way rotating $\mathbf{Z}$ and then weighting its dimensions affects the resulting configuration?

Exercise 21.3 Consider the group space in Figure 21.6 on p. 463.

(a) Interpret this configuration and its four clusters.

(b) Interpret what weighting the X- and the Y-axis of this configuration means in terms of the perceived similarity of the breakfast items.

(c) Assume that one particular person assigns a vector weight of -1 to item B, and weights of about $+1$ to all other items. What do you conclude from this?

(d) Assume that we decided to adopt an interpretation for this group space with two orthogonal dimensions whose ends are defined by the characteristics of the items in the four clusters in Figure 21.6. In that case, we may also decide to translate the group space to a more meaningful origin. Sketch in Figure 21.6 what seems the most reasonable origin to you. Discuss what implications such a shift of origin has for dimensional- and for vector-weighting models.

22
Three-Way MDS Models

In the Procrustean context, the dimension-weighting model was used in order to better match a set of K given configurations $\mathbf{X}_k$ to each other. We now ask how a solution of the dimension-weighting model can be found directly from the set of K proximity matrices without first deriving individual MDS spaces $\mathbf{X}_k$ for each individual k. We discuss how dimension weighting can be incorporated into a framework for minimizing Stress. Another popular algorithm for solving this problem, INDSCAL, is considered in some detail. Then, some algebraic properties of dimension-weighting models are investigated. Finally, matrix-conditional and -unconditional approaches are distinguished, and some general comments on dimension-weighting models are made. Table 22.1 gives an overview of the (three-way) Procrustean models discussed so far and the three-way MDS models of this chapter.

22.1 The Model: Individual Weights on Fixed Dimensions

We now return to procedures that find a solution to dimension-weighting models directly. That is, given K proximity matrices, each of order $n \times n$, a group space and its associated subject space are computed without any intermediate analyses. This situation is depicted in Figure 3.10.

TABLE 22.1. Overview of models for three-way data in Chapters 21 and 22.

Distance	$\mathbf{X}_k$	Model	Chapter/ Section
Equal dimension weights	Given	Generalized Procrustes	21.3
Equal dimension weights	Derived from $\mathbf{P}_k$	Identity model Stress	8.6
Equal dimension weights	Derived from $\mathbf{P}_k$	Classical scaling	12
Weighted Euclidean	Given	PINDIS	21.4
Weighted Euclidean	Derived from $\mathbf{P}_k$	Three-way Stress	22.1
Weighted Euclidean	Derived from $\mathbf{P}_k$	INDSCAL	22.1
Generalized Euclidean	Given	Oblique Procrustes	21.4
Generalized Euclidean	Derived from $\mathbf{P}_k$	Three-way Stress	22.2
Generalized Euclidean	Derived from $\mathbf{P}_k$	IDIOSCAL	22.2

The Weighted Euclidean Model

The problem consists of representing the dissimilarity δ_{ijk} between objects i and j as seen by individual (or replication) k by the distance d_{ijk}:

$$\begin{aligned} d_{ijk}(\mathbf{G}\mathbf{W}_k) &= \left[\sum_{a=1}^{m}(w_{aak}g_{ia} - w_{aak}g_{ja})^2\right]^{1/2} \\ &= \left[\sum_{a=1}^{m} w_{aak}^2(g_{ia} - g_{ja})^2\right]^{1/2}, \end{aligned} \tag{22.1}$$

where $i, j = 1, \ldots, n; k = 1, \ldots, K; a = 1, \ldots, m$; $\mathbf{W}_k$ is an $m \times m$ diagonal matrix of nonnegative weights w_{aak} for every dimension a for individual k; and $\mathbf{G}$ is the matrix of coordinates of the *group stimulus space* $\mathbf{G}$. Note that $\mathbf{G}$ does not have subscript k: individual differences are possible only in the weights on the dimensions of $\mathbf{G}$. The group stimulus space is also called a *common space* (Heiser, 1988b). Equation (22.1) is called the *weighted Euclidean distance*, which we encountered before in (21.7).

In terms of an individual k, the weighted Euclidean model says that

$$\mathbf{X}_k = \mathbf{G}\mathbf{W}_k, \tag{22.2}$$

where $\mathbf{X}_k$ is the individual configuration. Because distances do not change under translation, we may assume that $\mathbf{G}$ is column centered. $\mathbf{X}_k = \mathbf{G}\mathbf{W}_k$ is similar to $\mathbf{Z}\mathbf{S}\mathbf{W}_k$ in (21.11), where $\mathbf{Z}$ was defined as the average configuration of N individual configurations $\mathbf{X}_k$ transformed to an optimal fit in the sense of the generalized Procrustean loss function in (21.1). However, in this chapter there are no individual configurations $\mathbf{X}_k$ to begin with, and thus $\mathbf{G}$ must be computed differently.

There is an inherent indeterminacy in the weighted Euclidean model: the dimension weights depend on the particular definition of the group space. Let $\mathbf{D}$ be any diagonal matrix with full rank. Then

$$\mathbf{X}_k = \mathbf{G}\mathbf{W}_k = \mathbf{G}\mathbf{D}\mathbf{D}^{-1}\mathbf{W}_k = (\mathbf{G}\mathbf{D})(\mathbf{D}^{-1}\mathbf{W}_k) = \mathbf{G}^*\mathbf{W}_k^*; \tag{22.3}$$

that is, if $\mathbf{G}$ is stretched by $\mathbf{D}$, and the weights in $\mathbf{W}_k$ are subjected to the inverse transformation, the product remains the same. For the group space $\mathbf{G}$, no restriction was defined yet, except for the irrelevant centering convention. Yet, in order to make $\mathbf{G}$ identifiable, it must be normed somehow. One such norming is to require that $\mathbf{G}\mathbf{G}' = \mathbf{I}$. Although this norming is a purely formal requirement, it nevertheless affects the interpretation of the weights in each $\mathbf{W}_k$: they are conditional to $\mathbf{G}$, as 22.3 makes clear. Hence, care must be taken with claims that, for example, a person weights dimension X twice as much as dimension Y. This assertion is only true relative to the given group space $\mathbf{G}$. However, it is possible to compare the weights of different persons on each dimension in turn without restrictions.

The weighted Euclidean model can be implemented in several ways. First, we discuss a method that minimizes Stress to find a group space $\mathbf{G}$ and dimension weights $\mathbf{W}_k$ from K proximity matrices. Then, we discuss the popular INDSCAL algorithm, which finds $\mathbf{G}$ and the $\mathbf{W}_k$s from the scalar product matrices derived from the K proximity matrices.

Fitting the Dimension-Weighting Model via Stress

Dimension weights can be implemented fairly easily in the Stress framework by applying the constrained MDS theory (De Leeuw & Heiser, 1980) from Section 10.3. Let us assume that the proximities are dissimilarities. Then, the Stress that needs to be minimized equals

$$\sigma_r(\mathbf{X}_1, \ldots, \mathbf{X}_k) = \sum_{k=1}^{K} \sum_{i<j} \left(\delta_{ijk} - d_{ij}(\mathbf{X}_k)\right)^2, \tag{22.4}$$

subject to the constraints that $\mathbf{X}_k = \mathbf{G}\mathbf{W}_k$ as required for the dimension-weighting model. This minimization can also be viewed as doing MDS on a $Kn \times Kn$ dissimilarity supermatrix $\mathbf{\Delta}^*$ (with the individual K dissimilarity matrices $\mathbf{\Delta}_k$ on the diagonal blocks, and other blocks missing) and a configuration supermatrix $\mathbf{X}^*$ (with the individual configuration matrices $\mathbf{X}_k$ stacked under each other); that is,

$$\mathbf{\Delta}^* = \begin{bmatrix} \mathbf{\Delta}_1 & & & \\ & \mathbf{\Delta}_2 & & \\ & & \ddots & \\ & & & \mathbf{\Delta}_K \end{bmatrix} \text{ and } \mathbf{X}^* = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \vdots \\ \mathbf{X}_K \end{bmatrix}$$

subject to the constraints that $\mathbf{X}_k = \mathbf{G}\mathbf{W}_k$. The theory of Section 10.3 says that every iteration of the majorization algorithm for confirmatory MDS consists of the following two steps.

1. Compute the unconstrained update $\overline{\mathbf{X}}^*$ by the Guttman transform (8.28).

2. Minimize tr $(\mathbf{X}-\overline{\mathbf{X}}^*)'\mathbf{V}^*(\mathbf{X}-\overline{\mathbf{X}}^*)$ over $\mathbf{X}$ subject to the constraints to obtain the update $\mathbf{X}^u$, where here $\mathbf{V}^*$ is a block-diagonal matrix with $n\mathbf{J}$ on the diagonal blocks and where $\mathbf{J} = \mathbf{I} - n^{-1}\mathbf{1}\mathbf{1}'$ is the centering matrix.

Minimizing tr $(\mathbf{X}-\overline{\mathbf{X}}^*)'\mathbf{V}^*(\mathbf{X}-\overline{\mathbf{X}}^*)$ in the second step is equal to minimizing

$$\sum_k \text{tr } n(\mathbf{X}_k-\overline{\mathbf{X}}_k)'\mathbf{J}(\mathbf{X}-\overline{\mathbf{X}}_k) = \sum_k \text{tr } n(\mathbf{G}\mathbf{W}_k-\overline{\mathbf{X}}_k)'(\mathbf{G}\mathbf{W}_k-\overline{\mathbf{X}}_k) \tag{22.5}$$

over $\mathbf{G}$ and $\mathbf{W}_k$. The centering matrix $n\mathbf{J}$ may be removed from (22.5), because $\overline{\mathbf{X}}_k$ is already column centered. De Leeuw and Heiser (1980) give a solution that is based on dimensionwise solving (22.5). Let $\overline{\mathbf{X}}_a$ denote the $n \times K$ matrix with column a of each unconstrained update $\overline{\mathbf{X}}_k$ stacked next to each other. Let $\mathbf{g}_a$ be column a of $\mathbf{G}$ and $\mathbf{w}_a$ the $K \times 1$ vector of the dimension weight w_{aak} for individual k in dimension a. Then, minimizing (22.5) is the same as minimizing

$$\sum_a \text{tr } (\mathbf{g}_a\mathbf{w}_a'-\overline{\mathbf{X}}_a)'(\mathbf{g}_a\mathbf{w}_a'-\overline{\mathbf{X}}_a). \tag{22.6}$$

This problem can be solved for each dimension separately by an alternating least squares algorithm, where in each iteration (22.6) is minimized over $\mathbf{g}_a$, keeping $\mathbf{w}_a$ fixed, followed by the minimization over $\mathbf{w}_a$, keeping $\mathbf{g}_a$ fixed. Alternatively, the analytic minimum is obtained by computing the singular value decomposition of $\overline{\mathbf{X}}_a = \mathbf{P}\mathbf{\Phi}\mathbf{Q}'$ and setting $\mathbf{g}_a = \mathbf{p}_1$ and $\mathbf{w}_a = \phi_1\mathbf{q}_1$. The PROXSCAL program implements the dimension-weighting model for Stress with more options (such as fixing coordinates and allowing for missing proximities). For the detailed mathematics of that approach, we refer to Heiser (1988b) and Commandeur and Heiser (1993). A different algorithm for dimension weighting with constrained dimensions is given by Winsberg and De Soete (1997).

If all weights $\mathbf{w}_a$ are constrained to be equal, we get the *identity* model for three-way proximities (Commandeur & Heiser, 1993). Then, the only thing that needs to be estimated is the group stimulus space $\mathbf{G}$. This allows (22.4) to be written as

$$\begin{aligned}\sigma_r(\mathbf{G}) &= \sum_k^K \sum_{i<j} (\delta_{ijk} - d_{ij}(\mathbf{G}))^2 \\ &= K\sum_{i<j}\left(\overline{\delta}_{ij} - d_{ij}(\mathbf{G})\right)^2 + \sum_{i<j}\sum_k^K \left(\overline{\delta}_{ij} - \delta_{ijk}\right)^2,\end{aligned}$$

where $\bar{\delta}_{ij} = K^{-1} \sum_k^K \delta_{ijk}$. The first term of $\sigma_r(\mathbf{G})$ amounts to simple MDS of the average dissimilarity matrix, and the second term measures the difference of the individual dissimilarity matrices to their average.

Heiser (1989b) discusses the minimization of the weighted Euclidean model for Stress with city-block distances. The minimization can be done by a combinatorial approach (similar to combinatorial methods used for unidimensional scaling) combined with a majorizing approach that accommodates negative disparities, or by majorization of city-block distances (Groenen et al., 1995).

The INDSCAL *Algorithm*

A popular algorithm for solving the dimension-weighting model is based on the scalar-product matrix, similar to classical scaling. Let $\mathbf{B}_{\mathbf{\Delta}_k} = -\frac{1}{2}\mathbf{J}\mathbf{\Delta}_k^{(2)}\mathbf{J}$ be the $n \times n$ scalar-product matrix for individual k derived from the distances via (12.2). Classical scaling for individual k minimizes

$$\frac{1}{4}||\mathbf{J}[\Delta_k^{(2)} - \mathbf{D}^{(2)}(\mathbf{X})]\mathbf{J}||^2 = \frac{1}{4}4||\mathbf{B}_{\mathbf{\Delta}_k} - \mathbf{X}\mathbf{X}'||^2.$$

This is extended by including dimension weights in the INDSCAL loss function; that is,

$$L_{IND}(\mathbf{G}, \mathbf{W}_1, \ldots, \mathbf{W}_K) = \sum_k^K ||\mathbf{B}_{\mathbf{\Delta}_k} - \mathbf{G}\mathbf{W}_k^2\mathbf{G}'||^2 \tag{22.7}$$

$$= \sum_{k=1}^K \sum_{i,j} \left(b_{ijk} - \sum_{a=1}^m g_{ia} g_{ja} w_{aak}^2 \right)^2. \tag{22.8}$$

It is assumed that the scalar-product matrices $\mathbf{B}_{\mathbf{\Delta}_k}, k = 1, \ldots, K$, are given. In the case of interval-scale proximities, an additive constant that leads to Euclidean distances must be computed, and scalar products are then derived from these distances. If only ordinal proximities (possibly even with missing data values) are given as data, one often proceeds as in PINDIS, that is, by first computing the individual configurations $\mathbf{X}_k, k = 1, \ldots, K$, via ordinal MDS, and then from these deriving the needed scalar products (e.g., Krantz & Tversky, 1975). We now describe a solution for (22.7).

The INDSCAL procedure (Carroll & Chang, 1970) proceeds as follows. The INDSCAL loss function L_{IND} has to be solved over two sets of parameters, $\mathbf{G}$ and the $\mathbf{W}_k$s. Unfortunately, this loss function does not have an analytical solution, except in the error-free case (Schönemann, 1972). INDSCAL uses the alternating update strategy in which an update of $\mathbf{G}$ for fixed $\mathbf{W}_k$s is followed by an update of the $\mathbf{W}_k$s for fixed $\mathbf{G}$. These updates are iterated until convergence. The two steps are computed as follows.

1. The update for the $\mathbf{W}_k$s for fixed $\mathbf{G}$ is found by standard regression. However, L_{IND} has to be rewritten. First, we string out each $\mathbf{B}_{\mathbf{\Delta}_k}$

into one column vector with n^2 elements and then form an $n^2 \times K$ matrix $\mathbf{B}^*$ by stacking these column vectors next to each other. In a similar fashion, we then stack the diagonals of the K weight matrices $\mathbf{W}_k^2$ in an $m \times K$ matrix $\mathbf{W}$. Finally, we compute the products $\mathbf{g}_a \mathbf{g}_a'$, string out its elements into one column vector, and place them for each dimension $a = 1, \ldots, m$ next to each other in the $n^2 \times m$ matrix $\mathbf{V}$. This leads to a compact way of writing L_{IND} as

$$L_{IND} = \text{tr } (\mathbf{B}^* - \mathbf{VW})'(\mathbf{B}^* - \mathbf{VW}). \tag{22.9}$$

The update for $\mathbf{W}$ is found by differentiating (22.9) with respect to $\mathbf{W}$ and setting the result equal to the null matrix $\mathbf{0}$, which yields

$$\mathbf{W} = (\mathbf{V}'\mathbf{V})^{-1}\mathbf{V}'\mathbf{B}^*. \tag{22.10}$$

The columns of $\mathbf{W}$ are the diagonals of the individual weight matrices $\mathbf{W}_k^2$. Note, however, that some elements of $\mathbf{W}$ may be negative, so that the corresponding dimension weight is not a real number. This problem of negative squared dimension weights in the INDSCAL algorithm could be avoided by minimizing (22.9) over $\mathbf{W}$ under the constraint that $\mathbf{W} \geq \mathbf{0}$, as suggested by Ten Berge, Kiers, and Krijnen (1993), who used nonnegative least-squares (Lawson & Hanson, 1974). De Soete, Carroll, and Chaturvedi (1993) imposed these constraints using the alternating least-squares method discussed in Section 9.6.

2. A better $\mathbf{G}$, relative to the given $\mathbf{W}_k^2$s, is computed by INDSCAL as follows. With fixed $\mathbf{W}_k$s, we minimize

$$\begin{aligned} L_{IND}(\mathbf{G}, \mathbf{H}) &= \sum_k^K ||\mathbf{B}_{\mathbf{\Delta}_k} - \mathbf{H}\mathbf{W}_k^2\mathbf{G}'||^2 \\ &= \sum_k^K \text{tr } \mathbf{B}_{\mathbf{\Delta}_k}^2 + \text{tr } \mathbf{G}\left[\sum_k \mathbf{W}_k^2\mathbf{H}'\mathbf{H}\mathbf{W}_k^2\right]\mathbf{G})' \\ &\quad -2\text{tr } \mathbf{G}\left[\sum_k \mathbf{W}_k^2\mathbf{H}'\mathbf{B}_{\mathbf{\Delta}_k}\right] \end{aligned} \tag{22.11}$$

over both $\mathbf{G}$ and $\mathbf{H}$, the so-called CANDECOMP algorithm (Carroll & Chang, 1970). After convergence, it turns out that $\mathbf{G}$ and $\mathbf{H}$ are equal or can be made equal. Differentiating (22.11) with respect to $\mathbf{G}$ and setting the result equal to $\mathbf{0}$ gives the update for $\mathbf{G}$; that is,

$$\mathbf{G} = \left(\sum_k \mathbf{B}_{\mathbf{\Delta}_k}\mathbf{H}\mathbf{W}_k^2\right)\left(\sum_k \mathbf{W}_k^2\mathbf{H}'\mathbf{H}\mathbf{W}_k^2\right)^{-1}.$$

$\mathbf{H}$ is updated with the same update formula by reversing the roles of $\mathbf{G}$ and $\mathbf{H}$.

These two steps are repeated until the process converges to a final solution for $\mathbf{W}$ and $\mathbf{G}$, which "almost always" is the global optimum, according to Carroll and Wish (1974a, p. 90) and Ten Berge and Kiers (1991).

22.2 The Generalized Euclidean Model

The weighted Euclidean distance can be extended by the *generalized Euclidean distance*, where the individual space is defined as $\mathbf{X}_k = \mathbf{G}\mathbf{T}_k$, with $\mathbf{T}_k$ an $m \times m$ (real-valued) matrix that need not be diagonal.

Interpreting the Generalized Euclidean Model

The generalized Euclidean model can be interpreted as follows. Consider the singular value decomposition of $\mathbf{T}_k$, $\mathbf{T}_k = \mathbf{P}\mathbf{\Lambda}\mathbf{Q}'$. Then, the transformation $\mathbf{G}\mathbf{T}_k = \mathbf{G}\mathbf{P}\mathbf{\Phi}\mathbf{Q}'$ can be interpreted as: take group space $\mathbf{G}$, rotate it by $\mathbf{P}$, and stretch it along its dimensions by $\mathbf{\Phi}$. Because we are concerned with the distances of $\mathbf{G}\mathbf{T}_k$, the final rotation by $\mathbf{Q}'$ is irrelevant. This shows that in the generalized Euclidean model every individual k transforms the group space first by a rotation and/or a reflection, and then by stretching. In contrast to the weighted Euclidean model, each individual may weight a different set of dimensions of the group space. Therefore, this model is somewhat less restrictive than the dimension-weighting model.

Other interpretations are possible. For example, Tucker (1972) and Harshman (1972) proposed decomposing $\mathbf{T}_k = \mathbf{D}_k\mathbf{M}_k$, where the diagonal matrix $\mathbf{D}_k$ contains the standard deviation of the column elements of $\mathbf{T}_k$ [so that $\text{diag}(\mathbf{D}_k^2) = \text{diag}(\mathbf{T}_k'\mathbf{T}_k)$]. Thus, $\mathbf{M}_k'\mathbf{M}_k$ has diagonal elements 1 and can be seen as a correlation matrix or as a matrix of cosines of angles among oblique dimensions. The interpretation of the generalized Euclidean model using this decomposition is that the individual space $\mathbf{X}_k$ can be obtained from the group space $\mathbf{G}$ by first stretching its dimensions by $\mathbf{D}_k$ and then applying an oblique rotation by $\mathbf{M}_k$. Harshman and Lundy (1984) proposed a model with only one $\mathbf{M}$ that is common to all individuals. However, this model is not equivalent to the generalized Euclidean model.

Whether or not there are applications for the generalized Euclidean model, there is nothing that rules it out formally. Indeed, even more exotic interpretations derived from other decompositions (Carroll & Wish, 1974a, 1974b) are possible.

If generalized Euclidean models are interpreted as a distance model in $\mathbf{G}$,

$$\begin{aligned} d_{ijk}^2(\mathbf{G}) &= (\mathbf{g}_i\mathbf{T}_k - \mathbf{g}_j\mathbf{T}_k)'(\mathbf{g}_i\mathbf{T}_k - \mathbf{g}_j\mathbf{T}_k) \\ &= (\mathbf{g}_i - \mathbf{g}_j)'\mathbf{C}_k(\mathbf{g}_i - \mathbf{g}_j), \end{aligned}$$

then $\mathbf{C}_k$ must be positive definite, not just positive semidefinite, as Carroll and Wish (1974b) declare, because otherwise one may obtain $d_{ijk}(\mathbf{G}) = 0$ even though $i \neq j$. That is, if we want to interpret the model in such a way that each individual k picks his or her own particular distance function from the family of weighted Euclidean distances or, as mathematicians sometimes call it, from the family of *elliptical distances* (Pease, 1965, p. 219) on the group space $\mathbf{G}$, then all dimension weights must be positive. If some of these weights are zero, then this interpretation has to be changed slightly to one in which individual k first reduces $\mathbf{G}$ to a subspace and then computes distances in this, possibly further transformed, subspace of $\mathbf{G}$. The first model has been called a *subjective metrics model* (Schönemann & Borg, 1981a), and the latter, due to Schulz (1972, 1975, 1980), may be called a *subjective transformations model*. From a practical point of view, however, these distinctions are irrelevant because, in the subjective metrics model, $\mathbf{G}$ may be almost reduced to a lower rank by choosing extremely small weights for some of its dimensions.

Fitting the Generalized Euclidean Model via Stress

The method for minimizing Stress with the generalized Euclidean model is the same as for the weighted Euclidean model via Stress, except that $\mathbf{X}_k$ is restricted as $\mathbf{X}_k = \mathbf{G}\mathbf{T}_k$, where $\mathbf{T}_k$ may be *any* real-valued $m \times m$ matrix. In the second step of the algorithm, the restrictions are imposed by minimizing

$$\sum_k \text{tr } (\mathbf{G}\mathbf{T}_k - \overline{\mathbf{X}}_k)'(\mathbf{G}\mathbf{T}_k - \overline{\mathbf{X}}_k) \tag{22.12}$$

over $\mathbf{G}$ and the $\mathbf{T}_k$s. Let the $n \times mK$ matrix $\overline{\mathbf{X}}$ contain the $\overline{\mathbf{X}}_k$s stacked next to each other, and the $m \times mK$ matrix $\mathbf{T}$ the $\mathbf{T}_k$s stacked next to each other. Then, (22.12) is equal to

$$\text{tr } (\mathbf{G}\mathbf{T} - \overline{\mathbf{X}})'(\mathbf{G}\mathbf{T} - \overline{\mathbf{X}}),$$

which is solved analytically (De Leeuw & Heiser, 1980) by taking the singular value decomposition of $\overline{\mathbf{X}} = \mathbf{P}\mathbf{\Phi}\mathbf{Q}'$ and setting $\mathbf{G} = \mathbf{P}_m$ and $\mathbf{T} = \mathbf{\Phi}_m\mathbf{Q}'_m$, where the subscript m implies taking only the first m singular values and vectors.

The IDIOSCAL *Model*

The generalized Euclidean model gained its popularity in the framework of scalar products. This idiosyncratic weighting model is also called the IDIOSCAL *model* (Carroll & Wish, 1974a, 1974b; Schulz, 1980). In scalar

product notation, this model minimizes

$$\begin{aligned} L_{IDIO}(\mathbf{G}, \mathbf{T}_1, \mathbf{T}_2, \ldots, \mathbf{T}_K) &= \sum_{k=1}^{K} ||\mathbf{B}_k - (\mathbf{G}\mathbf{T}_k)(\mathbf{G}\mathbf{T}_k)'||^2 \\ &= \sum_{k=1}^{K} ||\mathbf{B}_k - \mathbf{G}\mathbf{T}_k\mathbf{T}_k'\mathbf{G}'||^2 \\ &= \sum_{k=1}^{K} ||\mathbf{B}_k - \mathbf{G}\mathbf{C}_k\mathbf{G}'||^2, \end{aligned} \tag{22.13}$$

where $\mathbf{T}_k\mathbf{T}_k' = \mathbf{C}_k$ and $\mathbf{C}_k$ is positive semidefinite.

Apart from the many possibilities for factoring $\mathbf{C}_k$, it is of interest to ask whether there is only one $\mathbf{C}_k$ and one $\mathbf{G}$ that solve (22.13). This is not so, because

$$\begin{aligned} \mathbf{B}_k &= \mathbf{G}\mathbf{C}_k\mathbf{G}' \\ &= \mathbf{G}(\mathbf{A}\mathbf{A}^{-1})\mathbf{C}_k(\mathbf{A}\mathbf{A}^{-1})'\mathbf{G}' \\ &= (\mathbf{G}\mathbf{A})[\mathbf{A}^{-1}\mathbf{C}_k(\mathbf{A}')^{-1}](\mathbf{G}\mathbf{A})' \\ &= \mathbf{G}^*\mathbf{C}_k^*(\mathbf{G}^*)', \end{aligned} \tag{22.14}$$

where $\mathbf{A}$ is an arbitrary $m \times m$ matrix with full rank. In comparison to (22.3), the more general IDIOSCAL model is less unique. This has the practical implication that if this model is applied to a set of data matrices, many quite different group spaces $\mathbf{G}$ can be derived, and it is impossible to say which one is the *true* common structure. Schönemann (1972) proposed imposing the restriction that the $\mathbf{C}_k$s average to $\mathbf{I}$; that is,

$$\frac{1}{K}\sum_{k=1}^{K}\mathbf{C}_k = \mathbf{I}. \tag{22.15}$$

Given a set of $k = 1, \ldots, K$ arbitrary $\mathbf{C}_k^*$ as in (22.14), property (22.15) can be imposed by choosing a transformation matrix $\mathbf{A}$ such that

$$\begin{aligned} \mathbf{I} &= K^{-1}[\mathbf{A}^{-1}\mathbf{C}_1^*(\mathbf{A}')^{-1} + \ldots + \mathbf{A}^{-1}\mathbf{C}_K^*(\mathbf{A}')^{-1}] \\ &= K^{-1}\mathbf{A}^{-1}(\mathbf{C}_1^* + \ldots + \mathbf{C}_K^*)(\mathbf{A}')^{-1}, \end{aligned}$$

whence $K\mathbf{A}\mathbf{A}' = \mathbf{C}_1^* + \ldots + \mathbf{C}_K^*$. Because each $\mathbf{C}_k^*$ is symmetric by (22.13), $\mathbf{C}_1^* + \ldots + \mathbf{C}_K^*$ is also symmetric, so $\mathbf{A}$ is found by factoring the average of all $\mathbf{C}_k$s into $\mathbf{A}\mathbf{A}'$. If (22.15) holds, then

$$\frac{1}{K}\sum_{k=1}^{K}\mathbf{B}_k = \frac{1}{K}\sum\mathbf{G}\mathbf{C}_k\mathbf{G} = \frac{1}{K}\mathbf{G}(\mathbf{C}_1 + \ldots + \mathbf{C}_K)\mathbf{G}' = \mathbf{G}\mathbf{G}'. \tag{22.16}$$

For error-free data, this equation can be solved immediately (by classical scaling) to yield the group space $\mathbf{G}$, or, more properly, one possible $\mathbf{G}$

because each such $\mathbf{G}$ can be arbitrarily rotated and/or reflected and would still satisfy (22.16).

To find each individual $\mathbf{C}_k$ is also simple. We just solve the following equation for $\mathbf{C}_k$,

$$\begin{aligned} \mathbf{B}_k &= \mathbf{G}\mathbf{C}_k\mathbf{G}', && (22.17)\\ \mathbf{G}'\mathbf{B}_k\mathbf{G} &= \mathbf{G}'\mathbf{G}\mathbf{C}_k\mathbf{G}'\mathbf{G}, && (22.18)\\ (\mathbf{G}'\mathbf{G})^{-1}\mathbf{G}'\mathbf{B}_k\mathbf{G}(\mathbf{G}'\mathbf{G})^{-1} &= \mathbf{C}_k. && (22.19) \end{aligned}$$

Note that the pre- and postmultiplications in (22.18) serve the purpose of generating the matrix $\mathbf{G}'\mathbf{G}$, which, assuming that $\text{rank}(\mathbf{G}) = m$, is invertible, whereas $\mathbf{G}$ generally is not. Thus, for error-free $\mathbf{B}_k$s, the IDIOSCAL loss function (22.13) can be solved analytically (Schönemann, 1972).

Chaturvedi and Carroll (1994) imposed the additional restriction on $\mathbf{G}$ that every row only contains a single 1 and the rest 0, which makes $\mathbf{G}$ an indicator matrix. Thus, $\mathbf{G}$ classifies each stimulus i to one of M clusters. This model, called INDCLUS, falls somewhere between clustering and MDS.

22.3 Overview of Three-Way Models in MDS

To develop some geometric feeling for the various three-way models discussed in this chapter, let us demonstrate with the help of a simple example how they relate a common space to the individual space of each subject.[1]. An overview of these models is given in Figure 22.1 The *identity* model is trivial boundary case: every subject space should be equal to the common space, that is, $\mathbf{X}_k = \mathbf{G}$. This model is equivalent to computing the average dissimilarity and doing an ordinary MDS (see Section 22.1). Note that the weight plot shows dimension weights of one for all subjects on all dimensions.

However, we also know that only the relative distances between the points in a configuration are of importance, not the absolute distances. Therefore, instead of the identity model, it is better to fit the *dilation* model that allows for a dilation factor for each subject; that is, $\mathbf{X}_k = w_k\mathbf{G}$. This model is shown in the second row of Figure 22.1. Inserting the dilation factors ensures that the size of the individual configuration reflects the fit (see Section 11.1). As the weights do not differ per dimension, the points for individuals in the weights plot are on a line.

The third row in Figure 22.1 shows the weighted Euclidean model that allows each subject to weight the fixed dimensions of the common space, that is, $\mathbf{X}_k = \mathbf{G}\mathbf{W}_k$. In this example, the weights are $w_{111} = 1.5$ and $w_{221} =$

[1]Note that we are not discussing models with idiosyncratic origins or with vector weightings, as discussed in Chapter 21. Rather, the models considered here are all within the dimension-weighting family for group spaces centered at the origin.

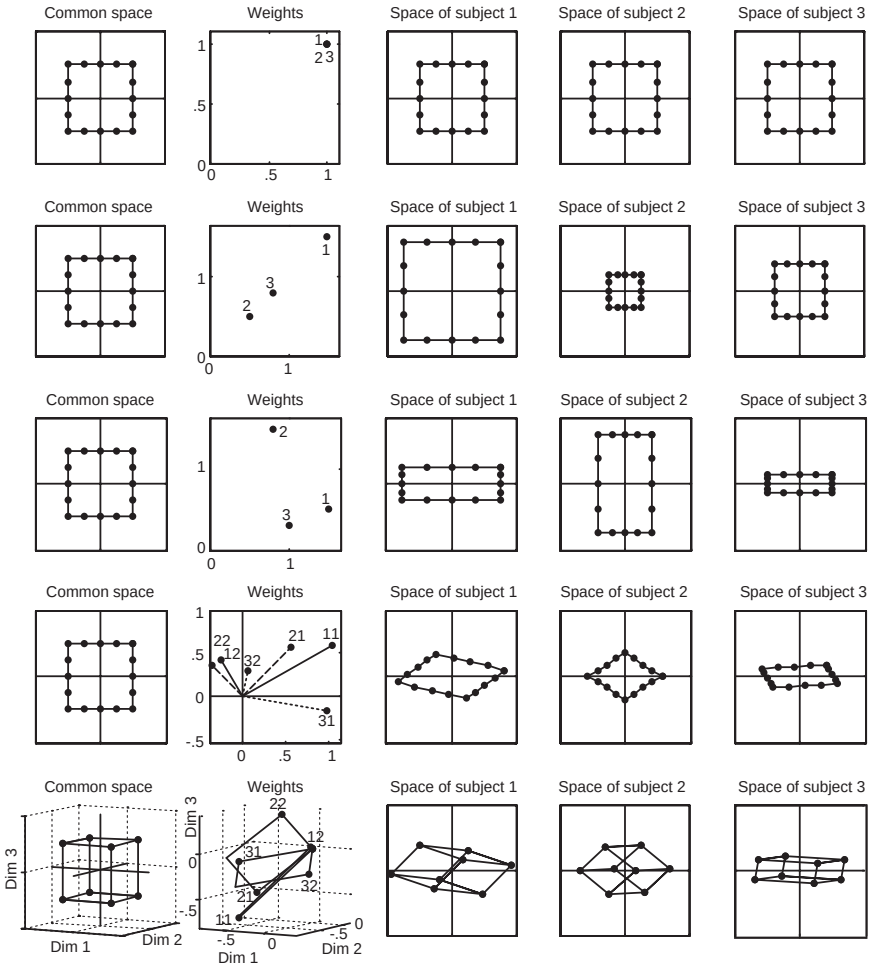


FIGURE 22.1. Overview of five three-way models for MDS. For each model the common space, the weights, and three individual spaces are given. The first row considers the identity model, the second row the dilation model, the third row the weighted Euclidean distance model, the fourth row the generalized Euclidean distance model, and the fifth row the reduced rank model.

.5 for Subject 1, $w_{112} = .8$ and $w_{222} = 1.5$ for Subject 2, and $w_{113} = 1$ and $w_{223} = .3$ for Subject 3. The weights plot shows each subject as a point with its dimension weights as coordinates. The weighted Euclidean model generalizes the dilation model by allowing for each subject to have unequal weights per dimension. In this example, we see that the Subjects 1 and 3 emphasize the first dimension in their individual spaces and Subject 2 the second dimension.

The generalized Euclidean model is given in the fourth row of Figure 22.1. In this model, the common space is first rotated to each individual's *principal directions*, as Young (1984) calls this orientation, and subsequently weighted to obtain the individual space, that is, $\mathbf{X}_k = \mathbf{G}\mathbf{T}_k\mathbf{W}_k$, where $\mathbf{T}_k$ is a rotation matrix and $\mathbf{W}_k$ is again a diagonal matrix with weights.

In our example, we choose

$$\mathbf{T}_1 = \begin{bmatrix} .866 & -.500 \\ .500 & .866 \end{bmatrix}, \quad \mathbf{W}_1 = \begin{bmatrix} 1.2 & .0 \\ .0 & .5 \end{bmatrix},$$

$$\mathbf{T}_2 = \begin{bmatrix} .707 & -.707 \\ .707 & .707 \end{bmatrix}, \quad \mathbf{W}_2 = \begin{bmatrix} .8 & .0 \\ .0 & .5 \end{bmatrix},$$

$$\mathbf{T}_3 = \begin{bmatrix} .985 & .174 \\ -.174 & .985 \end{bmatrix}, \quad \mathbf{W}_3 = \begin{bmatrix} 1.0 & .0 \\ .0 & .3 \end{bmatrix},$$

where the rotation matrices $\mathbf{T}_1, \mathbf{T}_2$, and $\mathbf{T}_3$ correspond to rotation by $30°, 45°$, and $-10°$. The weight plot is different than before. It shows how to obtain the subject space from the common space. For example, the solid vectors 11 and 12 show that the space of subject 1 is obtained by rotating the common space by 30° and to obtain dimension 1, stretch in the direction of vector 11 by a factor 1.2 (i.e., the length of vector 11) and for dimension 2, shrink in the direction of vector 12 by a factor 0.5 (the length of vector 12). Thus, the weight vectors belonging to each subject always have an angle of 90°. These vectors are obtained by $\mathbf{T}_k'\mathbf{W}_k$.

The last row of Figure 22.1 displays the *reduced rank* model. In this case, the individual spaces are allowed to have a lower rank than the common space, that is, $\mathbf{X}_k = \mathbf{G}\mathbf{T}_k\mathbf{W}_k$, where $\mathbf{G}$ is $n \times m$, $\mathbf{T}_k$ is an $m \times q$ rotation projection matrix with $q < m$, and $\mathbf{W}_k$ a diagonal $q \times q$ matrix with dimension weights. Thus, the dimensionality m of the common space is reduced to q for each subject space by the $\mathbf{T}_k$s. The example shows the common space of a cube in 3D and the individual spaces in 2D (thus $m = 3$ and $q = 2$). Then, the weights plot is interpreted in the same way as for the generalized Euclidean model. For example, the vectors 21 and 22 for subject 2 are connected to form a rectangle so that it is easy to see which 2D plane is associated with the space of Subject 2. Again, the length of each vector indicates the weighting factor for stretching or shrinking along its direction to obtain the dimension for this subject. Unless the common space and the

subject spaces are very structured, it may be hard to interpret the reduced rank model in empirical applications.

22.4 Some Algebra of Dimension-Weighting Models

The fit measure provided by INDSCAL is the correlation between the original scalar products, given in the K matrices $\mathbf{B}_k$, and the reproduced scalar products, computed by $\widehat{\mathbf{B}}_k = \mathbf{G}\mathbf{W}_k^2\mathbf{G}'$. These correlations are usually extremely high, even if the model is not adequate. This was shown by MacCallum (1976). He generated synthetic data from a group space that was stretched not only differentially for each $k = 1, \ldots, K$, but also, in violation of the INDSCAL model, along different directions for each k. He observed fit coefficients that were not lower than $r = .97$ and commented that "one must wonder whether this index provides in any sense a measure of the appropriateness of the INDSCAL model to a given set of data" (pp. 181–182).

Finer fit indices can, however, be derived from an algebraic analysis of the model. Such an analysis starts out by assuming the ideal case, where data are given that perfectly satisfy the model. Of course, this is unrealistic, because data always have error components. Hence, one can never expect to satisfy a deterministic model strictly, except in trivial cases. Nevertheless, by studying the ideal case, one can derive certain properties that data must possess if they are to be accounted for by a particular model. Real data should then also satisfy these conditions "more or less". They may also violate the model conditions systematically, and this provides potentially very informative insights into the structure of the data.

The Common-Space Condition

What does the dimension-weighting model imply for the data? That is, what properties must the data possess so that they can be explained by such models? For the subjective metrics interpretation, it is necessary that $\text{rank}(\mathbf{B}_k) = m$, for all k, because $\text{rank}(\mathbf{C}_k) = m$ and $\text{rank}(\mathbf{G}) = m$. Moreover, because $\mathbf{C}_k = \mathbf{T}_k\mathbf{T}_k'$, $\text{rank}(\mathbf{T}_k) = m$. Even if $\text{rank}(\mathbf{C}_k) < m$ (as may be the case in the subjective transformations model), it must hold that each individual space, $\mathbf{X}_k = \mathbf{G}\mathbf{T}_k$, lies in the column space of $\mathbf{G}$; that is, the columns of $\mathbf{X}_k$ must be linear combinations of the columns of $\mathbf{G}$. For example, column 1 of $\mathbf{X}_k$ should result from adding the columns of $\mathbf{G}$ with the weights of column 1 of $\mathbf{T}_k$.

In practical applications of the model, we typically would not use the $\mathbf{G}$ resulting from (22.16) as the group space but only its first few dimensions. But, with only a subspace of the complete $\mathbf{G}$, each $\mathbf{B}_k$ can only be approximated by the model, because the model requires $\text{rank}(\mathbf{B}_k) = \text{rank}(\mathbf{G}) = m$. However, if the dropped dimensions represent just error,

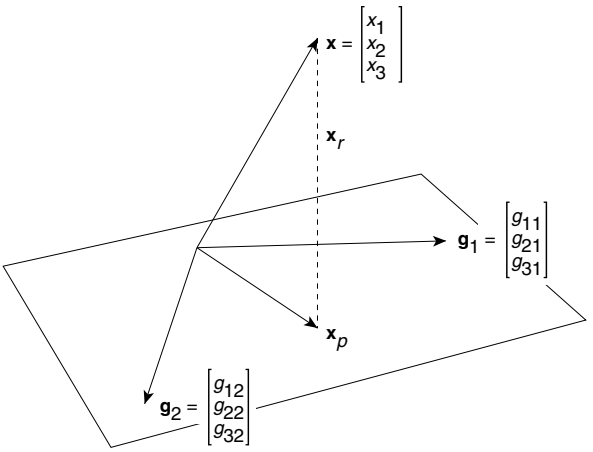


FIGURE 22.2. Geometric view of projecting a vector $\mathbf{x}$ onto the column space of a matrix with column vectors $\mathbf{g}_1$ and $\mathbf{g}_2$.

then the low-dimensional $\mathbf{G}$ should account for most of the variance of each $\mathbf{B}_k$ or, at least, for more variance than could be expected by chance.

This is easier to understand geometrically. In Figure 22.2, the column vectors $\mathbf{g}_1$ and $\mathbf{g}_2$ span a plane onto which the vector $\mathbf{x}$ is projected. The vectors $\mathbf{g}_1$ and $\mathbf{g}_2$ together form the 3×2 matrix $\mathbf{G}$. The projection of $\mathbf{x}$ onto the $\mathbf{G}$-plane, $\mathbf{x}_p$, is equal to some linear combination $w_1 \cdot \mathbf{g}_1 + w_2 \cdot \mathbf{g}_2$ or $\mathbf{x}_p = \mathbf{G}\mathbf{w}$, where $\mathbf{w}' = (w_1, w_2)$ is the weight or coordinate vector of $\mathbf{x}_p$. The residual vector (i.e., the component of $\mathbf{x}$ not contained in $\mathbf{G}$) is $\mathbf{x}_r = \mathbf{x} - \mathbf{x}_p = \mathbf{x} - \mathbf{G}\mathbf{w}$. As Figure 22.2 shows, $\mathbf{x}_r$ is orthogonal to $\mathbf{x}_p$. Thus, $\mathbf{x}_p'\mathbf{x}_r = 0$ or $(\mathbf{G}\mathbf{w})'(\mathbf{x} - \mathbf{G}\mathbf{w}) = 0$ or $\mathbf{w}'(\mathbf{G}'\mathbf{x} - \mathbf{G}'\mathbf{G}\mathbf{w}) = 0$. Because $\mathbf{w} \neq \mathbf{0}$, in general, we have $\mathbf{G}'\mathbf{x} - \mathbf{G}'\mathbf{G}\mathbf{w} = \mathbf{0}$ and $\mathbf{w} = (\mathbf{G}'\mathbf{G})^{-1}\mathbf{G}'\mathbf{x}$. With this weight vector, we obtain $\mathbf{x}_p = \mathbf{G}\mathbf{w} = \mathbf{G}[(\mathbf{G}'\mathbf{G})^{-1}\mathbf{G}'\mathbf{x}] = [\mathbf{G}(\mathbf{G}'\mathbf{G})^{-1}\mathbf{G}']\mathbf{x} = \mathbf{P}_G\mathbf{x}$, where $\mathbf{P}_G$ denotes the matrix that effects the projection of $\mathbf{x}$ onto the column space of $\mathbf{G}$.

Now, the common-space index is constructed as follows. The portion of $\mathbf{B}_k$ that can be reproduced from $\mathbf{P}_G\mathbf{X}_k$ is $\widehat{\mathbf{B}}_k = (\mathbf{P}_G\mathbf{X}_k)(\mathbf{P}_G\mathbf{X}_k)' = \mathbf{P}_G\mathbf{B}_k\mathbf{P}_G'$. The sum-of-squares of its elements can be expressed as tr $(\mathbf{P}_G\mathbf{B}_k\mathbf{P}_G)^2$, and the sum-of-squares of $\mathbf{B}_k$s elements is tr $\mathbf{B}_k^2$. The ratio of these two sums-of-squares is a possible measure for how well the common-space condition is satisfied empirically:

$$v_k = \frac{\text{tr } (\mathbf{P}_G\mathbf{B}_k\mathbf{P}_G)^2}{\text{tr } \mathbf{B}_k^2}, \tag{22.20}$$

the common-space index for individual k (Schönemann, James, & Carter, 1979). We would require, of course, that this index be close to 1 or "high" before any of the models in the IDIOSCAL family could be considered seriously as an explanation for an individual's data.

The Diagonality Condition

The common-space condition is a rather weak criterion the data must satisfy so that they can be represented by a model of the IDIOSCAL type. This weakness is simply a consequence of the generality of the models, which, without many more additional constraints on $\mathbf{G}$ and/or $\mathbf{C}_k$, are not likely to lead to much scientific insight. Thus, we now go on to the more restrictive dimension-weighting model $\mathbf{B}_k = \mathbf{G}\mathbf{W}_k^2\mathbf{G}'$ and investigate what further properties must hold in the $\mathbf{B}_k$s for such a representation to be possible.

We first impose a condition similar to the one in (22.15),

$$\frac{1}{K}\sum_{k=1}^{K}\mathbf{W}_k = \mathbf{I}, \tag{22.21}$$

which leads to

$$\frac{1}{K}\sum_{k}\mathbf{B}_k = \mathbf{G}\mathbf{G}' \tag{22.22}$$

and thus to a direct solution[2] for $\mathbf{G}$. To compute $\mathbf{W}_k$ is somewhat more demanding than to find $\mathbf{C}_k$ in (22.19) because $\mathbf{W}_k$ must be diagonal. Thus, we first find $\mathbf{C}_k$ and then try to "diagonalize" it. This is done as follows. We note again that $\mathbf{G}$ is determined only up to an orthogonal matrix $\mathbf{S}$, because $\mathbf{G}^*(\mathbf{G}^*)' = (\mathbf{G}\mathbf{S})(\mathbf{G}\mathbf{S})' = \mathbf{G}\mathbf{S}\mathbf{S}'\mathbf{G}' = \mathbf{G}\mathbf{G}'$. Hence, we want to find that $\mathbf{S}$ which diagonalizes $\mathbf{C}_k$; that is,

$$\mathbf{B}_k = (\mathbf{G}\mathbf{S})\mathbf{C}_k(\mathbf{G}\mathbf{S})' = \mathbf{G}\mathbf{S}\mathbf{C}_k\mathbf{S}'\mathbf{G}' = \mathbf{G}(\mathbf{S}\mathbf{C}_k\mathbf{S}')\mathbf{G}', \tag{22.23}$$

so that

$$\mathbf{S}\mathbf{C}_k\mathbf{S}' = \mathbf{W}_k^2. \tag{22.24}$$

If we write

$$\mathbf{C}_k = \mathbf{S}'\mathbf{W}_k^2\mathbf{S}, \tag{22.25}$$

we see that $\mathbf{S}$ and $\mathbf{W}_k^2$ are the eigenvector and eigenvalue matrices of $\mathbf{C}_k$. Because $\mathbf{C}_k$ is symmetric and positive definite, $\mathbf{S}$ is orthogonal or can be so constructed, and $\mathbf{W}_k^2$ is positive definite. Note, however, that $\mathbf{S}$ does not have a subscript, and thus (22.25) cannot be guaranteed to hold for every set of $\mathbf{B}_k$s. Rather, these $\mathbf{B}_k$s must have a common set of eigenvectors. Otherwise, the data cannot be explained by the dimension-weighting model. Geometrically, the reason for this condition is apparent and simply expresses that the model requires one fixed dimension system for all individuals.

---

[2]This $\mathbf{G}$ is taken as a rational starting configuration in the dimension-weighting option of ALSCAL (Takane et al., 1977). For ALSCAL, see Appendix A.

With an $\mathbf{S}$ computed from one particular $\mathbf{C}_k$ or from the average of all $\mathbf{C}_k$s, we can check how well it does in generating a diagonal matrix $\mathbf{W}_k^2$ from $\mathbf{S}\mathbf{C}_k\mathbf{S}'$. An index for how much the data violate this *diagonality condition* is provided by the sum-of-squares of the nondiagonal elements of all $\mathbf{W}_k^2$s, computed with this one $\mathbf{S}$, appropriately normed to make the index independent of the size of $\mathbf{G}$. Schönemann et al. (1979) define a diagonality index

$$\delta_k = \frac{\text{tr } [\tilde{\mathbf{W}}_k^2 - \mathbf{I}]^2}{(m-1)m}, \tag{22.26}$$

where $\tilde{\mathbf{W}}_k^2$ is a normalized[3] form of $\mathbf{W}_k^2$. If $\mathbf{W}_k$ is diagonal, then $\delta_k = 0$. Otherwise, $\delta_k > 0$, and we then must decide whether it is still acceptably small.

An Empirical Application: Helm's Color Similarities

To illustrate, we scale the Helm color data from Table 21.1 with COSPA (Schönemann, James, & Carter, 1978), a program that also computes a common-space and a diagonality index for each k. Table 22.2 shows these indices. If the model were strictly adequate, we should have $v_k = 1$ and $\delta_k = 0$ for all individuals. Even though this is not true, it holds that all v_ks are high and most δ_ks are small. Moreover, the v_k-indices of the color-deficient subjects are generally lower than those of the color-normal subjects. This could be expected from the results in Table 22.2, because the former persons have relatively much more variance accounted for by the "small" dimensions, possibly due to a greater error variance in their data. Also, s_{13} has the worst δ_k value, which mirrors this person's relatively low communality values from Table 21.3.

Schönemann et al. (1979) report some statistical norms for these indices, derived by computer simulations under various error conditions. In the least restrictive or—relative to the model—the "nullest" case, each individual scalar-product matrix $\mathbf{B}_k$ is generated by forming the product $\mathbf{X}_k\mathbf{X}_k'$ with a random $\mathbf{X}_k$. For $m = 2, n = 10$, and $N = 16$, it is found that 90% of the v_k-values are less than 0.40. Hence, common-space values of the magnitude of those in Table 22.2 are extremely unlikely if the null-hypothetical situation is true. For the diagonality index, 90% of the values obtained were greater than 0.04. Some of the δ_ks in Table 22.2 are greater than this value, and, if taken by themselves, would not lead to a rejection of the null hypothesis. But if all of the diagonality indices are taken together, then a value distribution like the one observed for the Helm data is highly improbable under this random condition. These tests provide just rough guidelines, because it is not clear when we should assume such a

[3]The normalization of $\mathbf{W}_k^2$ is achieved by pre- and postmultiplying it by $\text{diag}[(\mathbf{W}_k^2)'\mathbf{W}_k^2]^{-1}$.

TABLE 22.2. Model test, indices for Helm data of Table 21.1.

Subject	Common Space (v_k)	Diagonality (δ_k)
s_1	0.95	0.04
s_2	0.94	0.02
s_3	0.94	0.01
s_4	0.96	0.02
s_5	0.93	0.04
$s_6^{[1]}$	0.91	0.09
$s_6^{[2]}$	0.96	0.08
s_7	0.92	0.02
s_8	0.93	0.12
s_9	0.92	0.07
s_{10}	0.87	0.08
s_{11}	0.86	0.06
$s_{12}^{[1]}$	0.84	0.04
$s_{12}^{[2]}$	0.85	0.01
s_{13}	0.86	0.45
s_{14}	0.93	0.17

null hypothesis. In color perception, it is certainly not the incumbent hypothesis, which the null hypothesis should be (Guttman, 1977). Moreover, we are not really interested in "some" dimension-weighting model but in a model where a particular group space (i.e., the color circle) is expected, and where this configuration is individually transformed by weighting a particular dimension, not just any one. Because everything comes out as predicted (except that, for some individuals, there is some residual unspecified variance) it would be foolish to reject the model altogether, just because some formal norms are too high. Rather, it seems more fruitful to take this result as a reasonable approximation, modify and/or supplement the theory somewhat, and test it in further empirical studies.

22.5 Conditional and Unconditional Approaches

The dimension-weighting model $\mathbf{B}_k = \mathbf{G}\mathbf{W}_k^2\mathbf{G}$ comes in two variants. In one case, the individual scalar-product matrices are processed as they are; in the other, they first are normed so that their sum-of-squares is equal to 1 for each k. Some authors call the first case the *Horan model* and the latter the INDSCAL model (Schönemann et al., 1979). A more gripping distinction calls the first approach *unconditional* and the latter *matrix-conditional* (Takane et al., 1977). This reveals the similarity to the situation in unfolding, where we did not want to compare data values across the rows of the data matrix and so used a split-by-rows or row-conditional approach. Analogously, a matrix-conditional or split-by-matrices treatment

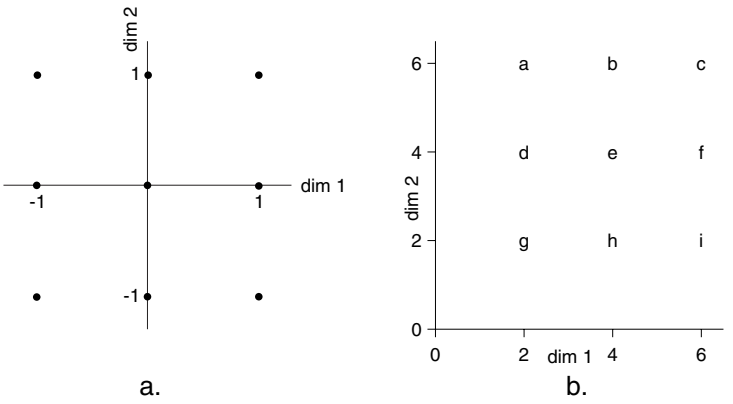


FIGURE 22.3. Synthetic (a) group space $\mathbf{G}$ and (b) subject space weights w_{11k} and w_{22k} for subjects a, b, ..., i used in the MacCallum (1977) study.

of the data implies that we do not want to compare the values of $\mathbf{B}_k$ and $\mathbf{B}_l$, for any $k \neq l$. The approach to be chosen depends on the particular data under investigation. For the Helm color data, for example, it seems that we should opt for the unconditional approach, because the data collection procedure suggests that all individuals used the same ratio scale for their proximity judgments. For the Green–Rao breakfast data (Table 14.1), on the other hand, the data were just ordinal, so ordinal MDS was used to arrive at ratio-scaled values. These values are the MDS distances, and they can be uniformly dilated or shrunk, of course, so that in this case we should prefer the matrix-conditional approach.

If the data are unconditionally comparable over individuals, then to norm all of the $\mathbf{B}_k$s in the same way leads to a loss of empirical information. This is apparent from the following demonstration due to MacCallum (1977). [Similar examples are given by Möbus (1975) and Schulz and Pittner (1978).] Figure 22.3 shows a group space $\mathbf{G}$ (panel a) and the associated subject space (panel b) that determines the weight matrices $\mathbf{W}_a, \mathbf{W}_b, \ldots, \mathbf{W}_i$. The nine $\mathbf{B}_k$s that can be derived from these figures as $\mathbf{B}_k = \mathbf{G}\mathbf{W}_k^2\mathbf{G}'$ differ in their sum-of-squares: for example, $\mathbf{B}_c$'s values are all much larger than the corresponding values in $\mathbf{B}_g$. Now, scaling the $\mathbf{B}_k$s with INDSCAL (which is always matrix-conditional) or with the matrix-conditional option of ALSCAL yields the subject space in Figure 22.4 (panel a). The different "size" of each individual's private perceptual space $\mathbf{X}_k$ is not represented. But because, for example, $\mathbf{X}_c$, $\mathbf{X}_e$, and $\mathbf{X}_g$ are perfectly similar and differ only in their sizes, the norming has the effect of projecting c, e, and g onto the same point in the subject space, as shown in Figure 22.4 (panel b). If the unconditional approach is used, the subject space is recovered perfectly.

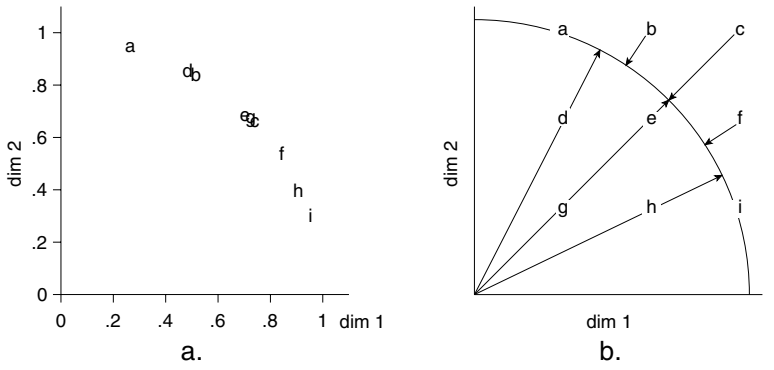


FIGURE 22.4. (a) INDSCAL reconstruction of subject space in Fig. 22.3b, and (b) visualization of norming effect.

22.6 On the Dimension-Weighting Models

The dimension-weighting models have received considerable attention in the literature. In their restrictive versions with fixed dimensions, they have been used in many applications because they promised to yield dimensions with a unique orientation while offering an intuitively appealing explanation for interindividual differences. The only other popular MDS model that accounts for interindividual differences is unfolding, but unfolding is for dominance data, not for similarity data. Unfolding assumes that the perceptual space is the same for all persons. It models different preferences by different ideal points in this space. The weighted Euclidean model allows for different perceptual spaces, related to each other by different weights attached to a set of fixed dimensions. Both models can be combined into one, an unfolding model with different dimensional weights for each person (see Chapter 16).

Some more recent developments should also be mentioned. They are motivated by practical and applied problems such as analyzing data sets where n is very large. The usual computer programs cannot handle such cases, or, more important, the subject space tends to be cluttered. "Marketing research suppliers often collect samples from thousands of consumers, and the ability of MDS procedures to fully portray the structure in such volumes of data is indeed limited. The resulting joint spaces or individual weight spaces become saturated with points/vectors, often rendering interpretation impossible ... Yet, marketeers are rarely interested in the particular responses of consumers at the individual level ... marketeers are more concerned with identifying and targeting market segments—homogeneous groups of consumers who share some designated set of characteristics (e.g.,

demographics, psychographics, consumption patterns, etc.) ..." (DeSarbo, Manrai, & Manrai, 1994, p. 191). In order to identify such segments, models were invented that combine a fuzzy form of cluster analysis with MDS. In essence, what one wants is an MDS solution where the subject space does not represent individual persons but types of persons. One procedure for that purpose is CLASCAL by Winsberg and De Soete (1993), a *latent class MDS model* (LCMDS). If the number of types of persons, S, is equal to 1, CLASCAL is but normal MDS. If $S = K$, CLASCAL corresponds to the INDSCAL model. For $1 < S < K$, CLASCAL estimates the probability that each person belongs to class S and, furthermore, computes an INDSCAL-like MDS solution for each class separately.

The dimension-weighting model therefore continues to be of interest. One may ask, however, whether it has led to noticeable substantive insights or to the establishment of scientific laws. In this regard, the model seems to have been much less successful, in contrast, for example, to the numerous regional laws established in the context of facet theoretical analyses of "normal" MDS data representations (see Chapter 5). Why is this so, even though the model certainly seems to be a plausible one? The answer may be found in the problems that we encountered with dimensional models in Chapter 17: if one takes a close look at dimensional models in the sense that the distance formula explains how dissimilarity judgments are generated from meaningful psychological dimensions, they are found to be less convincing, even in the case of stimuli as simple as rectangles. Adding interindividual differences to such models does not change things for the better. One should, therefore, be careful not to be misled by the dimension-weighting models: the dimensions they identify are not automatically meaningful ones, even though they may be rotationally unique.

22.7 Exercises

Exercise 22.1 Consider the three correlation matrices in Table 20.1 at p. 438.

(a) Without going into much theory, represent these data in the dimensionally weighted (DW) MDS model by using, for example, the PROXSCAL program in SPSS. How do you evaluate the outcome of this scaling effort?

(b) Scale each data matrix individually via MDS and then compare the configurations (by using Procrustean methods) and its Stress values with the DW solution.

(c) Use the DW configuration as a common starting configuration for an MDS scaling of each correlation matrix. How does this approach affect the MDS solutions?

Exercise 22.2 Consider Figure 17.7 at p. 373.

(a) Use the configuration of the 16 points on the *solid* grid to construct two different configurations, one by stretching this grid by factor 2 along the horizontal dimension (width), the other by stretching the grid by 2 along the vertical dimension (height). Compute the distances for the two resulting configurations.

(b) Use the two sets of distance as data in dimensional-weighting individual differences scaling. Check whether you succeed in recovering both the underlying configurations and the weights used in (a) to generate these distances.

(c) Add error to the distances and repeat the MDS analyses.

(d) Interpret the above weightings of the dimensions' width and height in substantive terms in the context of the perception of rectangles.

(e) Assume that you would generate more sets of distance matrices. This time, the configuration of points on the dashed grid in Figure 17.7 is stretched (or shrunk) along the dimensions width and height. Would these data lead to the same MDS configurations as the data generated above in (a)?

(f) Again assume that you would generate more sets of distances, this time by differentially stretching the configuration of points on the *dashed* grid in Figure 17.7 along a width-by-height coordinate system *rotated counterclockwise by 45 degrees*. Discuss what this would mean in terms of rectangle perception.

(g) Would you be able to discriminate persons using a weighted width-by-height model as in (a) and (e) from those using the rotated system in (f) by using INDSCAL or by using IDIOSCAL?

Exercise 22.3 Young (1987) reports the following hypothetical coordinates for four food stimuli and the dimension weights for five persons.

Food	I	II
Potato	-2	1
Spinach	-1	4
Lettuce	1	3
Tuna	4	-1

Person	I	II
1	.0	.9
2	.2	.8
3	.6	.6
4	.4	.4
5	.8	.2

.

(a) Interpret these data in the context of a dimensional salience model.

(b) Use the matrix equations of this model to compute the distance matrix for each person.

(c) Use a suitable MDS program to reconstruct the underlying configuration and weights from the set of distance matrices.

(d) Add an idiosyncratic rotation for each person, and repeat the above analyses with an MDS program that fits this model.

Exercise 22.4 The table below (Dunn-Rankin, Knezek, Wallace, & Zhang, 2004) shows ratings of five persons on the similarity of four handicaps: Learning Disability (LD), Mental Retardation (MR), Deafness (D), and Blindness (B).

Person	Handicap	LD	MR	D	B
1	LD	–			
1	MR	4	–		
1	D	4	5	–	
1	B	4	2	5	–
2	LD	–			
2	MR	6	–		
2	D	3	8	–	
2	B	2	2	4	–
3	LD	–			
3	MR	5	–		
3	D	4	6	–	
3	B	4	3	4	–

Person	Handicap	LD	MR	D	B
4	LD	–			
4	MR	2	–		
4	D	2	4	–	
4	B	6	2	5	–
5	LD	–			
5	MR	2	–		
5	D	6	7	–	
5	B	6	4	5	–

(a) Analyze these data with a dimensional salience model. Assess its fit.

(b) Interpret the dimensions of the solution space.

(c) Interpret the subject space (its meaning and how well it explains each person).

23
Modeling Asymmetric Data

Distances are always symmetric, but proximities may be asymmetric. Proximities, therefore, cannot always be fully represented by the distances among points in an MDS space. If one feels that the proximities deviate from being symmetric due to error only, this is not a problem. In that case, one may somehow symmetrize the proximities (e.g., by first averaging the corresponding p_{ij}s and p_{ji}s and then running the MDS on these averages). If one hypothesizes, however, that the nonsymmetries are meaningful, one needs special models for analyzing such data. In this chapter, we consider a number of such models. First, it is shown that an asymmetric proximity matrix can always be decomposed into a symmetric and a skew-symmetric component. The symmetric component can be then be subjected to ordinary MDS. For the skew-symmetric part, we discuss special visualization techniques for either the nonsymmetric component by itself or for embedding the nonsymmetric component into an MDS representation of the symmetric component. In the rest of the chapter, we treat a variety of models that analyze asymmetric proximities directly or indirectly. Many other models for asymmetric data exist. For a good overview of models for asymmetric data, we refer to Zielman and Heiser (1996).

23.1 Symmetry and Skew-Symmetry

We compare the values in row A of Table 4.2 with those in column A. Row A shows the confusion rate for code A, presented first, with codes B, C,

and so on, respectively, presented afterwards. In column A, the code A is always the second stimulus in the comparison. Comparing corresponding elements in column A and row A, we note, for example, that $p(\mathrm{A},\mathrm{R}) = .35$, and $p(\mathrm{R},\mathrm{A}) = .13$. Hence, A is definitely confused more often with R if it is presented before A than if it follows A in time.

Thus, the Morse code data are definitely not symmetric. However, in the analysis of these data so far, asymmetries played no further role. They were simply discarded as error, and only the symmetric part was analyzed. But is that good science? We know that asymmetries are not uncommon in cognition. A child, for example, is typically seen as similar to a parent, but one would not say that the parent resembles the child. This asymmetry is explained as a prototype–specimen relation: the specimen resembles the prototype, but the prototype does not resemble the specimen. Other examples and more theorizing are reported by Tversky (1977), for example. So, it is at least conceivable that the asymmetries in the Morse code data are not purely random but systematic.

To arrive at an answer to that question, we first note that every square matrix $\mathbf{P}$ can be uniquely decomposed into a symmetric matrix and a *skew-symmetric* matrix: That is, every asymmetric proximity matrix $\mathbf{P}$ can be uniquely decomposed into

$$\mathbf{P} = \mathbf{M} + \mathbf{N}, \tag{23.1}$$

where $\mathbf{M}$ is symmetric and $\mathbf{N}$ is skew-symmetric. This means that $\mathbf{M} = \mathbf{M}'$ and $\mathbf{N} = -\mathbf{N}'$. The two components of $\mathbf{P}$ are

$$\mathbf{M} = (\mathbf{P} + \mathbf{P}')/2, \text{ and} \tag{23.2}$$

$$\mathbf{N} = (\mathbf{P} - \mathbf{P}')/2. \tag{23.3}$$

Note that the diagonal elements of $\mathbf{N}$ are always zero, because for those elements it holds that $n_{ii} = (p_{ii} - p_{ii})/2 = 0$.

To demonstrate this decomposition numerically, consider the following example, where $\mathbf{P}$ is equal to the first four rows and columns of Table 4.2. Then, $\mathbf{P}$ has the decomposition

$$\begin{aligned}
\mathbf{P} &= \begin{bmatrix} 92 & 4 & 6 & 13 \\ 5 & 84 & 37 & 31 \\ 4 & 38 & 87 & 17 \\ 8 & 62 & 17 & 88 \end{bmatrix} = \mathbf{M} + \mathbf{N} \\
&= \begin{bmatrix} 92.0 & 4.5 & 5.0 & 10.5 \\ 4.5 & 84.0 & 37.5 & 46.5 \\ 5.0 & 37.5 & 87.0 & 17.0 \\ 10.5 & 46.5 & 17.0 & 88.0 \end{bmatrix} + \begin{bmatrix} 0.0 & -0.5 & 1.0 & 2.5 \\ 0.5 & 0.0 & -0.5 & -15.5 \\ -1.0 & 0.5 & 0.0 & 0.0 \\ -2.5 & 15.5 & 0.0 & 0.0 \end{bmatrix}.
\end{aligned} \tag{23.4}$$

To show that the decomposition is unique, assume that $\mathbf{P} = \mathbf{M}_1 + \mathbf{N}_1$ is another such decomposition with $\mathbf{M}_1 = \mathbf{M}_1'$ and $\mathbf{N}_1 = -\mathbf{N}_1'$. Then, we have $\mathbf{P}' = \mathbf{M}_1' + \mathbf{N}_1' = \mathbf{M}_1 - \mathbf{N}_1$, and it follows that $\mathbf{P} + \mathbf{P}' = 2\mathbf{M}_1$ and

$\mathbf{P} - \mathbf{P}' = 2\mathbf{N}_1$. Inserting this into (23.2) and (23.3), respectively, we find that $\mathbf{M} = \mathbf{M}_1$ and $\mathbf{N} = \mathbf{N}_1$, which proves uniqueness of the decomposition (23.1).

Furthermore, the decomposition in $\mathbf{M}$ and $\mathbf{N}$ allows one to partition the sum-of-squares into a part due to symmetry and a part due to skew-symmetry. That is,

$$\begin{aligned}
\sum_{i,j} p_{ij}^2 &= \sum_{i,j} \left[\tfrac{1}{2}(p_{ij} + p_{ji}) + \tfrac{1}{2}(p_{ij} - p_{ji})\right]^2 \\
&= \sum_{i,j} \tfrac{1}{4}\left[(p_{ij} + p_{ji})^2 + (p_{ij} - p_{ji})^2 + 2(p_{ij} + p_{ji})(p_{ij} - p_{ji})\right] \\
&= \sum_{i,j} m_{ij}^2 + \sum_{i,j} n_{ij}^2 + 2\sum_{i,j} m_{ij} n_{ij} \\
&= \sum_{i,j} m_{ij}^2 + \sum_{i,j} n_{ij}^2.
\end{aligned}$$

The cross-product term $\sum_{i,j} m_{ij} n_{ij}$ vanishes because

$$\begin{aligned}
\sum_{i,j} m_{ij} n_{ij} &= \tfrac{1}{4}\sum_{i,j} (p_{ij} + p_{ji})(p_{ij} - p_{ji}) \\
&= \tfrac{1}{4}\left[\sum_{i,j} p_{ij}^2 - \sum_{i,j} p_{ji}^2 + \sum_{i,j} p_{ij} p_{ji} - \sum_{i,j} p_{ij} p_{ji}\right] = 0.
\end{aligned}$$

Thus, $\mathbf{M}$ and $\mathbf{N}$ are orthogonal because $\operatorname{tr} \mathbf{MN} = 0$. The decomposition of the sum-of-squares suggests analyzing asymmetric data in two separate steps: the analysis of the symmetric part and the analysis of the skew-symmetric part. For the Morse code data, the sum of the squared proximities without the diagonal equals 698,309.0, from which 671,489.5 (96%) is due to symmetry and 26,819.5 (4%) is due to asymmetry. This implies that the symmetric part of the data is dominant, and asymmetry plays a minor role, but may still reveal interesting relations.

23.2 A Simple Model for Skew-Symmetric Data

The simplest model for representing skew-symmetric data $\mathbf{N}$ locates every object on a line such that the *signed* distance for every pair of coordinates represents the corresponding elements of $\mathbf{N}$. Expressed algebraically, this model postulates that $n_{ij} = x_i - x_j$, or, in matrix form,

$$\mathbf{N} = \mathbf{x}\mathbf{1}' - \mathbf{1}\mathbf{x}', \tag{23.5}$$

where $\mathbf{x}$ has sum zero. Choosing $\mathbf{x} = n^{-1}\mathbf{N}\mathbf{1}$ or, in other words, the averages of the n_{ij} values within each row i over all columns, is the least-squares

solution for (23.5). Obviously, this model is so restricted that it does not fit many data. Hence, we now turn to more general models.

23.3 The Gower Model for Skew-Symmetries

An interesting decomposition of a skew-symmetric matrix has been given by Gower (1977) and Constantine and Gower (1978). The singular value decomposition of any skew-symmetric matrix $\mathbf{N}$ has the special form

$$\mathbf{N} = \mathbf{PK\Phi P'}, \tag{23.6}$$

where $\mathbf{P}$ is orthonormal, $\mathbf{\Phi}$ has singular values ordered in pairs (ϕ_1, ϕ_1, ϕ_2, ϕ_2, ...), and $\mathbf{K}$ is a permutation-reflection matrix that has along its diagonal 2×2 blocks with off-diagonal values -1 and 1. The decomposition of $\mathbf{N}$ in (23.4) is

$$\left[\begin{array}{cc|cc} .16 & .00 & .00 & -.99 \\ -.97 & .19 & .00 & -.16 \\ -.01 & -.04 & 1.00 & -.00 \\ -.19 & -.98 & -.04 & -.03 \end{array}\right] \left[\begin{array}{cc|cc} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \end{array}\right] \left[\begin{array}{cc|cc} 15.72 & & & \\ & 15.72 & & \\ \hline & & .91 & \\ & & & .91 \end{array}\right] \left[\begin{array}{cc|cc} .00 & .19 & -.04 & -.98 \\ -.16 & .97 & .01 & .19 \\ \hline .99 & .16 & .00 & .03 \\ .00 & .00 & 1.00 & -.04 \end{array}\right].$$

The dimensions here come in pairs with equal singular values. Such a pair of dimensions is called a *bimension*. Each bimension spans a plane. Its points can be taken from the pairs of columns (1,2), (3,4), and so on of $\mathbf{PK}$ or from the respective columns of $\mathbf{P}$. The configurations are the same in both cases. Their interpretation hinges on how any two points are related to each other in terms of (a) the angle subtended by the vectors that they define, and (b) the area of the triangle spanned by these vectors. The area of the triangle represents the size of the asymmetry, and sense of the angle represents the sign of the asymmetry. Note that because the singular values in each bimension are equal, the bimension may be freely rotated or reflected without changing the fit. Thus, in the Gower model, the axes cannot be interpreted.

To illustrate these notions, let us apply the Gower decomposition to the skew-symmetric part of the Morse code confusion data (Rothkopf, 1957). Zielman and Heiser (1996) did the same analysis on the Morse codes that represent the 10 digits only. The first bimension (with singular value 67.37, showing 34% of the total skew-symmetry) of the full table is presented in Figure 23.1, a display we call a *Gower diagram*. In this display, all the rows of $\mathbf{P}$ are plotted as vectors and we have to use a clockwise rotation for positive estimates of n_{ij}. To understand how to interpret this plot, consider the triangle between the origin, point H, and point V. The area of the triangle is an estimate of the size of the asymmetry. Because clockwise rotations indicate positive estimates, going from H to V indicates that if H is the first stimulus in the pair, it is more often confused with V than vice versa.

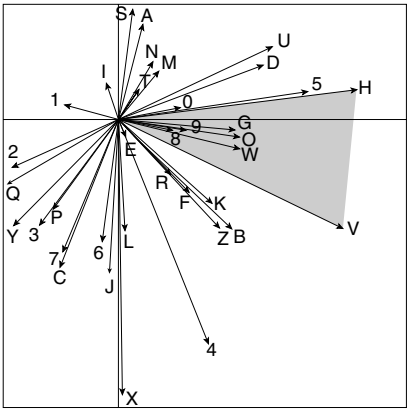


FIGURE 23.1. Gower diagram of the skew-symmetric part of the Morse code data (first bimension). The area of the triangle between the origin and the two points H and V is an estimate of the value n_{HV}. The clockwise rotation indicates that $n_{\mathrm{HV}} > 0$ and $n_{\mathrm{VH}} < 0$.

Once identified, an asymmetry like this one can be interpreted substantively. In this case, it is easy to understand, because the sequence HV is $\cdots\cdot$ followed by $\cdots -$, whereas VH is $\cdots -$ followed by $\cdots\cdot$. Clearly, the middle $-$ makes it easier for the subjects to distinguish the two signals.

We also note from Figure 23.1 that the big contributors to asymmetry are the signals X, 4, V, and H, where, for example, H4, VX, and V4 generate higher confusion rates than, respectively, 4H, XV, and 4V. In contrast, for E, I, and T it does not really matter whether they occur as the first or second signal, because they lie close to the origin and do not have much asymmetry with other signals.

Some of the properties of Gower's model are as follows.

- If n is even, then there are $n/2$ bimensions; if n is odd, then there are $n/2 - 1$ bimensions; that is, $\phi_n = 0$.
- Points that lie on the same line through the origin do not have asymmetry, thus spanning a triangle with zero area.
- A point close to the origin has little asymmetry with all other points, and, hence, triangles where these points together with the origin and any other point form the corners tend to be small, in general.
- If there is a line through the origin in a bimension such that all of the vectors project positively on this line, then reordering $\mathbf{N}$ by the order of the vectors of the bimension yields a matrix with all negative elements in the lower (or upper) triangular matrix and the positive elements in the upper (or lower) triangular matrix. No circular triads are present in the data in this case.

- If points lie on a line not through the origin of the vectors, then the points form an additive scale. Let the order of three points along such a line be A, B, C. Then, the area spanned by the triangle OAC equals $OAB + OBC$, which is clearly additive.
- When computing the solution for the Gower model, we do not know a priori what the direction of interpretation will be. If the submatrix of $\mathbf{K}$ equals $\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$, then an anticlockwise rotation from vector i to j indicates a positive estimate of n_{ij} and a negative estimate for n_{ij}. If the computational procedure gives $\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$ as a submatrix, then we have to apply a clockwise rotation to identify a positive estimate of n_{ij}.

23.4 Modeling Skew-Symmetry by Distances

The power of the Gower decomposition is that a graph of only n objects is obtained in a plane where the angles subtended by the vectors have a fixed meaning. A disadvantage is that areas represent the size of asymmetries, because judgments on areas are cognitively demanding. Interpreting distances is easier. Therefore, we propose a new model for visualizing a skew-symmetric matrix that expresses asymmetries by Euclidean distances.

The distance model for skew-symmetry uses Euclidean distances between points i and j to estimate the size of the skew-symmetric effect $|n_{ij}|$. In addition, similar to the Gower model, the direction of rotation is important. If the angle measured clockwise between the vector to point i and the vector to point j is less then 180°, then n_{ij} is estimated by $d_{ij}(\mathbf{X})$. Conversely, if this angle is between 180° and 360° measured clockwise (or, equivalently, between 0 and −180°, in the counterclockwise sense), then n_{ij} is estimated by $-d_{ij}(\mathbf{X})$. Thus, the model predicts that starting from a point i all points j that are in the half plane with positive angles between 0° and 180° (measured clockwise) have a positive estimate for n_{ij}. All points j that have negative angles between 0° and −180° (measured clockwise) produce a negative estimate for n_{ij}.

To fit the distance model to skew-symmetric data, we need to determine for point i whether point j lies in the positive or in the negative rotational half of the plane. Let $\mathbf{x}_i$ be the 2×1 vector with the row coordinates of point i. Now, a rotation of $\mathbf{x}_i$ by −90° is obtained by reflecting the first coordinate vector and then swapping the dimensions, that is, by $\mathbf{T}'\mathbf{x}_i$ with

$$\mathbf{T}' = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \text{ and } \mathbf{T} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}.$$

If another vector $\mathbf{x}_j$ is projected onto $\mathbf{T}'\mathbf{x}_i$, and the result is positive, then point j is on the positive side of the plane so that n_{ij} is estimated by

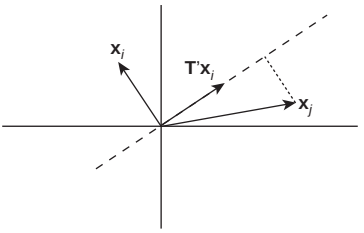


FIGURE 23.2. Illustration of determining whether point j gives a positive or negative contribution for a combination ij. Vector $\mathbf{x}_j$ is projected on $\mathbf{T}'\mathbf{x}_i$, the $-90°$ rotation of $\mathbf{x}_i$. In this example, the projection $\mathbf{x}_j'\mathbf{T}'\mathbf{x}_i$ is positive indicating a positive estimate for n_{ij}.

$d_{ij}(\mathbf{X})$. Figure 23.2 gives an example of this case. On the other hand, if the projection is negative, then point j is on the negative side of the plane so that n_{ij} is estimated by $-d_{ij}(\mathbf{X})$. The projection of $\mathbf{x}_j$ onto $\mathbf{T}'\mathbf{x}_i$ is given by $\mathbf{x}_j'\mathbf{T}'\mathbf{x}_i$. Let the sign function be defined by

$$\text{sign}(z) = \begin{cases} 1 & \text{if } z > 0, \\ 0 & \text{if } z = 0, \\ -1 & \text{if } z < 0. \end{cases}$$

Now, the estimate of n_{ij} can be obtained by $\text{sign}(\mathbf{x}_j'\mathbf{T}'\mathbf{x}_i)d_{ij}(\mathbf{X})$. A formal model for the distance model for skew-symmetry is obtained by minimizing the sum of squared differences between n_{ij} and its estimate, that is, by minimizing

$$L(\mathbf{X}) = \sum_{i=1}^{n}\sum_{j=1}^{n}\left[n_{ij} - \text{sign}(\mathbf{x}_i\mathbf{T}\mathbf{x}_j)d_{ij}(\mathbf{X})\right]^2 \tag{23.7}$$

over $\mathbf{X}$.

We applied this model to the skew-symmetric part of the Morse code data and the results are given in Figure 23.3. Small distances in the plot generally indicate small skew-symmetries. The plot is dominated by large asymmetries that are shown by large distances. For example, we see that H and X have a large distance. Because we are using a clockwise rotation, the sequence HX leads to higher confusion rates than presenting X first and H afterwards. Comparing this solution to the Gower diagram in Figure 23.1 shows that there is not so much difference. However, the advantage of the distance model for skew symmetry is that the distance of two points shows to what extent the proximities of two objects are asymmetric.

Some caution is required here. The loss function (23.7) was fitted with a general-purpose optimization routine in MatLab. Such a routine may not be optimal for this loss function. In particular, we expect that this loss function may be quite sensitive to local optima. Further study is needed to see how severe this problem is.

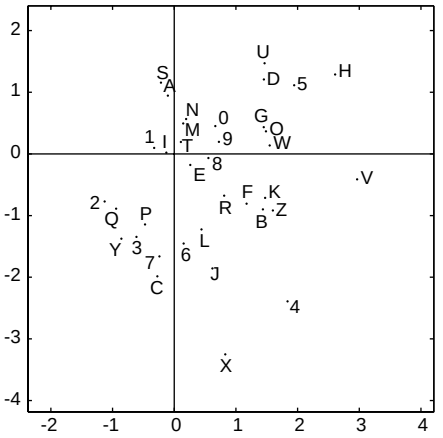


FIGURE 23.3. Plot of the distance model for skew-symmetry on the Morse code data. The distance in a clockwise direction estimates positive values for n_{ij} and those in an anticlockwise direction estimate minus the distance for n_{ij}.

As is the case for the Gower decomposition, the dimensions in the current model come in pairs, the so-called bimension. The distance model for skew-symmetry could be extended to two or more pairs of such dimensions. It seems natural to compute the distance for each bimension separately, inasmuch as the interpretation is done bimensionwise.

23.5 Embedding Skew-Symmetries as Drift Vectors into MDS Plots

Another simple method for asymmetries is simultaneously displaying the symmetric part and the skew-symmetric part of the data. This makes it possible to see how these two data components are related to each other. The skew-symmetric values are embedded into the MDS representation of the symmetrized data by drawing arrows (*drift vectors*) from each point i to any other point j in the configuration so that these vectors correspond in length and direction to the values in row i of the skew-symmetric matrix (Borg, 1979; Borg & Groenen, 1995). Thus, on point R in Figure 4.6 we would attach a vector of length $.11 = (.35 - .13)/2$ pointing towards A. The units for the arrows are chosen so that they can be represented most conveniently in the given configuration. The arrow's direction towards A is chosen to express that A is more often confused with R when presented first than vice versa.

To avoid a cluttered picture, we can draw only the resultant of the vector bundle thus attached to each point. The resultant averages out random nonsymmetries and shows the general drift (see Figure 23.4). Length and

direction angle of drift vectors are computed as follows. We use vector notation and show the 2D case.

1. Do for all points i.
2. Do for all points $j \neq i$.
3. Given vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ in terms of their MDS coordinates, $\mathbf{a}_{ij} = \mathbf{x}_j - \mathbf{x}_i$ is the vector from point i to point j in the MDS configuration.
4. Norm $\mathbf{a}_{ij}$ to unit length to get $\mathbf{b}_{ij}$; that is, $\mathbf{b}_{ij} = \mathbf{a}_{ij}/(\mathbf{a}'_{ij}\mathbf{a}_{ij})^{1/2}$.
5. Multiply $\mathbf{b}_{ij}$ by element n_{ij} of the skew-symmetric component of the proximity matrix to obtain $\mathbf{c}_{ij}$; that is, $\mathbf{c}_{ij} = n_{ij}\mathbf{b}_{ij}$.
6. End do.
7. Average the n vectors $\mathbf{c}_{ij}$ to obtain the (average) drift vector for point i, $\mathbf{d}_i$; that is, $\mathbf{d}_i = n^{-1}\sum_j \mathbf{c}_{ij}$.
8. For plotting $\mathbf{d}_i$, compute $\mathbf{d}_i$'s length as the root mean square of its elements and the direction angle relative to the Y-axis, $\alpha_i = \arccos(\mathbf{d}'_i\mathbf{u}/\sqrt{\mathbf{d}'_i\mathbf{d}_i})$, where $\mathbf{u}' = (0, 1)$.
9. End do.

Figure 23.4 shows the 2D MDS solution with the embedded drift vectors. It is obvious that the nonsymmetries in the confusion data are not random (see also Möbus, 1979). One notes that the arrows exhibit a definite vector field with a trend that, in substantive terms, indicates that shorter Morse code signals are more often confused with longer ones than vice versa. The vertical axis reflects the temporal length of the signals. The signal E, for example, is just one ·, while the signal for O is – – – – –. Moreover, because the trend is towards the North-West, the asymmetries also reflect the composition pattern of the signals: signals in the direction of the drift vectors tend to have more short components (see Figure 4.7).

23.6 Analyzing Asymmetry by Unfolding

We now change to models that directly analyze the entire asymmetric proximity matrix. One property that all these models share is that they somehow model both the symmetric part of the data and the skew-symmetric part. One of the simplest distance models that can be used is unfolding (see Chapters 14 to 16) as, for example, has been suggested by Gower (1977).

We look at an example of brand switching data. These data are derived from supermarket scanner data as described by Bell and Lattin (1998). In this example, we want to investigate how households change in buying 15 different cola soft drinks. The daily purchases of cola soft drinks were recorded for 488 U.S. households over a period of 104 weeks from June 1991

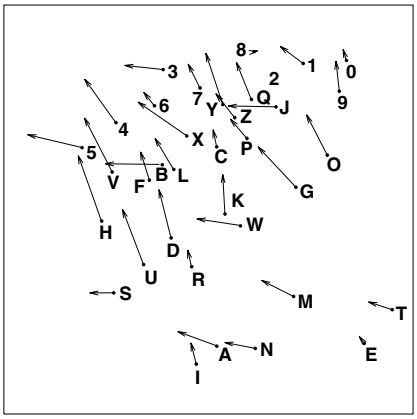


FIGURE 23.4. 2D Morse code MDS configuration with drift vectors to model the asymmetry in the Morse codes.

to June 1993. A household is considered to make a change whenever the products were of a different type or brand for two subsequent purchases. If a household has more than one purchase on a day, then we divide this switch evenly over all the products that have been bought that day. Table 23.1 shows the rounded brand switching data for colas. The rows of the table indicate the type of cola bought before and the column indicates the type of cola that is currently bought. Thus, the changes are made from the row product to the column product.

Brand switching data can be interpreted as similarities, because large values indicate that households easily switch between the two products and, hence, consider them similar. For unfolding, we need to transform these similarities into dissimilarities. One such transformation can be obtained by the gravity model discussed in Section 6.4. This model stems from astronomy and relates the gravitational force p_{ij} to a squared Euclidean distance d_{ij}^2 by the relation $p_{ij} = km_i m_j / d_{ij}^2$, where k is a constant and m_i and m_j are the masses of the two bodies. We equate m_i and m_j with the row and column sums of the brand switching matrix. Then,

$$\delta_{ij} = \left(\frac{m_i \cdot m_j}{p_{ij}} \right)^{1/2} \tag{23.8}$$

gives an asymmetric dissimilarity matrix on which unfolding can be performed. If p_{ij} equals zero, then δ_{ij} is declared missing. The brand switching data converted in the sense of the gravity model are given in Table 23.2. Note that Zielman and Heiser (1993) and Groenen and Heiser (1996) have used the gravity model in a similar context.

We have applied unfolding to the cola brand switching data. The joint representation is given in Figure 23.5, where the rows (the colas from which the change is made) are plotted as solid points and the columns (the colas

TABLE 23.1. Brand switching data among 15 different colas. The row indicates from which product the change is made, the column contains the product to which is changed.

		To													
	From	a.	b.	c.	d.	e.	f.	g.	h.	i.	j.	k.	l.	m.	n.	o.
a.	Coke decaf	41	11	2	8	0	2	15	8	14	0	9	11	0	6	2
b.	Coke diet decaf	9	341	32	3	4	8	55	78	31	1	63	16	17	14	4
c.	Pepsi diet decaf	3	27	160	15	8	2	18	15	32	2	31	13	2	12	7
d.	Pepsi decaf	7	3	17	89	2	3	16	8	4	0	3	27	1	6	3
e.	Canfield	1	7	6	2	119	6	20	8	19	0	16	15	2	21	7
f.	Coke	4	4	2	1	4	73	37	8	12	3	8	33	3	36	6
g.	Coke classic	14	53	16	16	22	38	675	98	56	10	48	187	33	172	20
h.	Coke diet	5	74	14	12	7	5	108	716	123	26	92	31	11	27	18
i.	Pepsi diet	14	35	36	3	15	11	56	120	422	20	86	82	29	38	10
j.	RC diet	0	5	0	1	3	3	6	30	5	12	17	6	4	14	1
k.	Rite diet	13	70	29	6	12	5	49	87	92	19	471	40	11	34	8
l.	Pepsi	8	18	9	26	19	29	204	26	91	5	29	663	24	217	51
m.	Private label	2	14	4	3	1	2	35	13	22	1	20	19	364	23	1
n.	RC	7	10	13	7	19	34	171	30	31	10	36	230	22	440	41
o.	Wildwood	3	3	7	3	10	9	26	22	11	2	4	48	2	35	215

TABLE 23.2. Brand switching data of Table 23.1 converted in the sense of the gravity model (23.8).

		To													
	From	a.	b.	c.	d.	e.	f.	g.	h.	i.	j.	k.	l.	m.	n.	o.
a.	Coke decaf	20	89	150	56	0	122	113	143	94	0	116	129	0	153	159
b.	Coke diet decaf	99	37	86	210	203	139	135	105	145	274	100	245	144	230	258
c.	Pepsi diet decaf	123	93	27	67	103	200	170	171	102	139	102	195	302	178	140
d.	Pepsi decaf	59	206	62	20	152	120	133	173	214	0	242	100	315	186	158
e.	Canfield	181	155	120	156	23	98	136	199	112	0	120	154	256	114	118
f.	Coke	88	199	201	214	120	27	97	193	137	93	165	100	202	84	124
g.	Coke classic	117	136	178	133	127	94	57	137	159	127	168	105	152	96	169
h.	Coke diet	182	108	177	144	211	242	132	47	100	74	113	241	246	227	167
i.	Pepsi diet	96	137	97	252	126	143	161	102	47	74	103	130	133	168	196
j.	RC diet	0	120	0	144	93	91	163	67	144	31	77	159	119	91	205
k.	Rite diet	98	96	106	175	139	209	170	117	100	74	43	183	212	175	216
l.	Pepsi	152	231	234	103	135	106	102	263	123	177	214	55	176	85	105
m.	Private label	185	159	213	185	358	245	149	226	152	241	156	198	27	158	454
n.	RC	144	273	171	175	119	86	98	216	185	111	169	82	162	52	103
o.	Wildwood	132	300	141	161	99	101	151	152	187	149	305	109	324	112	27

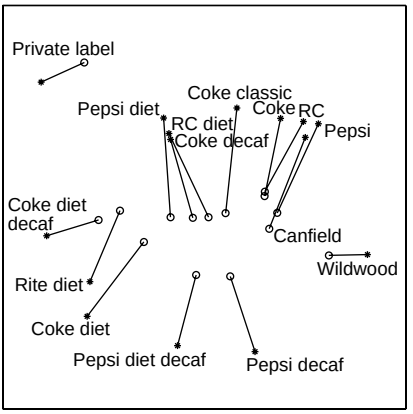


FIGURE 23.5. Unfolding on the brand switching data of Table 23.1 after being converted by the gravity model. The solid points denote the the colas chosen at time t, the open circles represent colas chosen at time $t+1$.

to which the change is made) as open circles. To interpret this diagram, one studies, for example, how Coke diet buyers change: note that they tend to move to Riet diet, Coke diet decaf, Pepsi diet, RC diet, and Pepsi diet decaf, because they are the nearest in the plot. They are not likely to change to Private label, Wildwood, and Canfield because these brands are far away. A striking feature of the solution is that there is not much changing going on for customers buying the Private label because it is located far away from all other colas. In the same manner, one can focus on any other cola and see how customers change to the competing colas.

23.7 The Slide-Vector Model

The unfolding model for asymmetry estimates many parameters. To reduce the number of parameters, a constrained form of unfolding can be used. One such model is the *slide-vector* model that constrains the row and column points to be equal up to a translation. This restriction implies that the solution consists of the points that represent the choice objects and one uniform shift of the entire space, the *slide-vector* $\mathbf{z}$, in a fixed direction. This model was first proposed by De Leeuw and Heiser (1982, who attributed it to a personal communication with Kruskal, 1973) and was thoroughly worked out by Zielman and Heiser (1993). Note that Carroll and Wish (1974b) refer to the same model when they speak of the drift vector model. The rationale behind the model is that the data can be thought to consist of symmetric distances augmented by a strong wind: changes against the wind direction will take more effort, whereas changes in the direction are easier. In fact, Figure 23.4 suggests just that model.

To lay out the slide-vector model formally, we define $\mathbf{X}$ to be the row coordinates and $\mathbf{Y}$ the column coordinates in the unfolding problem. Then, the restriction in the slide-vector model amounts to $y_{is} = x_{is} - z_s$. The definition of the distance in the slide-vector model is given by

$$d_{ij}(\mathbf{X}, \mathbf{Y}, \mathbf{z}) = \left(\sum_s (x_{is} - y_{js})^2 \right)^{1/2} = \left(\sum_s (x_{is} + z_s - x_{js})^2 \right)^{1/2}. \quad (23.9)$$

Thus, the row points $\mathbf{X}$ are equal to the column points $\mathbf{Y}$ translated by the slide-vector $\mathbf{z}$.

Clearly, if $\mathbf{z} = \mathbf{0}$, then (23.9) reduces to the ordinary symmetric Euclidean distance. For two objects that are at the same position, $d_{ij}(\mathbf{X}, \mathbf{Y}, \mathbf{z})$ reduces to $(\sum_s z_s^2)^{1/2}$, which again is symmetric. In fact, if diagonal values δ_{ii} are fitted in the unfolding problem, then these entries are all estimated by the length of the slide vector.

The slide-vector model can easily be fitted by considering unfolding as MDS with missing within-group dissimilarities (see Section 14.1) combined with external constraints (see Section 10.3). In matrix notation, the slide-vector restrictions are given by $\mathbf{Y} = \mathbf{X} - \mathbf{1}\mathbf{z}'$. Stacking the row and column coordinates underneath each other gives

$$\begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \end{bmatrix} = \begin{bmatrix} \mathbf{X} \\ \mathbf{X} - \mathbf{1}\mathbf{z}' \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{I} & -\mathbf{1} \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ \mathbf{z}' \end{bmatrix} = \mathbf{E} \begin{bmatrix} \mathbf{X} \\ \mathbf{z}' \end{bmatrix}.$$

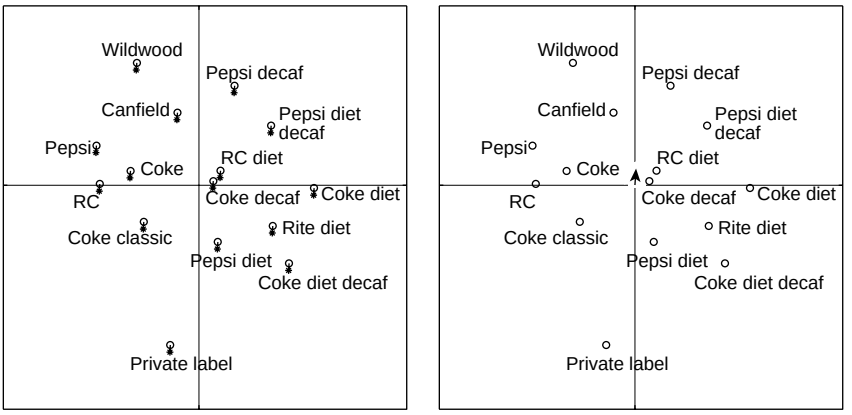
Thus, the slide-vector model is fitted by providing the external constraints $\mathbf{E}$ in this way.

Consider a small illustrative example of three objects for generating the matrix of external constraints $\mathbf{E}$. Suppose we want to apply the slide-vector model in two dimensions. Then, the full MDS matrix becomes

$$\boldsymbol{\Delta} = \begin{bmatrix} 0 & 0 & 0 & \delta_{11} & \delta_{12} & \delta_{13} \\ 0 & 0 & 0 & \delta_{21} & \delta_{22} & \delta_{23} \\ 0 & 0 & 0 & \delta_{31} & \delta_{32} & \delta_{33} \\ \delta_{11} & \delta_{21} & \delta_{31} & 0 & 0 & 0 \\ \delta_{12} & \delta_{22} & \delta_{32} & 0 & 0 & 0 \\ \delta_{13} & \delta_{23} & \delta_{33} & 0 & 0 & 0 \end{bmatrix} \text{ and } \mathbf{W} = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}.$$

Note that the matrix $\mathbf{W}$ indicates that the within-block dissimilarities are missing, so that we are dealing with unfolding. The between-block dissimilarities contain the asymmetric data. The matrix of coordinates is simply

$$\begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \\ x_{31} & x_{32} \\ y_{11} & y_{12} \\ y_{21} & y_{22} \\ y_{31} & y_{32} \end{bmatrix},$$



a. Joint representation. b. Slide vector representation.

FIGURE 23.6. The slide-vector model fitted to the brand switching data of Table 23.1 after being converted by the gravity model. Panel (a) shows the joint representation of rows (*) and columns (o). Panel (b) shows the representation with one set of points and the slide vector. The arrow in the center that indicates the slide vector is rather small.

that is, restricted to be equal to

$$\mathbf{E}\begin{bmatrix}\mathbf{X}\\ \mathbf{z}'\end{bmatrix} = \begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\1&0&0&-1\\0&1&0&-1\\0&0&1&-1\end{bmatrix}\begin{bmatrix}x_{11}&x_{12}\\x_{21}&x_{22}\\x_{31}&x_{32}\\z_1&z_2\end{bmatrix} = \begin{bmatrix}x_{11}&x_{12}\\x_{21}&x_{22}\\x_{31}&x_{32}\\x_{11}-z_1&x_{12}-z_2\\x_{21}-z_1&x_{22}-z_2\\x_{31}-z_1&x_{32}-z_2\end{bmatrix}.$$

Thus, the slide-vector model can be fitted by providing the matrix of external variables $\mathbf{E}$ above to an MDS program that allows for linear restrictions on the configuration and allows for missing values of the dissimilarities.

We now return to the colaswitching data of Table 23.1 and their transformations by the gravity model in Table 23.2. The slide-vector model was fitted by PROXSCAL in SPSS as it allows for external variables such as those defined by $\mathbf{E}$. The resulting configuration is given in Figure 23.6. Because the slide-vector model is a constrained version of unfolding, there are two possible representations. The joint representation in Figure 23.6a shows the row and column points together. It can be clearly seen that the column points are indeed equal to the row point up to a translation. The second representation (see Figure 23.6b) only shows a single set of coordinates together with the slide-vector $\mathbf{z}$.

The example shows that the slide vector is rather small for these data. The model mostly captures the symmetric part of the data and shows that a uniform trend in the asymmetries, however large they may be, is relatively small.

It seems that switching takes place mostly between colas of the same type. For example, there is switching with the group of Coke, Pepsi, RC, and Canfield, within the group Pepsi diet, Coke diet decaf, Rite diet, and Coke diet, but less switching between these groups. In this solution, too, we see that Private label is farthest away from all other colas indicating that those households do not switch easily to other colas.

A disadvantage of the slide-vector model is that it is quite restrictive compared to unconstrained unfolding. Instead of $n \times p$ coordinates for $\mathbf{Y}$ in unrestricted unfolding, the slide-vector model only estimates p parameters for the slide vector. It seems that the slide-vector model works better if asymmetries are large relative to the symmetric part of the data. We also expect the slide-vector model to perform better for small data sets because the restrictions on the coordinates are weaker than for large n.

23.8 The Hill-Climbing Model

The formal advantage of the slide-vector model is that it easily fits into the constrained unfolding framework so that it can be fitted by a standard program such as PROXSCAL. The joint representation of $\mathbf{X}$ and the constrained $\mathbf{Y}$ is easy to interpret even though it doubles the number of points. The more parsimonious representation of only $\mathbf{X}$ and $\mathbf{z}$ seems harder to interpret. To remedy the latter problem, we propose an adaptation of the slide-vector model. We are not aware of references in the literature that have proposed this model earlier.

The new model is based on the *hill-climbing* metaphor: walking uphill is more difficult than walking downhill, whereas on a plateau walking from point A to B takes the same effort as walking from B to A. This idea can be modeled by choosing the distance measure as

$$d_{ij}(\mathbf{X}, \mathbf{z}) = \left(\sum_s (x_{is} - x_{js})^2 \right)^{1/2} + \frac{\sum_s (x_{is} - x_{js}) z_s}{\left(\sum_s (x_{is} - x_{js})^2 \right)^{1/2}}, \tag{23.10}$$

or in matrix notation

$$d_{ij}(\mathbf{X}, \mathbf{z}) = \|\mathbf{x}_i - \mathbf{x}_j\| + \frac{(\mathbf{x}_i - \mathbf{x}_j)'\mathbf{z}}{\|\mathbf{x}_i - \mathbf{x}_j\|}. \tag{23.11}$$

A least-squares model estimating (23.11) is given by

$$L(\mathbf{X}, \mathbf{z}) = \sum_{i=1}^{n} \sum_{j=1}^{n} \left(\delta_{ij} - \left[\|\mathbf{x}_i - \mathbf{x}_j\| + \frac{(\mathbf{x}_i - \mathbf{x}_j)'\mathbf{z}}{\|\mathbf{x}_i - \mathbf{x}_j\|} \right] \right)^2. \tag{23.12}$$

The rationale behind this model is that the projection of the difference vector $\mathbf{x}_i - \mathbf{x}_j$ of going from point i to point j on a slope given by the

slope vector $\mathbf{z}$ measures to what extent it is more difficult or easier to go from point i to j than the Euclidean distance only. If the difference vector is orthogonal to the slope vector $\mathbf{z}$, then no asymmetry is modeled. If the difference vector is parallel to the slope vector, then maximum asymmetry is achieved. Note that $d_{ii}(\mathbf{X}, \mathbf{z}) = 0$ by definition so that the diagonal values cannot be modeled. The denominator $\|\mathbf{x}_i - \mathbf{x}_j\|$ in (23.11) is chosen so that the length of the Euclidean distance between i and j does not influence the amount of asymmetry.

The orientation of the difference vector, and thus the positioning of the points, is influenced by the asymmetry in the data because the orientation of the difference vector determines the projection on the slope vector. It may be verified that (23.12) can be decomposed into a symmetric part $\mathbf{M}$ with elements $m_{ij} = (\delta_{ij} + \delta_{ji})/2$ and a skew-symmetric part $\mathbf{N}$ with elements $n_{ij} = (\delta_{ij} - \delta_{ji})/2$; that is,

$$\begin{aligned} L(\mathbf{X}, \mathbf{z}) &= 2\sum_{i=1}^{n}\sum_{j=i+1}^{n}\left(m_{ij} - \|\mathbf{x}_i - \mathbf{x}_j\|\right)^2 \\ &\quad + \sum_{i=1}^{n}\sum_{j=1}^{n}\left(n_{ij} - \frac{(\mathbf{x}_i - \mathbf{x}_j)'\mathbf{z}}{\|\mathbf{x}_i - \mathbf{x}_j\|}\right)^2. \end{aligned} \tag{23.13}$$

This reformulation shows that the distances directly model the symmetric part of $\mathbf{\Delta}$, and the projections model the skew-symmetric part. The size of the symmetric part and the skew-symmetric part can be expressed in terms of the sum-of-squares $\|\mathbf{M}\|^2$ and $\|\mathbf{N}\|^2$. The relative difference between those measures influences the solution (23.13) in how much symmetry and how much skew-symmetry of the data is fitted. Consider the following adaptation of (23.13); that is,

$$\begin{aligned} L_w(\mathbf{X}, \mathbf{z}) &= \frac{2\alpha}{\|\mathbf{M}\|^2}\sum_{i=1}^{n}\sum_{j=i+1}^{n}\left(m_{ij} - \|\mathbf{x}_i - \mathbf{x}_j\|\right)^2 \\ &\quad + \frac{1-\alpha}{\|\mathbf{N}\|^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\left(n_{ij} - \frac{(\mathbf{x}_i - \mathbf{x}_j)'\mathbf{z}}{\|\mathbf{x}_i - \mathbf{x}_j\|}\right)^2, \end{aligned} \tag{23.14}$$

where $0 \leq \alpha \leq 1$ is a fixed weight that sets the relative importance of the two parts. Choosing $\alpha = 1$ fits only the symmetric part $\mathbf{M}$ as in regular MDS. For $\alpha = 0$, only the skew-symmetric part $\mathbf{N}$ is fitted. If $\alpha = .5$, then both parts are equally important in the solution. Note that the length of $\mathbf{z}$ only reflects the amount of skew-symmetry that is captured from $\mathbf{N}$. For the interpretation, it is the direction of $\mathbf{z}$ that shows how the difference vectors project on the hill slope $\mathbf{z}$.

We have fitted the hill-climbing model to the cola data. Our model cannot be estimated by a constrained form of MDS and a specialized algorithm had to be developed. Here we have used a general-purpose minimization

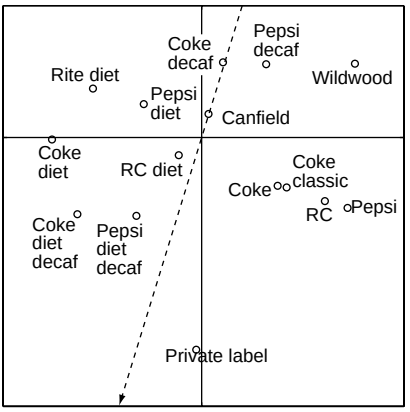


FIGURE 23.7. The hill-climbing model fitted to the brand switching data of Table 23.1 after being converted by the gravity model. The dotted line gives the direction of the slope up the hill.

function in MatLab to compute a solution. Figure 23.7 presents the results. The slope vector gives the uphill direction. The symmetric part of the data can be interpreted as usual. For example, there seems to be much switching between Coke, Coke classic, RC, and Pepsi because they are close together. To a lesser extent this also holds for the group of Coke decaf and Pepsi decaf, the group Coke diet, Pepsi diet and Rite diet, and the group of Coke diet decaf and Pepsi diet decaf.

To see how much skew-symmetry is present, the hill-climbing model predicts that changing from Private label to most other colas is easier than changing from those colas to Private label. The reason is that starting from Private label to any other cola it is downhill and the other way is uphill. We also see that there are several groups of colas whose difference vectors are almost orthogonal to the slope direction. Those groups lie at the same altitude on the hill and hardly display asymmetry in how often people change from one cola to the other or vice versa. One such group is Coke, Coke classic, RC, and Pespi. Other groups that are mostly symmetric consist of Coke decaf and Pepsi decaf, a group with Pepsi diet and Rite diet, and a group of Coke diet decaf and Pepsi diet decaf. In a similar way, more relations could be deferred from the hill-climbing representation in Figure 23.7.

The hill-climbing model resembles to some extent the *jet-stream* model proposed by Gower (1977). This model uses the metaphor of flying times taking the jet-stream into account. Using our notation, the distances in the jet-stream model are defined by

$$d_{ij}(\mathbf{X}, \mathbf{z}) = \frac{\|\mathbf{x}_i - \mathbf{x}_j\|}{1 + \frac{(\mathbf{x}_i - \mathbf{x}_j)'\mathbf{z}}{\|\mathbf{x}_i - \mathbf{x}_j\|}}. \tag{23.15}$$

The difference from the hill-climbing model is that the asymmetry factor appears in the jet-stream model in the denominator, whereas in the hill-climbing model it turns up as a separate term.

23.9 The Radius-Distance Model

Another model for fitting asymmetry directly is based on a representation of objects by circles with different radii (Okada & Imaizumi, 1987). The asymmetric dissimilarity δ_{ij} is modeled along the line connecting the centers i and j of two circles. The *radius-distance* from i to j is defined as the Euclidean distance d_{ij} between the centers of the circles, subtracting the starting radius of object i and adding the ending radius of object j. Thus, the radius-distance model can be fitted by minimizing

$$\begin{aligned} L(\mathbf{X},\mathbf{r}) &= \sum_{i=1}^{n}\sum_{j=1}^{n}[\delta_{ij} - (d_{ij}(\mathbf{X}) - r_i + r_j)]^2 \\ &= \|\boldsymbol{\Delta} - (\mathbf{D}(\mathbf{X}) - \mathbf{1}\mathbf{r}' + \mathbf{r}\mathbf{1}')\|^2, \end{aligned} \tag{23.16}$$

where $\mathbf{D}(\mathbf{X})$ has elements $d_{ij}(\mathbf{X})$ which refer to the usual Euclidean distance and $\mathbf{r}$ is a vector containing nonnegative radii r_i. As the hill-climbing model, the radius distance model always fits the diagonal elements by 0 because for the symmetric part $d_{ii}(\mathbf{X}) = 0$ and for the skew-symmetric part $r_i - r_i = 0$.

An algorithm to minimize $L(\mathbf{X},\mathbf{r})$ can be easily formulated by recognizing that the loss can be decomposed in a symmetric and skew-symmetric part (Bove & Critchley, 1993). This property means that $L(\mathbf{X},\mathbf{r})$ may be written as

$$\begin{aligned} L(\mathbf{X},\mathbf{r}) &= \|(\boldsymbol{\Delta} + \boldsymbol{\Delta}')/2 - \mathbf{D}(\mathbf{X})\|^2 \\ &\quad + \|(\boldsymbol{\Delta} - \boldsymbol{\Delta}')/2 - (\mathbf{r}\mathbf{1}' - \mathbf{1}\mathbf{r}')\|^2. \end{aligned} \tag{23.17}$$

Therefore, the symmetric part can be fitted by a regular MDS on $(\boldsymbol{\Delta} + \boldsymbol{\Delta}')/2$. The solution for the skew-symmetric part $\mathbf{N} = (\boldsymbol{\Delta} - \boldsymbol{\Delta}')/2$ requires a bit more care because of the nonnegativity constraints on the radii r_i. Some rewriting allows the skew-symmetric term of (23.17) to be expressed as

$$\|\mathbf{N} - (\mathbf{r}\mathbf{1}' - \mathbf{1}\mathbf{r}')\|^2 = \text{tr}\ \mathbf{N}'\mathbf{N} + 2n\mathbf{r}'\mathbf{J}\mathbf{r} - 4\mathbf{r}'\mathbf{J}\mathbf{N}\mathbf{1}, \tag{23.18}$$

where $\mathbf{N} = (\boldsymbol{\Delta} - \boldsymbol{\Delta}')/2$ and $\mathbf{J} = \mathbf{I} - n^{-1}\mathbf{1}\mathbf{1}'$ is the centering matrix. There is a simple analytic solution to (23.18). In Section 23.2, the unconstrained minimizer for (23.18) was given as $\mathbf{r}_u = n^{-1}\mathbf{J}\mathbf{N}\mathbf{1}$. Note that the centering matrix $\mathbf{J}$ can be left out because $\mathbf{N}\mathbf{1}$ has column sum zero due to the skew-symmetry of $\mathbf{N}$. It is clear that $\mathbf{r}_u$ does not satisfy the restriction that all

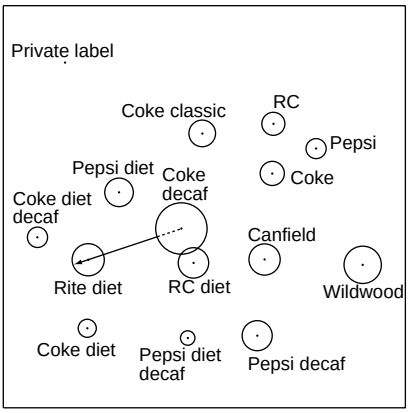


FIGURE 23.8. The radius-distance model of Okada and Imaizumi (1987) fitted to the brand switching data of Table 23.1 after being converted by the gravity model.

$r_i \geq 0$. However, any $\mathbf{r}_c = \mathbf{r}_u + c\mathbf{1}$ with a $c \geq \min_i r_i$ will be a feasible solution. The reason is that adding a constant does not change (23.18), because $\mathbf{r}$ is premultiplied by $\mathbf{J}$ and the nonnegativity constraints will be satisfied. Therefore, we choose $c = \min_i r_i$ so that the smallest radius equals zero.

Figure 23.8 shows the results of the distance radius model to these data. The symmetric relations can be easily interpreted by considering the centers of the circles. Large distances indicate little mutual switching whereas colas at close distance imply more mutual switching. The asymmetric part is taken care of by the differences in circle sizes. The fitted distance going from Coke decaf to Rite diet is indicated by the arrow in Figure 23.8. Going the other way around, from Rite diet to Coke decaf, the distance is computed from the border of the Rite diet circle to the far end of the Coke decaf circle. Because the circle of Coke decaf is larger than the circle of Rite diet, the distance Rite diet to Coke decaf is larger than the distance Coke decaf to Rite diet. This indicates that more households are changing from Coke decaf to Rite diet than vice versa. In a similar way, all the relations can be interpreted.

A nonmetric version of the radius-distance model was proposed by Okada and Imaizumi (1987) who included a gradient-based algorithm. However, the algorithmic approach outlined here can still be followed. First, replace the δ_{ij} by $\widehat{d}_{ij}$ as was done in Chapter 9 when going from metric to nonmetric MDS. Then alternatingly update one set of parameters while keeping the other fixed. Thus, given $\mathbf{X}$ and $\mathbf{r}$, update $\widehat{\mathbf{d}}$. Normalize $\widehat{\mathbf{d}}'\widehat{\mathbf{d}}$ to n^2 so that the trivial solution of $\widehat{\mathbf{d}} = \mathbf{0}$, $\mathbf{X} = \mathbf{0}$, and $\mathbf{r} = \mathbf{0}$ is avoided. Next, update $\mathbf{X}$ and $\mathbf{r}$ given $\widehat{\mathbf{d}}$. The update of $\mathbf{X}$ is given by (8.29) where $(\widehat{\mathbf{D}} + \widehat{\mathbf{D}}')/2$ should used instead of the dissimilarities. The update of $\mathbf{r}$ can be obtained by $\mathbf{r}_c$ discussed above. Remain iterating until convergence is obtained.

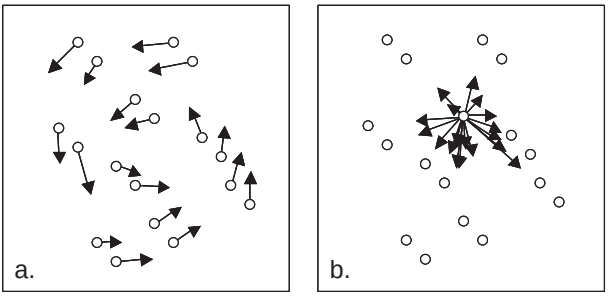


FIGURE 23.9. A circular vector field (panel a) and a configuration with a dominant point that "feeds" into all other points (panel b).

Okada (1990) discusses how the radius-distance model can be extended to ellipsoids instead of circles. This relaxation has the advantage that for each dimension and each object the radius can be different thereby allowing to estimate the skew-symmetric component in a better way. Okada (1990) also presents a gradient based algorithm.

23.10 Using Asymmetry Models

The MDS user should be aware of the simple but important fact that any asymmetry model is, as all models are, designed for a particular purpose only. Each such model represents one particular form of asymmetry only. It helps to detect only those patterns in the data on which it focuses. If the chosen model does not show this particular form of asymmetry for the given data, it does not imply that there are no other systematic asymmetries in the data. Consider, for example, the slide-vector model. It is made to show to what extent the data contain a general asymmetry in one direction of the space. As we can see in Figure 23.6, this form of asymmetry can be rather small. However, the data may contain other interesting asymmetries. Figure 23.9 shows two particular forms of asymmetries that would go unnoticed by the slide-vector model: the circular vector field in panel A and the configuration with one "dominant" element in panel B. (Neither of these cases is inconceivable for real data.) The circular field would be detected by the drift-vector model, for example, but the case in panel B would not lead to a (resultant) drift vector that adequately describes the asymmetries. Rather, in this case, it would be more revealing to show all the drift vectors attached to this one particular point, provided that the asymmetries can be considered big enough (relative to the symmetric part of the data) and, of course, reliable enough to warrant further studies.

When analyzing asymmetries, the user should experiment with different models to avoid missing systematic patterns that exist in the data. Models that identify the extent of one global and linear trend, in particular,

should be complemented with models that represent the more fine-grained asymmetries. To use such a hierarchy of models also allows one to assess whether it is worth it, relative to the quality of the data, to pursue models with many free fitting parameters.

23.11 Overview

In this section, we give a summary of the models for asymmetry and skew-symmetry discussed in this chapter. The reader should know that the discussion in this chapter is not exhaustive. Other models for asymmetry exist in the literature, some aimed at specific applications. In this chapter, we have restricted ourselves to mostly distance-based models for asymmetry or skew-symmetry.

Many of the models for asymmetry can be decomposed into a symmetric part and a skew-symmetric part. Some models only estimate the skew-symmetric part. Others fit the asymmetric data directly. Table 23.3 gives an overview of the models discussed in this chapter. Analyzing the skew-symmetric component separately from the symmetric part has the advantage that for the interpretation one only cares about the skew-symmetry. On the other hand, it may be useful to see the symmetric and skew-symmetric relations simultaneously. An important issue in deciding for an asymmetry model is the way of representing the asymmetry and how easy it is to interpret it. The latter remains a subjective matter.

Throughout this chapter, we only discussed the analysis of two-way asymmetric data. For three-way data, several models have been proposed in the literature. For example, Okada and Imaizumi (1997) extend the radius-distance model to the case of replications of two-way asymmetry data. De Rooij and Heiser (2000) extend distance measures to deal with the case of one-mode three-way asymmetric data.

23.12 Exercises

Exercise 23.1 Consider the data in the table below. They represent preceding–following contingencies for certain types of threat display behaviors shown by a common bird, the great tit (Blurton Jones, 1968). The numbers correspond to the proportion of times that the behavior in column j followed the behavior in row j. For example, feeding follows fluffing 3% of the time. Spence (1978) argues that these data are "a measure of how 'close' behavior j is to behavior i" and uses MDS to "visually detect" possible groupings of behaviors. The asymmetry of the data is noticed by Spence, but not studied.

TABLE 23.3. Summary of the properties of the models for asymmetric data discussed in this chapter. A + or a − in the columns **P**, **M**, and **N** indicate whether the model fits asymmetric proximities directly (column **P**), the symmetric part separately (column **M**), and the skew-symmetric part separately (column **N**).

Section	Model	**P**	**M**	**N**	Graphical Representation
23.3	Signed-distance model	–	–	+	Signed distances between points on line
23.3	Gower decomposition	–	–	+	Areas between vectors plus orientation
23.4	Distance model for skew-symmetry	–	–	+	Distance between points plus orientation
23.5	Scaling the skew-symmetry	–	+	+	Symmetry by distance between points, skew-symmetry by a summary vector
23.6	Unfolding	+	–	–	Distances between row and column objects
23.7	Slide-vector model	+	–	–	As unfolding, but row and column points are equal up to a translation
23.8	Hill-climbing model	+	–	–	Symmetry by distance between points, skew-symmetry modeled by projection of difference vector onto the slope direction.
23.9	Radius-distance model	–	+	+	Distance between two points with the radius from the starting circle removed and the radius of the arriving circle added

	Type of Behavior	1	2	3	4	5	6	7	8	9	10	11	12	13
1	Attack	4	17	16	11	10	13	11	0	6	0	0	9	4
2	Head down	26	0	5	14	4	13	2	8	5	0	0	5	18
3	Horizontal	25	3	0	12	13	11	3	2	10	8	0	4	9
4	Head up	5	9	8	8	14	15	5	4	13	0	2	5	12
5	Wings out	22	13	10	5	2	10	2	7	7	0	0	2	19
6	Feeding	2	5	18	13	11	3	3	5	13	8	1	16	1
7	Incomplete feeding	4	10	15	4	4	13	7	22	0	0	12	8	0
8	Hopping around	1	10	0	4	2	4	46	0	3	6	11	11	3
9	Hopping away	0	4	6	9	5	1	8	4	1	6	31	15	10
10	Crest raising	0	0	0	6	7	3	0	11	17	1	30	13	12
11	Fluffing	0	4	5	6	3	3	0	23	13	35	0	6	3
12	Looking around	5	0	5	0	3	6	12	12	11	30	8	0	9
13	Hopping towards	5	25	12	8	21	4	2	2	2	7	5	6	0

(a) Assess, by matrix decomposition, just how asymmetric these data are.

(b) Use the symmetric portion of the data for a two-dimensional MDS analysis. Then, add the skew-symmetric portion as vectors to a few behaviors that are strongly asymmetric, and to a few others that are only mildly asymmetric.

(c) How would you interpret the symmetric portion of these data? (Hint: Blurton Jones speculates that behaviors within each "group" may have certain causal factors in common.)

(d) Scale these data by the slide vector model, using PROXSCAL.

Exercise 23.2 Consider the data matrix below (Coombs, 1964). It shows the frequencies with which an article that appeared in the journal shown as a row entry cites an article in the column journal.

Journal	AJP	JASP	JAP	JCPP	JCP	JEdP	JexP	Pka
Am. J. Psy.	119	8	4	21	0	1	85	2
J. Abnorm. Soc. Psy.	32	510	16	11	73	9	119	4
J. Applied Psy.	2	8	84	1	7	8	16	10
J. Comp. Physiol. Psy.	35	8	0	533	0	1	126	1
J. Consulting Psy.	6	116	11	1	225	7	12	7
J. Educ. Psy.	4	9	7	0	3	52	27	5
J. Exp. Psy.	125	19	6	70	0	0	586	15
Psychometrika	2	5	5	0	13	2	13	58

To study the interaction behavior of these journals, we may follow Coombs, Dawes, and Tversky (1970) by first subtracting the column and the row means from the matrix entries. This leaves pure interaction values. Then, proceed as follows.

(a) Split the matrix of interaction values into its symmetric and skew-symmetric component.

(b) Scale the symmetric part via MDS. Interpret the solution.

(c) Attach drift vectors to the points of the MDS configuration by hand or by using an appropriate graphics package (see, e.g., Borg & Groenen, 1995).

(d) How do you interpret these drift vectors?

Exercise 23.3 Consider Table 23.2 on p. 505 with the asymmetric dissimilarities obtained from the brand switching between 15 colas.

(a) Compute the skew-symmetric matrix of this table.

(b) Compute the unidimensional skew-symmetry model (23.5). Plot the results on a line. How do you interpret this solution?

(c) Apply Gower's decomposition to these data. Plot the first bimension. Interpret the solution.

(d) Which model do you expect to recover the skew-symmetry the best? Why do you think so?

(e) Compute for both models how much of the sum of squared skew-symmetry is recovered by the unidimensional skew-symmetry model and by Gower's decomposition using the first bimension. Does your computation coincide with your expectations?

(f) Do the two models differ in their interpretation? If so, how?

24
Methods Related to MDS

In this chapter, two other techniques are discussed that have something in common with MDS. First, we discuss the analysis of a variables-by-objects data matrix by principal components analysis and show how it is related to MDS. Then, we discuss correspondence analysis, a technique particularly suited for the analysis of a contingency table of two categorical variables.

24.1 Principal Component Analysis

Principal component analysis (PCA) is a technique that goes back to Pearson (1901) and Hotelling (1933). It begins with a data matrix of n cases (often: persons) and k variables (often: items, tasks). The objective of the method is to explain the k variables by a much smaller set of m "new" variables that are linear combinations of the original variables. Thus, *new variable* $i = w_1 \cdot (\textit{variable } 1) + w_2 \cdot (\textit{variable } 2) + \cdots + w_k \cdot (\textit{variable } k)$, where the weights, w_j, are the unknowns. The hypothesis is that only a few ($m \ll k$) of these new variables suffice to explain most of the variance of the data. For example, in intelligence testing, the testees are typically asked to work through test batteries with many items. One assumes, however, that not every item requires a special ability to solve it. Rather, only a few abilities should be needed, and each item requires a different mixture of these abilities. Somewhat more formally, one thus wants to (a) find these underlying mixtures of more general components, and then (b) assign each case a score on them. For example, a test battery of an intelligence test may

require essentially only verbal and numerical reasoning (the components), and each testee is assigned a score on these components on the basis of his or her test results. The components are, of course, not identified directly: rather, PCA shows which variables combine with high weights to form one particular component, and then one has to infer from the content of these variables what the component means. This approach is similar to interpreting dimensions in MDS on the basis of the points that have the longest projections onto these dimensions.

Consider an example. Assume that $\mathbf{M}$ is the usual person-by-variable data matrix. We begin by standardizing $\mathbf{M}$ so that its columns all sum to zero and have norms equal to 1. This leads to matrix $\mathbf{Z}$; that is,

$$\mathbf{M} = \begin{bmatrix} 8 & 9 & 1 \\ 5 & 5 & 5 \\ 4 & 4 & 5 \\ 8 & 7 & 2 \\ 7 & 1 & 4 \\ 4 & 5 & 7 \\ 5 & 3 & 6 \\ 2 & 6 & 8 \end{bmatrix} \quad \rightarrow \quad \mathbf{Z} = \begin{bmatrix} .46 & .62 & -.60 \\ -.07 & .00 & .04 \\ -.24 & -.15 & .04 \\ .46 & .31 & -.44 \\ .29 & -.62 & -.12 \\ -.24 & .00 & .36 \\ -.07 & -.31 & .20 \\ -.60 & .15 & .52 \end{bmatrix}. \tag{24.1}$$

To see what PCA does *geometrically*, we plot in Figure 24.1a the persons (=rows) of $\mathbf{Z}$ as points in a 3D space. The axes are formed by the three variables (=columns) of $\mathbf{Z}$. If we rotate the axes, the variance of the projections of the points on the rotated axes will change in general. We know from Section 7.10 that there exists one particular rotation to *principal axes*. These axes are characterized by the property that they are closest to the points or, expressed differently, that the projections of all points onto the principal axes have maximal length, axis by axis in decreasing order. The principal axes give us what we are looking for: the coordinates of the points on the principal axes are the principal components. The variance of the elements of the first principal component (denoted by $\mathbf{k}_1$) is maximal. The second principal axis gives rise to the second PC, $\mathbf{k}_2$, and the third principal axis to the last PC, $\mathbf{k}_3$. Note that each principal axis $\mathbf{k}_a$ may be reflected without changing the variance of the corresponding PC. Thus, any PCA solution is unique up to reflections of its components.

To see how PCA works computationally, consider the (full rank) singular value decomposition $\mathbf{Z} = \mathbf{P}\mathbf{\Lambda}\mathbf{Q}'$; that is,

$$\mathbf{Z} = \begin{bmatrix} -.64 & .32 & -.10 \\ .05 & .03 & -.08 \\ .16 & -.05 & -.80 \\ -.49 & .04 & .22 \\ -.02 & -.75 & .03 \\ .27 & .16 & .48 \\ .20 & -.25 & .25 \\ .46 & .49 & -.00 \end{bmatrix} \begin{bmatrix} 1.46 & 0 & 0 \\ 0 & .92 & 0 \\ 0 & 0 & .20 \end{bmatrix} \begin{bmatrix} -.64 & -.38 & .67 \\ -.38 & .91 & .16 \\ .67 & .15 & .72 \end{bmatrix}, \tag{24.2}$$

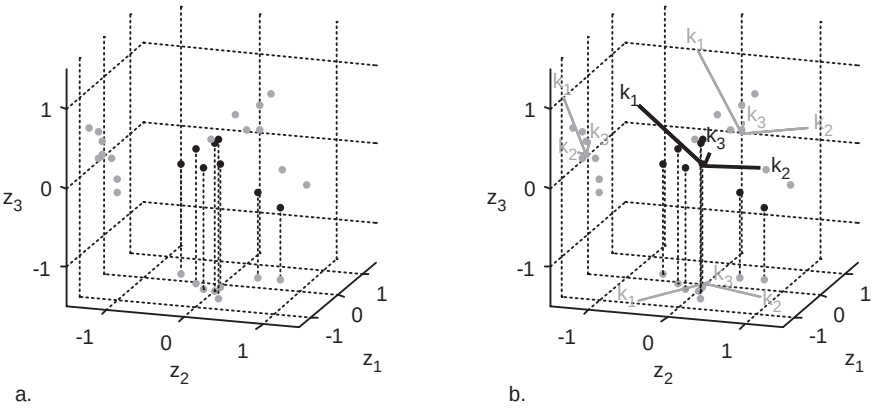


FIGURE 24.1. Plot of persons of $\mathbf{Z}$ in space spanned by variables of $\mathbf{Z}$ (panel a); same space with principal axes shown as tripod $\mathbf{K}$ (panel b).

where $\mathbf{P}$ contains the *standardized* principal components (PCs). The PCs are orthogonal to each other, because $\mathbf{P}'\mathbf{P} = \mathbf{I}$ in any singular value decomposition. The columns of $\mathbf{K} = \mathbf{P}\mathbf{\Lambda}$ are the unstandardized principal components. Figure 24.1b shows the principal axes that generate these PCs—$\mathbf{k}_1, \mathbf{k}_2$, and $\mathbf{k}_3$—in the space of the original variables, the columns of $\mathbf{Z}$. The PCs are related to the original $\mathbf{Z}$ by a rotation/reflection, $\mathbf{K} = \mathbf{Z}\mathbf{Q}$, because $\mathbf{Q}'\mathbf{Q} = \mathbf{Q}\mathbf{Q}' = \mathbf{I}$ in any singular value decomposition.

We can also directly look at the space spanned by the principal axes, where the elements of $\mathbf{K}$ are the coordinates of points that represent the persons. This view is shown in Figure 24.2a, where the tripod of $\mathbf{z}_1, \mathbf{z}_2$, and $\mathbf{z}_3$ indicates how the original variables are oriented in this principal axes space.

The norms of the principal components $\mathbf{k}_1, \mathbf{k}_2$, and $\mathbf{k}_3$ are equal to the respective singular values λ_a on the diagonal of $\mathbf{\Lambda}$. The squared singular values indicate how much variance is accounted for by the various principal components. In our small example, we see that the third PC is very small so that the various person points are almost all located at the same height on the third dimension of Figure 24.2a. Expressed algebraically, the data matrix $\mathbf{Z}$ is decomposed into a sum of matrices each with rank 1, $\lambda_1\mathbf{p}_1\mathbf{q}_1' + \lambda_2\mathbf{p}_2\mathbf{q}_2' + \lambda_3\mathbf{p}_3\mathbf{q}_3'$ (with $\mathbf{q}_a$ column a of $\mathbf{Q}$), so that the first $k < 3$ terms are the best approximation of $\mathbf{Z}$ by a matrix of lower rank k. The singular value λ_a is the weight of the information in the term $\lambda_a\mathbf{p}_a\mathbf{q}_a'$ (see Section 7.6, item 4).

Standardizing $\mathbf{K}$ amounts to adjusting the components $\mathbf{k}_1, \mathbf{k}_2$, and $\mathbf{k}_3$ to length one by dividing each column of $\mathbf{K}$ by the corresponding λ_a. That is, $\mathbf{P} = \mathbf{K}\mathbf{\Lambda}^{-1}$. Geometrically, this operation means that the configuration is stretched or compressed along the axes of Figure 24.2a. The result of this transformation is shown in Figure 24.2b.

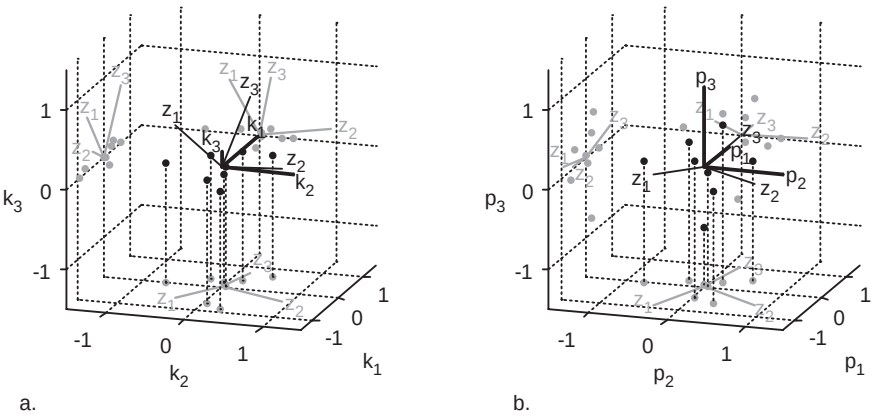


FIGURE 24.2. Persons as points in space that have a principal axes orientation; the axes that correspond to the original variables are shown as the tripod **Z** (panel a). Panel (b) shows same as panel (a), except that space is spanned here by the standardized principal components.

Figure 24.2b can be obtained from the original data as follows. Start with the original row-points configuration **Z**, rotate it to principal axes orientation by **Q**, and then stretch or compress the configuration along the principal axes by the weights in $\mathbf{\Lambda}^{-1}$. Algebraically, this corresponds to computing $\mathbf{P} = \mathbf{ZQ\Lambda}^{-1}$, where the columns of **P** are obviously weighted sums of **Z**'s columns, as intended.

The matrices **Q** and $\mathbf{\Lambda}$ can also be found from an eigendecomposition of the intercorrelation matrix of the original variables, $\mathbf{R} = \mathbf{Z}'\mathbf{Z}$, because $\mathbf{R} = \mathbf{Q\Lambda P}'\mathbf{P\Lambda Q}' = \mathbf{Q\Lambda}^2\mathbf{Q}'$. Thus, the eigenvalues of **R** are equal to the squared singular values of **Z**. A graphical representation of the first two principal axes for our small example is given in Figure 24.3.

Once the components **P** are found, one can reverse the perspective and ask how they explain the original variables. Assuming here that the decomposition has full rank, we can simply reverse the equation $\mathbf{P} = \mathbf{ZQ\Lambda}^{-1}$ to get **Z** from **P** via $\mathbf{Z} = \mathbf{PL}'$ with $\mathbf{L} = \mathbf{Q\Lambda}$. The coefficients in **L** are called *component loadings* and can be interpreted as the correlations between the variables and the components. This property can be seen as follows. The correlations between the variables (columns) of **Z** and **P** are $\mathbf{Z}'\mathbf{P}$, because both **Z** and **P** are standardized. Thus, we get $\mathbf{Z}'\mathbf{P} = \mathbf{Z}'\mathbf{ZQ\Lambda^{-1}} = \mathbf{RQ\Lambda^{-1}} = \mathbf{Q\Lambda}^2\mathbf{Q}'\mathbf{Q\Lambda}^{-1} = \mathbf{Q\Lambda}$. This yields for the example above

$$\mathbf{L} = \begin{bmatrix} -.93 & -.35 & .13 \\ -.55 & .84 & .03 \\ .98 & .14 & .14 \end{bmatrix}.$$

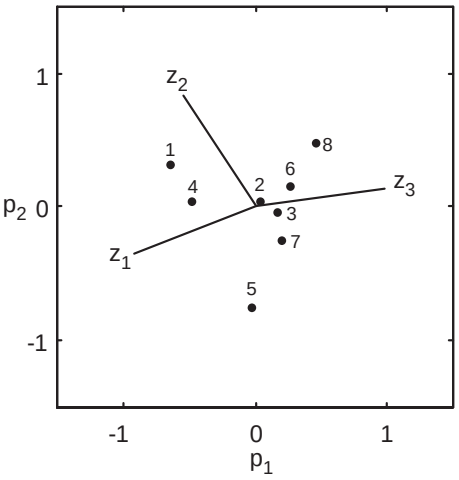


FIGURE 24.3. Persons (labeled by row numbers) in the space of the first two standardized principal components, together with projections of the original variables $\mathbf{z}_1, \mathbf{z}_2$, and $\mathbf{z}_3$ onto this space.

$\mathbf{L}$ shows that the column vector 1 of the data matrix correlates with the first PC with -0.93. It is therefore almost fully explained by this PC. The second variable of the data matrix correlates with the first PC with -0.55, and the third variable with 0.98. Overall, the component loadings make clear that the three variables of our data matrix are essentially only two-dimensional (as Figure 24.2a shows graphically). They correlate most with the first PC, and almost not at all with the third PC.

The loadings can also be interpreted geometrically as the lengths of the projections of the vectors that represent the variables onto the standardized PCs $\mathbf{P}$. The squares of the elements of the component loadings in $\mathbf{L}$ are a measure of fit for the variables (see Table 24.1). The sum of the squared component loadings for dimension a is equal to the eigenvalue λ_a^2. Because the sum-of-squares of the loadings in matrix $\mathbf{L}$ above is 1 in each row, each variable is fully accounted for in the 3D space spanned by the PCs and for 98.7% in 2D.

The simultaneous representation of objects and variables in one plot as in Figure 24.3 is called a *biplot* (Gabriel, 1971). The term *bi* in biplot refers to the representation of the two modes (the objects and the variables) in one plot but not to the dimensionality, although the plots are usually made in two dimensions. The two sets of points, the object points that correspond to the rows of $\mathbf{P}$ and the variables whose coordinates are the component loadings $\mathbf{L} = \mathbf{Q}\mathbf{\Lambda}$, are related as scalar products. This means that we can only interpret the projection of object points on the vector that represents a variable (similar to Figure 16.3), not the distance between an object point and the variable-vector. This projection predicts the value of the object on

TABLE 24.1. Squared component loadings of the example data in (24.1). The last row contains the proportion of variance accounted for (VAF).

	Dimension			Total	
Variable	1	2	3	$1+2$	$1+2+3$
1	.860	.122	.018	.982	1.000
2	.299	.700	.001	.999	1.000
3	.961	.019	.021	.980	1.000
λ_a^2	2.120	.841	.040	2.961	3.000
VAF	.707	.280	.013	.987	1.000

the variable. For more details and some examples of biplots, we refer to Gabriel (1971), Gower and Hand (1996), and Gifi (1990).

A Typical Application of PCA

In many applications, only the structure of the variables is of interest. Then, PCA becomes quite similar to (metric) MDS, because it then reduces to the question of analyzing the structure of a correlation matrix. As an illustration, consider the correlation matrix in Table 5.1. Rather than taking these numbers as similarities and attempting to represent them by distances in a Euclidean space, in PCA we look at the correlations as scalar products. An optimal solution for a PCA representation is easy to find, as was shown above. The loadings of the intelligence test items of Table 5.1 are exhibited in Table 24.2. Overall, these eight variables have a total variance of 8 (geometrically expressed: a total length of 8). Hence, for example, the first three PCs account for $\lambda_1^2 + \lambda_2^2 + \lambda_3^2 = 3.37 + 1.35 + 1.05 = 5.77$ or $(5.77/8) \cdot 100 = 72\%$ of the variance. This follows from the spectral decomposition theorem [see (7.11)], and the convention to norm the eigenvectors to length 1. Note also that the ath PC accounts for a maximum of the variance of the original variables that has not been explained already by the PCs $1, \ldots, a-1$.

Geometrically, we see that the configuration of the variables in the space spanned by the first three PCs, as shown in Figure 24.4, is similar to Figure 5.1. Both exhibit a circular arrangement of the points and vector endpoints, respectively. The PCA representation, however, is higher-dimensional. The (ordinal) MDS representation of Figure 5.1 essentially corresponds to a plane that captures the vector endpoints in Figure 24.4, because in MDS it is the distance of the vectors' endpoints that we want to represent, not the angles that the vectors subtend.

MDS and PCA (in the sense of metrically analyzing a correlation matrix) are, therefore, closely related. However, one cannot always expect similar results. PCA not only leads to higher-dimensional representation spaces than MDS. PCA is also almost always done metrically, whereas most MDS

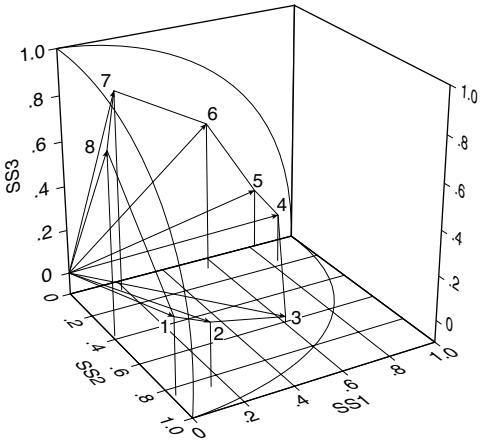


FIGURE 24.4. 3D principal component representation of correlations in Table 5.1, rotated to simple structure.

TABLE 24.2. Loading of variables in Table 5.1 on principal components (PC1, ..., PC8) and on dimensions rotated to simple structure in the space spanned by first three PCs (SS1, SS2, SS3).

Test	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	SS1	SS2	SS3
1	0.63	-0.59	0.15	-0.14	-0.22	-0.02	-0.22	0.33	0.02	0.85	0.24
2	0.69	-0.50	0.24	-0.13	-0.11	-0.04	0.06	-0.41	0.15	0.86	0.18
3	0.70	0.02	0.50	0.06	0.35	0.15	0.31	0.15	0.60	0.61	-0.07
4	0.68	0.48	0.22	0.18	0.11	0.17	-0.42	-0.10	0.84	0.16	0.12
5	0.60	0.57	0.10	-0.20	-0.10	-0.49	0.06	0.04	0.82	0.00	0.18
6	0.70	0.30	-0.29	0.15	-0.44	0.27	0.21	0.03	0.56	0.08	0.58
7	0.57	0.02	-0.61	-0.46	0.28	0.13	-0.02	-0.01	0.18	0.07	0.81
8	0.59	-0.31	-0.46	0.50	0.18	-0.25	0.01	0.00	0.01	0.38	0.72
λ_a^2	3.37	1.35	1.05	0.59	0.51	0.45	0.37	0.31			
Explained variance (%)	42.1	16.9	13.1	7.4	6.4	5.59	4.7	3.9	26.4	25.1	20.7

applications are ordinal ones, in particular those in exploratory data analysis where one wants data representations that are as simple as possible. Moreover, the PCA solution is seldom studied geometrically. Rather, typically only the loadings of the vectors on the components are interpreted, similar to traditional dimension-oriented MDS. In our illustrative application, that means that one would interpret the values of the various tests on the rotated components SS1, SS2, and SS3 but not the circular manifold that we see in Figure 24.4.

Principal Coordinates Analysis

A closely related technique with the same algebraic results as PCA, called *principal coordinates analysis* (PCO), emphasizes the representation of the objects (Gower, 1966). Consider the rows of the data matrix $\mathbf{Z}$ with k variables as points in the k-dimensional space. The aim is to approximate the distances $d_{ij}(\mathbf{Z})$ in a low-dimensional $m < k$ space $\mathbf{X}$. If this is done with classical scaling, then we have to do the following computations. First, compute the matrix of squared distances $\mathbf{D}^{(2)}(\mathbf{Z}) = \mathbf{1c}' + \mathbf{c1}' - 2\mathbf{ZZ}'$, with $\mathbf{c}$ the vector of the diagonal elements of $\mathbf{ZZ}'$; see (7.5). Then pre- and postmultiply $\mathbf{D}^{(2)}(\mathbf{Z})$ with the centering matrix $\mathbf{J}$ and multiply the result with $-\frac{1}{2}$. These operations lead to

$$\begin{aligned} -\tfrac{1}{2}\mathbf{J}\mathbf{D}^{(2)}(\mathbf{Z})\mathbf{J} &= -\tfrac{1}{2}\mathbf{J}(\mathbf{1c}' + \mathbf{c1}' - 2\mathbf{ZZ}')\mathbf{J} \\ &= -\tfrac{1}{2}\mathbf{J}(-2\mathbf{ZZ}')\mathbf{J} = \mathbf{ZZ}'. \end{aligned}$$

Then, the eigendecomposition of $\mathbf{ZZ}' = \mathbf{P}\mathbf{\Lambda}^2\mathbf{P}'$ is computed. The configuration $\mathbf{X}$ for the object points obtained by classical scaling equals the first m columns of $\mathbf{P}\mathbf{\Lambda}$. The configuration obtained from PCO is exactly the same as $\mathbf{K}$ obtained by PCA. Thus, using the normalization $\mathbf{P}\mathbf{\Lambda}$, this equivalence shows that PCA may be seen as MDS that tries to reconstruct distances in a high-dimensional space by a low-dimensional representation.

Of course, instead of using the classical MDS criterion, the high-dimensional distances can also be approximated by using the Stress function in MDS. This approach has been advocated by Meulman (1986, 1992) and is called distance-based PCA. It turns out that the Stress values at a minimum can also be interpreted as a ratio of variances, similar to PCA (Groenen & Meulman, 2004).

24.2 Correspondence Analysis

Correspondence analysis (CA) can be seen as an equivalent of PCA on a contingency table of two categorical variables. In such a table, every entry gives the frequency of each combination of categories of the two variables. The objective of CA is to show the interaction in this table graphically.

TABLE 24.3. A hypothetical contingency table of the distributions of seats by country and political faction (Groenen & Gifi, 1989).

	Political Faction			
Country	Christian Democrats	Socialists	Other	Total
Belgium	8	9	7	24
Germany	39	30	6	75
Italy	25	11	39	75
Luxembourg	3	2	1	6
The Netherlands	13	10	2	25
Total	88	62	55	205

Consider the following hypothetical example. Assume we are interested in the political similarity of some European countries. One set of data that speaks to this issue is the distribution of the seats of these countries in the European Parliament over the political factions. Let Table 24.3 be the hypothetical contingency table of sets for five countries and three political factions. Figure 24.5 shows the result of the correspondence analysis of Table 24.3. In the figure, both the row points (the countries) and the column points (political factions) are plotted. The distance between row points is a particular form of similarity of the countries. For example, The Netherlands and Germany have the same relative distribution of seats over the political factions (see Table 24.4). That is, they have the same data "profile" (Greenacre, 1984, p. 55). Zero distances in CA always occur for profiles that are exactly the same. The properties of these two countries are similar to the profile of Luxembourg and thus are located close to each other but not at zero distance. The centroid can be interpreted as the average country, so that the closer a country is located towards the centroid, the more similar the country is to the average country. Italy and Belgium differ from the other countries because they are not located close together. Note that the scatter of the country points is almost exclusively along the first dimension, indicating that the second dimension is of minor importance. The distance between column points along each axis can be interpreted in a similar way, but the distance between country points and party points has to be interpreted with some care. We return to this later when discussing the example at the end of this section.

Although CA is often applied to contingency tables, the method can in principle be used on any rectangular table with nonnegative similarity values. For example, CA can be used on preference rankings and could be used as an alternative to unfolding. (If used this way, the entries in the table should be similarities, though.)

CA is known under different names, such as reciprocal averaging, dual scaling, canonical correlation analysis (applied to qualitative data), and simultaneous regression, because it has been discovered independently in dif-

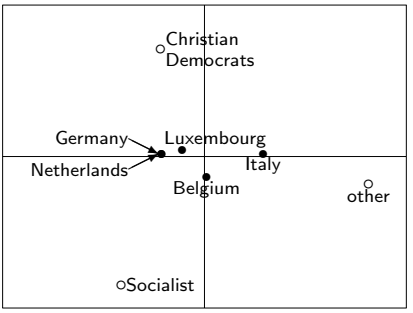


FIGURE 24.5. The correspondence analysis solution of Table 24.3. Note that the points for Germany and the Netherlands are located on top of each other.

TABLE 24.4. Row profiles of Table 24.3.

	Political Faction			
Country	Christian Democrats	Socialists	Other	Total
Belgium	.333	.375	.292	1
Germany	.520	.400	.080	1
Italy	.333	.147	.520	1
Luxembourg	.500	.333	.167	1
The Netherlands	.520	.400	.080	1
Mean row profile	.429	.302	.268	

ferent areas (Hotelling, 1933; Richardson & Kuder, 1933; Hirschfeld, 1935). Guttman (1941) presented a comprehensive treatment of the algebra of CA. The graphical and geometric emphasis in CA has been largely due to Benzécri et al. (1973), a book that also contains a historical overview.[1] There is a wide literature on CA, and standard textbooks are: Nishisato (1980, 1994), Lebart, Morineau, and Warwick (1984), Greenacre (1984, 1994), and Gifi (1990). Developments on CA can be found in Greenacre and Blasius (1994) and Blasius and Greenacre (1998). For a discussion of the relation of CA with MDS, we refer to Heiser and Meulman (1983a). Groenen and Van de Velden (2004) discuss the inverse CA problem, that is, given a CA solution what data sets would have produced the same CA solution.

The remainder of this section is organized as follows. First, we consider the geometry of CA following the example of Groenen and Gifi (1989), also discussed in SPSS (1990). Then, it is shown how the CA solution can be computed. Also, several algebraic properties of CA are discussed such as the inertia, the contribution of a point to the inertia of a dimension, and the proportion of total distance of a point shown in a dimension. Next, we apply CA to crime rates in 10 US states. Finally, we end with some remarks on the relation of CA and MDS.

Geometry of Correspondence Analysis

To measure the similarity between two countries, correspondence analysis uses (row) profiles normed to sum to one in each row. For example, the Christian Democrats occupy 52% (39/75 = .520) of Germany's seats in the European Parliament. Table 24.4 contains the row profiles of Table 24.3. From the row profiles, we see that the Netherlands and Germany have the same relative distribution of seats over the factions, irrespective of their difference in the total number of seats. Now, we discuss how to reconstruct geometrically the CA solution of Figure 24.5 in three steps.

1. Consider Table 24.4 as coordinates in a 3D space (Figure 24.6). The mean row profile is represented as the centroid. Because the profiles sum to one, all of the points lie in the 2D subspace spanned by the points representing the political factions: point (1, 0, 0) for Christian Democrats, point (0, 1, 0) for Socialist, and point (0, 0, 1) for Other. This 2D triangle is shown in Figure 24.7.

2. The next step in correspondence analysis is to assign weights to the dimensions. Let $\mathbf{F} = (f_{ij})$ be the contingency table, such as Table 24.3. In CA, a weighted Euclidean distance is used, where the dimen-

[1] Other historical overviews can be found in Nishisato (1980), Van Rijckevoorsel and Tijssen (1987), Van Rijckevoorsel (1987), and Gifi (1990).

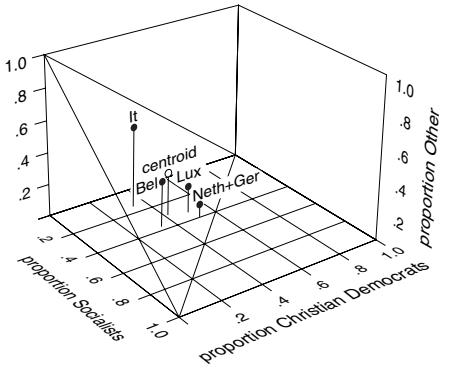


FIGURE 24.6. 3D representation of the row profiles from Table 24.4.

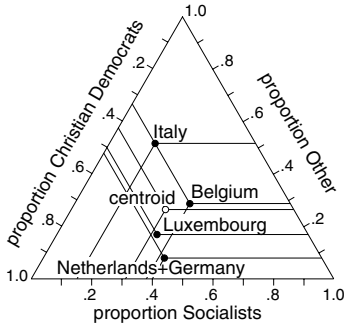


FIGURE 24.7. 2D representation of the row profiles from Table 24.4.

sion weights are equal to $(\sum_j f_{ij}/\sum_{ij} f_{ij})^{-1/2}$, that is, the inverse of the square root of the column means of Table 24.4. In CA, the columns with small means are considered to be more discriminating than the columns with large means. Hence, the weight for column 1 is $1/\sqrt{.429} = 1.527$, for column 2 it is $1/\sqrt{.302} = 1.820$, and for column 3 it is $\sqrt{.268} = 1.932$. The weighted configuration is shown in Figure 24.8. This configuration is the same as the solution obtained by CA in Figure 24.5, apart from the rotation.

3. The final step is to rotate to principal axes such that maximum variance is shown in the first dimension, the second dimension maximizing the remaining variance, and so on.

These three steps show geometrically how a correspondence analysis solution is obtained. The emphasis in these steps was on the row points. The role of the rows and columns can be reversed by simply transposing the correspondence table. Next, we discuss some of the algebraic properties of correspondence analysis.

Algebraic Properties

The weighted Euclidean distance used in CA has a close relation with the χ^2-statistic and so-called χ^2-distances, provided the entries in the correspondence table are frequencies. Let $f_{i+} = \sum_j f_{ij}$ be the row sum of $\mathbf{F}$, $f_{+j} = \sum_i f_{ij}$ the column sum, and $n = \sum_{ij} f_{ij}$ the total sum. The weighted Euclidean distance of row profiles k and l (the distances between the points in Figure 24.8) is given by

$$d_{kl} = \left(\sum_j \frac{(f_{kj}/f_{k+} - f_{lj}/f_{l+})^2}{f_{+j}/n} \right)^{1/2}, \tag{24.3}$$

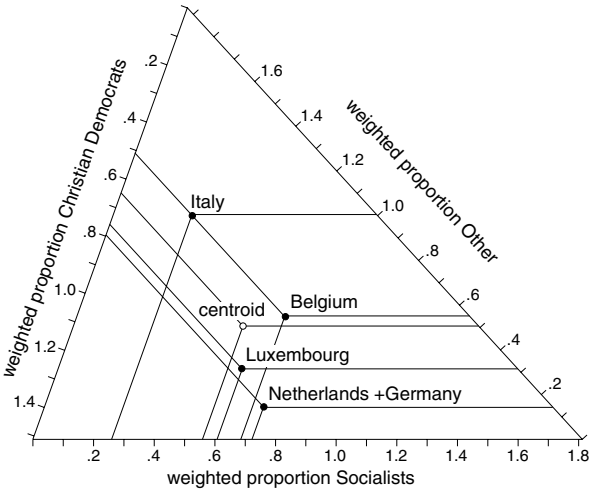


FIGURE 24.8. The weighted Euclidean space used by correspondence analysis.

and the weighted Euclidean distance of row profile k to the average profile z by

$$d_{kz} = \left(\sum_j \frac{(f_{kj}/f_{k+} - f_{+j}/n)^2}{f_{+j}/n} \right)^{1/2}. \tag{24.4}$$

These distances are called χ^2-*distances* because

$$\begin{aligned} \sum_i \frac{f_{i+}}{n} d_{iz}^2 &= \left(\sum_{i,j} \frac{(f_{i+}/n)(f_{ij}/f_{i+} - f_{+j}/n)^2}{f_{+j}/n} \right) \\ &= n^{-1} \left(\sum_{i,j} \frac{(f_{ij} - f_{i+}f_{+j}/n)^2}{f_{i+}f_{+j}/n} \right) = \frac{\chi^2}{n}. \end{aligned}$$

Thus, n times the weighted sum of the squared distances of the row points to their centroid (in full dimensionality) is equal to the χ^2-statistic. Expression (24.5) is called *total inertia*.

We continue discussing how the coordinates in correspondence analysis are obtained. Let $\mathbf{D}_r$ be the diagonal matrix of row marginals (with diagonal elements f_{i+}) and $\mathbf{D}_c$ the diagonal matrix of column marginals (with diagonal elements f_{+j}). Let matrix $\mathbf{E}$ be the matrix of expected values under the *independence model*, which has elements $e_{ij} = f_{i+}f_{+j}/n$. Then, correspondence analysis requires the singular value decomposition of

$$\mathbf{D}_r^{-1/2}(\mathbf{F} - \mathbf{E})\mathbf{D}_c^{-1/2} = \mathbf{P}\mathbf{\Phi}\mathbf{Q}', \tag{24.5}$$

with the usual properties $\mathbf{P}'\mathbf{P} = \mathbf{Q}'\mathbf{Q} = \mathbf{I}$ and $\mathbf{\Phi}$ the diagonal matrix of singular values. The rank of the decomposed matrix in (24.5) is at most $M = \min(\text{number of row points, number of column points}) - 1$. The row scores $\mathbf{R}$ and column scores $\mathbf{C}$ are given by

$$\mathbf{R} = n^{1/2}\mathbf{D}_r^{-1/2}\mathbf{P}\mathbf{\Phi} \quad \text{and} \quad \mathbf{C} = n^{1/2}\mathbf{D}_c^{-1/2}\mathbf{Q}. \tag{24.6}$$

This normalization implies $\mathbf{R}'\mathbf{D}_r\mathbf{R} = \mathbf{\Phi}^2$, $\mathbf{C}'\mathbf{D}_c\mathbf{C} = n\mathbf{I}$, and is called the *row principal* by SPSS (1990), because the squared singular values are the weighted sum of the squared row coordinates (after the principal coordinates normalization of Greenacre, 1984, p. 88). For a discussion of other normalizations, we refer to Greenacre (1984) and Gifi (1990).

Properties of this decomposition are:

- The weighted sum of the row scores (weights $\mathbf{D}_r$) and the weighted sum of the column scores (weights $\mathbf{D}_c$) are equal to zero. The origin is the average row (and column) profile.

- The term $\sum_a \phi_a^2$ is called the *inertia*. In our example, we have perfect fit, so that all of the inertia is shown in 2D. Inertia is related to the χ^2-statistic by $\chi^2/n = \sum_a \phi_a^2$. Therefore, the proportion of total inertia recovered in m dimensions equals $(\sum_{a=1}^m \phi_a^2)/(\chi^2/n)$.

- The contribution of row point i in recovering the inertia in dimension a is $(f_{i+}/n)r_{ia}^2/\phi_a^2$. For column points, this contribution is $(f_{+j}/n)c_{ja}^2$. The difference in formulas for row and column points stems from the row principal normalization that is used. These relative contributions are important to find those points that are important on dimension a.

- Another interesting measure is the proportion of the χ^2-distance of row i to the centroid that is represented by the coordinate in dimension a. This proportion is given by r_{ia}^2/d_{iz}^2 for the row objects, and $c_{ja}^2\phi_a^2/(\sum_l c_{jl}^2\phi_l^2)$ for the column objects.

- Using the normalization above, the row scores are the weighted centroid of the column scores, which is called the *barycentric principle* (Benzécri et al., 1973). The *transition formulas* allow the transformation of the column scores into row scores and the row scores into column scores by

$$\mathbf{R} = \mathbf{D}_r^{-1}\mathbf{F}\mathbf{C}, \tag{24.7}$$
$$\mathbf{C} = \mathbf{D}_c^{-1}\mathbf{F}'\mathbf{R}\mathbf{\Phi}^{-2}. \tag{24.8}$$

  Note that (24.7) computes the weighted centroid of the column points.

- Because (24.5) is a dimensionwise decomposition, the elements of $\mathbf{F}$ can be *reconstituted* in m dimensions by

$$\widehat{f}_{ij} = (f_{i+}f_{+j}/n)\left[1 + \sum_{a=1}^{m} r_{ia}c_{ja}\right]. \tag{24.9}$$

  If $m = M$ (full dimensionality), then (24.9) reconstitutes $\mathbf{F}$ perfectly.

- In our example, we saw that the Netherlands and Germany have the same row profile which gave yielding equal scores in CA. It turns out that CA also gives the same results if these two rows are aggregated. This principle is called *distributional equivalence* (Benzécri et al., 1973). For our example, this principle implies that the matrix with aggregated frequencies for the Netherlands and Germany,

$$\mathbf{F} = \begin{bmatrix} 8 & 9 & 7 \\ 52 & 40 & 8 \\ 25 & 11 & 39 \\ 3 & 2 & 1 \end{bmatrix},$$

  yields exactly the same correspondence analysis solution as the one obtained in Figure 24.5.

CA can be viewed as the residual analysis of the independence model for a contingency table. If the χ^2-value is significant (the independence model does not hold), then the residuals contain more than noise alone, so that it makes sense to analyze the remaining structure in the residuals by CA. However, if the χ^2-value of the independence model is *not* significant, then the residuals are simply the result of noise, so that CA should be avoided. The view of CA as residual analysis of loglinear models has been advocated by Van der Heijden and De Leeuw (1985) and Van der Heijden, De Falguerolles, and De Leeuw (1989). A maximum likelihood version of CA was proposed by Goodman (1985, 1986), and Gilula and Haberman (1986). For a comparison of these methods, see Van der Heijden, Mooijaart, and Takane (1994).

Crime Rates

To illustrate how CA works, consider Table 24.5 with crime rates of seven offenses of 10 U.S. states (U.S. Statistical Abstract 1970, Bureau of Census: Crime rates per 100,000 people). The 50 states were used in an MDS analysis in Chapter 1, but here we restrict ourselves to the 10 states reported in Table 24.5. The main question is how similar or different the states are with respect to their crime statistics. What criminal offenses characterize the states?

CA on Table 24.5 yields the inertia reported in Table 24.6. The first two dimensions show 72% of the total inertia, 47% in the first dimension

TABLE 24.5. Crime rates per 100,000 people for 10 U.S. states. The rows entries are the criminal offenses and the column entries are the states.

State	AK	AL	AR	HI	IL	MA	NE	NY	TN	WY
Murder	12.2	11.7	10.1	3.6	9.6	3.5	3.0	7.9	8.8	5.7
Rape	26.1	18.5	17.1	11.8	20.4	12.0	9.3	15.5	15.5	12.3
Robbery	71.8	50.3	45.6	63.3	251.1	99.5	57.3	443.3	82.0	22.0
Assault	168.0	215.0	150.0	43.0	187.0	88.0	115.0	209.0	169.0	73.0
Burglary	790.0	763.0	885.0	1456.0	765.0	1134.0	505.0	1414.0	807.0	646.0
Larceny	2183.0	1125.0	1211.0	3106.0	2028.0	1531.0	1572.0	2025.0	1025.0	2049.0
Auto theft	551.0	223.0	109.0	581.0	518.0	878.0	292.0	682.0	289.0	165.0

TABLE 24.6. Singular values ϕ_a and percentage of reconstructed inertia of correspondence analysis on crime rates in Table 24.5.

Dim.	ϕ_a	Inertia ϕ_a^2	Perc. Inertia	Cum. Inertia
1	.195	.038	46.8	46.8
2	.143	.020	25.1	72.0
3	.123	.015	18.6	90.6
4	.086	.007	9.1	99.7
5	.014	.000	0.3	100.0
6	.002	.000	0.0	100.0
Total		.081	100.0	

and 25% in the second dimension. The coordinates for the points are displayed in Figure 24.9. Because the row principal normalization is used, the crimes are the weighted average of the points representing the states. The predicted profile (or reconstructed profile) for a state consists of the projections of the criminal offenses points onto the line through the origin and a state. For example, the projections on the line through the origin and MA (Massachusetts) (see Figure 24.9) show that auto theft and robbery happen more often than average. Because larceny and burglary project almost on the origin, they occur at an average rate in Massachusetts, whereas murder, rape, and assault are below average. Robbery (and to a lesser extent assault) happens in New York (NY) more often than average, and larceny less than average. In contrast, Nebraska (NE), Wyoming (WY), and Hawaii (HI) have the opposite profile compared to NY. Murder happens more often than average in the Southern states of Arkansas (AR), Alabama (AL), and Tennessee (TN). The first dimension seems to be dominated by states with robbery (on the right) versus states with more than average larceny. The second axis shows crimes with physical violence (bottom) versus property crimes (top).

Detailed results for the row and column points are given in Table 24.7. The second column gives the so-called mass ($\mathbf{D}_r/n$ and $\mathbf{D}_c/n$ for the rows

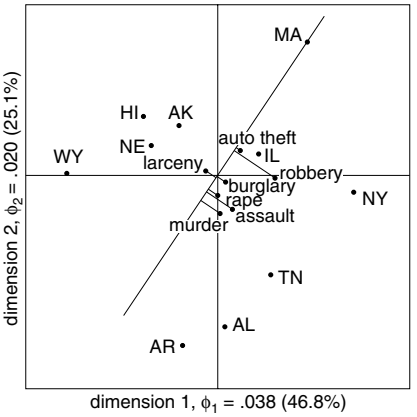


FIGURE 24.9. The correspondence analysis solution of the crime data for 10 U.S. states reported in Table 24.5.

and columns, respectively) weighting the importance of each point in the CA solution. The next two columns give the row scores **R** and column scores **C**. Then, column d_{iz} and d_{jz} shows how much of the each point contributes to the total inertia of .018 in the full dimensional space. We see that the crimes murder and rape and the states AK, IL, and NE hardly determine the CA solution as their contributions to the total inertia are very low. The next two columns show the contribution of each point to the total inertia of a dimension. For example, the first dimension is mostly determined by the crimes robbery and larceny in the states NY, WY, and to a lesser extent in HI and MA. The second dimension is mainly determined by the crimes assault and auto theft in the states AL, AR, and MA. Even though points may not determine the dimension, it may still be that a reasonable proportion of the inertia of a point is shown in that dimension. The last column shows the proportion of the inertia d_{iz} and d_{jz} that is shown in both dimensions. We see that the inertia of all crimes is reasonably well recovered in these two CA dimensions, because their total proportion of inertia d_{iz} and d_{jz} recovered in two dimensions is varying from 42.1% to 87.9%. The same is true for the states with the exception of IL of which only 19.2% of its inertia is shown in these two dimensions. Therefore, IL should be excluded from the interpretation of this CA solution.

The MDS analysis in Chapter 1 (Figure 1.1) yields similar results. In both analyses, we find that violent crimes (rape, assault, murder) are close together as opposed to the property crimes. These results seem more pronounced in the ordinal MDS solution. We have to bear in mind, though, that the MDS solution was based on the full data set, whereas the correspondence analysis solution was based on 10 states only.

TABLE 24.7. Results for row and column points of the correspondence analysis on crime rates in Table 24.5.

Crime	$\mathbf{D}_r/n$	$\mathbf{r}_1$	$\mathbf{r}_2$	d_{iz}	Contr. to Dim. Dim 1	Contr. to Dim. Dim 2	Prop. d_{iz} Shown Dim 1	Prop. d_{iz} Shown Dim 2	Prop. d_{iz} Shown Total
Murder	.002	.012	-.470	.001	.000	.024	.000	.642	.642
Rape	.005	-.015	-.258	.001	.000	.015	.001	.420	.421
Robbery	.035	.656	-.055	.023	.393	.005	.648	.005	.653
Assault	.041	.157	-.421	.012	.027	.360	.086	.621	.708
Burglary	.268	.077	-.101	.011	.042	.133	.150	.253	.403
Larceny	.523	-.156	.029	.015	.333	.021	.848	.029	.876
Auto theft	.126	.249	.268	.019	.205	.442	.408	.471	.879
Total	1.000			.081	1.000	1.000			

State	$\mathbf{D}_c/n$	$\mathbf{c}_1$	$\mathbf{c}_2$	d_{jz}	Contr. to Dim. Dim 1	Contr. to Dim. Dim 2	Prop. d_{jz} Shown Dim 1	Prop. d_{jz} Shown Dim 2	Prop. d_{jz} Shown Total
AK	.111	-.469	.561	.003	.024	.035	.273	.210	.483
AL	.070	.068	-1.797	.006	.000	.227	.002	.740	.742
AR	.071	-.427	-2.016	.008	.013	.289	.064	.772	.836
HI	.154	-.886	.668	.008	.121	.069	.547	.167	.713
IL	.111	.464	.228	.005	.024	.006	.170	.022	.192
MA	.110	1.035	1.540	.015	.118	.260	.303	.360	.663
NE	.075	-.794	.327	.003	.047	.008	.579	.053	.632
NY	.140	1.578	-.220	.017	.350	.007	.785	.008	.793
TN	.070	.606	-1.188	.004	.026	.099	.245	.507	.752
WY	.087	-1.784	.001	.011	.277	.000	.930	.000	.930
Total	1.000			.081	1.000	1.000			

Comparing CA and MDS

Correspondence analysis has several properties in common with MDS but differs on other aspects. Both techniques graphically display the objects as points in a low-dimensional space. In its basic form, MDS is a one-mode technique (only one set of objects is analyzed), whereas CA is a two-mode technique (row and column objects are displayed, as in unfolding). The data in CA are restricted to be nonnegative, whereas MDS can process more types of data: nonnegative or negative, frequencies, correlations, ratings, rankings, and so on. In addition, MDS can optimally transform the data. For contingency tables (the most widely used type of data for CA) the nonnegativity restriction does not pose a problem, because frequencies between two categorical variables are always nonnegative. CA uses the χ^2-distance as a dissimilarity measure, whereas MDS can accept any dissimilarity or similarity measures (see Chapter 6). In MDS (and unfolding), the distances between all points can be directly interpreted, but in CA this is so only for either the row or the column points. The relation between row and column points can only be assessed by projection (as in Figure 24.9). Therefore, a CA solution has to be interpreted with some care, analogous to non-Euclidean MDS solutions.

There exists a close relation between CA and Classical Scaling. Let $\mathbf{\Delta}$ contain the χ^2-distances between the rows. Let the centering matrix $\mathbf{J}$ be replaced by the weighted centering matrix $\mathbf{J}_w = \mathbf{I} - (\mathbf{1}\mathbf{D}_r\mathbf{1})^{-1}\mathbf{1}\mathbf{1}'\mathbf{D}_r$, so that $\mathbf{J}_w\mathbf{X}$ has weighted mean zero (see Section 12.3). Then, the eigendecomposition in classical scaling of $-(1/2)\mathbf{J}_w\mathbf{\Delta}^{(2)}\mathbf{J}_w'$ yields exactly the same solution for the row scores as does correspondence analysis.

Applying MDS to this $\mathbf{\Delta}$ gives an even higher proportion of explained inertia than CA. Gifi (1990) discusses the decomposition of the χ^2-distances with Stress for binary $\mathbf{F}$. For a general $\mathbf{F}$, setting $\delta_{ik} = d_{ik}$ as defined in (24.3) and setting the weight $w_{ik} = n/(f_{i+}f_{+k})$ gives a decomposition of the χ^2-distances by MDS. If the MDS algorithm uses the CA solution as a start configuration, then the final MDS solution always gives a better reconstruction of the χ^2-distances than CA. One drawback of using MDS (on the matrix of χ^2-distances) instead of CA is that the MDS solution only displays the row points, not the column points. If $\mathbf{X}$ has weighted sum zero, that is, $\sum_i f_{i+}x_{ia} = 0$ for each dimension a, then the origin represents the average row profile, just as in correspondence analysis.

24.3 Exercises

Exercise 24.1 Consider the matrix below. It shows correlations (multiplied by 100) among 13 work value items described in Table 5.2. The lower (upper) half of the matrix is based on a representative survey of the East

(West) German workforce. Note that in this study, no data were gathered on work value item 10.

No.	Work Value	1	2	3	4	5	6	7	8	9	11	12	13	14
1	Interesting job		47	43	38	28	37	29	28	27	16	15	21	28
2	Independent work	51		53	31	27	34	23	25	28	25	16	15	26
3	Much responsibility	42	57		39	32	42	38	38	41	24	16	09	25
4	Meaningful job	37	30	33		20	33	38	44	29	24	13	08	33
5	Chances for advancement	28	29	33	18		43	19	25	15	39	52	27	34
6	Respected job	18	23	34	24	43		37	39	29	37	29	21	35
7	Can help others	20	19	31	33	17	32		48	49	16	10	14	26
8	Useful job	20	17	28	40	18	37	56		32	23	16	18	30
9	Contact with other people	31	34	39	31	21	24	43	34		16	11	10	19
11	Secure position	14	17	18	19	39	37	24	25	17		40	18	38
12	High income	20	26	25	05	54	32	05	08	11	32		27	29
13	Much spare time	25	22	13	09	19	30	13	18	19	16	30		25
14	Healthy working cond.	32	31	23	37	25	20	25	23	24	33	16	23	

(a) Analyze both of these correlation matrices via PCA (varimax rotation) and interpret the resulting component loadings. Do the components correspond to any of the facets of Table 5.2?

(b) What type of facets—axial, modular, or polar facets—can or cannot be seen in a PCA of item correlations?

(c) Take the facet "Alderfer" in Table 5.2, for example. Imagine we had many "material" work values, but only very few "growth" and "relational" work values, respectively. How would this affect an attempt to verify a facet classification via PCA and via MDS analysis with regional interpretations, respectively? Which approach is less robust against uneven item sampling, and why?

(d) Fit the two PCA solutions by Procrustean methods to each other. Which transformations are admissible for PCA solutions and why are they? Which ones are not and why not?

Exercise 24.2 Assume that it takes two abilities, AR = "ability to read well" and AM = "ability for mathematics", to perform well in the tasks $T_1, \ldots, T_4$. The table below shows the ability scores for five persons, their respective performance in four tasks (T_i), and the measured performance in these tests (T_i^*). The scores were constructed as follows: let $T_i = a \cdot AR + b \cdot AM$, where $a + b = 1$; let $T_i^* = T_i + \text{error}$.

Person	AR	AM	T_1	T_2	T_3	T_4	T_1^*	T_2^*	T_3^*	T_4^*
1	10	4	8.80	8.20	5.80	4.00	8.06	7.50	7.01	3.69
2	5	3	4.60	4.40	3.60	3.00	3.93	3.86	4.63	0.93
3	2	7	3.00	3.50	5.50	7.00	3.24	3.91	6.43	5.11
4	5	9	5.80	6.20	7.80	9.00	4.19	4.48	8.70	8.76
5	9	10	9.20	9.30	9.70	10.00	9.51	9.93	11.04	9.92

(a) Take the observed task scores, $T_1^*, \ldots, T_4^*$, intercorrelate them, and find out by using PCA how these scores can best be explained through (possibly rotated) latent factors.

(b) Compute the components and compare them with the true AR- and AM-scores, respectively.

(c) Sketch the vector configuration of the four tasks in 2D, both in principal axes orientation and in varimax orientation.

(d) Intercorrelate $T_1^*, \ldots, T_4^*$ and do an MDS analysis of these correlations. Compare the result with the PCA solution.

Exercise 24.3 Bendixen (1996) reports frequencies of 14 statements on 8 breakfast items judged by a sample of 100 housewives (see the table below). The breakfast items are Cereals (CER), Muesli (MUE), Porridge (POR), Bacon and eggs (B&E), Toast and tea (T&T), Fresh fruit (FRF), Stewed fruit (STF), and Yoghurt (YOG). Note that each respondent could choose more than one statement for each breakfast item.

		Breakfast Item							
no.	Statement	CER	MUE	POR	B&E	T&T	FRF	STF	YOG
1	Healthy	14	38	25	18	8	31	28	34
2	Nutritious	14	28	25	25	7	32	26	31
3	Good in summer	42	22	11	13	7	37	16	35
4	Good in winter	10	10	32	26	6	11	19	8
5	Expensive	6	33	5	27	3	9	18	10
6	Quick and easy	54	33	8	2	15	26	8	20
7	Tasty	24	21	16	34	11	33	26	26
8	Economical	24	3	20	3	16	7	3	7
9	For a treat	5	3	3	31	4	4	16	17
10	For weekdays	47	24	15	9	13	11	6	10
11	For weekends	12	5	8	56	16	10	23	18
12	Tasteless	8	6	2	2	0	0	2	1
13	Takes too long to prepare	0	0	9	35	1	0	10	0
14	Family favorite	14	4	10	31	5	7	2	5

(a) What items and statements do you expect to influence the CA solution most? Why do you think so?

(b) Apply CA to the matrix of frequencies above. (You can use, for example, the CORRESPONDENCE program in SPSS.) How many dimensions do you choose? How much of the total inertia is accounted for by these dimensions?

(c) Interpret the most important relations in the CA solution. (Hint: focus on a statement and look which breakfast items are more and less than average characterized by this statement.)

(d) What items and statements are not well represented in this CA solution? Do you need to revise your interpretation at (c)?

(e) Remove the bad fitting items and statements and redo CA. Interpret the solution. Are the relations different from the CA solution in (b)? If so, explain the differences.

Exercise 24.4 Consider the data on the interpretations of Rorschach inkblot pictures reported in Exercise 15.3 on p. 332.

(a) Do CA on these data. How many dimensions do you choose?

(b) Identify good and bad fitting row and column points. What measures do you use for doing so?

(c) Interpret the CA solution. What are the most important relations in these data?

Exercise 24.5 PCA can also be attempted "by hand".

(a) Consider the correlation matrix 5.1. Convert the correlations into angles among pairs of vectors. [Hint: For example, for $r_{12} = .67$ in Table 5.1, the angle of the corresponding geometric vectors for items 1 and 2 is $\arccos(.67) = 47.9°$.]

(b) With this angle information, construct a vector representation of the correlations. First, take eight knitting needles, straws, sticks, or the like. Then stick needle 1 into a styrofoam ball and then needle 2 such that it forms an angle of 47.9° with 1. Then proceed with needle 3, and so on.

(c) Compare your result to a solution arrived at by computation.

Part VI

Appendices

Appendix A
Computer Programs for MDS

Several computer programs for doing MDS exist, some of which are included in major software packages, and others are in the public domain. The list of programs we discuss in this appendix is not exhaustive, although we have tried to find the most important ones in terms of options and availability at the time of writing. Each program is described briefly. We also give an example of how a simple MDS job is run in each program. Where possible, we provide a subset of the commands and keywords needed for running a simple MDS analysis.

MDS is very much a visualization technique. Fortunately, the graphical capabilities of modern PCs have improved drastically over the years. Therefore, we place more emphasis on the graphical representations provided by MDS programs. In addition, we found two programs (GGVIS and PERMAP) freely available on the Internet that show interactively how the MDS solution is obtained. To reflect the development, we have organized the remainder of this appendix into three sections: the first section discusses two interactive MDS programs, the second section is focused mainly on commercial statistical packages that have high-resolution graphics, and the third section treats MDS programs that do not have high resolution graphics and have mostly been developed in the early days of MDS.

Table A.1 gives an overview of the properties of each of the programs. The MDS models in Table A.1 denote: (a) ordinal MDS with the primary approach to ties; (b) ordinal MDS with the secondary approach to ties; (c) ordinal MDS, using rank-image transformations; (d) interval MDS, $a + b \cdot p_{ij} = d_{ij}(\mathbf{X})$; (e) ratio MDS, $b \cdot p_{ij} = d_{ij}(\mathbf{X})$; (f) splines; (g) polynomial regression, $a+b\cdot p_{ij}(\mathbf{X})+c\cdot p_{ij}^2(\mathbf{X})+\cdots = d_{ij}(\mathbf{X})$; (h) power, $p_{ij} = a\cdot d_{ij}^b(\mathbf{X})$,

which is equivalent to a linear MDS on logged proximities; (i) mixed models, for example, ordinal (unconditional) MDS for matrix 1 and linear MDS for a copy of matrix 1 stored in matrix 2. Note that (h) is but a linear MDS on logged proximities.

One general warning for the use of all programs is in place: many programs have rather weak convergence criteria, which may cause the program stopping the iterations too early and give suboptimal solutions. To be on the safe side, we advise to stop the iterative process only if the difference in two subsequent loss function values (usually Stress, or S-Stress) is smaller than 10^{-6} and setting the maximum number of iterations to 100 or more.

To illustrate the setup of a program, we use the artificial data on the ranking of pairs of politicians in Tables 9.4 and 9.3.

A.1 Interactive MDS Programs

With improving speed and graphical capabilities of modern computers, it becomes possible to animate the way in which MDS solutions are obtained. In this section, we discuss two of these programs, GGVIS and PERMAP. We call these programs an interactive form of MDS because they allow us to manipulate the MDS options by an easy user interface. Any change of MDS options usually leads to animations showing the changes leading to an optimal configuration. In such a way, you can test the stability of the solution interactively, for example, by eliminating points, changing the MDS model, rearranging points to check for local minima, and so on. These programs stay close to the exploratory nature of MDS with an emphasis on visualization.

GGVIS

GGVIS is a standard plug-in for MDS that comes with the GGOBI visualization software. It is freely available from the Internet and can be run as a standalone application or within the statistical programming environment R (also freely available). GGOBI visualizes rectangular two-way-two-mode data allowing an interactive grand tour through high-dimensional spaces, labeling, glyphing, connecting edges, and the like. GGVIS uses many of these options but is tailored for MDS. For an extensive discussion of GGVIS, we refer to Buja and Swayne (2002).

A nice feature of GGVIS is that a change of options has immediate effects on the solution. Thus, the user can see in real-time, for example, how the configuration changes from a metric to a nonmetric solution. Although the emphasis of GGVIS is on metric MDS, it also allows for ordinal transformations. Two options in GGVIS are unique. First, you can set interactively a power transformation of the dissimilarities and the resulting distribution is

TABLE A.1. A summary of several MDS programs; + stands for Yes or indicates that option is available, – shows that option is not available, n.a. means not applicable, and mem indicates that the maximum number of objects depends on memory available.

	Ggvis	Permap	Alscal	Newmdsx	Proxscal	SAS	Statistica	Systat	Fssa	Kyst	Minissa	Multiscale
Platforms												
Version	1.0.0b	11.3	n.a.	4.0.4	1.0	n.a.	4.5	11	3	2a	1	n.a.
Standalone program	+	+	+	+	–	–	–	–	+	+	+	+
In larger package	+	–	+	–	+	+	+	+	–	–	+	–
Commercial	–	–	+	+	+	+	+	+	–	–	+	–
MS-Windows	+	+	+	+	+	+	+	+	–	+	–	+
Macintosh	–	–	+	–	+	+	+	+	–	+	–	+
Graphical user interface	+	+	+	+	+	+	+	+	–	–	–	–
High resolution graphics	+	+	+	+	+	+	+	+	–	–	–	–
Dynamic graphics	+	+	–	–	–	–	–	–	–	–	–	–
General features												
Minimizes Stress	+	+	–	+	+	+	+	+	–	+	+	+
Minimizes S-Stress	–	+	+	–	–	+	–	+	–	–	–	–
Minimizes alienation	–	–	–	+	–	–	–	+	+	–	+	–
Maximizes likelihood	–	+	–	–	–	–	–	–	–	–	–	+
Max. number of objects	mem	200	100	100	mem	mem	90	mem	50	60	100	mem
Min. number of objects	2	2	4	2	2	2	2	2	3	3	2	3
Max. dimensionality	n-1	4	6	10	n-1	n-1	9	5	10	6	10	n-1
Processes rectangular data	+	+	+	+	+	–	–	+	–	+	+	+
Allows for missing data	+	+	+	+	+	+	+	+	+	+	+	+
Offers Minkowski distances	+	+	–	+	–	–	–	+	–	+	+	–
Allows for weights w_{ij}	+	+	–	–	+	+	–	–	–	+	–	–
MDS models												
Ordinal, prim. appr. ties	–	+	+	+	+	+	+	+	–	+	+	–
Ordinal, sec. appr. ties	+	–	+	+	+	+	–	–	+	+	+	+
Ordinal, rank-image	–	–	–	+	–	–	–	+	+	–	+	–
Interval	–	+	+	–	+	+	–	+	–	+	–	+
Ratio	+	+	+	–	+	+	–	–	–	+	–	+
Absolute	+	–	+	–	–	+	–	–	–	–	–	+
Splines	–	–	–	–	+	–	–	–	–	–	–	+
Polynomial regression	–	–	+	–	+	–	–	–	–	+	–	+
Power	–	–	–	–	+	–	–	+	–	–	–	–
Mixed models	–	–	–	–	–	–	–	–	–	+	–	–
Special models												
Split, by row	–	–	+	+	–	+	+	+	+	+	+	+
Split, by row and by col.	–	–	–	+	–	–	–	–	–	+	+	–
Split, by matrix	–	–	+	–	+	+	–	+	–	+	–	–
Asymmetry models	–	–	+	–	–	–	–	–	–	–	–	–
Weighted Euclidean model	–	–	+	+	+	+	–	+	–	–	+	+
Generalized Euclidean model	–	–	+	+	+	–	–	–	–	–	+	–
External variables	–	–	–	–	+	–	–	–	+	–	–	–
Constrained solutions	–	+	+	+	+	+	–	–	–	+	+	–

FIGURE A.1. Three different situations of the MDS settings windows of GGVIS with four tabs at the top.

shown in a histogram. A power may be chosen such that the transformed dissimilarities have a uniform distribution (for an application, see Section 9.7). The second unique option is to set the weights that are applied to the errors to a power of the dissimilarities. Choosing a large positive power emphasizes the correct display of large dissimilarities, whereas a large negative power mostly ignores large dissimilarities and emphasizes the proper representation of small dissimilarities (see Section 11.3).

We tested a beta version of GGVIS. To use GGVIS, you first have to set up a data file that GGVIS can process. Below, we present a sample file in XML. Once the data are read, move to the MDS module by choosing Tools > GGVIS (MDS).... This brings up the window shown in Figure A.1.

Here, you can move to the fourth tab (or directly click on the "Run" button). It opens a window (see middle panel of Figure A.1) where you can change the default parameter settings for the dimensionality of the MDS space and the stepsize for the iterations. For our politicians data, we would set the dimensionality to 2, changing it from the default value of 3. To get an ordinal rather than a metric MDS solution, we would then press the pull-down menu "Metric MDS" in the middle of the window, where we can click on "Nonmetric MDS". Then, click on the "Run MDS" button which starts the program showing how Stress is minimized and how the data are weighted (see windows in the middle of the right panel of Figure A.2). The ordinal MDS solution is given in Figure A.2.

The "Run MDS" button is a toggle. You may, for example, experiment with different stepsizes, metric vs. nonmetric MDS, different weights and so on, and rerun the MDS. From the "Reset" menu, you can reinitialize or

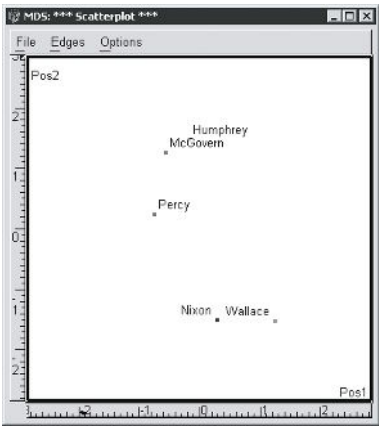


FIGURE A.2. Solution obtained by GGVIS.

scramble the MDS configuration, and from the "View" menu you can view the Shepard diagram.

GGVIS reads its data in XML format. Below you find an example. XML commands are written between <>-signs. Every command that is opened, for example, `<description>`, should also be closed again `</description>`. The first two lines define that these data belong to GGOBI.

```
 <?xml version="1.0"?> <!DOCTYPE
ggobidata SYSTEM "ggobi.dtd">

<ggobidata count="2">
<data name="Politicians">
<description>
Example data set to illustrate {\sc Ggvis}
</description>
<variables count="0">
</variables>
<records count="5" glyph="fr 1" color="3">
<record id="1"  label="Humphrey"  color="1">  </record>
<record id="2"  label="McGovern"  color="3">  </record>
<record id="3"  label="Percy"     color="3">  </record>
<record id="4"  label="Wallace"   color="2">  </record>
<record id="5"  label="Nixon"     color="0">  </record>
</records>
</data>

<data name="dissimilarity">
<description>
 Dissimilarities (rank orders)
</description>
<variables count="1">
  <realvariable name="Dissimilarity" nickname="D" />
</variables>
<records count="10" glyph="fr 1" color="0">
```

```
<record source="1" destination="2">  1 </record>
<record source="1" destination="3">  5 </record>
<record source="1" destination="4">  7 </record>
<record source="1" destination="5">  6 </record>
<record source="2" destination="3">  2 </record>
<record source="2" destination="4"> 10 </record>
<record source="2" destination="5">  8 </record>
<record source="3" destination="4">  9 </record>
<record source="3" destination="5">  4 </record>
<record source="4" destination="5">  3 </record>
</records>
</data>

</ggobidata>
```

GGVIS uses the following commands:

- `<ggobidata count="2">` says that what follows are two data sets specific for GGOBI.
- `<data name="Politicians">` specifies that the data defined here are called 'Politicians'.
- `<description>` allows a description of the data.
- `<variables count="0">` indicate that the rows defined below have no variables. If, for example, `count=1` then one lines should follow defining the variable name and nickname by `<realvariable name="Variable 1" nickname="V1" />`. For more variables, add more lines.
- `<records count="5" glyph="fr 1" color="3">` specifies that five records follow with a certain form of glyph type and color.
- `<record id="1" label="Humphrey" color="1"> </record>` defines the first record to have label 'Humphrey' and a specific color. If there is at least one variable, then their values should be specified before `</record>`.
- `<record source="1" destination="2"> 1 </record>` defines a single dissimilarity for objects 1 and 2. For all available dissimilarities, a single record should be specified indicating the row number and their column number. Missing dissimilarities are obtained by omitting the records for the missing pairs of objects.

More information can be found on the GGOBI website http://www.ggobi.org; E-mail: ggobi-help@ggobi.org

PERMAP

PERMAP is one of the few interactive MDS packages available. It allows users to interact directly with an MDS solution, move objects in the solution space, remove certain objects, and change MDS options. PERMAP is not built on any of the previously existing MDS software and can be freely downloaded from the Internet.

The program has a wide range of options, some of which are unique to PERMAP. It allows for ratio, interval, and ordinal MDS, the latter using the primary approach to ties. It can minimize several MDS loss functions,

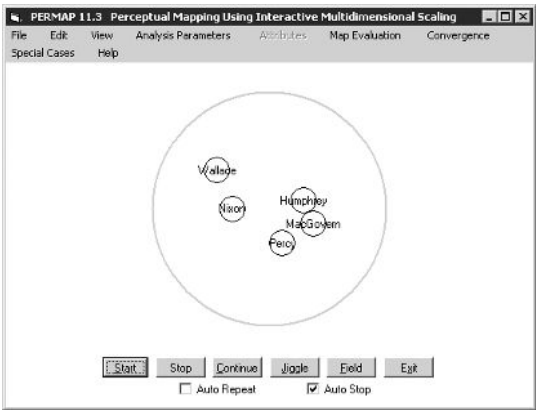


FIGURE A.3. Screen shot of PERMAP.

including Stress, Stress-1, S-Stress, and MULTISCALE. In addition, weights can be specified for every dissimilarity. PERMAP uses the general Minkowski distances that include Euclidean, city-block, and dominance distances. The program can also compute dissimilarities from a rectangular data matrix. A summary of the MDS solution can be saved into a text file. PERMAP comes with an extensive documentation aimed at the nonexpert.

A user-friendly option of PERMAP is to drag objects away from the solution into a "parking lot" to exclude the objects from the current MDS configuration. PERMAP will recompute the solution without these points. This option enables the user to test the influence of these points on the solution. If you want to use the object again, then you can drag the object back from the parking lot to the MDS configuration. In addition, points can be moved by dragging them around in the MDS solution. It is also possible to lock certain points that will keep them at a fixed location. Figure A.3 shows a screen shot of PERMAP using the politicians data. Text labels can be attached to the points by providing them as the first entry on a line with dissimilarities. The program can compute solutions in 1 to 4 dimensions. However, for three- or four-dimensional solutions, PERMAP shows a 2D projection of the 4D space. Note that these 2D projections may differ when PERMAP reruns the analysis.

The data input for PERMAP comes from a text file that can be written with any editor. The data file must be structured by certain specific keywords that instruct PERMAP how to read the information. Other text is simply ignored and can be used to explain the data. A simple setup for our politicians data looks like this:

```
TITLE= Example setup: politicians
NOBJECTS= 5
DISSIMILARITYLIST
Humphrey  0
MacGovern 1  0
```

```
Percy     5  2  0
Wallace   7 10  9  0
Nixon     6  8  4  3  0
```

A nonexhaustive list of subcommands of PERMAP is given below.

- NOBJECTS sets the number of objects.
- TITLE and SUBTITLE set the title and subtitle to be used in the output. If the MESSAGE and SUBMESSAGE are specified, then these are used in the output.
- DISSIMILARITYLIST or SIMILARITYLIST indicate that a triangular matrix with the dissimilarities or similarities are specified below. Note that the diagonal elements should be given as well and that NA specifies a missing value. If a dissimilarity line starts with a text entry, then it is used as a label for the points.
- WEIGHTLIST announces to PERMAP that the weights are specified below.
- ATTRIBUTELIST specifies that the two-way data to be used to compute dissimilarities follow below. Note that NATTRIBUTES (indicating the number of columns of the two-way data) has to be defined before.
- LOCATIONLIST gives the initial configuration.
- STARTMDSANALYSISTYPE defines the transformations. Choose 0 for a ratio transformation, 2 for interval, and 4 for ordinal primary approach to ties. The program cannot do the secondary approach to ties.
- STARTBADNESSFUNCTION defines what MDS loss function is used. Choose 0 for Stress, 1 for Stress-1, 2 for S-Stress, and 3 for MULTISCALE.
- STARTDISTANCEFUNCTION defines the distance to be used. Choose 0 for Euclidean distances, 1 for city-block, and 2 for Minkowski.
- STARTATTRIBUTEFUNCTIONNUM defines how the two-way data should be transformed into dissimilarities. Choose 0 for one minus the cosine of the angle between the vectors defined by the columns, 1 for Euclidean distances between the rows, 2 for city-block distances between the rows, 3 for one minus Guttman's μ_2 coefficient, 4 for the Pearson correlation between the columns, 5 for the Spearman rank correlation between the columns, 6 for the proportion of different categories between the rows (to be used for nominal variables). Options 7 to 12 are used for binary variables: 7 for the Jaccard indexes, 8 for Gower/Russel/Rao, 9 for Sokal-Michener distances, 10 for Hamman, 11 for Yule, and 12 for Askin/Charles.
- STARTDIMENSIONNUM allows to specify the dimensionality of the solution between 1 and 4.

Apart from the NOBJECTS command and a command to read the data, all other commands are optional.

For more information, contact Ron B. Heady, University of Louisiana at Lafayette, U.S.A. E-mail: ron@heady.us; Internet: http://www.ucs.louisiana.edu/~rbh8900

A.2 MDS Programs with High-Resolution Graphics

Current computers are able to provide high-resolution graphics, which is particularly important for a visualization technique such as MDS. All ma-

jor statistical packages provide these high-resolution graphics. Below we discuss several MDS procedures available in commercial statistical packages and a package called NEWMDSX© that provides a shell for text-based MDS programs that produce high-resolution graphics.

ALSCAL

ALSCAL (Takane et al., 1977) is one of the current MDS modules in SPSS. ALSCAL differs from other MDS programs in minimizing S-Stress rather than Stress, thereby fitting squared distances to squared dissimilarities. As a result, in ALSCAL the large dissimilarities are much better represented than the small dissimilarities. ALSCAL is a flexible MDS program that also provides models for asymmetric data, unfolding, and three-way analyses (by the weighted or generalized Euclidean model). Many options can be combined. ALSCAL also allows coordinates to be fixed, which is especially useful for external unfolding.

ALSCAL can be started in SPSS by choosing the menu "Analyze > Scale > Multidimensional Scaling...". Using dialogue boxes, the ALSCAL options can be specified. In addition, (dis)similarity matrices can be created from rectangular data matrices. Alternatively, ALSCAL can be run through SPSS-syntax allowing for some more options. Some care has to be taken when adapting a configuration plot in ALSCAL. If you change the range of the axes or resize the plot differently for the two axes, then the horizontal units can be different from the vertical units so that the distances you see may be misleading. In addition, the default convergence criterion is far too weak and should be manually tightened to, say, .000001 or smaller.

A sample setup for an ordinal MDS analysis with ALSCAL of a 5×5 matrix of dissimilarity scores on five politicians is this:

```
TITLE 'Alscal in SPSS example setup: politicians'.
MATRIX DATA /VARIABLES Humphrey McGovern Percy Wallace Nixon
  /CONTENTS PROX /FORMAT LOWER NODIAGONAL.
BEGIN DATA
 1
 5  2
 7 10  9
 6  8  4  3
END DATA.
ALSCAL /VARIABLES Humphrey McGovern Percy Wallace Nixon
 /CRITERIA CONVERGE(0.000001) ITER(100) STRESSMIN(0.000001)
 /LEVEL ORDINAL.
```

Commands in SPSS are ended by a dot (.); subcommands start with a slash (/) and usually have one or more keywords; keywords are printed in caps.

The ALSCAL job above first formulates a TITLE. It then defines the data setup in the MATRIX DATA command and lists the proximities between BEGIN DATA and END DATA. In the MATRIX DATA command, /VARIABLES should be followed by a list of variable names,

one for each variable (object, point). This list can also be abbreviated by 'VAR1 TO VAR5' or 'OBJECT1 TO OBJECT5'. The variable names are at most eight characters long. The subcommand /CONTENTS PROX indicates that the contents is a proximity matrix. /FORMAT LOWER NODIAGONAL indicates that only the lower triangular elements of the proximities are to be read. Finally, the desired MDS model is specified in the ALSCAL command: The VARIABLES option lists the variables that are to be mapped into points; the CRITERIA option specifies a number of technical requests for the optimization; the LEVEL option requests an ordinal MDS.

Some optional subcommands of ALSCAL are:

- /SHAPE specifies the shape of the dissimilarity matrix. Valid keywords are SYMMETRIC, ASYMMETRIC, and RECTANGULAR. SHAPE = RECTANGULAR defines unfolding.
- /LEVEL indicates the allowed transformation of the dissimilarities. Default is ORDINAL, which does monotone regression with the secondary approach to ties. For the primary approach specify, ORDINAL(UNTIE). If the proximities are similarities instead of dissimilarities, you can specify ORDINAL(SIMILAR), which may be combined with UNTIE. The keyword INTERVAL indicates interval transformations. For example, INTERVAL(3) specifies polynomial regression of the order 3. RATIO excludes the intercept and followed by '(2)' indicates quadratic polynomial regression.
- /CONDITION specifies conditionality of the transformations. In three-way scaling, MATRIX indicates that for each replication a separate transformation of the proximities has to be found (default). UNCONDITIONAL specifies that there is only one transformation for all replications. ROW means that the proximities in every row may have a different transformation, which is useful for unfolding.
- /MODEL indicates which model has to be used. EUCLIDEAN indicates the ordinary Euclidean distance (default), INDSCAL specifies the individual differences (weighted) Euclidean distance model.
- /CRITERIA controls the stopping conditions of the algorithm. CONVERGENCE (.000001) causes the program to stop whenever the difference in S-Stress between subsequent iterations is less than .000001. ITER(100) sets the maximum number of iterations to 100. STRESSMIN(.0001) causes the iterations to stop whenever S-Stress is less than .0001. NEGATIVE allows negative dimension weights in the INDSCAL model. CUTOFF(0) causes negative proximities to be treated as missing (default). DIMENS(2,5) causes ALSCAL to compute a solution in 5 dimensions, then 4, 3, and 2 dimensions. Default is DIMENS(2,2).
- /PRINT specifies print options. DATA prints the proximities. INTERMED prints intermediate results, which can generate a huge amount of output. HEADER prints a summary of options specified.
- /PLOT controls the plots made by ALSCAL. Defaults are the plots for the object configuration, the weight matrix (for INDSCAL) and Shepard plots. In addition, ALL generates a transformation plot for every replication or row (depending on CONDITION) and a plot of the weighted object coordinates for every replication (when appropriate).

For more information, contact: worldwide headquarters SPSS Inc. 233 S. Wacker Drive, 11th Floor, Chicago, IL 60606-6307, U.S.A. Phone: (312) 651-3000; Fax: (312) 651-3668; Internet: http://www.spss.com

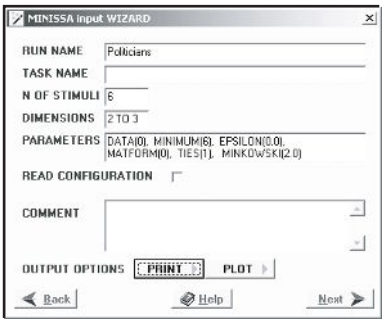


FIGURE A.4. NEWMDSX© wizard for constructing MINISSA input.

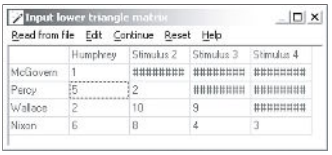


FIGURE A.5. NEWMDSX© data entry window for MINISSA.

NEWMDSX©

Many of the first-generation MDS programs have text-based input and output, and no graphical user interface nor high-resolution graphics. The package NEWMDSX© is aimed to fill this gap. It offers a shell with an easy graphical user interface to run a variety of programs reported in the literature. In this shell, you can start a wizard to construct the input file for the program you want; see Figure A.4 (based on a beta version of NEWMDSX©, version 4.0.4.). It also has several ways to read data, including a spreadsheet-like data entry window; see Figure A.5. The MDS program included in NEWMDSX© is MINISSA (discussed separately in Section A.3).

NEWMDSX© has high-resolution graphics for configurations in one to three dimensions. In addition, it provides Shepard diagrams and Stress plots. For hierarchical clustering it provides a dendrogram. The graphics the program produces can be edited.

In Figure A.6, an example is given of the graphics windows. NEWMDSX© is the only package currently available that provides a relatively easy interface for the MDS programs developed from 1960 to 1980. Table A.2 contains an overview of the MDS programs included, many of which are discussed in this book. NEWMDSX© is a package with not too many options but with a rich amount of MDS programs.

More info at: E-mail: enquiries@newmdsx.com; Internet: http://www.newmdsx.com

PROXSCAL

PROXSCAL is a program for least-squares MDS minimizing Stress available in SPSS (Commandeur & Heiser, 1993; Meulman, Heiser, & SPSS, 1999). It builds on the majorizing algorithm of De Leeuw and Heiser (1980) (see Chapter 8), which guarantees convergence of Stress. PROXSCAL offers a large variety of options for MDS analysis. One of the unique features of PROXSCAL is that the user can impose external constraints on the MDS

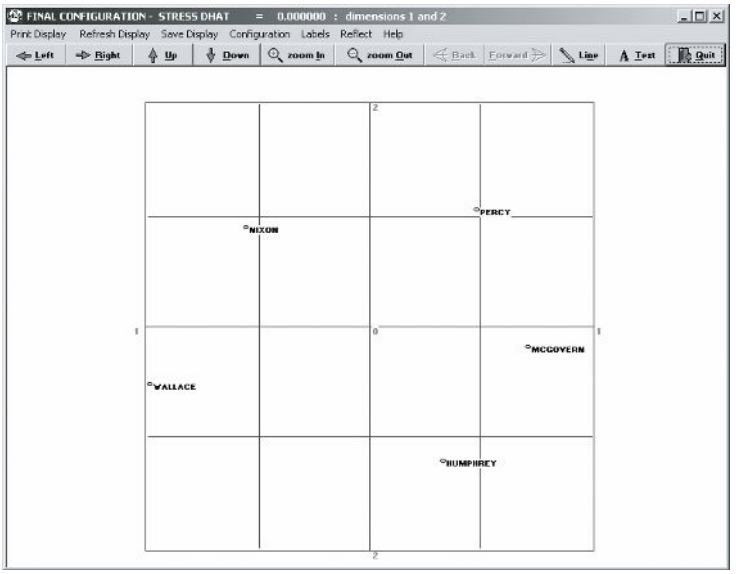


FIGURE A.6. Graphics window obtained by NEWMDSX© .

TABLE A.2. Overview of programs that can be run within NEWMDSX©.

Program	Remarks
CANDECOMP	Three-way decomposition for three-way data.
CONJOINT	Performs unidimensional conjoint analysis.
CORRESP	Correspondence analysis.
HICLUS	Performs hierarchical clustering on dissimilarity data using single or complete linkage.
INDSCAL-S	INdividual Differences SCALing for fitting the weighted Euclidean distances.
MDSORT	MDS of sorting data.
MDPREF	MultiDimensional PREFerence for the vector model of unfolding.
MINI-RSA	Ideal point unfolding model.
MINISSA	Nonmetric MDS program.
MRSCAL	MetRic SCALing for metric MDS with Minkowski distances.
PARAMAP	For maximizing local monotonicity.
PINDIS	Procrustean INdividual Differences Scaling for doing Procrustes analysis.
PREFMAP	PREFerence MAPping for external unfolding using the ideal point or vector model for unfolding.
PRO-FIT	PROperty FITting for external unfolding using the vector model.
TRISOSCAL	TRIadic Similarities Ordinal SCALing for MDS analysis of triadic dissimilarities.
WOMBATS	Work Out Measures Before Attempting To Scale converts a two-way two-mode data matrix into a dissimilarity matrix.

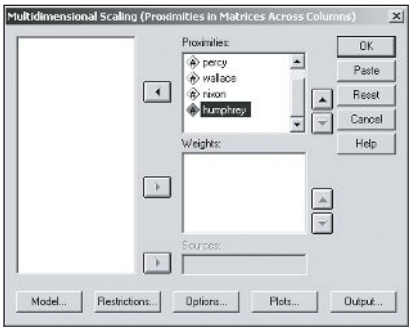


FIGURE A.7. Main dialogue boxes in SPSS PROXSCAL.

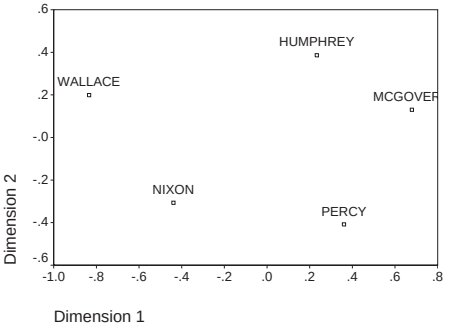


FIGURE A.8. PROXSCAL solution of the politicans data.

configuration, such as the restriction that the coordinates are a linear combination of some external variables (see Section 10.3). It also allows fixing some of the coordinates. These options can be combined with the weighted Euclidean model (Section 22.1) or the generalized Euclidean model (Section 22.2). The implementation of these models in PROXSCAL never gives negative dimension weights.

PROXSCAL can attach user-defined weights to every proximity. The transformations of the proximities in PROXSCAL are ordinal, interval, power, and monotone spline. PROXSCAL is the only program that avoids negative disparities that may arise when specifying an interval transformation of the dissimilarities as shown by Heiser (1990).

PROXSCAL can be specified in SPSS by choosing the menu "Analyze > Scale > Multidimensional Scaling (PROXSCAL)...". The main dialogue box of PROXSCAL is given in Figure A.7. Most options can be accessed through dialogue boxes. However, some options can only be specified using syntax.

```
TITLE 'Proxscal in SPSS example setup: politicians'.
MATRIX DATA /VARIABLES Humphrey McGovern Percy Wallace Nixon
  /CONTENTS MAT /FORMAT LOWER NODIAGONAL.
BEGIN DATA
 1
 5  2
 7 10  9
 6  8  4  3
END DATA.
PROXSCAL /VARIABLES Humphrey McGovern Percy Wallace Nixon
 /SHAPE = LOWER
 /INITIAL = TORGERSON
 /CRITERIA = DIMENSION(2) DIFFSTRESS(0.000001)
             MAXITER(100) MINSTRESS(0.000001)
 /TRANSFORMATION = ORDINAL.
```

PROXSCAL has many different options, some of which are only accessible from syntax. The commands in the example above up to `PROXSCAL` correspond to those discussed

above for the ALSCAL program. A nonexhaustive list of subcommands of PROXSCAL is given below.

- /VARIABLES specifies the columns of the dissimilarity matrix. Unless the /TABLE subcommand is used, the number of rows should be a multiple of the number of variables specified. In the simple case of a single dissimilarity matrix, the number of rows is equal to the number of columns. For three-way analyses, the replications should be stacked underneath each other. The labels used in the plot are either the variable names or their variable labels.
- /TABLE allows to read dissimilarities from a single column. To identify a dissimilarity, two extra variables are needed: its row and column number. If three-way data are present, a third variable is needed to define the replication (called a source in PROXSCAL). The value labels to the row, column, and replication variable are used as labels in the plots.
- /SHAPE specifies the shape of the dissimilarity matrix. Valid keywords are LOWER, UPPER, and BOTH to specify respectively the lower part of the dissimilarity matrix, the upper part, or the symmetrized table.
- /WEIGHTS is used to specify nonnegative weights for weighting the residuals. The weights are read similarly as the dissimilarities as specified by the VARIABLES subcommand.
- /PROXIMITIES defines whether the data are assumed to be DISSIMILARITIES (default) or SIMILARITIES.
- /INITIAL specifies what initial configuration is used. SIMPLEX (default) uses a PROXSCAL specific start assuming that all objects are at distance one of each other. TORGERSON defines classical scaling as a start configuration. This option tends to give better quality solutions than SIMPLEX. RANDOM(N) computes a PROXSCAL solution for `n` random starts and reports the best.
- /TRANSFORMATION indicates the allowed proximities of the dissimilarities. Default is RATIO where the dissimilarities are only multiplied such that the disparities have a specific sum of squares. The keyword INTERVAL indicates interval transformations. ORDINAL does monotone regression with the secondary approach to ties and ORDINAL(UNTIE) specifies the primary approach to ties. SPLINE(DEGREE,INKNOT) specifies a spline transformation of order `degree` and `inknot` interior knots.
- /CONDITION specifies the conditionality of the transformations. In three-way scaling, MATRIX indicates that for each replication a separate transformation of the proximities has to be found (default). UNCONDITIONAL specifies that there is only one transformation for all replications.
- /MODEL has the IDENTITY model as default thereby modeling all replications by the same configuration. For three-way data, WEIGHTED indicates the weighted Euclidean distance so that dimensions may differ in the third way according to their individual dimension weight. GENERALIZED specifies the individual differences (weighted) Euclidean distance model. REDUCED specifies the reduced rank model.
- /CRITERIA controls the stopping conditions of the algorithm. CONVERGENCE (.000001) causes the program to stop whenever the difference in S-Stress between subsequent iterations is less than .000001. MAXITER(100) sets the maximum number of iterations to 100. MINSTRESS(.0001) causes the iterations to stop whenever Stress is less than .0001. In addition, by DIMENSIONS(DMIN,DMAX) you can let PROXSCAL compute solutions from `dmax` dimensions to `dmin` dimensions.
- /PRINT specifies print options. INPUT prints the proximities, HISTORY the history of iterations, STRESS several standard Stress measures, DECOMPOSITION a decomposition of Stress into Stress per object and possibly per replication, COMMON the

coordinates, DISTANCES the distances, WEIGHTS the weights used by the generalized or weighted Euclidean distances found by three-way models, and TRANSFORMATION the transformations.

- /PLOT controls the plots made by PROXSCAL. Defaults are the plots for the object configuration (COMMON), and the weight matrix (WEIGHTS) for three-way models. In addition, a STRESS-plot is available for plotting Stress against the number of dimensions of the solution, TRANSFORMATION for plotting the the data against the disparities, RESIDUALS for plotting the disparities against the distances, and INDIVIDUAL for plotting the coordinates of replications in three-way models. Note that PROXSCAL does not provide a Shepard plot.

For more information contact: Frank M.T.A. Busing, Dept. of Psychometrics, Univ. of Leiden, P.O. Box 9555, 2300 RB Leiden, The Netherlands. E-mail: busing@fsw.leidenuniv.nl; Internet: http://www.spss.com

SAS

SAS is a comprehensive system of software products for managing, analyzing, and presenting data. SAS has versions for virtually all platforms. SAS can be run in batch mode, submit mode, interactive line mode, and display manager (windows) mode. SAS used to offer ALSCAL as its MDS module but now has replaced ALSCAL by 'PROC MDS'. This program has many options, some of them rather tricky ones, with effects that are difficult to predict.

In batch mode, our sample MDS job can be set up in the SAS command language as follows. SAS commands and options are printed in capital letters. Commands are ended with a ;.

```
DATA polit;
TITLE Politicians Example;
INPUT (var1-var5)(3.) @21 name $ 8.;
CARDS;
  0                Humphrey
  1  0             McGovern
  5  2  0          Percy
  7 10  9  0       Wallace
  6  8  4  3  0    Nixon
;
PROC MDS DIM=2 LEVEL=ORDINAL PFINAL PCONFIG OUT=OUT OUTRES=RES;
OBJECT name;
RUN;
```

The job first reads five numeric variables in fixed format (fields of length 3) and one alphanumeric variable, "name", in a field of length 8, starting in column 21. Then, the MDS procedure is called, together with a set of options. PROC MDS analyzes the proximities among all variables at the ordinal measurement level (LEVEL=ORDINAL) in two dimensions (DIMENSION=2. The fit values and the configuration are printed (PFINAL and PCONFIG, respectively). The estimates and fitted values are saved in output data files

("out" and "res", respectively). PROC MDS produces no plots. For plotting, one must utilize special plotting procedures with the output files.

SAS has its own command language. The following summarizes the most important commands for PROC MDS. The general syntax is:

```
PROC MDS  options;
   VAR  variables;
   INVAR  variables;
   OBJECT  variable;
   SUBJECT  variable;
   WEIGHT  variable;
   BY  variables;
```

The PROC MDS statement is required. All other statements are optional. The VAR statement defines the numeric variables in the file DATA = xyz that contain the proximities. Each variable corresponds to one object/point. If VAR is omitted, all numeric variables not specified in another statement are used. The INVAR statement defines the numeric variables in the file INITIAL=xyz, the initial configuration, where the first variable contains the coordinates on the first dimension, ..., the mth variable the coordinate on the mth dimension. The WEIGHT statement specifies a numeric variable in the file DATA=xyz that contains weights for each proximity. The number of WEIGHT variables must be the same as the number of VAR variables, and the variables in the WEIGHT statement must be in the same order as the corresponding variables in the VAR statement. If no WEIGHT statement is used, all data within a partition are assigned equal weight. The BY statement is used to obtain separate MDS analyses on groups of proximity data defined by the BY variables. The OBJECT statement defines a variable that contains descriptive labels for the points. The SUBJECT statement specifies a variable in the file DATA = xyz that contains descriptive labels for the data matrices or "subjects".

The options for PROC MDS are:

- The Proximities
  - DATA = SAS file name, the data set containing one or more *square* matrices to be analyzed. (The requirement to input square proximity matrices makes the procedure clumsy for unfolding applications, because off-diagonal matrices cannot be processed directly. Rather, they have to be defined as submatrices within a square matrix with missing values.) Usually, there is one matrix per person. Data are generally assumed to be dissimilarities unless (a) there are diagonal elements that are generally larger than the off-diagonal elements or (b) one uses the SIMILAR option.
  - SIMILAR causes the data to be treated as similarities.
  - SHAPE = TRIANGULAR | SQUARE determines whether the entire data matrix for each subject is stored and analyzed or only one triangle of the matrix. Default is triangle, unless CONDITION = ROW.
- The MDS Model
  - LEVEL = ABSOLUTE | RATIO | INTERVAL | LOGINTERVAL | ORDINAL specifies the admissible transformation on the proximities.
  - CONDITION = UN | MATRIX | ROW. Conditionalities of proximities. Default is MATRIX.
  - DIMENSION = $m_{\min}$ [TO $m_{\max}$], where $1 \leq m_{\min} \leq m_{\max} < n$. Skipping the TO term leads to $m_{\min} = m$.

  - CUTOFF = k, causes data less than k to be treated as missing values. Default value is 0.
  - UNTIE specifies the primary approach to ties for LEVEL = ORDINAL.
  - NEGATIVE allows slopes or powers to be negative with LEVEL = RATIO, INTERVAL, or LOGINTERVAL.
  - COEFF = IDENTITY | DIAGONAL, yields Euclidean distances and weighted Euclidean distances ("INDSCAL"), respectively.

- The Loss Function
  - FORMULA = 0 | 1 | 2 determines how the badness-of-fit criterion is standardized. 0 fits a regression model by ordinary least squares, not for level=ordinal. 1 is Stress-1 (for FIT = DISTANCE and LEVEL = ORDINAL) or S-Stress (for FIT = SQUARED and LEVEL = ORDINAL). 2 standardizes each partition specified by CONDITION; corresponds to Stress-2 for FIT = DISTANCE and LEVEL = ORDINAL. Default is 1 unless FIT = LOG.
  - ALTERNATE = NONE | MATRIX | ROW. Determines what form of alternating least-squares algorithm is used. NONE causes all parameters to be adjusted simultaneously on each iteration; best for small n (=objects) and N (=matrices). MATRIX adjusts the parameters for the first proximity matrix, then for the second, etc.; best for large N and small n. ROW adds further stages; best for large n.
  - FIT = DISTANCE | SQUARED | LOG | n specifies a fixed transformation to apply to *both* $f(p_{ij})$s and d_{ij}s before the error is computed. This leads to different weighting of large/small values. The default is DISTANCE or, equivalently, 1 which fits $f(p_{ij})$'s to d_{ij}'s. FIT = n fits nth power $f(p_{ij})$s to nth power d_{ij}s.

- Some technical options
  - MAXITER = k, the maximum number of iterations. Default is 100.
  - CONVERGENCE = k, the convergence criterion. Default is $k = .01$. Values of less than .0001 may be impossible because of machine precision.
  - RANDOM = k, causes initial coordinate values to be pseudorandom numbers with seed=k.

- Some output options
  - PFIT and PCONFIG: print the fit values and the MDS configuration, respectively. Various other print options exists, e.g., PINIT, which prints the initial values, and PTRANS, which prints the estimated transformation parameters if any are computed in metric models.
  - OUT = xyz. Creates the SAS data file "xyz" containing the estimates of all parameters of the MDS model and the value of the badness-of-fit criterion.
  - OUTRES = xyz. Creates file that contains original proximities, MDS distances, transformed proximities/distances, residuals.

For more information, contact: SAS Institute Inc., 100 SAS Campus Drive, Cary, NC 27513-2414, U.S.A. Phone: (919) 677-8000. Fax: (919) 677-4444. Internet: http://www.sas.com

STATISTICA

STATISTICA is a comprehensive package for statistics and statistical graphics that includes an MDS module. Doing MDS, STATISTICA yields windows of numerical output, high-resolution plots of the MDS configuration (also 3D configurations that can be rotated in space), fit plots such as data vs. d-hats or data vs. rank-images, and so on. The graphics windows can be modified by changing fonts, changing the line thickness, resizing points, moving point labels as objects, and the like, or by drawing into the plots with the built-in drawing tools.

STATISTICA's MDS module uses the Guttman–Lingoes initial configuration or a user-supplied external initial configuration. It then employs the MINISSA algorithm (Roskam & Lingoes, 1981), which does ordinal MDS with rank-images in the initial iterations and monotone regression later on to ensure convergence. As fit indices, Stress and Raw Stress are computed with both monotone regression values and rank-images. The coefficient of alienation is also reported.

STATISTICA's MDS only offers ordinal MDS and the Euclidean metric. Although this is sufficient for most applications, models like interval MDS, say, are needed for some data to avoid degeneracies.

STATISTICA can be run interactively (via mouse clicks that can be recorded), by submitting a program of previously stored mouse clicks, or by executing a file of STATISTICA's SCL command language. In SCL, our politicians example is set up as shown below, assuming that a system file "proxpol.sta" has been created beforehand. There exists only one further option: ITERATIONS $= k$.

```
FILE = "c:\proxpol.sta"
MDS
 / VARIABLES = ALL
 / DIMENSIONS = 2
```

For more information, contact: StatSoft Inc., 2300 East 14th St., Tulsa, OK 74104, U.S.A. Phone: (918) 749-1119. Fax: (918) 749-2217. E-mail: info@statsoft.com. Internet: http://www.statsoft.com

SYSTAT

SYSTAT is a comprehensive package for statistics, including graphics (Wilkinson & Hill, 1994). SYSTAT can be run in batch mode, submit mode, interactive line mode, and in a pull-down menu mode. As is true for all statistics packages, it is best to first set up a system file containing the proximity matrix. System files are defined via a spreadsheet, reading data from an ASCII file, or by computing proximities internally from other data. However, data can also be input from within a command file. To do a SYSTAT MDS analysis, one calls, from within the MDS module, the system file "polit.syd", say,

by typing "USE polit" (return), followed by "MODEL var1..var5" (return) and "ESTIMATE" (return). Alternatively, one first defines a command file containing these three lines and then submits this command file. (The first three letters of the commands are sufficient.) This will do an ordinal MDS in 2D, using Euclidean distances. A more explicit batch mode command job is this:

```
BASIC
SAVE polit
TYPE=DISSIMILARITY
INPUT Humphrey McGovern Percy Wallace Nixon
DIAGONAL=ABSENT
RUN
1
5  2
7 10  9
6  8  4  3
~
MDS
USE polit
MODEL Humphrey..Nixon
ESTIMATE / LOSS=KRUSKAL, REG=MONOTONIC, DIM=2, R=2
```

If the batch job is called "mds.cmd", it is run by typing "systat < mds.cmd" from the DOS prompt. The resulting configuration and its Shepard diagram are shown on the computer screen, where they can be fine-tuned, previewed, and sent to a printer. The results can also be saved, along with distances, d-hats, and residuals for further analyses. In combination with a built-in spreadsheet editor, the points in the configuration plot can be labeled. Using this feature, one can, for example, create a facet diagram, where the points are labeled by their codings on a particular facet (see Chapter 5). Three-dimensional plots with embedded axes, labels, surfaces, and the like, are also available.

For an experienced data analyst, SYSTAT is best used with its command language and submit files. All SYSTAT jobs can be documented and easily modified if needed. However, even using pull-down menus or Windows, command language files are automatically generated for previewing and saving.

A SYSTAT MDS job generally looks like this:

```
MDS
 MODEL variables / options
 CONFIG arguments
 SAVE filename / arguments
 ESTIMATE / options
```

- The MODEL options: ROWS = N and SHAPE = RECT | SQUARE: when doing unfolding, one needs to specify that the proximity matrix is "rectangular" with N rows; the number of columns corresponds to the number of objects n.

- The CONFIG arguments: CONFIG = [coordinates of first point; coordinates of second point; ...] or CONFIG = LAST. The LAST argument allows using the configuration from the previous scaling. An example for an external 3-point configuration in 2D is CONFIG = [1 2; 3.3 5; 4 5].
- The SAVE arguments: Specifying DISTANCES saves the MDS distances; CONFIG saves the final MDS configuration; RESID saves the proximities, distances, d-hats, and residuals, together with row and column subscripts.
- The ESTIMATE options:
  - DIMENSION = m specifies the dimensionality m of the MDS solution.
  - REGRESSION = MONOTONIC | LINEAR | LOG | POWER uses ordinal, interval, log-interval, or power-function MDS, respectively.
  - SPLIT = ROW is split-by-rows (unfolding); SPLIT =MATRIX is split-by-matrix conditionality for stacked proximity matrices.
  - WEIGHT is individual differences scaling with dimension weights for each matrix.
  - R = r specifies the exponent of the Minkowski metric. For example, R = 2 requests Euclidean distances.
  - LOSS = KRUSKAL | GUTTMAN | YOUNG specifies the loss function. KRUSKAL is Stress-1; GUTTMAN is the coefficient of alienation, K; YOUNG is S-Stress.
  - ITERATIONS = k sets the maximum number of iterations to k.
  - CONVERGE = k causes MDS to stop when the maximum absolute difference between any coordinate in the solution $\mathbf{X}$ at iteration i and iteration $i + 1$ is less than k.

The program defaults are DIM =2, REGR = MONO, no split, no weight, LOSS = KRUS, R = 2, ITER = 50, DECREMENT = 0.005.

For more information, contact: Systat Software, Inc.; 501 Canal Blvd; Suite E; Point Richmond, CA 94804-2028; U.S.A. Phone: (800) 797-7401. Fax: (800) 797-7406. E-mail: info-usa@systat.com. Internet: http://www.systat.com

A.3 MDS Programs without High-Resolution Graphics

Most programs from the early days of MDS lack high-resolution graphics. Nevertheless, these programs are usually well documented in the literature and often used in simulation studies. For completeness, we discuss a few of these programs.

The Guttman–Lingoes Nonmetric PC Series

The GL Series is a collection of 32 individual programs for the analysis of qualitative and ordinal data. The philosophy of the GL Series is to have compact programs that are good for particular purposes rather

than one jumbo program that does everything. For MDS, the main program is MINISSA-I for unconditional and row/column conditional proximities, but there are also special programs for proximities that are both row and column conditional at the same time (SSA-II), for imposing a number of independent sets of order constraints onto the distances (CMDA), for individual differences scaling (PINDIS), or for representing proximities in a vector model (SSA-III). A complete documentation of the source code (FORTRAN IV) of most of the programs is available (Lingoes, 1973).

The GL programs are essentially batch programs. The user is required to "punch" his or her specifications as numbers in four-field columns of the parameter "cards" of the batch job (see below, fourth row of batch job). A typical set-up for MINISSA-I for our politicians data looks as follows.

```
SSA-I.OUT (name of file for numerical output)
SSA-I.PLT (name of file for printer-plot output)
1 MDS of five politicians ("title card")
   5   2   2   0   0   1   0   0   0   0   0   0   0   0   0
(5F3.0)
  1
  5  2
  7 10  9
  6  8  4  3
```

The various numerical entries on the fourth "card" specify the number of objects (5), the lowest MDS dimensionality (2), the highest MDS dimensionality (2), the type of proximities (0=dissimilarities), a request to print out the distance matrix (0=no), and a request to minimize Kruskal's Stress (1=yes). The remaining zeros refer to the usual defaults: Adding points to a fixed configuration (no=0, yes=1), external initial configuration (no=0, yes=1), special switch for program options (0=no special setting,1=ignore cells in SSA-I/MDS, number of column fields for SSAR-I/unfolding), ignore cells in data matrix (no=0, yes=1), missing data (no=0, yes=1), distance formula (Euclidean=0, city-block=1), differential weighting of small and large distances (global weighting=0, local weighting=1), type of analysis (SSA-I/MDS=0, SSAR-I/unfolding=1), missing data code (real value), value above/below which all input data are considered tied (real value).

Most of the GL programs can be downloaded from http://www.newmdsx.com

FSSA: *Faceted Smallest Space Analysis*

A special MDS program is FSSA by Shye (1991). FSSA is a stand-alone program that is public domain for scientists. It analyzes from 3 to 50 objects in two to ten dimensions, using the Guttman algorithm. What makes FSSA unique is that it allows one to code the objects with respect to several facets (see Chapter 5). FSSA then partitions the resulting 2D planes, facet by facet, in three ways (axial, polar, modular), using parallel straight lines, concentric circles, and rays emanating from a common origin, respectively. The partitionings are shown as screen diagrams.

The program can be obtained from: Samuel Shye, Dept. of Psychology, Hebrew University of Jerusalem, Jerusalem 91905, Israel. E-mail: msshye@pluto.mscc.huji.ac.il

KYST

KYST (Kruskal et al., 1978) is an exceedingly flexible MDS program. It allows one to set up virtually any MDS model, except MDS with side constraints on coordinates or distances. KYST is noncommercial software, available on the Internet at http://www.netlib.org/mds/. The program is written in FORTRAN and can, in principle, be compiled on any machine. KYST allows the user to specify Minkowski distances other than the Euclidean distance. Moreover, weights can be assigned to each proximity, for example, for weighting each datum by its reliability. Another feature is the possibility of polynomial regression for specifying the transformation function of the proximities. Because KYST is an old program, it provides only printer-type graphics with 132 characters per line. KYST's manual is not made for today's occasional user of MDS, but the logic of KYST's command language is straightforward.

An example setup for a KYST2e job is the following.

```
DIMMAX = 2, DIMMIN = 2
REGRESSION = ASCENDING
COORDINATES = ROTATE
ITERATIONS = 100
PRINT = HISTORY
PRINT = DATA
PLOT = SCATTER = ALL
PLOT = CONFIGURATION
TORSCA
DATA, LOWERHALFMATRIX, DIAGON = ABSENT
KYST example setup: politicians
  5  1  1
(5F3.0)
  1
  5  2
  7 10  9
  6  8  4  3
COMPUTE
STOP
```

The setup consists of control lines and the data definition lines (data deck). The order of the control lines mostly does not matter. The data definition lines start in this example with the line "DATA, LOWERHALFMATRIX,..." and end with the line " 6 8 4 3". These lines are in strict order and may not be reordered. Also, the REGRESSION control lines should precede the data definition lines. Some of the control commands on the control lines are:

- Analysis options: DIMMAX = 3 sets the maximum dimensionality to 3, DIMMIN = 2 sets the minimum dimensionality to 2. SFORM1 makes the program use Kruskal's Stress formula 1, SFORM2 requests Kruskal's Stress formula 2. PRIMARY requests

the primary approach to ties, and SECONDARY the secondary approach. ITERATIONS = 100 sets the maximum number of iterations to 100, SRATST = 0.9999 causes KYST to stop the iterations whenever the ratio of subsequent Stress values is smaller than 0.9999. STRMIN = .00001 stops the iterations if the Stress becomes smaller than .00001. TORSCA takes the classical scaling solution as the initial configuration. COORDINATES = ROTATION causes the solution to be rotated to principal axes (recommended).

- REGRESSION specifies the type of admissible transformation of the proximities. REGRESSION = ASCENDING indicates that the proximities are dissimilarities, REGRESSION = DESCENDING that they are similarities. REGRESSION = POLYNOMIAL = 3 specifies a third-degree polynomial regression transformation. Adding REGRESSION = CONSTANT (or NOCONSTANT) makes the program pick an optimal additive constant. Thus, an interval transformation is specified by REGRESSION = POLYNOMIAL = 1, REGRESSION = CONSTANT.
- Print and plot options. PRINT = (NO)DATA prints the proximities, PRINT = (NO)HISTORY outputs the history of iterations, and PRINT = (NO)DISTANCES prints the distances and the disparities. When a range of dimensions is specified, a plot of Stress against the number of dimensions is given. PLOT = SCATTER = ALL (or = NONE for no plots) produces a plot of distances and disparities against the proximities. PLOT = CONFIGURATION = ALL (or =SOME or = NONE) causes KYST to plot all principal components projection planes of the MDS configuration. SOME only plots the first principal component against all other components.
- Data definition. The data are defined by five parts: (1) a line ("card") specifying the data; (2) a title line; (3) a line of parameters; (4) a line with the format of the data; (5) lines specifying the data. Line 1 of the data definition part starts with DATA followed by LOWERHALFMATRIX to indicate the lower triangular elements, UPPERHALFMATRIX indicates the upper triangular elements, and MATRIX specifies the full matrix. For triangular matrices, one specifies either DIAGONAL = PRESENT or = ABSENT. For unfolding, one sets LOWERCORNERMATRIX (or UPPERCORNERMATRIX). Line 3 expects 3 or 4 numbers that should end in columns 3, 6, 9, and 12, respectively. The parameters specify number of objects, number of replications (usually 1), number of groups (usually 1), respectively. For corner matrices, the parameters are: number of rows; number of columns; the number of replications; number of groups. Line 4 contains a FORTRAN format, for example, (5F10.3). Line 5 (and further) contains the data, possibly followed by weights. The weights definition block is specified in the same way as the data definition block, except that the word DATA has to be replaced by WEIGHTS.
- COMPUTE causes the program to start computing. After the COMPUTE command, further MDS jobs can be set up, before a final STOP.

For more information contact: Scott M. Smith, Ph.D., Dept. of Marketing, 634 TNRB Brigham Young University, Provo, Utah 84602, U.S.A. E-mail: smsmith@byu.edu. Phone: (801) 376-1339. Fax: (801) 705-9430. Internet: http://marketing.byu.edu

MULTISCALE

MULTISCALE (Ramsay, 1977) is one of the few MDS programs that offers a Maximum Likelihood (ML) approach. For ML, it can assume normal

or lognormal distributions of the residuals and assumes that the errors are independent. MULTISCALE is one of the few programs that yields confidence regions for each point. The program has various options, including, for example, spline transformations, power transformations, and the weighted Euclidean model. The output of this program is somewhat hard to read because it contains much statistical information. However, the program has a clearly written manual. The MS-DOS version of MULTISCALE has high-resolution graphics and can output postscript plots. MULTISCALE runs on several operating systems.

A sample setup for MULTISCALE is

```
@TITLE LINES=2;
  Example setup of {\sc Multiscale}:
  similarity of 5 politicians,
@PARAMETERS NSTIM=5, NSUB=1, NDIM=2,
            TRAN=SPLINES,  DISTRIB=LOG;
@DISDATA VECTOR FORMAT=FREE;
1
5 2
7 10 9
6 8 4 3
@STIMLABS FORMAT=FREE;
Humphrey McGovern Percy Wallace Nixon
@COMPUTE;
```

MULTISCALE has its own command language. The following summarizes the most important commands. The input of MULTISCALE is organized in blocks. Each block starts with an @ and ends with a semicolon (;). For example, there exists a block for the title, a block for the parameters of the analysis, and a block for reading the data. For the most part, these blocks can appear in any order, with the following exceptions: the PARAMETER block must be the first block or the second block just after the TITLE block, and the COMPUTE block is the last block before the analysis takes place. Several runs can be specified in one file by repeating a (possibly different) PARAMETER (and TITLE) block ended by the next COMPUTE block.

Here is a short description of some of the blocks:

- The TITLE block indicates that the next line is a title. LINES = 2 specifies that the title has two lines (maximum is five lines).
- The PARAMETER block sets all the important options of the analysis. NSTIMULI defines the number of objects, NDIMENSIONS sets the number of dimensions, NSUBJECTS specifies the number of replications, NKNOTS sets the number of interior knots when using spline transformation, PROBABILITY sets the confidence level for the confidence regions for points, METRIC = IDENTITY (default) for weighting every dimension equal and METRIC = DIAGONAL for weighting each dimension separate (needed for the weighted Euclidean model). TRANSFORM sets the transformation of the proximities: the keyword SCALE sets the disparity (optimally) to the sum of the proximity and an additive constant only, POWER assumes interval level of the proximities, SPLINE specifies monotone splines. With DISTRIBUTION = LOGNORMAL, we assume that the error distribution is lognormal (default); with NORMAL, we simply obtain the Stress function. DCOMPLETE indicates that the complete proximity matrix is available for each replication instead of the lower triangular elements, LISTDATA prints input matrices, NOSTATS avoids printing statistics

for replications, NOASYMPT avoids printing asymptotic variance estimates, NODIST suppresses the matrix of distances, and TABLES prints the matrix of disparities, distances and normalized residuals for each replication. DPLOT plots the disparities against the distances, TPLOT plots the disparities against the proximities, and QPLOT plots the normalized residuals against quantiles of the normal distribution. For preference data, specify PCOMPLETE for complete preference matrix instead of lower triangular elements, PPLOT for plotting the preference against predicted preference.

- The DISDATA block inputs the proximities. MULTISCALE assumes that the proximities are entered in lower triangular format. However, VECTOR indicates that the proximities are in one large vector, not in lower triangular format. DIAGONAL specifies that diagonal values are present. FORMAT = fixed specifies that the data are read by a FORTRAN format, which is entered before the first line of data. FORMAT = FREE lets MULTISCALE read the data in free format, which means that the proximities should be separated by a space or a comma.
- The STIMLABELS block allows you to input labels of the objects. The labels should be on the lines following this block. By specifying FORMAT=FREE, the labels have to be separated by a space or a comma. In the same way, the SUBLABELS block allows you to specify labels for the replications.
- The COMPUTE block starts the analysis. ITMAX=100 sets the maximum number of iterations to 100, and CONV=.005 sets a convergence criterion dependent on the log-likelihood.

The program can be obtained free of charge from: James O. Ramsay, Dept. of Psychology, McGill University, 1205 Docteur Penfield Avenue, Montreal, Québec H4A 1B1, Canada. E-mail: ramsay@psych.mcgill.ca, Internet: ftp://ego.psych.mcgill.ca/pub/ramsay/multiscl/

Appendix B
Notation

For convenience, we summarize the notation used throughout this book. We use the following conventions: a lowercase italic character denotes a scalar, a lowercase bold character denotes a vector, and a uppercase bold character denotes a matrix. Elements of vectors or matrices are denoted by a subscripted scalar. A function is usually denoted by a character followed by an argument in parentheses, for example, $f(\mathbf{x})$ is a scalar function of the vector $\mathbf{x}$, and $\mathbf{A}(\mathbf{x})$ is a matrix function of the vector $\mathbf{x}$. Some explicit notation follows below.

n	Number of objects, persons, and so on.
i, j	Running index for objects, $i, j = 1, \ldots, n$.
m	Number of dimensions.
a	Running index for dimensions, $a = 1, \ldots, m$.
$\mathbf{X}$	Matrix of coordinates x_{ia} of n objects on m dimensions.
p_{ij}	Proximity between object i and j. It could be either a similarity or a dissimilarity measure.
δ_{ij}	Nonnegative dissimilarity between object i and j.
$\boldsymbol{\Delta}$	Symmetric matrix of nonnegative dissimilarities δ_{ij} of size $n \times n$, with $\delta_{ii} = 0$.

$d_{ij}(\mathbf{X})$ The Euclidean distance distance between row i and row j of $\mathbf{X}$; that is, $d_{ij}^2(\mathbf{X}) = \sum_{a=1}^{m} (x_{ia} - x_{ja})^2$.

d_{ij} A shorter notation for the Euclidean distance $d_{ij}(\mathbf{X})$.

$\widehat{d}_{ij}$ Disparity between objects i and j. Disparities are admissibly transformed proximities that optimally approximate given distances.

w_{ij} A nonnegative weight used to (down)weight the residual in the Stress function.

$\mathbf{W}$ Symmetric matrix of weights w_{ij} with zero diagonal.

$\mathbf{D}(\mathbf{X})$ Matrix of Euclidean distances between the rows of $\mathbf{X}$.

$\mathbf{A}'$ The transpose of $\mathbf{A}$.

$\mathbf{I}$ The identity matrix, which is a square matrix with diagonal elements equal to 1 and off-diagonal elements equal to 0.

$\mathbf{1}$ A column vector with all elements equal to 1.

$\mathbf{J}$ The $n \times n$ centering matrix, $\mathbf{J} = \mathbf{I} - n^{-1}\mathbf{1}\mathbf{1}'$, where all elements of the matrix $\mathbf{1}\mathbf{1}'$ are equal to 1.

$\mathbf{A}^q$ The qth power of a square matrix $\mathbf{A}$. For example, $\mathbf{A}^3 = \mathbf{A}\mathbf{A}\mathbf{A}$.

$\mathbf{A}^{-1}$ The matrix inverse of a square matrix $\mathbf{A}$ assuming that $\mathbf{A}$ is of full rank, so that $\mathbf{A}^{-1}\mathbf{A} = \mathbf{A}\mathbf{A}^{-1} = \mathbf{I}$.

$\mathbf{A}^-$ A generalized inverse of a square matrix $\mathbf{A}$ where $\mathbf{A}$ may be rank deficient, so that $\mathbf{A}\mathbf{A}^-\mathbf{A} = \mathbf{A}$ and $\mathbf{A}^-\mathbf{A}\mathbf{A}^- = \mathbf{A}^-$ holds. Usually, we choose the Moore–Penrose inverse $\mathbf{A}^+$ as the generalized inverse.

$\mathbf{A}^{(2)}$ Matrix $\mathbf{A}$ with squared elements.

$\mathbf{A}(\mathbf{X})$ Any matrix function of $\mathbf{X}$ with elements $a_{ij}(\mathbf{X})$.

tr $\mathbf{A}$ The trace operator sums the diagonal elements of $\mathbf{A}$; that is, tr $\mathbf{A} = \sum_{i=1}^{n} a_{ii}$.

$\widehat{\phi}(\mathbf{x}, \mathbf{y})$ Majorizing function of $\phi(\mathbf{x})$ for which $\phi(\mathbf{x}) \leq \widehat{\phi}(\mathbf{x}, \mathbf{y})$ and $\phi(\mathbf{x}) = \widehat{\phi}(\mathbf{x}, \mathbf{x})$ holds for all feasible $\mathbf{x}$ and $\mathbf{y}$.

$\|\mathbf{X}\|$ The Euclidean norm of matrix $\mathbf{X}$; that is, $\|\mathbf{X}\|^2 = \sum_{i=1}^{n} \sum_{a=1}^{m} x_{ia}^2$.

$\|\mathbf{X}\|_{\mathbf{V}}$ The weighted Euclidean norm of matrix $\mathbf{X}$; that is, $\|\mathbf{X}\|_{\mathbf{V}}^2 = \text{tr } \mathbf{X}'\mathbf{V}\mathbf{X}$.

σ_r Raw Stress, that is, the sum of the squared differences between the optimally transformed proximities $f(p_{ij})$ (i.e., the disparities $\widehat{d}_{ij}$) and the corresponding distances $d_{ij}(\mathbf{X})$ of the MDS configuration $\mathbf{X}$.

σ_n Normalized Stress, that is, raw Stress divided by the sum of squared dissimilarities or disparities.

σ_1 Stress formula 1, that is, the square root of raw Stress divided by the sum of squared distances.

σ_2 Stress formula 2, that is, the square root of raw Stress divided by the sum of squares of the distances minus the average distance.

References

Abelson, R. P., & Sermat, V. (1962). Multidimensional scaling of facial expressions. *Journal of Experimental Psychology*, *63*, 546–554.

Ahrens, H. J. (1972). Zur Verwendung des Metrikparameters multidimensionaler Skalierungen bei der Analyse von Wahrnehmungsstrukturen. *Zeitschrift für experimentelle und angewandte Psychologie*, *19*, 173–195.

Ahrens, H. J. (1974). *Multidimensionale Skalierung.* Weinheim, Germany: Beltz.

Andrews, F. M., & Inglehart, R. F. (1979). The structure of subjective well-being in nine Western societies. *Social Indicators Research*, *6*, 73–90.

Arabie, P. (1991). Was Euclid an unnecessarily sophisticated psychologist? *Psychometrika*, *56*, 567–587.

Attneave, F. (1950). Dimensions of similarity. *American Journal of Psychology*, *3*, 515–556.

Bagozzi, R. P. (1993). Assessing construct validity in personality research: Applications to measures of self-esteem. *Journal of Research in Personality*, *27*, 49–87.

Bailey, R. A., & Gower, J. C. (1990). Approximating a symmetric matrix. *Psychometrika*, *55*, 665–675.

Baird, J. C., & Noma, E. (1978). *Fundamentals of scaling and psychophysics.* New York: Wiley.

Barnes, S. H., Kaase, M., Allerback, K. R., Farah, B., Heunks, F., Inglehart, R., Jennings, M. K., Klingemann, H. D., Marsh, A., & Rosenmayr, L. (Eds.). (1979). *Political action: Mass participation in five western democracies.* Beverly Hills, CA: Sage.

Beaton, A. E., & Tukey, J. (1974). The fitting of power series, meaningful polynomials, illustrated on band-spectroscopic data. *Technometrics*, *16*, 147–185.

Bell, D. R., & Lattin, J. M. (1998). Shopping behavior and consumer preference for store price format: Why 'large basket' shoppers prefer EDLP. *Marketing Science*, *17*, 66–88.

Bendixen, M. (1996). A practical guide to the use of correspondence analysis in marketing research. *Marketing Research On-Line*, *1*, 16–38.

Bentler, P. M., & Weeks, D. G. (1978). Restricted multidimensional scaling models. *Journal of Mathematical Psychology*, *17*, 138–151.

Benzécri, J. P., et al.. (1973). *L'Analyse des données.* Paris: Dunod.

Beuhring, T., & Cudeck, R. (1985). *Development of the culture fair interest inventory: Experimental junior version* (Tech. Rep. No. Final Report on Project No. 85/1). Pretoria, South Africa: Human Sciences Research Council.

Bijleveld, C. C. J. H., & De Leeuw, J. (1991). Fitting longitudinal reduced-rank regression models by alternating least squares. *Psychometrika*, *56*, 433–447.

Bijmolt, T. H. A., & Wedel, M. (1995). The effects of alternative methods of collecting similarity data for multidimensional scaling. *Research in Marketing*, *12*, 363–371.

Bilsky, W., Borg, I., & Wetzels, P. (1994). Assessing conflict tactics in close relationships: A reanlysis of a research instrument. In J. J. Hox, P. G. Swanborn, & G. J. Mellenbergh (Eds.), *Facet theory: Analysis and design* (pp. 39–46). Zeist, The Netherlands: Setos.

Blasius, J., & Greenacre, M. J. (1998). *Visualization of categorical data.* San Diego: Academic.

Bloxom, B. (1968). *Individual differences in multidimensional scaling* (Tech. Rep. No. ETS RM 68-45). Princeton, NJ: Educational Testing Service.

Bloxom, B. (1978). Constrained multidimensional scaling in n spaces. *Psychometrika*, *43*, 397–408.

Blurton Jones, N. G. (1968). Observations and experiments on causation of threat displays of the great tit (parus major). *Animal Behaviour Monographs*, *27*, 74–158.

Boender, C. G. E. (1984). *The generalized multinomial distribution: A Bayesian analysis and applications.* Unpublished doctoral dissertation, Erasmus University, Rotterdam.

Borg, I. (1977a). Some basic concepts of facet theory. In J. C. Lingoes, E. E. Roskam, & I. Borg (Eds.), *Geometric representations of individual differences.* Ann Arbor, Michigan: Mathesis.

Borg, I. (1977b). SFIT: Matrix fitting when points correspond in some substantive sense only. *Journal of Marketing Research*, *14*, 556–558.

Borg, I. (1978a). Einige metrische und nichtmetrische Untersuchungen über den Gesamtzusammenhang der Items im ABB. In O. Neuberger & M. Allerbeck (Eds.), *Messung und Analyse von Arbeitszufriedenheit.* Bern, Switzerland: Huber.

Borg, I. (1978b). PAL: Point-wise alienation coefficients in multidimensional scaling. *Journal of Marketing Research*, *15*, 478–479.

Borg, I. (1979). Ein Verfahren zur Analyse metrischer asymmetrischer Proximitätsmatrizen. *Archiv für Psychologie*, *131*, 183–194.

Borg, I. (1988). Revisiting Thurstone's and Coombs' scales on the seriousness of crimes and offences. *European Journal of Social Psychology*, *18*, 53–61.

Borg, I. (1999). MTMM, FT, MDS, and FA. In R. Meyer Schweizer (Ed.), *Proceedings of the seventh facet theory conference* (pp. 3–18). Bern, Switzerland: Bern University.

Borg, I., & Bergermaier, R. (1981). Some comments on The structure of subjective wellbeing in nine Western societies by Andrews and Inglehart. *Social Indicators Research*, *9*, 265–278.

Borg, I., & Bergermaier, R. (1982). Degenerationsprobleme im Unfolding und ihre Lösung. *Zeitschrift für Sozialpsychologie*, *13*, 287–299.

Borg, I., & Groenen, P. J. F. (1995). Asymmetries in multidimensional scaling. In F. Faulbaum (Ed.), *Softstat '95* (pp. 31–35). Stuttgart: Gustav Fischer.

Borg, I., & Groenen, P. J. F. (1997). Multitrait-multimethod by multidimensional scaling. In F. Faulbaum & W. Bandilla (Eds.), *Softstat '97* (pp. 59–66). Stuttgart: Lucius.

Borg, I., & Groenen, P. J. F. (1998). Regional interpretations in multidimensional scaling. In J. Blasius & M. Greenacre (Eds.), *Visualizing categorical data.* New York: Academic.

Borg, I., & Leutner, D. (1983). Dimensional models for the perception of rectangles. *Perception and Psychophysics*, *34*, 257–269.

Borg, I., & Leutner, D. (1985). Measuring the similarity of MDS configurations. *Multivariate Behavioral Research*, *20*, 325–334.

Borg, I., & Lingoes, J. (1987). *Multidimensional similarity structure analysis.* Berlin: Springer.

Borg, I., & Lingoes, J. C. (1977). Geometric representations of individual differences. In J. C. Lingoes (Ed.), *Geometric representations of relational data* (1st ed., pp. 216–284). Ann Arbor, MI: Mathesis.

Borg, I., & Lingoes, J. C. (1980). A model and algorithm for multidimensional scaling with external constraints on the distances. *Psychometrika*, *45*, 25–38.

Borg, I., Schönemann, P. H., & Leutner, D. (1982). Merkmalsüberdeterminierung und andere Artefakte bei der Wahrnehmung einfacher geometrischer Figuren. *Zeitschrift für experimentelle und angewandte Psychologie*, *24*, 531–544.

Borg, I., & Shye, S. (1995). *Facet theory: Form and content.* Newbury Park, CA: Sage.

Borg, I., & Staufenbiel, T. (1984). Zur Beziehung von optimalem Stress-Wert und richtiger Minkowski-Metrik. *Zeitschrift für experimentelle und angewandte Psychologie*, *31*, 376–390.

Borg, I., & Staufenbiel, T. (1986). The MBR metric. *Journal of Mathematical Psychology*, *30*, 81—84.

Borg, I., & Staufenbiel, T. (1993). Facet theory and design for attitude measurement and its application. In D. Krebs & P. Schmidt (Eds.), *New directions in attitude measurement* (pp. 206–237). New York: De Gruyter.

Borg, I., & Tremmel, L. (1988). Compression effects in pairwise ratio scaling. *Methodika*, *2*, 1–15.

Bortz, J. (1974). Kritische Bemerkungen über den Einsatz nichteuklidischer Metriken im Rahmen der multidimensionalen Skalierung. *Archiv für Psychologie*, *126*, 194–212.

Bove, G., & Critchley, F. (1993). Metric multidimensional scaling for asymmetric proximities when the asymmetry is one-dimensional. In R. Steyer, K. F. Wender, & K. F. Widamann (Eds.), *Pcychometric methodology: Proceedings of the 7th European Meeting of the Psychometric Society in Trier* (pp. 55–60). Stuttgart: Gustav Fischer Verlag.

Broderson, U. (1968). *Intra- und interindividuelle mehrdimensionale Skalierung eines nach objektiven Kriterien variierenden Reizmaterials.* Unpublished master's thesis, Christian-Albrechts-Universität.

Brokken, F. B. (1983). Orthogonal Procrustes rotation maximizing congruence. *Psychometrika*, *48*, 343–352.

Browne, M. W. (1969). On oblique procrustes rotations. *Psychometrika*, *34*, 375–394.

Browne, M. W. (1972a). Oblique rotation to a partially specified target. *British Journal of Mathematical and Statistical Psychology*, *25*, 207–212.

Browne, M. W. (1972b). Orthogonal rotation to a partially specified target. *British Journal of Mathematical and Statistical Psychology*, *25*, 115–120.

Browne, M. W. (1987). The Young-Housholder algorithm and the least squares multidimensional scaling of squared distances. *Journal of Classification*, *4*, 175–190.

Browne, M. W., & Kristof, W. (1969). On the oblique rotation of a factor matrix to a specified pattern. *Psychometrika*, *34*, 237–248.

Brusco, M. J. (2001). A simulation annealing heuristic for unidimensional and multidimensional (city-block) scaling of symmetric proximity matrices. *Journal of Classification*, *18*, 3–33.

Brusco, M. J. (2002). Integer programming methods for seriation and unidimensional scaling of proximity matrices: A review and some extensions. *Journal of Classification*, *19*, 45–67.

Brusco, M. J., & Stahl, S. (2000). Using quadratic assignment methods to generate initial permutations for least quares unidimensional scaling of symmetric proximity matrices. *Journal of Classification*, *17*, 197–223.

Buja, A., Logan, B. F., Reeds, J. R., & Shepp, L. A. (1994). Inequalities and positive-definite functions arising from a problem in multidimensional scaling. *The Annals of Statistics*, *22*, 406–438.

Buja, A., & Swayne, D. F. (2002). Visualization methodology for multidimensional scaling. *Journal of Classification*, *19*, 7–44.

Burton, M. L. (1975). Dissimilarity measures for unconstained sorting data. *Multivariate Behavioral Research*, *10*, 409–424.

Busing, F. M. T. A. (2004). Personal communication.

Busing, F. M. T. A. (2005). Avoiding degeneracy in metric unfolding by penalizing the intercept. (Unpublished manuscript)

Busing, F. M. T. A., Groenen, P. J. F., & Heiser, W. J. (2005). Avoiding degeneracy in multidimensional unfolding by penalizing on the coefficient of variation. *Psychometrika*, *70*, 71–98.

Cailliez, F. (1983). The analytical solution to the additive constant problem. *Psychometrika*, *48*, 305–308.

Carroll, J. D. (1972). Individual differences and multidimensional scaling. In R. N. Shepard, A. K. Romney, & S. B. Nerlove (Eds.), *Geometric representations of individual differences.* New York: Academic.

Carroll, J. D. (1980). Models and methods for multidimensional analysis of preferential choice or other dominance data. In E. D. Lantermann & H. Feger (Eds.), *Similarity and choice.* Bern, Switzerland: Huber.

Carroll, J. D., & Arabie, P. (1980). Multidimensional scaling. *Annual Review of Psychology*, *31*, 607–649.

Carroll, J. D., & Chang, J. J. (1970). Analysis of individual differences in multidimensional scaling via an N-way generalization of 'Eckart-Young' decomposition. *Psychometrika*, *35*, 283–320.

Carroll, J. D., & Chang, J. J. (1972, March). *IDIOSCAL (Individual Differences In Orientation SCALing): A generalization of INDSCAL allowing idiosyncratic reference systems as well as an analytic approximation to INDSCAL.* Paper presented at the spring meeting of the Psychometric Society. Princeton, NJ.

Carroll, J. D., Green, P. E., & Carmone, F. J. (1976). *CANDELINC: A new method for multidimensional scaling with constrained solutions.* Paper presented at the International Congress of Psychology. Paris.

Carroll, J. D., & Wish, M. (1974a). Models and methods for three-way multidimensional scaling. In D. H. Krantz, R. C. Atkinson, R. D. Luce, & P. Suppes (Eds.), *Contemporary developments in mathematical psychology* (Vol. II, pp. 57–105). San Francisco: Freeman.

Carroll, J. D., & Wish, M. (1974b). Multidimensional perceptual models and measurement methods. In E. C. Carterette & M. P. Friedman (Eds.), *Handbook of perception* (Vol. II, pp. 391–447). New York: Academic.

Chang, J. J., & Carroll, J. D. (1969). *How to use MDPREF, a computer program for multidimensional analysis of preference data.* Computer Manual. Murray Hill, NJ: Bell Labs.

Chaturvedi, A., & Carroll, J. D. (1994). An alternating combinatorial optimization approach to fitting the INDCLUS and generalized INDCLUS models. *Journal of Classification*, *11*, 155–170.

Cliff, N. (1966). Orthogonal rotation to congruence. *Psychometrika*, *31*, 33–42.

Cliff, N. (1973). Scaling. *Annual Review of Psychology*, *24*, 473–506.

Cohen, H. S., & Davison, M. L. (1973). Jiffy-scale: A FORTRAN IV program for generating Ross-ordered pair comparisons. *Behavioral Science*, *18*, 76.

Cohen, H. S., & Jones, L. E. (1973). The effects of random error and subsampling of dimensions on recovery of configurations by non-metric multidimensional scaling. *Psychometrika*, *39*, 69–90.

Commandeur, J. J. F. (1991). *Matching configurations.* Leiden, The Netherlands: DSWO.

Commandeur, J. J. F. (1993). *Missing data in the distance approach to principal component analysis* (Tech. Rep. No. RR-92-07). Leiden, The Netherlands: Department of Data Theory, Leiden University.

Commandeur, J. J. F., & Heiser, W. J. (1993). *Mathematical derivations in the proximity scaling (PROXSCAL) of symmetric data matrices* (Tech. Rep. No. RR-93-03). Leiden, The Netherlands: Department of Data Theory, Leiden University.

Constantine, A. G., & Gower, J. C. (1978). Graphic representations of asymmetric matrices. *Applied Statistics*, *27*, 297–304.

Coombs, C. H. (1950). Psychological scaling without a unit of measurement. *Psychological Review*, *57*, 145–158.

Coombs, C. H. (1964). *A theory of data.* New York: Wiley.

Coombs, C. H. (1967). Thurstone's measurement of social values revisited forty years later. *Journal of Personality and Social Psychology*, *6*, 85–91.

Coombs, C. H. (1975). A note on the relation between the vector model and the unfolding model for preferences. *Psychometrika*, *40*, 115–116.

Coombs, C. H., & Avrunin, G. S. (1977). Single-peaked functions and the theory of preference. *Psychological Review*, *84*, 216–230.

Coombs, C. H., Dawes, R. M., & Tversky, A. (1970). *Mathematical psychology: An elementary introduction.* Englewood-Cliffs, NJ: Prentice-Hall. (Reprinted by Mathesis, Ann Arbor, MI, 1981)

Cooper, L. G. (1972). A new solution to the additive constant problem in metric multidimensional scaling. *Psychometrika*, *37*, 311–322.

Cox, M. A. A., & Cox, T. F. (1992). Interpretation of Stress in non-metric multidimensional scaling. *Statistica Applicata*, *4*, 611–618.

Cox, T. F., & Cox, M. A. A. (1990). Interpreting Stress in multidimensional scaling. *Journal of Statistical Computation and Simulation*, *37*, 211–223.

Cox, T. F., & Cox, M. A. A. (1991). Multidimensional scaling on a sphere. *Communications in Statistics A, Theory and Methods*, *20*, 2943–2953.

Cox, T. F., & Cox, M. A. A. (1994). *Multidimensional scaling.* London: Chapman & Hall.

Coxon, A. P. M., & Jones, C. L. (1978). *The images of occupational prestige: A study in social cognition.* London: Macmillan.

Critchley, F. (1986). Dimensionality theorems in multidimensional scaling and hierarchical cluster analysis. In E. Diday, Y. Escoufier, L. Lebart, J. Lepage, Y. Schektman, & R. Tomassone (Eds.), *Informatics* (Vol. IV, pp. 45–70). Amsterdam: North-Holland.

Critchley, F., & Fichet, B. (1994). The partial order by inclusion of the principal classes of dissimilarity on a finite set and some of their basic properties. In B. V. Cutsem (Ed.), *Classification and dissimilarity analysis* (Vol. 93, pp. 5–65). New York: Springer.

Cross, D. V. (1965a). *Metric properties of multidimensional stimulus control.* Unpublished doctoral dissertation, University of Michigan.

Cross, D. V. (1965b). Metric properties of multidimensional stimulus generalization. In D. I. Mostofsky (Ed.), *Stimulus generalization* (pp. 72–93). San Francisco: Stanford University Press.

Daniels, H. E. (1944). The relation between measures of correlation in the universe of sample permutations. *Biometrika*, *33*, 129–135.

Davison, M. L. (1983). *Multidimensional scaling.* New York: Wiley.

De Boor, C. (1978). *A practical guide to splines.* New York: Springer.

Defays, D. (1978). A short note on a method of seriation. *British Journal of Mathematical and Statistical Psychology*, *31*, 49–53.

De Leeuw, J. (1977). Applications of convex analysis to multidimensional scaling. In J. R. Barra, F. Brodeau, G. Romier, & B. van Cutsem (Eds.), *Recent developments in statistics* (pp. 133–145). Amsterdam, The Netherlands: North-Holland.

De Leeuw, J. (1984). Differentiability of Kruskal's Stress at a local minimum. *Psychometrika*, *49*, 111–113.

De Leeuw, J. (1988). Convergence of the majorization method for multidimensional scaling. *Journal of Classification*, *5*, 163–180.

De Leeuw, J. (1993). *Fitting distances by least squares* (Tech. Rep. No. 130). Los Angeles, CA: Interdivisional Program in Statistics, UCLA.

De Leeuw, J. (1994). Block relaxation algorithms in statistics. In H.-H. Bock, W. Lenski, & M. M. Richter (Eds.), *Information systems and data analysis* (pp. 308–324). Berlin: Springer.

De Leeuw, J., & Groenen, P. J. F. (1997). Inverse multidimensional scaling. *Journal of Classification*, *14*, 3–21.

De Leeuw, J., & Heiser, W. J. (1977). Convergence of correction-matrix algorithms for multidimensional scaling. In J. C. Lingoes, E. E. Roskam, & I. Borg (Eds.), *Geometric representations of relational data* (pp. 735–752). Ann Arbor, MI: Mathesis.

De Leeuw, J., & Heiser, W. J. (1980). Multidimensional scaling with restrictions on the configuration. In P. R. Krishnaiah (Ed.), *Multivariate analysis* (Vol. V, pp. 501–522). Amsterdam, The Netherlands: North-Holland.

De Leeuw, J., & Heiser, W. J. (1982). Theory of multidimensional scaling. In P. R. Krishnaiah & L. N. Kanal (Eds.), *Handbook of statistics* (Vol. 2, pp. 285–316). Amsterdam, The Netherlands: North-Holland.

De Leeuw, J., & Stoop, I. (1984). Upper bounds of Kruskal's Stress. *Psychometrika*, *49*, 391–402.

De Rooij, M., & Heiser, W. J. (2000). Triadic distance models for the analysis of asymmetric three-way proximity data. *British Journal of Mathematical and Statistical Psychology*, *53*, 99–119.

DeSarbo, W. S., Manrai, A. K., & Manrai, L. A. (1994). Latent class multidimensional scaling: A review of recent developments in the marketing and psychometric literature. In R. P. Bagozzi (Ed.), *Advanced methods of marketing research* (pp. 190–222). Cambridge, MA: Blackwell.

DeSarbo, W. S., & Rao, V. R. (1984). GENFOLD2: A set of models and algorithms for the GENeral UnFOLDing analysis of preference/dominance data. *Journal of Classification*, *1*, 147–186.

De Soete, G., Carroll, J. D., & Chaturvedi, A. D. (1993). A modified CANDECOMP method for fitting the extended INDSCAL model. *Journal of Classification*, *10*, 75–92.

De Soete, G., Hubert, L., & Arabie, P. (1988). On the use of simulated annealing for combinatorial data analysis. In W. Gaul & M. Schader (Eds.), *Data, expert knowledge and decisions* (pp. 329–340). Berlin: Springer.

Diederich, G., Messick, S. J., & Tucker, L. R. (1957). A general least squares solution for successive intervals. *Psychometrika*, *22*, 159–173.

Dijksterhuis, G., & Gower, J. C. (1991). The interpretation of generalized Procrustes analysis and allied methods. *Food Quality and Preference*, *3*, 67–87.

Drösler, J. (1979). Foundations of multidimensional metric scaling in Cayley-Klein geometries. *British Journal of Mathematical and Statistical Psychology*, *32*, 185–211.

Drösler, J. (1981). The empirical validity of multidimensional scaling. In I. Borg (Ed.), *Multidimensional data representations: when and why* (pp. 627–651). Ann Arbor, MI: Mathesis.

Dunn-Rankin, P. (1983). *Scaling methods.* Hillsdale, NJ: Lawrence Erlbaum.

Dunn-Rankin, P., Knezek, G. A., Wallace, S., & Zhang, S. (2004). *Scaling methods.* Manwah, NJ: Lawrence Erlbaum.

Eckart, C., & Young, G. (1936). Approximation of one matrix by another of lower rank. *Psychometrika*, *1*, 211–218.

Eisler, H., & Lindman, R. (1990). Representations of dimensional models of similarity. In H. G. Geisler (Ed.), *Psychophysical explorations of mental structures* (pp. 165–171). Toronto: Hogrefe & Huber.

Eisler, H., & Roskam, E. E. (1977). Multidimensional similarity: An experimental and theoretical comparison of vector, distance, and set theoretical models. *Acta Psychologica*, *41*, 1–46 and 335–363.

Ekehammar, B. (1972). Multidimensional scaling according to different vector models for subjective similarity. *Acta Psychologica*, *36*, 79–84.

Ekman, G. (1954). Dimensions of color vision. *Journal of Psychology*, *38*, 467–474.

Ekman, G., Engen, T., Künnapas, T., & Lindman, R. (1964). A quantitative principle of qualitative similarity. *Journal of Experimental Psychology*, *68*, 530–536.

Ekman, G., Goude, G., & Waern, Y. (1961). Subjective similarity in two perceptual continua. *Journal of Experimental Psychology*, *61*, 222–227.

Ekman, G., & Lindman, R. (1961). *Multidimensional ratio scaling and multidimensional similarity* (Reports from the Psychological Laboratories No. 103). University of Stockholm.

Ekman, G. A. (1963). A direct method for multidimensional ratio scaling. *Psychometrika*, *28*, 3–41.

Elizur, D., Borg, I., Hunt, R., & Magyari-Beck, I. (1991). The structure of work values: A cross cultural comparison. *Journal of Organizational Behavior*, *12*, 21–38.

Engen, T., Levy, N., & Schlosberg, H. (1958). The dimensional analysis of a new series of facial expressions. *Journal of Experimental Psychology*, *55*, 454–458.

England, G., & Ruiz-Quintanilla, S. A. (1994). How working is defined: Structure and stability. In I. Borg & S. Dolan (Eds.), *Proceedings of the Fourth International Conference on Work and Organizational Values* (pp. 104–113). Montreal: ISSWOV.

Fechner, G. T. (1860). *Elemente der Psychophysik* (Vol. I and II). Leipzig, Germany: Breitkopf und Hartel.

Feger, H. (1980). Einstellungsstruktur und Einstellungsänderung: Ergebnisse, Probleme und ein Komponentenmodell der Einstellungsobjekte. *Zeitschrift für Sozialpsychologie*, *10*, 331–349.

Fichet, B. (1994). Dimensionality problems in l_1-norm representations. In B. V. Cutsem (Ed.), *Classification and dissimilarity analysis* (Vol. 93, pp. 201–224). New York: Springer.

Fischer, W., & Micko, H. C. (1972). More about metrics of subjective spaces and attention distributions. *Journal of Mathematical Psychology*, *9*, 36–54.

Fletcher, R. (1987). *Practical methods of optimization.* Chichester, England: Wiley.

Foa, U. (1965). New developments in facet design and analysis. *Psychological Review*, *72*, 262–238.

Foa, U. G. (1958). The contiguity principle in the structure of interpersonal relations. *Human Relations*, *11*, 229–238.

Fréchet, M. (1910). Les dimensions d'un ensemble abstrait. *Math. Ann.*, *68*, 145–168.

Gabriel, K. R. (1971). The biplot-graphic display of matrices with applications to principal components analysis. *Biometrika*, *58*, 453–467.

Gaffke, N., & Mathar, R. (1989). A cyclic projection algorithm via duality. *Metrika*, *36*, 29–54.

Galinat, W., & Borg, I. (1987). On symbolic temporal information: beliefs about the experience of duration. *Memory and Cognition*, *15*, 308–317.

Garmize, L. M., & Rychlak, J. F. (1964). Role-play validation of a sociocultural theory of symbolism. *Journal of Consulting Psychology*, *28*, 107–115.

Garner, W. R. (1962). *Uncertainty and structure as psychological concepts.* New York: Wiley.

Garner, W. R. (Ed.). (1974). *The processing of information and structure.* Potomac, MD: Erlbaum.

Gati, I., & Tversky, A. (1982). Representations of qualitative and quantitative dimensions. *Journal of Experimental Psychology: Human Perception and Performance*, *8*, 325–340.

Gifi, A. (1990). *Nonlinear multivariate analysis.* Chichester, England: Wiley.

Gilula, Z., & Haberman, S. J. (1986). Canonical analysis of contingency tables by maximum likelihood. *Journal of the American Statistical Association*, *81*, 780–788.

Gleason, T. C. (1967). *A general model for nonmetric multidimensional scaling.* (University of Michigan, unpublished mimeograph)

Glunt, W., Hayden, T. L., & Liu, W.-M. (1991). The embedding problem for predistance matrices. *Bulletin of Mathematical Biology*, *53*, 769–796.

Glunt, W., Hayden, T. L., & Raydan, M. (1993). Molecular confirmations from distance matrices. *Journal of Computational Chemistry*, *14*, 114–120.

Glushko, R. J. (1975). Pattern goodness and redundancy revisited: Multidimensional scaling and hierarchical clustering analyses. *Perception & Psychophysics*, *17*, 158–162.

Goodman, L. A. (1985). The analysis of cross-classified data having ordered and/or unordered categories: Association models, correlation models and asymmetry models for contingency tables with or without missing values. *The Annals of Statistics*, *13*, 10–69.

Goodman, L. A. (1986). Some useful extensions to the usual correspondence analysis approach and the usual loglinear approach in the analysis of contingency tables (with comments). *International Statistical Review*, *54*, 243–309.

Goude, G. (1972). A multidimensional scaling approach to the perception of art. *Scandinavian Journal of Psychology*, *13*, 258–271 and 272–284.

Gower, J. C. (1966). Some distance properties of latent root and vector methods used in multivariate analysis. *Biometrika*, *53*, 325–338.

Gower, J. C. (1971). A general coefficient of similarity and some of its properties. *Biometrics*, *27*, 857–874.

Gower, J. C. (1975). Generalized Procrustes analysis. *Psychometrika*, *40*, 33–51.

Gower, J. C. (1977). The analysis of asymmetry and orthogonality. In F. Brodeau, G. Romier, & B. Van Cutsem (Eds.), *Recent developments in statistics* (pp. 109–123). Amsterdam: North-Holland.

Gower, J. C. (1985). Measures of similarity, dissimilarity, and distance. In S. Kotz, N. L. Johnson, & C. B. Read (Eds.), *Encyclopedia of statistical sciences* (Vol. 5, pp. 397–405). New York: Wiley.

Gower, J. C., & Dijksterhuis, G. B. (2004). *Procrustes problems.* Oxford: Oxford University Press.

Gower, J. C., & Groenen, P. J. F. (1991). Applications of the modified Leverrier-Faddeev algorithm for the construction of explicit matrix spectral decompositions and inverses. *Utilitas Mathematica*, *40*, 51–64.

Gower, J. C., & Hand, D. J. (1996). *Biplots.* London: Chapman & Hall.

Gower, J. C., & Legendre, P. (1986). Metric and Euclidean properties of dissimilarity coefficients. *Journal of Classification*, *3*, 5–48.

Graef, J., & Spence, I. (1979). Using distance information in the design of large multidimensional scaling experiments. *Psychological Bulletin*, *86*, 60–66.

Gramlich, G. (2004). *Anwendungen der linearen Algebra.* München, Germany: Fachbuchverlag Leipzig.

Green, B. F. (1952). The orthogonal approximation of an oblique structure in factor analysis. *Psychometrika*, *17*, 429–440.

Green, P. E. (1974). *On the robustness of multidimensional scaling techniques.* Unpublished paper. University of Pennsylvania, Wharton Business School.

Green, P. E., & Carmone, F. J. (1970). *Multidimensional scaling and related techniques in marketing analysis.* Boston, MA: Allyn & Bacon.

Green, P. E., & Carroll, R. D. (1976). *Mathematical tools for applied multivariate analysis.* New York: Academic.

Green, P. E., & Rao, V. (1972). *Applied multidimensional scaling.* Hinsdale, IL: Dryden.

Green, P. E., & Wind, Y. (1973). *Multivariate decisions in marketing: A measurement approach.* Hinsdale, IL: Dryden.

Greenacre, M., & Blasius, J. (Eds.). (1994). *Correspondence analysis in the social sciences.* London: Academic.

Greenacre, M. J. (1984). *Theory and applications of correspondence analysis.* New York: Academic.

Greenacre, M. J. (1994). *Correspondence analysis in practice.* London: Academic.

Greenberg, M. J. (1980). *Euclidean and non-Euclidean geometries.* San Francisco: Freeman.

Groenen, P. J. F. (1993). *The majorization approach to multidimensional scaling: Some problems and extensions.* Leiden, The Netherlands: DSWO.

Groenen, P. J. F., De Leeuw, J., & Mathar, R. (1996). Least squares multidimensional scaling with transformed distances. In W. Gaul & D. Pfeifer (Eds.), *Studies in classification, data analysis, and knowledge organization* (pp. 177–185). Berlin: Springer.

Groenen, P. J. F., & Franses, P. H. (2000). Visualizing time-varying correlations across stock markets. *Journal of Empirical Finance*, *7*, 155–172.

Groenen, P. J. F., & Gifi, A. (1989). *Anacor* (Tech. Rep. No. UG-89-01). Leiden, The Netherlands: Department of Data Theory, Leiden University.

Groenen, P. J. F., & Heiser, W. J. (1991). *An improved tunneling function for finding a decreasing series of local minima* (Tech. Rep. No. RR-91-06). Leiden, The Netherlands: Department of Data Theory, Leiden University.

Groenen, P. J. F., & Heiser, W. J. (1996). The tunneling method for global optimization in multidimensional scaling. *Psychometrika*, *61*, 529–550.

Groenen, P. J. F., & Heiser, W. J. (2000). Iterative majorization algorithms in statistical computing. *Journal of Computational and Graphical Statistics*, *9*, 44–48.

Groenen, P. J. F., Heiser, W. J., & Meulman, J. J. (1999). Global optimization in least-squares multidimensional scaling by distance smoothing. *Journal of Classification*, *16*, 225–254.

Groenen, P. J. F., Mathar, R., & Heiser, W. J. (1995). The majorization approach to multidimensional scaling for Minkowski distances. *Journal of Classification*, *12*, 3–19.

Groenen, P. J. F., Mathar, R., & Trejos, J. (2000). Global optimization methods for multidimensional scaling applied to mobile communication. In W. Gaul, O. Opitz, & M. Schader (Eds.), *Data analysis* (p. 459-469). Heidelberg: Springer.

Groenen, P. J. F., & Meulman, J. J. (2004). A comparison of the ratio of variances in distance-based and classicial multivariate analysis. *Statistica Neerlandica*, *58*, 428–439.

Groenen, P. J. F., & Van de Velden, M. (2004). Inverse correspondence analysis. *Linear Algebra and its Applications*, *388*, 221–238.

Gulliksen, H. (1946). Paired comparisons and the logic of measurement. *Psychological Review*, *53*, 199–213.

Guthrie, J. T. (1973). Models of reading and reading disability. *Journal of Educational Psychology*, *65*, 9–18.

Guttman, L. (1941). The quantification of a class of attributes: A theory and method of scale construction. In P. Horst et al. (Eds.), *The prediction of personal adjustment* (pp. 319–348). New York: Social Science Research Council.

Guttman, L. (1954). A new approach to factor analysis: The radex. In P. Lazarsfeld (Ed.), *Mathematical thinking in the behavioral sciences* (pp. 258–348). New York: Free Press.

Guttman, L. (1959). A structural theory for intergroup beliefs and attitude. *American Sociological Review*, *24*, 318–328.

Guttman, L. (1965). A faceted definition of intelligence. *Scripta Hierosolymitana*, *14*, 166–181.

Guttman, L. (1968). A general nonmetric technique for finding the smallest coordinate space for a configuration of points. *Psychometrika*, *33*, 469–506.

Guttman, L. (1971). Measurement as structural theory. *Psychometrika*, *36*, 329–347.

Guttman, L. (1976). *Faceted SSA* (Tech. Rep.). Jerusalem, Israel: The Israel Institute of Applied Social Research.

Guttman, L. (1977). What is not what in statistics. *The Statistician*, *26*, 81–107.

Guttman, L. (1981). Efficacy coefficients for differences among averages. In I. Borg (Ed.), *Multidimensional data representations: when and why* (pp. 1–10). Ann Arbor, MI: Mathesis.

Guttman, L. (1982). Facet theory, smallest space analysis, and factor analysis. *Perceptual and Motor Skills*, *54*, 491–493.

Guttman, L. (1985). Coefficients of polytonicity and monotonicity. In S. Kotz & N. L. Johnson (Eds.), *Encyclopedia of statistical sciences* (Vol. 7, pp. 80–87). New York: Wiley.

Guttman, L. (1991). *Louis Guttman: In Memoriam - Chapters from an unfinished textbook on facet theory.* Jerusalem: The Israel Academy of Sciences and Humanities.

Guttman, L., & Levy, S. (1991). Two structural laws for intelligence tests. *Intelligence*, *15*, 79–103.

Guttman, N., & Kalish, H. I. (1956). Discriminability and stimulus generalization. *Journal of Experimental Psychology*, *51*, 79–88.

Harshman, R. A. (1972). Determination and proof of minimum uniqueness conditions for PARAFAC-I. *UCLA Working Papers in Phonetics*, *22*.

Harshman, R. A., & Lundy, M. E. (1984). The Parafac model for three-way factor analysis and multidimensional scaling. In H. G. Law, C. W. Snyder, J. A. Hattie, & R. P. McDonald (Eds.), *Research methods for multimode data analysis* (pp. 122–215). New York: Praeger.

Hefner, R. A. (1958). *Extension of the law of comparative judgment to discriminable and multidimensional stimuli.* Unpublished doctoral dissertation, University of Michigan.

Heiser, W. J. (1981). *Unfolding analysis of proximity data.* Unpublished doctoral dissertation, Leiden University.

Heiser, W. J. (1985). *Multidimensional scaling by optimizing goodness-of-fit to a smooth hypothesis* (Tech. Rep. No. RR-85-07). Leiden, The Netherlands: Department of Data Theory, Leiden University.

Heiser, W. J. (1988a). Multidimensional scaling with least absolute residuals. In H. H. Bock (Ed.), *Classification and related methods* (pp. 455–462). Amsterdam: North-Holland.

Heiser, W. J. (1988b). PROXSCAL, multidimensional scaling of proximities. In A. Di Ciaccio & G. Bove (Eds.), *International meeting on the analysis of multiway data matrices, software guide* (pp. 77–81). Rome: C.N.R.

Heiser, W. J. (1989a). Order invariant unfolding analysis under smoothness restrictions. In G. De Soete, H. Feger, & K. C. Klauer (Eds.), *New developments in psychological choice modeling* (pp. 3–31). Amsterdam: Elsevier Science.

Heiser, W. J. (1989b). The city-block model for three-way multidimensional scaling. In R. Coppi & S. Bolasco (Eds.), *Multiway data analysis* (pp. 395–404). Amsterdam: Elsevier Science.

Heiser, W. J. (1990). A generalized majorization method for least squares multidimensional scaling of pseudodistances that may be negative. *Psychometrika*, *56*, 7–27.

Heiser, W. J. (1993). Clustering in low-dimensional space. In O. Opitz, B. Lausen, & R. Klar (Eds.), *Information and classification: Concepts, methods, and applications* (pp. 162–173). Berlin: Springer.

Heiser, W. J. (1995). Convergent computation by iterative majorization: Theory and applications in multidimensional data analysis. In W. J. Krzanowski (Ed.), *Recent advances in descriptive multivariate analysis* (pp. 157–189). Oxford: Oxford University Press.

Heiser, W. J., & De Leeuw, J. (1979). *How to use SMACOF-III: A program for metric multidimensional unfolding* (Tech. Rep.). Leiden University, Department of Data Theory.

Heiser, W. J., & Groenen, P. J. F. (1997). Cluster differences scaling with a within-cluster loss component and a fuzzy successive approximation strategy to avoid local minima. *Psychometrika*, *62*, 529–550.

Heiser, W. J., & Meulman, J. J. (1983a). Analyzing rectangular tables by joint and constrained MDS. *Journal of Econometrics*, *22*, 193–167.

Heiser, W. J., & Meulman, J. J. (1983b). Constrained multidimensional scaling including confirmation. *Applied Psychological Measurement*, *7*, 381–404.

Helm, C. E. (1959). *A multidimensional ratio scaling analysis of color relations* (Tech. Rep.). Princeton, NJ: Princeton University and Educational Testing Service.

Helm, C. E. (1964). Multidimensional ratio scaling analysis of perceived color relations. *Journal of the Optical Society of America*, *54*, 256–262.

Hirschfeld, H. O. (1935). A connection between correlation and contingency. *Proceedings Cambridge Philosophical Society*, *31*, 520–524.

Horan, C. B. (1969). Multidimensional scaling: Combining observations when individuals have different perceptual structures. *Psychometrika*, *34*, 139–165.

Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, *24*, 417–441, 498–520.

Huber, P. J. (1964). Robust estimation of a location parameter. *Annals of Mathematical Statistics*, *35*, 73–101.

Hubert, L., Arabie, P., & Hesson-McInnis, M. (1992). Multidimensional scaling in the city-block metric: A combinatorial approach. *Journal of Classification*, *9*, 211–236.

Hubert, L. J., & Arabie, P. (1986). Unidimensional scaling and combinatorial optimization. In J. De Leeuw, W. J. Heiser, J. J. Meulman, & F. Critchley (Eds.), *Multidimensional data analysis* (pp. 181–196). Leiden, The Netherlands: DSWO.

Hubert, L. J., Arabie, P., & Meulman, J. J. (1997). Linear and circular unidimensional scaling for symmetric proximity matrices. *British Journal of Mathematical and Statistical Psychology*, *50*, 253–284.

Hubert, L. J., & Golledge, R. G. (1981). Matrix reorganisation and dynamic programming: Applications to paired comparison and unidimensional seriation. *Psychometrika*, *46*, 429–441.

Hunter, D. R., & Lange, K. (2004). A autorial on MM algorithms. *The American Statistician*, *39*, 30–37.

Hurley, J. R., & Cattell, R. B. (1962). The Procrustes program: Producing direct rotation to test a hypothesized factor structure. *Behavioral Science*, *7*, 258–262.

Indow, T. (1974). Applications of multidimensional scaling in perception. In E. C. Carterette & M. P. Friedman (Eds.), *Handbook of perception* (Vol. II, pp. 493–531). New York: Academic.

Isaac, P. D., & Poor, D. (1974). On the determination of appropriate dimensionality in data with error. *Psychometrika*, *39*, 91–109.

Kaiser, H. F. (1958). The varimax criterion for analytic rotation in factor analysis. *Psychometrika*, *23*, 187–200.

Kearsley, A. J., Tapia, R. A., & Trosset, M. W. (1998). The solution of the metric Stress and S-Stress problems in multidimensional scaling using Newton's method. *Computational Statistics*, *13*, 369–396.

Kendall, D. G. (1971). Seriation from abundance matrices. In F. R. Hodson, D. G. Kendall, & P. Tautu (Eds.), *Mathematics in the archaelogical and historical sciences.* Edinburgh: Edinburgh University Press.

Kiers, H. A. L. (1990). Majorization as a tool for optimizing a class of matrix functions. *Psychometrika*, *55*, 417–428.

Kiers, H. A. L. (2002). Setting up alternating least squares and iterative majorization algorithms for solving various matrix optimization problems. *Computational Statistics and Data Analysis*, *41*, 157–170.

Kiers, H. A. L., & Groenen, P. J. F. (1996). A monotonicaly convergent algorithm for orthogonal congruence rotation. *Psychometrika*, *61*, 375–389.

Kim, C., Rangaswamy, A., & DeSarbo, W. S. (1999). A quasi-metric approach to multidimensional unfolding for reducing the occurrence of degenerate solutions. *Multivariate Behavioral Research*, *34*, 143–180.

Klahr, D. (1969). A Monte-Carlo investigation of the statistical significance of Kruskal's nonmetric scaling procedure. *Psychometrika*, *34*, 319–330.

Kloek, T., & Theil, H. (1965). International comparisons of prices and quantities consumed. *Econometrica*, *33*, 535–556.

Krantz, D. H. (1967). Rational distance functions for multidimensional scaling. *Journal of Mathematical Psychology*, *4*, 226–245.

Krantz, D. H. (1972). Integration of just noticeable differences. *Journal of Mathematical Psychology*, *8*, 591–599.

Krantz, D. H., Luce, R. D., Suppes, P., & Tversky, A. (1971). *Foundations of measurement* (Vol. 1). New York: Academic.

Krantz, D. H., & Tversky, A. (1975). Similarity of rectangles: An analysis of subjective dimensions. *Journal of Mathematical Psychology*, *12*, 4–34.

Kristof, W. (1970). A theorem on the trace of certain matrix products and some applications. *Journal of Mathematical Psychology*, *7*, 515–530.

Kristof, W., & Wingersky, B. (1971). Generalization of the orthogonal Procrustes rotation procedure for more than two matrices. In *Proceedings of the 79th annual convention of the american psychological association* (pp. 89–90).

Krumhansl, C. L. (1978). Concerning the applicability of geometric models to similarity data: The interrelationship between similarity and spatial density. *Psychological Review*, *85*, 445–463.

Kruskal, J. B. (1964a). Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, *29*, 1–27.

Kruskal, J. B. (1964b). Nonmetric multidimensional scaling: A numerical method. *Psychometrika*, *29*, 115–129.

Kruskal, J. B. (1968). *How to use M-D-SCAL, a program to do multidimensional scaling and multidimensional unfolding (Version 4 and 4M Fortran IV)* (Tech. Rep.). Murray Hill, NJ: Bell Telephone Laboratories.

Kruskal, J. B. (1977). Multidimensional scaling and other methods for discovering structure. In K. Enslein, A. Ralston, & H. S. Wilf (Eds.), *Statistical methods for digitial computers* (Vol. 2, pp. 296–339). New York: Wiley.

Kruskal, J. B., & Carmone, F. J. (1969). *How to use M-D-SCAL Version 5M and other useful information* (Tech. Rep.). Murray Hill, NJ: Bell Labs.

Kruskal, J. B., & Carroll, J. D. (1969). Geometrical models and badness-of-fit functions. In P. R. Krishnaiah (Ed.), *Multivariate analysis* (Vol. 2, pp. 639–671). New York: Academic.

Kruskal, J. B., & Wish, M. (1978). *Multidimensional scaling.* Beverly Hills, CA: Sage.

Kruskal, J. B., Young, F. W., & Seery, J. B. (1978). *How to use KYST-2, a very flexible program to do multidimensional scaling and unfolding* (Tech. Rep.). Murray Hill, NJ: Bell Labs.

Lange, K., Hunter, D. R., & Yang, I. (2000). Optimization transfer using surrogate objective functions. *Journal of Computational and Graphical Statistics*, *9*, 1–20.

Langeheine, R. (1980a). *Approximate norms and significance tests for the Lingoes-Borg Procrustean Individual Differences Scaling PINDIS* (Tech. Rep. No. 39). Kiel: Institut für die Pädagogik der Naturwissenschaften, Universität Kiel.

Langeheine, R. (1980b). Erwartete Fitwerte für Zufallskonfigurationen in PINDIS. *Zeitschrift für Sozialpsychologie*, *11*, 38–49.

Langeheine, R. (1982). Statistical evaluation of measures of fit in the Lingoes-Borg Procrustean individual differences scaling. *Psychometrika*, *47*, 427–442.

Lawler, E. (1967). The multitrait-multimethod approach to measuring managerial job performance. *Journal of Applied Psychology*, *51*, 369–381.

Lawson, C. L., & Hanson, R. J. (1974). *Solving least squares problems.* Englewood Cliffs, NJ: Prentice-Hall.

Lazarte, A., & Schönemann, P. H. (1991). Saliency metric for subadditive dissimilarity judgments of rectangles. *Perception and Psychophysics*, *49*, 142–158.

Lebart, L., Morineau, A., & Warwick, K. M. (1984). *Multivariate descriptive statistical analysis.* New York: Wiley.

Leutner, D., & Borg, I. (1983). Zufallskritische Beurteilung der Übereinstimmung von Faktor- und MDS-Konfigurationen. *Diagnostica*, *24*, 320–335.

Levelt, W. J. M., Geer, J. P. Van de, & Plomp, R. (1966). Triadic comparisons of musical intervals. *British Journal of Mathematical and Statistical Psychology*, *19*, 163–179.

Levy, S. (1976). Use of the mapping sentence for coordinating theory and research: A cross-cultural example. *Quality and Quantity*, *10*, 117–125.

Levy, S. (1983). A cross-cultural analysis of the structure and levels of atttitudes towards acts of political protest. *Social Indicators Research*, *12*, 281–309.

Levy, S., & Guttman, L. (1975). On the multivariate structure of wellbeing. *Social Indicators Research*, *2*, 361–388.

Lindman, H., & Caelli, T. (1978). Constant curvature Riemannean scaling. *Journal of Mathematical Psychology*, *17*, 89–109.

Lingoes, J. C. (1965). An IBM-7090 program for Guttman-Lingoes smallest space analysis-I. *Behavioral Science*, *10*, 183–184.

Lingoes, J. C. (1971). Some boundary conditions for a monotone analysis of symmetric matrices. *Psychometrika*, *36*, 195–203.

Lingoes, J. C. (1973). *The Guttman-Lingoes nonmetric program series.* Ann Arbor, MI: Mathesis.

Lingoes, J. C. (1981). Testing regional hypotheses in multidimensional scaling. In I. Borg (Ed.), *Multidimensional data representations: when and why* (pp. 280–310). Ann Arbor, MI: Mathesis.

Lingoes, J. C. (1989). *Guttman-Lingoes nonmetric PC series manual.* Ann Arbor, MI: Mathesis.

Lingoes, J. C., & Borg, I. (1977). Identifying spatial manifolds for interpretation. In E. E. Roskam, J. C. Lingoes, & I. Borg (Eds.), *Geometric representations of relational data.* Ann Arbor, MI: Mathesis.

Lingoes, J. C., & Borg, I. (1978). A direct approach to individual differences scaling using increasingly complex transformations. *Psychometrika*, *43*, 491–519.

Lingoes, J. C., & Borg, I. (1983). A quasi-statistical model for choosing between alternative configurations derived from ordinally constrained data. *British Journal of Mathematical and Statistical Psychology*, *36*, 36–53.

Lingoes, J. C., & Guttman, L. (1967). Nonmetric factor analysis: A rank reducing alternative to linear factor analysis. *Multivariate Behavioral Research*, *2*, 485–505.

Lingoes, J. C., & Roskam, E. E. (1973). A mathematical and empirical analysis of two multidimensional scaling algorithms. *Psychometrika*, *38*.

Luce, R. D. (1959). *Individual choice behavior.* New York: Wiley.

Luce, R. D. (1961). A choice theory analysis of similarity judgments. *Psychometrika*, *26*, 151–163.

Luce, R. D., & Suppes, P. (1963). Preference, utility, and subjective probability. In R. D. Luce, R. R. Bush, & E. Galanter (Eds.), *Handbook of mathematical psychology* (Vol. 3, pp. 249–410). New York: Wiley.

Luengo, F., Raydan, M., Glunt, W., & Hayden, T. L. (2002). Preconditioned spectral gradient method. *Numerical Algorithms*, *30*, 241-258.

Lüer, G., & Fillbrandt, H. (1970). Interindividuelle Unterschiede bei der Beurteilung von Reizähnlichkeiten. *Zeitschrift für experimentelle und angewandte Psychologie*, *18*, 123–138.

Lüer, G., Osterloh, W., & Ruge, A. (1970). Versuche zur multidimensionalen Skalierung von subjektiven Reizähnlichkeiten auf Grund von Schätzurteilen und Verwechselungshäufigkeiten beim Wiedererkennen. *Psychologische Forschung*, *33*, 223–241.

MacCallum, R. C. (1976). Effects on INDSCAL of non-orthogonal perceptions of object space dimensions. *Psychometrika*, *41*, 177–188.

MacCallum, R. C. (1977). Effects of conditionality on INDSCAL and ALSCAL weights. *Psychometrika*, *42*, 297–305.

Mathar, R. (1989). Algorithms in multidimensional sclaing. In O. Opitz (Ed.), *Conceptual and numerical analysis of data* (pp. 159–177). Berlin: Springer.

Mathar, R. (1994). Multidimensional scaling with l_p-distances: A unifying approach. In H. H. Bock, W. Lenski, & M. M. Richter (Eds.), *Information systems and data analysis* (pp. 325–331). Heidelberg, Germany: Springer.

Mathar, R. (1997). *Multidimensionale Skalierung.* Stuttgart: B. G. Teubner.

Mathar, R., & Groenen, P. J. F. (1991). Algorithms in convex analysis applied to multidimensional scaling. In E. Diday & Y. Lechevallier (Eds.), *Symbolic-numeric data analysis and learning* (pp. 45–56). Commack, NY: Nova Science.

Mathar, R., & Meyer, R. (1993). Preorderings, monotone functions, and best rank r approximations with applications to classical MDS. *Journal of Statistical Planning and Inference*, *37*, 291–305.

Mathar, R., & Žilinskas, A. (1993). On global optimization in two dimensional scaling. *Acta Aplicandae Mathematica*, *33*, 109–118.

McGee, V. E. (1966). The multidimensional scaling of "elastic" distances. *British Journal of Mathematical and Statistical Psychology*, *19*, 181–196.

Merkle, E. (1981). *Die Erfassung und Nutzung von Informationen über den Sortimentsverbund in Handelsbetrieben.* Berlin: Duncker & Humbolt.

Messick, S. J., & Abelson, R. P. (1956). The additive constant problem in multidimensional scaling. *Psychometrika*, *21*, 1–15.

Meulman, J. J. (1986). *A distance approach to nonlinear multivariate analysis.* Leiden, The Netherlands: DSWO.

Meulman, J. J. (1992). The integration of multidimensional scaling and multivariate analysis with optimal transformations. *Psychometrika*, *57*, 539–565.

Meulman, J. J., Heiser, W. J., & De Leeuw, J. (1983). *Progress notes on SMACOF-II* (Tech. Rep. No. UG-83). Leiden, The Netherlands: Department of Data Theory, Leiden University.

Meulman, J. J., Heiser, W. J., & SPSS. (1999). *Spss categories 10.0.* Chicago: SPSS.

Meyer, R. (1993). *Matrix-Approximation in der Multivariaten Statistik.* Aachen, Germany: Verlag der Augustinus Buchhandlung.

Mezzich, J. E. (1978). Evaluating clustering methods for psychiatric diagnosis. *Biological Psychiatry*, *13*, 265–281.

Micko, H. C., & Fischer, W. (1970). The metric of multidimensional psychological spaces as a function of the differential attention to subjective attributes. *Journal of Mathematical Psychology*, *7*, 118–143.

Miller, G. A. (1956). The magical number seven, plus minus one: Some limits on our capacity for processing information. *Psychological Review*, *63*, 81–97.

Möbus, C. (1975). Bemerkungen zur Skalierung interindividueller Urteilsdifferenzen: Simulation des INDSCAL-Modells von Carroll & Chang mit der 'Points-of-View' Analyse von Tucker & Messick. *Archiv für Psychologie*, *127*, 189–209.

Möbus, C. (1979). Zur Analyse nichtsymmetrischer Ähnlichkeitsurteile: Ein dimensionales Driftmodell, eine Vergleichshypothese, Tversky's Kontrastmodell und seine Fokushypothese. *Archiv für Psychologie*, *131*, 105–136.

Mooijaart, A., & Commandeur, J. J. F. (1990). A general solution of the weighted orthonormal Procrustes problem. *Psychometrika*, *53*, 657–663.

Moorhead, G., & Griffin, R. W. (1989). *Organizational behavior.* Boston: Houghton Mifflin.

Mulaik, S. A. (1972). *The foundations of factor analysis.* New York: McGraw-Hill.

Müller, C. (1984). *Mehrdimensionale nicht-numerische Skalierung in der Psychologie.* Göttingen, Germany: Hogrefe.

Nishisato, S. (1980). *Dual scaling and its applications* (Vol. 24). Toronto: University of Toronto Press.

Nishisato, S. (1994). *Elements of dual scaling: An introduction to practical data analysis.* Hillsdale, NJ: Erlbaum.

Noma, E., & Johnson, J. (1977). *Constraining nonmetric multidimensional scaling configurations* (Tech. Rep. No. 60). Human Performance Center, University of Michigan.

Norpoth, H. (1979a). Dimensionen des Parteienkonflikts und Präferenzordnungen der deutschen Wählerschaft: Eine Unfoldinganalyse. *Zeitschrift für Sozialpsychologie*, *10*, 350–362.

Norpoth, H. (1979b). The parties come to order! Dimensions of preferential choice in the West German electorate, 1961-1976. *American Political Science Review*, *73*, 724–736.

Okada, A. (1990). A generalization of asymmetric multidimensional scaling. In M. Schader & W. Gaul (Eds.), *Knowledge, data and computer-assited decisions* (pp. 127–138). Berlin: Springer.

Okada, A., & Imaizumi, T. (1987). Geometric models for asymmetric similarity data. *Behaviormetrika*, *21*, 81–96.

Okada, A., & Imaizumi, T. (1997). Asymmetric multidimensional scaling of two-mode three-way proximities. *Journal of Classification*, *14*, 195–224.

Ortega, J. M., & Rheinboldt, W. C. (1970). *Iterative solutions of nonlinear equations in several variables.* New York: Academic.

Pearson, K. (1896). Regression, heridity, and panmixia. *Philosophical Transactions of the Royal Society of London, Ser. A*, *187*, 269–318.

Pearson, K. (1901). On lines and planes of closest fit to a system of points in space. *Philosophical Magazine*, *2*, 557–572.

Pease, I., M. C. (1965). *Methods of matrix algebra.* New York: Academic.

Peay, E. R. (1988). Multidimensional rotation and scaling of configurations to optimal agreement. *Psychometrika*, *53*, 199–208.

Pliner, V. (1996). Metric, unidimensional scaling and global optimization. *Journal of Classification*, *13*, 3–18.

Poole, K. T. (1984). Least squares metric, unidimensional unfolding. *Psychometrika*, *49*, 311–323.

Poole, K. T. (1990). Least squares metric, unidimensional scaling of multivariate linear models. *Psychometrika*, *55*, 123–149.

Poor, D., & Wherry, R. J. (1976). Invariance of multidimensional configurations. *British Journal of Mathematical and Statistical Psychology*, *26*, 114–125.

Porrat, R. (1974). *A laboratory manual for the Guttman-Lingoes nonmetric computer programs* (Tech. Rep. No. RP 456 E). The Israel Institute of Applied Social Research.

Rabinowitz, G. (1976). A procedure for ordering object pairs consistent with the multidimensional unfolding model. *Psychometrika*, *41*, 349–373.

Rabinowitz, G. B. (1975). An introduction to nonmetric multidimensional scaling. *American Journal of Political Science*, *19*, 343–390.

Ramsay, J. O. (1969). Some statistical considerations in multidimensional scaling. *Psychometrika*, *34*, 167–182.

Ramsay, J. O. (1977). Maximum likelihood estimation in multidimensional scaling. *Psychometrika*, *42*, 241–266.

Ramsay, J. O. (1988). Monotone regression splines in action. *Statistical Science*, *3*(4), 425–461.

Restle, F. (1959). A metric and an ordering on sets. *Psychometrika*, *24*, 207–220.

Richardson, M., & Kuder, G. F. (1933). Making a rating scale that measures. *Personnel Journal*, *12*, 36–40.

Richardson, M. W. (1938). Multidimensional psychophysics. *Psychological Bulletin*, *35*, 659–660.

Rosenberg, S., & Kim, M. P. (1975). The method of sorting as a data gathering procedure in multivariate research. *Multivariate Behavioral Research*, *10*, 489–502.

Roskam, E. E. (1972). *An algorithm for multidimensional scaling by metric transformations of data* (Program Bulletin No. 23 106). Nijmegen, The Netherlands: Universiteit Nijmegen.

Roskam, E. E. (1979a). A general system for nonmetric data analysis. In J. C. Lingoes, E. E. Roskam, & I. Borg (Eds.), *Geometric representations of relational data* (pp. 313–347). Ann Arbor, MI: Mathesis.

Roskam, E. E. (1979b). The nature of data: Interpretation and representation. In J. C. Lingoes, E. E. Roskam, & I. Borg (Eds.), *Geometric representations of relational data* (pp. 149–235). Ann Arbor, MI: Mathesis.

Roskam, E. E., & Lingoes, J. C. (1981). Minissa. In S. S. Schiffman, M. L. Reynolds, & F. W.Young (Eds.), *Introduction to multidimensional scaling* (pp. 362–371). New York: Academic.

Ross, R. T. (1934). Optimum orders for presentations of pairs in paired comparisons. *Journal of Educational Psychology*, *25*, 375–382.

Rothkopf, E. Z. (1957). A measure of stimulus similarity and errors in some paired-associate learning. *Journal of Experimental Psychology*, *53*, 94–101.

Sammon, J. W. (1969). A non-linear mapping for data structure analysis. *IEEE Transactions on Computers*, *18*, 401–409.

SAS. (1999). Electronic helpfiles, retrieved October 22, 2004, from `http://rocs.acomp.usf.edu/sas/sashtml/stat/chap53/sect25.htm`

Schönemann, P. H. (1966). A generalized solution of the orthogonal Procrustes problem. *Psychometrika*, *31*, 1–10.

Schönemann, P. H. (1970). On metric multidimensional unfolding. *Psychometrika*, *35*, 349–366.

Schönemann, P. H. (1972). An algebraic solution for a class of subjective metrics models. *Psychometrika*, *37*, 441–451.

Schönemann, P. H. (1982). A metric for bounded response scales. *Bulletin of the Psychonomic Society*, *19*, 317–319.

Schönemann, P. H. (1983). Some theory and results for metrics for bounded response scales. *Journal of Mathematical Psychology*, *27*, 311–324.

Schönemann, P. H. (1985). On the formal differentiation of traces and determinants. *Journal of Multivariate Behavioral Research*, *20*, 113–139.

Schönemann, P. H. (1990). Psychophysical maps for rectangles. In H. G. Geissler (Ed.), *Psychophysical explorations of mental structures* (pp. 149–164). Toronto: Hogrefe & Huber.

Schönemann, P. H. (1994). Measurement: The reasonable ineffectiveness of mathematics in the social sciences. In I. Borg & P. P. Mohler (Eds.), *Trends and perspectives in empirical social research* (pp. 149–160). New York: De Gruyter.

Schönemann, P. H., & Borg, I. (1981a). Measurement, scaling, and factor analysis. In I. Borg (Ed.), *Multidimensional data representations: When and why* (pp. 380–419). Ann Arbor, MI: Mathesis.

Schönemann, P. H., & Borg, I. (1981b). On the interaction between area and shape. In I. Borg (Ed.), *Multidimensional data representations: When and why* (pp. 432–440). Ann Arbor, MI: Mathesis.

Schönemann, P. H., & Borg, I. (1983). Grundlagen der metrischen mehrdimensionalen Skaliermethoden. In H. Feger & J. Bredenkamp (Eds.), *Enzyklopädie der Psychologie: Messen und Testen* (pp. 257–345). Göttingen: Hofgrefe.

Schönemann, P. H., & Carroll, R. M. (1970). Fitting one matrix to another under choice of a central dilation and a rigid motion. *Psychometrika*, *35*, 245–256.

Schönemann, P. H., Dorcey, T., & Kienapple, K. (1985). Subadditive concatenation in dissimilarity judgments. *Perception and Psychophysics*, *38*, 1–17.

Schönemann, P. H., James, W. L., & Carter, F. S. (1978). COSPA, Common Space Analysis - a program for fitting and testing Horan's subjective metrics model. *Journal of Marketing Research*, *15*, 268–270.

Schönemann, P. H., James, W. L., & Carter, F. S. (1979). Statistical inference in multidimensional scaling: A method for fitting and testing Horan's Model. In J. C. Lingoes, E. E. Roskam, & I. Borg (Eds.), *Geometric representations of relational data* (pp. 791–826). Ann Arbor, MI: Mathesis.

Schönemann, P. H., & Lazarte, A. (1987). Psychophysical maps for subadditive dissimilarity ratings. *Perception and Psychophysics*, *42*, 342–354.

Schönemann, P. H., & Wang, M. M. (1972). An individual differences model for multidimensional scaling of preference data. *Psychometrika*, *37*, 275–311.

Schulz, U. (1972). Ein Euklidisches Modell der multidimensionalen Skalierung unter Berücksichtigung individueller Differenzen. In L. Eckensberger (Ed.), *Bericht über den 28. Kongress der Deutschen Gesellschaft für Psychologie* (pp. 75–89). Saarbrücken, Germany.

Schulz, U. (1975). Zu einem Dekompositionsmodell der multidimensionalen Skalierung mit individueller Gewichtung der Dimensionen. *Psychologische Beiträge*, *17*, 167–187.

Schulz, U. (1980). An alternative procedure for the analysis of similarity data and its comparison to the Idioscal- and Indscal- procedure. In H. Feger & E. D. Lantermann (Eds.), *Similarity and choice* (pp. 140–149). Bern, Switzerland: Huber.

Schulz, U., & Pittner, P. M. (1978). Zur multidimensionalen Skalierung individueller Differenzen. *Psychologische Beiträge*, *20*, 294–315.

Schumaker, L. (1981). *Spline functions: Basic theory.* New York: Wiley.

Schwarz, S., & Bilsky, W. (1987). Toward a universal psychological structure of human values. *Journal of Personality and Social Psychology*, *53*, 550–562.

Searle, S. R. (1982). *Matrix algebra useful for statistics.* New York: Wiley.

Shepard, R. N. (1957). Stimulus and response generalization: A stochastic model for relating generalization to distance in psychological space. *Psychometrika*, *22*, 325–345.

Shepard, R. N. (1958a). Stimulus and response generalization: Deduction of the generalization gradient from a trace model. *Psychological Review*, *65*, 242–256.

Shepard, R. N. (1958b). Stimulus and response generalization: Tests of a model relating generalization to distance in psychological space. *Journal of Experimental Psychology*, *55*, 509–523.

Shepard, R. N. (1963). Analysis of proximities as a technique for the study of information processing in man. *Human Factors*, *5*, 33–48.

Shepard, R. N. (1964). Attention and the metric structure of the stimulus space. *Journal of Mathematical Psychology*, *1*, 54–87.

Shepard, R. N. (1965). Approximation to uniform gradients of generalization by monotone transformations of scale. In D. I. Mostofsky (Ed.), *Stimulus generalization* (pp. 94–110). Stanford, CA: Stanford University Press.

Shepard, R. N. (1966). Metric structure in ordinal data. *Journal of Mathematical Psychology*, *3*, 287–315.

Shepard, R. N. (1969, September). *Computer explorations of psychological space.* Paper presented at the Annual Meeting of the American Psychological Association.

Shepard, R. N. (1974). Representation of structure in similarity data: Problems and prospects. *Psychometrika*, *39*, 373–421.

Shepard, R. N., & Carroll, J. D. (1966). Parametric representation of nonlinear data structures. In P. R. Krishnaiah (Ed.), *Multivariate analysis*. New York: Academic.

Sherman, C. R. (1972). Nonmetric multidimensional scaling: A Monte-Carlo study of the basic parameters. *Psychometrika*, *37*, 323–355.

Shinew, K. J., Floyd, M., McGuire, F., & Noe, F. (1995). Gender, race, and subjective social class and their association with leisure preferences. *Leisure Studies*, *17*, 75–89.

Shye, S. (1981). Comparing structures of multidimensional scaling (SSA): A response to Kashti's study and to Langeheine's critique. *Studies in Educational Evaluation*, *7*, 105–109.

Shye, S. (1985). Nonmetric multivariate models for behavioral action systems. In D. Canter (Ed.), *Facet theory: Approaches to social research* (pp. 97–148). New York: Springer.

Shye, S. (1991). *Faceted SSA: A computer program for the PC* (Tech. Rep.). Jerusalem: The Louis Guttman Israel Institute of Applied Social Research.

Shye, S., Elizur, D., & Hoffman, M. (1994). *Facet theory.* Newbury Park, CA: Sage.

Sibson, R. (1979). Studies in the robustness of multidimensional scaling: Perturbation analysis of classical scaling. *Journal of the Royal Statistical Society*, *41*, 217–229.

Sixtl, F. (1967). *Messmethoden der Psychologie.* Weinheim, Germany: Beltz.

Sjöberg, L. (1975). Models of similarity and intensity. *Psychological Bulletin*, *82*, 191–206.

Snow, R. E., Kyllonen, P. C., & Marshalek, B. (1984). The topography of ability and learning correlations. In R. J. Sternberg (Ed.), *Advances in the psychology of human intelligence* (Vol. 3, pp. 47–103). Hillsdale, NJ: Erlbaum.

Spence, I. (1972). *An aid to the estimation of dimensionality in nonmetric multidimensional scaling* (Tech. Rep. No. 229). University of Western Ontario Research Bulletin.

Spence, I. (1978). Multidimensional scaling. In P. W. Colgan (Ed.), *Quantitative ethology* (pp. 175–217). New York: Wiley.

Spence, I. (1979). A simple approximation for random rankings stress values. *Multivariate Behavioral Research*, *14*, 355–365.

Spence, I. (1983). Monte Carlo simulation studies. *Applied Psychological Measurement*, *7*(4), 405–425.

Spence, I., & Domoney, D. W. (1974). Single subject incomplete designs for nonmetric multidimensional scaling. *Psychometrika*, *39*, 469–490.

Spence, I., & Graef, J. (1974). The determination of the underlying dimensionality of an empirically obtained matrix of proximities. *Multivariate Behavioral Research*, *9*, 331–341.

Spence, I., & Ogilvie, J. C. (1973). A table of expected stress values for random rankings in nonmetric multidimensional scaling. *Multivariate Behavioral Research*, *8*, 511–517.

SPSS. (1990). *Categories.* Chicago: SPSS Inc.

Srinivasan, V., & Shocker, A. D. (1973). Linear programming techniques for multidimensional analysis of preferences. *Psychometrika*, *38*, 337–369.

Staufenbiel, T. (1987). Critical values and properties of μ_2. *Methodika*, *1*, 60–67.

Staufenbiel, T., & Borg, I. (1987). Dimensionen der Änlichkeit einfacher Reize: Breite-Höhe oder Grösse-Form? *Psychologische Beiträge*, *29*, 403–422.

Stenson, H. H., & Knoll, R. L. (1969). Goodness of fit for random rankings in Kruskal's nonmetric scaling procedure. *Psychological Bulletin*, *71*, 122–126.

Stephenson, W. (1953). *The study of behavior.* Chicago: University of Chicago Press.

Steverink, M., Van der Kloot, W., & Heiser, W. J. (2002). *Avoiding degenerate solutions in multidimensional unfolding by using additional distance information* (Tech. Rep. No. PRM 2002-01). Leiden University, the Netherlands: Department of Psychometrics.

Steyvers, M., & Busey, T. (2000). Predicting similarity ratings to faces using physical descriptions. In M. Wenger & J. Townsend (Eds.), *Computational, geometric, and process perspectives on facial cognition: Contexts and challenges* (pp. 115–146). New Jersey: Lawrence Erlbaum Associates.

Strang, G. (1976). *Linear algebra and its applications.* New York: Academic.

Suppes, P., & Zinnes, J. L. (1963). Basic measurement theory. In R. D. Luce, R. R. Bush, & E. Galanter (Eds.), *Handbook of mathematical psychology* (Vol. I, pp. 2–76). New York: Wiley.

Takane, Y., & Carroll, J. D. (1981). Nonmetric metric maximum likelihood multidimensional scaling from directional rankings of similarities. *Psychometrika*, *46*, 389–405.

Takane, Y., Young, F. W., & De Leeuw, J. (1977). Nonmetric individual differences multidimensional scaling: An alternating least-squares method with optimal scaling features. *Psychometrika*, *42*, 7–67.

Ten Berge, J. M. (1977). Orthogonal procrustes rotation for two or more matrices. *Psychometrika*, *42*, 267–276.

Ten Berge, J. M. F. (1991). A general solution for a class of weakly constrained linear regression problems. *Psychometrika*, *56*, 601–609.

Ten Berge, J. M. F. (1993). *Least squares optimization in multivariate analysis.* Leiden, The Netherlands: DSWO.

Ten Berge, J. M. F., & Kiers, H. A. L. (1991). Some clarifications of the CANDECOMP algorithm applied to INDSCAL. *Psychometrika*, *56*, 317–326.

Ten Berge, J. M. F., Kiers, H. A. L., & Commandeur, J. J. F. (1993). Orthogonal Procrustes rotation for matrices with missing values. *British Journal of Mathematical and Statistical Psychology*, *46*, 119–134.

Ten Berge, J. M. F., Kiers, H. A. L., & Krijnen, W. P. (1993). Computational solutions for the problem of negative saliences and nonsymmetry in INDSCAL. *Journal of Classification*, *10*, 115–124.

Tenenbaum, J. B. (1998). Mapping a manifold of perceptual observations. In M. Jordan, M. Kearns, & S. Solla (Eds.), *Advances in neural information processing 10* (pp. 682–687). Cambridge, MA.: MIT Press.

Tenenbaum, J. B., De Silva, V., & Langford, J. C. (2000). A global geometric framework for nonlinear dimension reduction. *Science*, *290*, 2319–2323.

Ter Braak, C. J. F. (1992). Multidimensional scaling and regression. *Statistica Applicata*, *4*, 577–586.

Thurstone, L. L. (1927). A law of comparative judgment. *Psychological Review*, *34*, 273–286.

Thurstone, L. L. (1935). *The vectors of mind.* Chicago: University of Chicago Press.

Thurstone, L. L. (1947). *Multiple factor analysis.* Chicago: University of Chicago Press.

Tobler, W., & Wineburg, S. (1971). A cappadocian speculation. *Nature*, *231*, 39–41.

Torgerson, W. S. (1952). Multidimensional scaling: I. Theory and method. *Psychometrika*, *17*, 401–419.

Torgerson, W. S. (1958). *Theory and methods of scaling.* New York: Wiley.

Torgerson, W. S. (1965). Multidimensional scaling of similarity. *Psychometrika*, *30*, 333–367.

Trosset, M. W. (1993). *A new formulation of the nonmetric Strain problem in multidimensional scaling* (Tech. Rep. No. TR93-30). Houston, TX: Department of Computational and Applied Mathematics, Rice University.

Tucker, L. R. (1951). *A method for the synthesis of factor analysis studies* (Tech. Rep. No. 984). Washington, DC: Department of the Army.

Tucker, L. R. (1960). Intra-individual and inter-individual multidimensionality. In H. Gulliksen & S. Messick (Eds.), *Psychological scaling: Theory and applications* (pp. 155–167). New York: Wiley.

Tucker, L. R. (1972). Relations between multidimensional scaling and three-mode factor analysis. *Psychometrika*, *37*, 3–27.

Tversky, A. (1977). Features of similarity. *Psychological Review*, *84*, 327–352.

Tversky, A., & Gati, I. (1982). Similarity, separability, and the triangle inequality. *Psychological Review*, *89*, 123–154.

Tversky, A., & Krantz, D. (1970). The dimensional representation and the metric structure of similarity data. *Journal of Mathematical Psychology*, *7*, 572–597.

Van der Heijden, P. G. M., De Falguerolles, A., & De Leeuw, J. (1989). A combined approach of contingency table analysis using correspondence analysis and loglinear analysis (with discussion). *Applied Statistics*, *38*, 249–292.

Van der Heijden, P. G. M., & De Leeuw, J. (1985). Correspondence analysis used complementary to loglinear analysis. *Psychometrika*, *50*, 429–447.

Van der Heijden, P. G. M., Mooijaart, A., & Takane, Y. (1994). Correspondence analysis and contingency table analysis. In M. J. Greenacre & J. Blasius (Eds.), *Correspondence analysis in the social sciences* (pp. 79–111). London: Academic.

Van der Lans, I. A. (1992). *Nonlinear multivariate analysis for multiattribute preference data.* Leiden, The Netherlands: DSWO.

Van Rijckevoorsel, J. L. A. (1987). *The application of fuzzy coding and horseshoes in multiple correspondence analysis.* Leiden, The Netherlands: DSWO.

Van Rijckevoorsel, J. L. A., & Tijssen, R. J. W. (1987). *Historical correspondence analysis: The history of correspondence analysis based on (co-)citations* (Tech. Rep. No. RR-87-11). Leiden, The Netherlands: Department of Data Theory, Leiden University.

Van Schuur, W. H. (1989). Unfolding the German political parties: A description and application of multiple unidimensional unfolding. In G. DeSoete, H. Feger, & K. C. Klauer (Eds.), *New developments in psychological choice modeling* (pp. 259–290). Amsterdam: Elsevier.

Van Schuur, W. H., & Post, W. J. (1990). *MUDFOLD manual.* Groningen, The Netherlands: IecProGAMMA.

Van Deun, K., Groenen, P. J. F., & Delbeke, L. (2005). *Vipscal: A combined vector ideal point model for preference data* (Tech. Rep. No. EI 2005-03). Econometric Institute Report.

Van Deun, K., Groenen, P. J. F., Heiser, W. J., Busing, F. M. T. A., & Delbeke, L. (2005). Interpreting degenerate solutions in unfolding by use of the vector model and the compensatory distance model. *Psychometrika*, *70*, 45–69.

Van Deun, K., Heiser, W. J., & Delbeke, L. (2004). Multidimensional unfolding by nonmetric multidimensional scaling of Spearman distances in the extended permutation polytope. (Unpublished manuscript)

Velleman, P., & Wilkinson, L. (1994). Nominal, ordinal, interval, and ratio typologies are misleading. In I. Borg & P. P. Mohler (Eds.), *Trends and perspectives in empirical social research* (pp. 161–177). New York: De Gruyter.

Verboon, P. (1994). *A robust approach to nonlinear multivariate analysis.* Leiden, The Netherlands: DSWO.

Verboon, P., & Heiser, W. J. (1992). Resistant orthogonal Procrustes analysis. *Journal of Classification*, *9*, 237–256.

Wagenaar, W. A., & Padmos, P. (1971). Quantitative interpretation of stress in Kruskal's multidimensional scaling technique. *British Journal of Mathematical and Statistical Psychology*, *24*, 101–110.

Wang, M. M., Schönemann, P. H., & Rusk, J. B. (1975). A conjugate gradient algorithm for the multidimensional analysis of preference data. *Multivariate Behavioral Research*, *10*, 45–79.

Weeks, D. G., & Bentler, P. M. (1979). A comparison of linear and monotone multidimensional scaling models. *Psychological Bulletin*, *86*, 349–354.

Wender, K. (1969). *Die psychologische Interpretation nichteuklidischer Metriken in der multidimensionalen Skalierung.* Unpublished doctoral dissertation, Technische Hochschule Darmstadt, Germany.

Wender, K. (1971). Die Metrik der multidimensionalen Skalierung als Funktion der Urteilsschwierigkeit. *Zeitschrift für experimentelle und angewandte Psychologie*, *18*, 166–187.

Wilkinson, L. (1990). *SYSTAT: The system for statistics.* Evanston, IL: SYSTAT Inc.

Wilkinson, L. (1996). Multidimensional scaling. In L. Wilkinson (Ed.), *Systat 6.0 for Windows: Statistics* (pp. 573–606). Chicago, IL: SPSS Inc.

Wilkinson, L., & Hill, M. (1994). *SYSTAT for DOS: Using SYSTAT, version 6 edition.* Evanston, IL: SYSTAT Inc.

Winsberg, S., & Carroll, J. D. (1989). A quasi- nonmetric method for multidimensional scaling via an extended Euclidean model. *Psychometrika*, *54*, 217–229.

Winsberg, S., & De Soete, G. (1993). A latent class approach to fitting the weighted Eulidean model, CLASCAL. *Psychometrika*, *58*, 315–330.

Winsberg, S., & De Soete, G. (1997). Multidimensional scaling with constrained dimensions. *British Journal of Mathematical and Statistical Psychology*, *50*, 55–72.

Wish, M. (1967). A model for the perception of Morse code like signals. *Human Factors*, *9*, 529–539.

Wish, M. (1971). Individual differences in perceptions and preferences among nations. In C. W. King & D. Tigert (Eds.), *Attitude research reaches new heights.* Chicago: American Marketing Association.

Wish, M., Deutsch, M., & Biener, L. (1970). Differences in conceptual structures of nations: An exploratory study. *Journal of Personality and Social Psychology*, *16*, 361–373.

Wish, M., Deutsch, M., & Biener, L. (1972). Differences in perceived similarity of nations. In A. K. Romney, R. N. Shepard, & S. B. Nerlove (Eds.), *Multidimensional scaling: Theory and applications in the behavioral sciences* (pp. 289–313). New York: Academic.

Wolford, G. L., & Hollingsworth, S. (1974). Evidence that short-term memory is not the limiting factor in the tachistoscopic full-report procedure. *Memory and Cognition*, *2*, 796–800.

Wolfrum, C. (1976a). Zum Auftreten quasiäquivalenter Lösungen bei einer Verallgemeinerung des Skalierungsverfahrens von Kruskal auf metrische Räume mit einer Minkowski-Metrik. *Archiv für Psychologie*, *128*, 96–111.

Wolfrum, C. (1976b). Zur Bestimmung eines optimalen Metrikkoeffizienten r mit dem Skalierungsverfahren von Kruskal. *Zeitschrift für experimentelle und angewandte Psychologie*, *23*, 339–350.

Woodworth, R. S. (1938). *Experimental psychology.* New York: Holt.

Xue, G. (1994). Improvement on the Northby algorithm for molecular confirmation: Better solutions. *Journal of Global Optimization*, *4*, 425–440.

Young, F. W. (1970). Nonmetric multidimensional scaling: Recovery of metric information. *Psychometrika*, *35*, 455–473.

Young, F. W. (1981). Quantitative analysis of qualitative data. *Psychometrika*, *46*, 357–388.

Young, F. W. (1984). The general Euclidean model. In H. G. Law, C. W. Snyder, Jr., J. A. Hattie, & R. P. McDonald (Eds.), *Research methods for multimode data analysis* (pp. 440–469). New York: Prager.

Young, F. W. (1987). Weighted distance models. In F. W. Young & R. M. Hamer (Eds.), *Multidimensional scaling: History, theory, and applications* (pp. 117–158). Hillsdale, NJ: Lawrence Erlbaum.

Young, G., & Householder, A. S. (1938). Discussion of a set of point in terms of their mutual distances. *Psychometrika*, *3*, 19–22.

Zielman, B., & Heiser, W. J. (1993). Analysis of asymmetry by a slide-vector. *Psychometrika*, *58*, 101–114.

Zielman, B., & Heiser, W. J. (1996). Models for asymmetric proximities. *British Journal of Mathematical and Statistical Psychology*, *49*, 127–146.

Zinnes, J. L., & MacKay, D. B. (1983). Probabilistic multidimensional scaling: Complete and imcomplete data. *Psychometrika*, *48*, 27–48.

Author Index

Abelson and Sermat (1962), 74, 230, 232, 266
Ahrens (1972), 367
Ahrens (1974), 369
Andrews and Inglehart (1979), 437, 438
Arabie (1991), 369
Attneave (1950), 376
Bagozzi (1993), 105
Baird and Noma (1978), 347
Barnes et al. (1979), 6
Beaton and Tukey (1974), 445
Bell and Lattin (1998), 503
Bendixen (1996), 539
Bentler and Weeks (1978), 230, 236
Benzécri et al. (1973), 529, 532, 533
Beuhring and Cudeck (1985), 15
Bijleveld and De Leeuw (1991), 179
Bijmolt and Wedel (1995), 114
Bilsky, Borg, and Wetzels (1994), 127
Blasius and Greenacre (1998), 529
Bloxom (1968), 457
Bloxom (1978), 230
Blurton Jones (1968), 515
Boender (1984), 278
Borg (1977a), 460
Borg (1977b), 443
Borg (1978a), 443
Borg (1978b), 68
Borg (1979), 502
Borg (1988), 125, 130
Borg (1999), 288
Borg and Bergermaier (1981), 441
Borg and Bergermaier (1982), 308
Borg and Groenen (1995), 502, 518
Borg and Groenen (1997), 288
Borg and Groenen (1998), 243
Borg and Leutner (1983), 373
Borg and Leutner (1985), 440
Borg and Lingoes (1977), 463, 465
Borg and Lingoes (1980), 239
Borg and Lingoes (1987), 13, 243
Borg and Shye (1995), 87, 102, 241
Borg and Staufenbiel (1984), 370
Borg and Staufenbiel (1986), 387
Borg and Staufenbiel (1993), 93, 313
Borg and Tremmel (1988), 382
Borg, Schönemann, and Leutner (1982), 372
Bortz (1974), 370, 371
Bove and Critchley (1993), 512
Broderson (1968), 11
Brokken (1983), 444
Browne (1969), 444
Browne (1972a), 442
Browne (1972b), 444
Browne (1987), 252
Browne and Kristof (1969), 444
Brusco (2001), 281
Brusco (2002), 281
Brusco and Stahl (2000), 281

Buja and Swayne (2002), 221, 222, 224, 255, 544
Buja, Logan, Reeds, and Shepp (1994), 274
Burton (1975), 126
Busing (2004), 330
Busing (2005), 325
Busing, Groenen, and Heiser (2005), 325–329
Cailliez (1983), 419
Carroll (1972), 338, 340, 343–345
Carroll (1980), 303, 342, 345
Carroll and Arabie (1980), 58
Carroll and Chang (1970), 457, 460, 477, 478
Carroll and Chang (1972), 263
Carroll and Wish (1974a), 458, 479, 480
Carroll and Wish (1974b), 479, 480, 506
Carroll, Green, and Carmone (1976), 265
Chang and Carroll (1969), 340
Chaturvedi and Carroll (1994), 482
Cliff (1966), 431
Cliff (1973), 48
Cohen and Davison (1973), 115
Cohen and Jones (1973), 53
Commandeur (1991), 442, 454, 455, 469
Commandeur (1993), 248
Commandeur and Heiser (1993), 476, 553
Constantine and Gower (1978), 498
Coombs (1950), 295
Coombs (1964), 307, 517
Coombs (1967), 125, 130
Coombs (1975), 338
Coombs and Avrunin (1977), 313, 345
Coombs, Dawes, and Tversky (1970), 517
Cooper (1972), 424
Cox and Cox (1990), 54
Cox and Cox (1991), 240
Cox and Cox (1992), 54
Cox and Cox (1994), 128
Coxon and Jones (1978), 126
Critchley (1986), 282
Critchley and Fichet (1994), 367
Cross (1965a), 363, 365
Cross (1965b), 364, 365
Daniels (1944), 120
Davison (1983), 131, 314
De Boor (1978), 215, 218
Defays (1978), 278, 279
De Leeuw (1977, 1988), 179
De Leeuw (1977), 179, 187, 200, 248, 250
De Leeuw (1984), 253
De Leeuw (1988), 187, 194, 200, 249, 250
De Leeuw (1993), 180, 181, 250, 281
De Leeuw (1994), 179
De Leeuw and Groenen (1997), 254, 282
De Leeuw and Heiser (1977, 1980), 179
De Leeuw and Heiser (1977), 187, 200, 214, 278
De Leeuw and Heiser (1980), 191, 230, 231, 233, 249, 475, 476, 480, 553
De Leeuw and Heiser (1982), 13, 265, 506
De Leeuw and Stoop (1984), 274, 275
De Rooij and Heiser (2000), 515
DeSarbo and Rao (1984), 325
DeSarbo, Manrai, and Manrai (1994), 492
De Soete, Carroll, and Chaturvedi (1993), 478
De Soete, Hubert, and Arabie (1988), 280
Diederich, Messick, and Tucker (1957), 75
Dijksterhuis and Gower (1991), 456
Drösler (1979), 13, 385
Drösler (1981), 32
Dunn-Rankin (1983), 132
Dunn-Rankin, Knezek, Wallace, and Zhang (2004), 494
Eckart and Young (1936), 261
Eisler and Lindman (1990), 403
Eisler and Roskam (1977), 403
Ekehammar (1972), 403
Ekman (1954), 64, 65, 82, 238
Ekman (1963), 392–395, 400, 403, 405
Ekman, Engen, Künnapas, and Lindman (1964), 397, 401–403
Ekman, Goude, and Waern (1961), 401
Ekman and Lindman (1961), 403
Elizur, Borg, Hunt, and Magyari-Beck (1991), 95, 120
Engen, Levy, and Schlosberg (1958), 74, 75, 129, 230
England and Ruiz-Quintanilla (1994), 127
Fechner (1860), 361
Feger (1980), 468
Fichet (1994), 367
Fischer and Micko (1972), 366

Fletcher (1987), 181
Foa (1958), 241
Foa (1965), 241
Fréchet (1910), 367
Gabriel (1971), 523, 524
Gaffke and Mathar (1989), 252
Galinat and Borg (1987), 96, 99
Garmize and Rychlak (1964), 332
Garner (1962), 366
Garner (1974), 367
Gati and Tversky (1982), 379, 385
Gifi (1990), 233, 341, 524, 529, 532, 537
Gilula and Haberman (1986), 533
Gleason (1967), 300
Glunt, Hayden, and Liu (1991), 252
Glunt, Hayden, and Raydan (1993), 251
Glushko (1975), 125
Goodman (1985), 533
Goodman (1986), 533
Goude (1972), 403
Gower (1966), 162, 261, 263, 277, 526
Gower (1971), 124, 125
Gower (1975), 455
Gower (1977), 498, 503, 511
Gower (1985), 127, 128
Gower and Dijksterhuis (2004), 450
Gower and Groenen (1991), 191
Gower and Hand (1996), 524
Gower and Legendre (1986), 128, 321
Graef and Spence (1979), 117
Gramlich (2004), 153, 154
Green (1952), 431
Green (1974), 55–57
Green and Carmone (1970), 296, 297
Green and Carroll (1976), 430
Green and Rao (1972), 108, 293, 294, 345, 462
Green and Wind (1973), 118
Greenacre (1984), 527, 529, 532
Greenacre (1994), 529
Greenacre and Blasius (1994), 529
Greenberg (1980), 32
Groenen (1993), 179, 236, 278–280, 283, 284, 336
Groenen and Franses (2000), 60
Groenen and Gifi (1989), 527, 529
Groenen and Heiser (1991), 283
Groenen and Heiser (1996), 277, 283, 284, 369, 504
Groenen and Heiser (2000), 253
Groenen and Meulman (2004), 526
Groenen and Van de Velden (2004), 529
Groenen, Heiser, and Meulman (1999), 179, 285, 286, 369
Groenen, De Leeuw, and Mathar (1996), 253, 254
Groenen, Mathar, and Heiser (1995), 179, 369, 477
Groenen, Mathar, and Trejos (2000), 278
Gulliksen (1946), 14
Guthrie (1973), 270
Guttman (1941), 529
Guttman (1954), 92, 441
Guttman (1959), 87
Guttman (1965), 15, 88, 241
Guttman (1968), 13, 68, 187, 191, 214
Guttman (1971), 103
Guttman (1976), 243
Guttman (1977), 104, 489
Guttman (1981), 252
Guttman (1982), 105
Guttman (1985), 120
Guttman (1991), 87, 102
Guttman and Kalish (1956), 360
Guttman and Levy (1991), 88, 91, 92, 101, 106
Harshman (1972), 479
Harshman and Lundy (1984), 479
Hefner (1958), 51
Heiser (1981), 298
Heiser (1985), 208, 323
Heiser (1988a), 182, 250, 254
Heiser (1988b), 474, 476
Heiser (1989a), 208, 304, 323
Heiser (1989b), 369, 477
Heiser (1990), 199, 202, 555
Heiser (1993), 236
Heiser (1995), 179, 182
Heiser and De Leeuw (1979), 322
Heiser and Groenen (1997), 236, 243
Heiser and Meulman (1983a), 529
Heiser and Meulman (1983b), 233
Helm (1959), 451–453
Helm (1964), 451
Hirschfeld (1935), 529
Horan (1969), 457
Hotelling (1933), 519, 529
Huber (1964), 445
Hubert and Arabie (1986), 278, 280
Hubert and Golledge (1981), 280
Hubert, Arabie, and Hesson-McInnis (1992), 369, 375
Hubert, Arabie, and Meulman (1997), 240

Hunter and Lange (2004), 179
Hurley and Cattell (1962), 430
Indow (1974), 462
Isaac and Poor (1974), 53
Kaiser (1958), 161
Kearsley, Tapia, and Trosset (1998), 253
Kendall (1971), 124
Kiers (1990), 179
Kiers (2002), 179
Kiers and Groenen (1996), 444
Kim, Rangaswamy, and DeSarbo (1999), 323, 331
Klahr (1969), 48
Kloek and Theil (1965), 263
Krantz (1967), 346
Krantz (1972), 361
Krantz and Tversky (1975), 112, 118, 375, 381, 477
Krantz, Luce, Suppes, and Tversky (1971), 377
Kristof (1970), 431
Kristof and Wingersky (1971), 455
Krumhansl (1978), 385
Kruskal (1964a), 42, 47, 48, 55, 71, 251, 368
Kruskal (1964b), 171, 187, 206
Kruskal (1968), 302
Kruskal (1977), 199
Kruskal and Carmone (1969), 71, 300
Kruskal and Carroll (1969), 251, 302, 303
Kruskal and Wish (1978), 9, 47
Kruskal, Young, and Seery (1978), 71, 564
Lange, Hunter, and Yang (2000), 179
Langeheine (1980a), 467
Langeheine (1980b), 439
Langeheine (1982), 439, 440, 467
Lawler (1967), 16
Lawson and Hanson (1974), 221, 478
Lazarte and Schönemann (1991), 376, 377, 382
Lebart, Morineau, and Warwick (1984), 529
Leutner and Borg (1983), 440
Levelt, Geer, and Plomp (1966), 244
Levy (1976), 106, 442
Levy (1983), 6, 7, 245
Levy and Guttman (1975), 120
Lindman and Caelli (1978), 385
Lingoes (1965), 57
Lingoes (1971), 419
Lingoes (1973), 563
Lingoes (1981), 104
Lingoes (1989), 299, 314
Lingoes and Borg (1977), 462, 469
Lingoes and Borg (1978), 461, 462, 466, 469
Lingoes and Borg (1983), 244
Lingoes and Guttman (1967), 405
Lingoes and Roskam (1973), 214
Luce (1959), 346
Luce (1961), 346
Luce and Suppes (1963), 346
Luengo, Raydan, Glunt, and Hayden (2002), 253
Lüer and Fillbrandt (1970), 372
Lüer, Osterloh, and Ruge (1970), 372
MacCallum (1976), 485
MacCallum (1977), 490
Mathar (1989), 187, 250
Mathar (1994), 250
Mathar (1997), 254
Mathar and Groenen (1991), 187, 248, 250
Mathar and Meyer (1993), 266
Mathar and Žilinskas (1993), 250
McGee (1966), 255
Merkle (1981), 132, 133
Messick and Abelson (1956), 420–423
Meulman (1986), 179, 526
Meulman (1992), 179, 526
Meulman, Heiser, and De Leeuw (1983), 233
Meulman, Heiser, and SPSS (1999), 553
Meyer (1993), 250
Mezzich (1978), 17, 18
Micko and Fischer (1970), 366
Miller (1956), 118
Möbus (1975), 490
Möbus (1979), 503
Mooijaart and Commandeur (1990), 462
Moorhead and Griffin (1989), 93
Mulaik (1972), 161, 444
Müller (1984), 13, 111
Nishisato (1980), 529
Nishisato (1994), 529
Noma and Johnson (1977), 228, 229
Norpoth (1979a), 313, 316
Norpoth (1979b), 315
Okada (1990), 513, 514
Okada and Imaizumi (1987), 512, 513
Okada and Imaizumi (1997), 515
Ortega and Rheinboldt (1970), 179
Pearson (1896), 327
Pearson (1901), 519

Pease (1965), 480
Peay (1988), 442
Pliner (1996), 278, 280–282, 284, 369
Poole (1984), 280
Poole (1990), 280
Poor and Wherry (1976), 439
Porrat (1974), 47
Rabinowitz (1975), 47, 206, 348
Rabinowitz (1976), 322
Ramsay (1969), 52
Ramsay (1977), 52, 253, 255, 565
Ramsay (1988), 215, 218
Restle (1959), 124, 377, 397
Richardson (1938), 14
Richardson and Kuder (1933), 529
Rosenberg and Kim (1975), 83
Roskam (1972), 424
Roskam (1979a), 214
Roskam (1979b), 345
Roskam and Lingoes (1981), 560
Ross (1934), 115
Rothkopf (1957), 68, 70, 118, 234, 235, 498
SAS (1999), 354
SPSS (1990), 529, 532
Sammon (1969), 255
Schönemann (1966), 431
Schönemann (1970), 264, 347
Schönemann (1972), 477, 481, 482
Schönemann (1982), 382
Schönemann (1983), 383
Schönemann (1985), 178
Schönemann (1990), 383
Schönemann (1994), 376
Schönemann and Borg (1981a), 480
Schönemann and Borg (1981b), 381
Schönemann and Borg (1983), 340, 377
Schönemann and Carroll (1970), 434
Schönemann and Lazarte (1987), 376
Schönemann and Wang (1972), 347
Schönemann, Dorcey, and Kienapple (1985), 376, 383, 387
Schönemann, James, and Carter (1978), 488
Schönemann, James, and Carter (1979), 486, 488, 489
Schulz (1972), 480
Schulz (1975), 480
Schulz (1980), 480
Schulz and Pittner (1978), 490
Schumaker (1981), 215
Schwarz and Bilsky (1987), 103
Searle (1982), 148
Shepard (1957), 40, 361, 362
Shepard (1958a), 362
Shepard (1958b), 362
Shepard (1963), 71
Shepard (1964), 368
Shepard (1965), 361
Shepard (1966), 439
Shepard (1969), 372
Shepard (1974), 244, 371
Shepard and Carroll (1966), 260
Sherman (1972), 53, 54
Shinew, Floyd, McGuire, and Noe (1995), 108
Shye (1981), 95, 441
Shye (1985), 103, 120
Shye (1991), 101, 563
Shye, Elizur, and Hoffman (1994), 107
Sibson (1979), 263
Sixtl (1967), 397, 408
Sjöberg (1975), 402
Snow, Kyllonen, and Marshalek (1984), 92
Spence (1972), 71
Spence (1978), 515
Spence (1979), 49
Spence (1983), 115, 117
Spence and Domoney (1974), 115
Spence and Graef (1974), 51, 52, 71
Spence and Ogilvie (1973), 49, 51
Srinivasan and Shocker (1973), 345
Staufenbiel (1987), 120
Staufenbiel and Borg (1987), 375, 381, 382
Stenson and Knoll (1969), 48, 50
Stephenson (1953), 113
Steverink, Van der Kloot, and Heiser (2002), 319, 321
Steyvers and Busey (2000), 386
Strang (1976), 75, 140
Suppes and Zinnes (1963), 52
Takane and Carroll (1981), 53
Takane, Young, and De Leeuw (1977), 252, 253, 487, 489, 551
Ten Berge (1977), 455
Ten Berge (1991), 239
Ten Berge (1993), 431
Ten Berge and Kiers (1991), 479
Ten Berge, Kiers, and Commandeur (1993), 442, 470
Ten Berge, Kiers, and Krijnen (1993), 478
Tenenbaum (1998), 259

Tenenbaum, De Silva, and Langford (2000), 266, 267
Ter Braak (1992), 265
Thurstone (1927), 40, 51, 125, 130
Thurstone (1935), 441
Thurstone (1947), 405
Tobler and Wineburg (1971), 126
Torgerson (1952), 14, 261
Torgerson (1958), 261, 277, 365, 366, 398, 426
Torgerson (1965), 372
Trosset (1993), 266
Tucker (1951), 248
Tucker (1960), 336
Tucker (1972), 479
Tversky (1977), 115, 385, 496
Tversky and Gati (1982), 377, 381, 384, 385
Tversky and Krantz (1970), 381
Van der Heijden and De Leeuw (1985), 533
Van der Heijden, De Falguerolles, and De Leeuw (1989), 533
Van der Heijden, Mooijaart, and Takane (1994), 533
Van der Lans (1992), 179
Van Deun, Groenen, and Delbeke (2005), 341
Van Deun, Groenen, Heiser, Busing, and Delbeke (2005), 302
Van Deun, Heiser, and Delbeke (2004), 322
Van Rijckevoorsel (1987), 529
Van Rijckevoorsel and Tijssen (1987), 529
Van Schuur (1989), 313
Velleman and Wilkinson (1994), 103
Verboon (1994), 444, 445
Verboon and Heiser (1992), 179, 444
Wagenaar and Padmos (1971), 51
Wang, Schönemann, and Rusk (1975), 347–349, 351
Weeks and Bentler (1979), 56, 57
Wender (1969), 370
Wender (1971), 366
Wilkinson (1990), 4
Wilkinson (1996), 315
Wilkinson and Hill (1994), 560
Winsberg and Carroll (1989), 236
Winsberg and De Soete (1993), 492
Winsberg and De Soete (1997), 476
Wish (1967), 71
Wish (1971), 9, 10, 45, 46, 212
Wish, Deutsch, and Biener (1970), 212
Wish, Deutsch, and Biener (1972), 331, 333
Wolford and Hollingsworth (1974), 84
Wolfrum (1976a), 370
Wolfrum (1976b), 370
Woodworth (1938), 74
Xue (1994), 254
Young (1970), 53
Young (1981), 233
Young (1984), 484
Young (1987), 493
Young and Householder (1938), 261
Zielman and Heiser (1993), 504, 506
Zielman and Heiser (1996), 495, 498
Zinnes and MacKay (1983), 53

Subject Index

Additive constant, 69, 376, 418, *419*, 477
  algebraic, 424
  as part of a loss function, 424
  geometric effects, 423
  in interval MDS, 416
  minimal, 420
  nonminimal, 423
  statistical, 420
  too large/small, 422
  true, 422
Algorithm, *178*
  combinatorial, 375
  for GPA, 454
  for MDS, 250
  for monotone regression, 206
  for unfolding, 297
  for unidimensional scaling, 280
  gradient-based, 375
  majorization, *178*, 231, 297, 444
  spectral gradient, 251
Alienation coefficient, *251*, 300, 305, 560, 562
  norms for, 47
Alscal, 203, 252, 487, 490, *551*
Alternating least squares, 221, 344, 476
Analyzable stimuli, 366, 368
Anchor stimulus method, 114
Asymmetry, 69, 395, *495*
  drift vector model, 506
  Gower model, 498
  hill-climbing model, 509
  jet-stream model, 511
  radius-distance model, 512
  slide-vector model, 506
  unfolding, 503
  using models, 514
Attribute profiles, 120
Axioms, 31
  consistency, 379
  decomposability, 379
  dimensional, 379, 380
  distance, 33, 380
  dominance, 379
  interdimensional additivity, 380
  intradimensional subtractivity, 379
  metric, 380
  minimality, 381
  monotonicity, 379
  of algebraic groups, 414
  of measurement, 378
  of scalar products, 413
  segmental additivity, 381
  symmetry, 381
  transitivity, 379
  triangle inequality, 381

Barycentric principle, 532
Bimension, 498
Biplot, 523

Blind procedures, 104
BTL model, 345

Candecomp, 478, 554
Card sorting, 113
Cartesian coordinate system, 170
Cartesian plane, 38
Catpca, 341
Centered configuration, 75
Circle in MBR plane, 383
Circle in Minkowski plane, 365
Circular unidimensional scaling, 240
Circumplex, 100
City-block
  distance, 12, 32, *121*, 364, 367, 369, 477
  maximum dimensionality, 367
Clascal, 492
Classical scaling, *261*, 267, 282, 420, 451, 453, 477, 481
  and correspondence analysis, 537
  and principal coordinates analysis, 526
  constrained, 265
Cluster analysis, 104, 236, 492
Cluster differences scaling, 236
Co-occurrence data, 126
Coefficient of determination, 249, 252
Coefficient of variation, 327
Common elements correlation, 124
Common-space, 474
  condition, 485
  index, 486, 488
Communality, 460, 463
Composition rule, 13
Concave, 181
Conditional fit measure, 67
Conditional unfolding, 300
Conex, 100
Configurations, 20, 23
Confirmatory MDS, 82, *227–244*, 283
Congruence coefficient, 122, 248, 252, 440
Conjoint, 554
Constant dissimilarities, 274
Constrained MDS, *230–236*, 265
  for three-way models, 475
  weakly, 237
Constraint
  external, 230
  internal, 230
  on the configuration, 230, 265
Containment judgments, 389, 392, 393, 397
Content model, 403
Contiguity patterns, 95
Convex, 282
  analysis for minimizing σ_r, 187, 250
  curve, 48, 51
Coordinate
  axes, 38
  simple structure, 405
  vector, 38
Corner triples, 384
Correlation, 122
  and Euclidean distances, 130
Correlations
  all positive, 105
  product-moment, 391
Corresp, 554
Correspondence analysis, 153, 526
  maximum likelihood, 533
Cosine law, 390, 398
Cospa, 488
Cylindrex, 91, 100

Data
  ability scores, 538
  automobile preferences, 354
  bird threat behavior, 515
  brand switching of colas, 505
  brand switching of colas (gravity model), 505
  breakfast items (Green), 294
  breweries (Borg), 308
  co-occurrence of buying clothes, 132
  color (Torgerson), 426
  color preferences (Davison, 1983), 314
  color preferences (Wilkinson, 1996), 315
  colors v-data (Ekman), 394
  colors (Ekman), 65, 238
  colors (Helm), 452
  crime rates (U.S. census)
    correlations, 4
    frequencies, 534
  crimes and offenses (Borg), 131
  ellipses (Noma), 229
  European cities, 20
  faces (Abelson), 76
  faces scales (Engen), 75
  free sorting of words with a, 132
  intelligence tests (Guttman), 15, 90, 241
  job performance (Lawler), 16

journal citations (Coombs), 517
kinship terms (Rosenberg & Kim), 83
KIPT tests (Guthrie), 270
leisure activities, 108
letter confusion (Wolford & Hollingsworth), 84
Morse signals (Rothkopf), 70
multitrait-multimethod (Bagozzi), 105
nations, 333
political preferences (Norpoth, 1979), 315
politicians (Rabinowitz), 207
politicians BTL (Wang), 348
preferences family composition, 354
protest acts (Levy, 1983), 245
prototypical psychiatric patients (Mezzich), 18
rectangles (Borg), 374
rectangles (Schönemann), 387
Rorschach pictures, 332
satisfaction with life, 106
similarity of handicaps, 494
similarity of tones (Levelt et al.), 244
statements on breakfast items, 539
stock market returns (Groenen & Franses), 60
twelve nations (Wish), 10
external scales, 85
vocational interests (Beuhring & Cudeck), 15
Wechsler IQ-test, 107
well-being (Andrews), 438
wheels (Broderson), 12
work values, 107, 538
Degenerate solution, 200, 270
how to avoid, 272
in ordinal unfolding, 301
Degree of a spline, 216
Derivative
first, 173
of a function, 173
partial, 176
second, 175
d-hat, 42, 199, 214
Diagnostic
Shepard diagram, 44
Stress per point, 44
Diagonality condition, 487, 488
Diagonality index, 488
Differentiation, 174
Dilation, *23*, 429, 434
Dilation model, 482
Dimension-weighting
algebra of model, 485
conditional, 489
model, 473
Stress, 475
unconditional, 489
Dimensional
assumptions, 380
interaction, 375
Dimensionality
choice of, 64
proper, 47
true, 117
wrong, 54
Dimensions, 38
perpendicular, 38
physical, 360
psychological, 360
underlying, 9
Directed line, 77
Direction cosine, 78
Disparity, 42, 199, 200, 239
negative, 202, 555
Dissimilarities, 3, 37, 111, 170
and error, 419
and Euclidean distances, 416
constant, 274
interval transformations, 416
Distance
and origin of point space, 400
by straight-ruler, 413
city-block, 12, 32, *121*, 364, 367, 369, 374, 477
dominance, 364, 367
elliptical, 480
Euclidean, 14, *39*, 121, 144, 364, 367, 389, 411, 413
extended Euclidean, 236
general, 413
generalized Euclidean, 457, 458, 479, 484
Kemeny, 319, 320
minimality of, 378
Minkowski, 363
nonnegativity, 33
on a circle, 412
pseudo, 199
psychological, 361
symmetry of, 33, 378
target, 199
triangle inequality, 412

trivial, 33
weighted Euclidean, 457, 458, 474, 480, 529
Distance smoothing, 285
Distance-based PCA, 526
Distributional equivalence, 533
Dominance data, 112
Dominance distance
maximum dimensionality, 367
Dominance probabilities, 125
Double centering, 262
Drift vectors, 502
d-star, 214
Duplex, 100
Dynamic programming, 280

Eckart–Young theorem, 150
Effect hypothesis, 6
Eigendecomposition, 146
computation, 157
power method, 157
Eigenequation, 146
Eigenvalues, 146
Eigenvectors, 146
Elastic scaling, 255
Embedding in Euclidean space, 414
Error of representation, 41, 170
Euclidean
embedding, 411, 413–415
space, 411
Euclidean distance
approximate, 420
computing, 39
definition, 39, 413
maximum dimensionality, 367, 419
particular properties, 413
Euclidean metric
robustness, 372
Extended Euclidean model, 236
External scales, 76
embedded, 79
fitting, 77
External unfolding, 335

Facet, 88
and regions, 87
axial, 99
content, 96
diagram, 95
modular, 99
multiple, 93
nominal, 104
ordered, 103
ordinal, 103
polar, 99
purposiveness of, 89
qualitative, 103
roles in MDS, 99
scale level of, 103
separation index, 102
theory, 87, 88
Factor analysis, 92, 104, 161, 441
nonmetric, 405
Feature-set models, 401
Fit per point, 46
Free sorting, 114
FSSA, 563
Full rank decomposition, 152
Full-dimensional scaling, 281
Function
biweight, 445
gradient of, 176
Huber, 445
loss, 37
minimization by attainable lower bound, 431
minimization by iterative majorization, 178, 445
partial derivative, 176
representation, 39
robust, 445
slope of, 172
trace, 143, 144, *176*, 570

Generalization gradients, 360
Generalized Euclidean distance, 458, 479, 484
Generalized Euclidean model
subjective-metrics interpretation, 480, 485
subjective-transformation interpretation, 480, 485
Generalized inverse, 155, 570
Generalized Procrustes analysis, 454
Geodesic, 39, 412
Geometry
curved, 30, 32
Euclidean, 31
exotic, 13
flat, 30
natural, 31
GGOBI, 544
GGVIS, 544
Global minimum, *172*, 276, 277
Gower's decomposition, 498
Gradient, 176

Gravity model, 126, 504
Group algebra, 414
Group space, 460, 474
Guttman scale, 92
Guttman transform, *191*, 194, 230, 279, 298, 475

Hefner model, 50, 71
HICLUS, 554
Hill-climbing model, 509

Ideal point, 295
  anti, 345
Identity model, 482
IDIOSCAL, 458, *480–482*
IM, *see* (iterative) majorization
Incidence matrix, 126
Incomplete data designs, 115
INDCLUS, 482
Individual differences
  scaling, 474, 491
  subject space, 460
INDSCAL, 457, 465, *477–479*, 490, 554
Inequality
  Cauchy–Schwarz, *120*, 189, 248, 440
  Kristof, 431
  sandwich, 180
  triangle, *33*, 143, 376, 381, 412
Initial configuration, 41
  in ordinal MDS, 57, 425
Integral stimuli, 366
Internal scales, 76
Interpretation
  clusters, 5, 81
  dimensional, 5, 9, 73, 74, 82
  of manifolds, 5
  of MDS solution, 55, 80
  regional, 5, 9, 81
Interval MDS, 42, 201
  additive constant, 416
Intradimensional additivity, 376
Invariance, 23
Inverse MDS, 254
Isometry, 23
  partial, 371, 372
Isosimilarity curve, 364
Isotonic region, 29, 306
I-spline, 214
Iterative improvements, 41
Iterative majorization, *see* majorization

Jaccard similarity measure, 127
Jet-stream model, 511
Judgment model, 13

Just noticeable differences (JND), 361

Kemeny distance, 319, 320
Knots, 215
KYST, 71, 240, 253, 286, 308, 318, 319, 324, 331, 369, 376, 385, 564

Latent class MDS, 492
Law of comparative judgment, 125
Limit operator, 173
Linear equation systems, 154
  approximate solutions, 156
  inconsistent, 155
  solving, 155
  underdetermined, 156
Linear regression, 77, 311
Linearizing data, 383
Local minimum, *171*, 228, 276
  for MDS with Minkowski distances, 369

μ_2, 68, *120*, 130, 442
Majorization, 297, 444, 475
  iterative, 178, 445
  linear, 181
  majorizing function, 179, 570
  of Stress with Minkowski distances, 369
  quadratic, 182, 369
  supporting point, 179
Manifold, 81
Mapping sentence, 88, 96
MATCHALS, 442, 469
Matrix, 137
  and configurations, 148
  and transformations, 148
  centering, 149, 191, 262, 435, 537, 570
  diagonal, 140
  Euclidean scalar product, 392
  full rank, 140, 148
  g-inverse, 155
  generalized inverse, 570
  idempotent, 149, 264
  identity, 139, 570
  indicator, 236, 482
  inverse, 139, 150, 570
  irreducible, 187
  Moore–Penrose inverse, *157*, 191, 298, 570
  multiplication from the right, 139
  negative definite, 149
  negative semidefinite, 149
  order, 138

orthogonal, 140
orthonormal, 140
permutation, 213, 279
positive definite, 149
positive semidefinite, 149, 282, 417
postmultiplication, 139
power, 570
premultiplication, 139
projector, 264
pseudoinverse, 157
quadratic, 138
rank, 148
rank deficient, 152
rectangular, 137
reflection, 434
rotation, 162, 434
scalar product, 145, 160, 417
singular, 152
singular value decomposition, *150*, 444
skew-symmetric, 496
square, 138, 139
symmetric, 138
trace of, 143, 144, *176*
transpose, 138
two-mode, 294
Maximum dimensions
for Euclidean distances, 419
for Minkowski distances, 367
required, 418
Maximum likelihood, 53, 253
correspondence analysis, 533
MDS, 253
MBR metric, 382
MDPREF, 340, 554
MDS
absolute, 200
almost complete, 336
and factor analysis, 105
as a psychological model, 11
axiomatic approach, 111
by computation, 19
by geometric construction, 19
by hand, 5
circular constraints, 237
confirmatory, 475
exploratory, 4
interval, 40, 201, 415
inverse, 254
maximum likelihood, 53
metric, 55, 200, 203
model, 39, 200
non-Euclidean, 32
nonmetric, 200, 203, 251, 277
of scalar products, 403, 405
ordinal, 24, 40, 200
purpose, 3
ratio, 20, 23, 200, 201, 430
ratio vs. ordinal, 29
ruler-and-compass approach, 20
semi-complete, 336
solution, 20
true underlying space, 51
MDSORT, 554
Measurement
axiomatic, 377
Median, 182
Metric MDS, 55, 200
MINI-RSA, 554
Minimum
global, *172*, 277, 431
local, *171*, 228
MINISSA, 214, 299, 369, 554, 563
Minkowski
metric, 363
parameter, 242
Minkowski distances, 363, 413
and Stress, 367
exchangability, 367
family of, 364
maximum dimensionality, 367
true, 367
Minkowski spaces
axioms, 377, 379
distinguishing, 369
Missing data, 42, 169, 171
designs, 117
Model
bounds of, 385
building, 13
building vs. data analysis, 14
deterministic, 485
dimensional, 14
of judgment, 372
perspective, 468
Modes of data, 58
Monotone function
strong, 40
weak, 40
Monotone regression, 42, *205*, 206, 220, 233
geometry of, 209
ordered cone, 210
smoothed, 208, 323
Monotonicity coefficient, 252

Moore–Penrose inverse, *157*, 191, 298, 570
MRSCAL, 554
Multidimensional similarity structure analysis, 13
Multiplex, 100
  and dimensions, 100
MULTISCALE, 53, 253, 255, 565
Multistart, 277

Negative definite matrix, 149
Negative semidefinite matrix, 149
NEWMDSX©, 553
Non-Euclidean spaces
  interpreting, 371
Nonmetric MDS, 200
Nonnegative least-squares, 221, 478
Norm, 143
  Euclidean, 143, 390, 570
  function, 390
  orthonormal invariant, 266
Null space, 148

Object point, 295
Optimal scaling, 233
Order
  equivalence, 24
  of a spline, 216
Ordered cone, 210
Ordinal MDS, 24, 200
  maximum dimensionality, 419
  rank-images, 213
Orthogonal matrix, 140
Orthonormal matrix, 140
Outlier, 43
Over-compression, 47, 54
Over-fitting, 47

PA, *see* Principal, axes rotation
Pairwise comparison, 113
PARAMAP, 554
Partial derivative, 176
Partitioning, 73
  an MDS space, 90
  and dimensions, 99
  choosing lines, 90
  errors in, 97
  polar, 105
Partitioning lines, 72
  robust, 100
PCA, 519, *see* principal components analysis
  distance-based, 526
PCO, *see* principal coordinates analysis

Pearson's r
  and Euclidean distances, 130
Penalty term, 239
Perceptual space, 50
Perfect representation, 41
  in full-dimensional scaling, 282
PERMAP, 548
Perspective model, 468
PINDIS, 469, 554, 563
Plot
  bi, 523
  Shepard, 43
  transformation, 202
Polar coordinates, 99
Positive definite matrix, 149
Positive semidefinite matrix, 149, 417
Positively homogeneous function, 248
Power method, 157
PREFMAP, 340, 554
PREFSCAL, 327
PRINCALS, 341
Principal
  axes orientation, 162
  axes rotation, 75, 97, 161, 262, 442, 530
  axis, 403
  component, 407, 519
  components analysis, 64, 341, *519*
  coordinates analysis, 526
PRO-FIT, 554
Procrustean similarity transformation, 455
Procrustes rotation, 429, 430
  generalized, 454
  oblique, 444
  robust, 445
Projection
  orthogonal planes, 98
  plane, 97
Projection plane, 7
Projector, 149
Proximities, *37*, 169
  angular separation, 122
  Bray–Curtis distance, 122
  Canberra distance, 122
  choosing particular measures, 128
  chord distance, 122
  city-block distance, 122, 130
  coarse, 118
  conditional, 114
  congruence coefficient, 122
  correlations, 120, 122
  degraded, 118

derived, 112, 119, 125
direct, 112, 129
dominance distance, 122
Euclidean distance, 122, 130
forms of, 112
free sorting, 114
from attribute profiles, 121
from card sortings, 129
from co-occurrence data, 126
from common elements, 124
from conversions, 125
from dominance probabilities, 126
from feature set measures, 124
from index conversions, 125
from laboratory studies, 129
from surveys, 129
Gower's general similarity measure, 125
granularity of, 118
gravity model, 126, 504
incomplete, 115
incomplete designs for, 115
Jaccard measures, 127
μ_2, 120, 122
Minkowski distance, 122
missing, 117
monotonic correlations, 130
position effects, 115
profile distance, 121
profile similarity, 130
Q-sort, 113
rankings, 113
ratings, 113
s_2 coefficient, 127
s_3 coefficient, 127
s_4 coefficient, 128
simple matching coefficient, 128
Spearman's ρ, 120
timing effects, 115
types, 111
PROXSCAL, 235, 240, 476, 508, 509, 553
Pseudo distances, 199
Psychological space, 359
Psychophysical
maps, 360, 377
scaling, 14
Psychophysical maps, 377

Q-sort, 113
Quasi-equivalency, 370

Radex, 91, 95, 100, 400
Radius-distance model, 512
Rank of matrix, 148
Rank-images, 213, 251
Rank-interval MDS, 56
Ranking
complete, 113
numbers, 24
Ratio MDS, 430
Rational starting configuration, 425
Recovering distances, 53, 55
Recovery
metric, 54
Rectangles
psychology of, 372
Reduced rank model, 484
Reflection, 22, 23, 147, 429
Regional hypothesis, 89, 91
falsifiability of, 101
Regions, 25, 71
and clusters, 104
and factors, 104
and theory, 102
choosing, 100
interpreting, 87
laws, 91
ordered, 98
predicting, 102
simple, 100
Regression
linear, 42, 77, 311
monotone, 42, *205*, 206, 220, 233
polynomial, 219
Response function, 377
Rigid motion, 23
Root mean squared residual, 68
Rotation, 23, 429
idiosyncratic, 462
matrix, 162
oblique Procrustes, 444
principal axes, 75, 162, 262, 442, 530
Procrustes, 430
simple structure, 161, 405
varimax, 105, 161
Rotations, 160
Row-conditional unfolding, 300

σ, *see* Stress, formula 1
σ_1, *see* Stress, formula 1
σ_2, *see* Stress, formula 2
Sammon mapping, 255
SAS, 557
Scalar, 139
Scalar function, 142, 390

Scalar products, 142, 389
and distances, 400
and Euclidean distances, 398
and origin of point space, 400
and translations, 398
axioms of, 413
empirical, 392, 394
matrix, 392
Scale level of proximities, 40
Scales
factor, 20
rating, 30
Scree
plot, 48, 52, 71
test, 66
Set-theoretic models, 397
Shepard diagram, *43*, 44, 47, 65, 270, 275, 303, 308–311, 319, 359, 363
Similarity, 3, 37, 111
Similarity of MDS configurations, 95
Simple-structure rotation, 161, 405
Simplex, 92, 100, 274
Simulated annealing, 281
Singular value decomposition, *150*, 232, 432, 444, 476, 520, 531
and least-squares, 153
left singular vectors, 150
of symmetric matrix, 153
rank deficient case, 152
right singular vectors, 150
singular value, 150
Skew-symmetry
drift vectors, 502
matrix, 496
Slide-vector model, 506
Slope of a function, 172
SMACOF, *187–194*, 204, 205, 298, 369
Smallest space analysis, 13
Smoothing out noise, 41
σ_n, *see* Stress, normalized
Solution
avoiding degeneracy, 272
degenerate, 270
empty space, 27
set, 25, 27
space, 25, 27
Space
physical, 12
physical vs. psychological, 361
psychological, 11
subject, 460
Spearman correlation, 67
and Euclidean distances, 130
Spectral decomposition, 146, 407, 524
properties, 147
Spectral gradient algorithm, 251
Spherex, 100
Spline, 214, 326
degree of, 216
integrated, 214
interior knots, 215
knots, 215
monotone, 214
order of, 216
σ_r, *see* Stress, raw
SSA program, 563
SSA-III, 405
S-Stress, 252, 254, 282, 344
Starting configuration
in ordinal MDS, 425
Stationary point, 174
STATISTICA, 560
Straight line, 39
Strain, 263
Stress
constrained, 230
definition, 171
evaluating, 47
expected, 49
for Minkowski distances, 369
for random data, 48
formula 1, 42, 251
formula 2, 251, 302, 324
implicit normalization, 42
normalized, 247, 327
penalized, 327
per point, 44, 45, 235
raw, 42, *171*, 250
Structuple, 89
Subadditivity, 376, 381
Subject space, 460
SVD, *see* Singular value decomposition
Symmetrized data, 71
SYSTAT, 369, 375, 560

Tangent, 173
Target distances, 199
Three-way data, 58, 449
Three-way models
dilation, 482
generalized Euclidean, 484
identity, 482
reduced rank, 484
weighted Euclidean, 482
Thurstone case-V model, 51

Ties
in data, 211
primary approach, 40, 203, 204, 211, 212
secondary approach, 40, 211
Torgerson scaling, 261
Torgerson–Gower scaling, 261
Trace function, 143, 144, *176*, 570
Trace-eigenvalue theorem, 420
Transformation, 23, 170
admissible, 23, 29
alias interpretation, 23, 161
alibi interpretation, 23, 161
continuous, 362
inadmissible, 23
invertible, 362
isometric, 23, 29
isotonic, 28
plot, 202
power, 221
similarity, 23
smooth, 362
spline, 214, 326
split-by-row, 300
Translation, 23
Transpose of a matrix, 138
Triangle inequality, 33, 143, 376
TRISOSCAL, 554
Tunneling method, 283, 369
Two-dimensional triple, 384
Two-mode data, 58, 294

Unconditional unfolding, 299
Under-compression, 54
Unfolding, 153, 489, 491
analyzing asymmetry, 503
conditional, 300
external, 335
ideal-point model, 295
internal, 335
ordinal-interval approach, 318
ordinal-ratio approach, 317
unconditional, 299
vector model, 336
vector-ideal point model (VIPSCAL), 341
weighted, 342
Unidimensional scaling, 278
circular, 240
dynamic programming strategy, 280
local pairwise interchange strategy (LOPI), 280
Unidimensional triple, 384
Unit ball for Minkowski metrics, 370

Variation coefficient, 327
Varimax rotation, 105, 161
v-data, 393, 395, 400
and dissimilarities, 398
built-in restrictions, 407
Vector, 138
Abelian space, 414
and points, 390
configuration, 391–393
length of, 390
linear dependency, 152
norm, 143
normal, 143
orthogonal, 142
properties of norms, 143
space, 390, 414
unit, 143
weights, 466
Vector model, 393
appropriateness of, 395, 397
for unfolding, 336
interpretation of properties, 400
Vector-ideal point model for unfolding (VIPSCAL), 341
VIPSCAL (vector-ideal point model for unfolding), 341

Ways of data, 58
Weakly constrained MDS, 237
Weber–Fechner law, 374
Weighted Euclidean distance, 458, 474, 482
Weighted unfolding, 342
WOMBATS, 554