Lecture Notes in Computer Science 2859

Edited by G. Goos, J. Hartmanis, and J. van Leeuwen

Springer
Berlin
Heidelberg
New York
Hong Kong
London
Milan
Paris
Tokyo

Bruno Apolloni   Maria Marinaro
Roberto Tagliaferri (Eds.)

Neural Nets

14th Italian Workshop on Neural Nets, WIRN VIETRI 2003
Vietri sul Mare, Italy, June 4-7, 2003
Revised Papers

Springer

Volume Editors

Bruno Apolloni
Università di Milano, Dipartimento di Scienze dell'Informazione
Via Comelico 39/41, 20135 Milano, Italy
E-mail: apolloni@dsi.unimi.it

Maria Marinaro
Università di Salerno, Dipartimento di Fisica
Via S. Allende, 84081 Baronissi (Salerno), Italy
E-mail: iiass.vietri@tin.it

Roberto Tagliaferri
Università di Salerno, Dipartimento di Matematica ed Informatica
Via S. Allende, 84081 Baronissi (Salerno), Italy
E-mail: rtagliaferri@unisa.it

Cataloging-in-Publication Data applied for

A catalog record for this book is available from the Library of Congress.

Bibliographic information published by Die Deutsche Bibliothek
Die Deutsche Bibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data is available in the Internet at <http://dnb.ddb.de>.

CR Subject Classification (1998): F.1.1, I.2.6, B.6.1, C.1, J.3, I.4, G.3, H.2.8

ISSN 0302-9743
ISBN 3-540-20227-7 Springer-Verlag Berlin Heidelberg New York

Springer-Verlag Berlin Heidelberg New York
a member of BertelsmannSpringer Science+Business Media GmbH

http://www.springer.de

Typesetting: Camera-ready by author, data conversion by Olgun Computergrafik
Printed on acid-free paper SPIN: 10958469 06/3142 5 4 3 2 1 0

Preface

The proceedings of the 14th Italian Workshop on Neural Nets WIRN VIETRI 2003 are collected in this volume. The workshop, held in Vietri sul Mare (SA) June 4–7, 2003 was jointly organized by the International Institute for Advanced Scientific Studies "Eduardo R. Caianiello" (IIASS) and the Società Italiana Reti Neuroniche (SIREN). The volume covers up-to-date topics on neural nets and related fields. It contains invited review papers and selected original contributions presented in either oral or poster sessions by both Italian and non-Italian researchers. The contributions have been assembled, for reading convenience, into four sections: Models, Architectures and Algorithms, Image and Signal Processing, and Applications, plus two special sessions. The latter gives a fresh perspective in the fields of "Bioinformatics and Statistics" and "Formats Knowledge: Words, Images, Narratives", sharing the technical languages of the involved disciplines. The editors would like to thank the invited speakers and all the contributors whose highly qualified papers contributed towards the success of the workshop. Finally, special thanks go to the referees for their accurate work.

June 2003

Bruno Apolloni
Maria Marinaro
Roberto Tagliaferri

Organizing Scientific Committee

B. Apolloni (Univ. Milano), A. Bertoni (Univ. Milano), N. A. Borghese (Univ. Milano), D. D. Caviglia (Univ. Genova), P. Campadelli (Univ. Milano), A. Chella (Univ. Palermo), A. Colla (ELSAG Genova), A. Esposito (I.I.A.S.S.), F.M. Frattale Mascioli (Univ. Roma), C. Furlanello (ITC-IRST Trento), S. Giove (Univ. Venezia), M. Gori (Univ. Siena), M. Marinaro (Univ. Salerno), F. Masulli (Univ. Pisa), C. Morabito (Univ. Reggio Calabria), P. Morasso (Univ. Genova), G. Orlandi (Univ. Roma), T. Parisini (Univ. Trieste), E. Pasero (Politecnico Torino), A. Petrosino (CNR Napoli), V. Piuri (Politecnico Milano), R. Serra (CRA Montecatini Ravenna), F. Sorbello (Univ. Palermo), A. Sperduti (Univ. Padova), R. Tagliaferri (Univ. Salerno)

Referees

Apolloni, B.
Bertoni, A.
Borghese, N.A.
Burattini, E.
Burrascano, P.
Campadelli, P.
Caviglia, D.D.
Chella, A.
Ciaramella, A.
Colla, A.M.
Di Claudio, E.
Eleuteri, A.
Esposito, A.
Fiore, S.
Frattale Mascioli, F.
Frixione, M.
Furlanello, C.
Giove, S.
Haus, G.
Marinaro, M.
Masulli, F.
Morabito, F.C.
Morasso, P.
Palmieri, F.
Parisi, R.
Parisini, T.
Pasero, E.
Petrosino, A.
Piuri, V.
Raiconi, G.
Serra, R.
Sperduti, A.
Staiano, A.
Tagliaferri, R.
Valentini, G.

Sponsoring Institutions

International Institute for Advanced Scientific Studies (IIASS) “E.R. Caianiello”
Dip. di Fisica “E.R. Caianiello”, University of Salerno
Dip. di Matematica ed Informatica, University of Salerno
Dip. di Scienze dell’Informazione, University of Milano
Società Italiana Reti Neuroniche (SIREN)
IEEE Neural Network Society, Italian Chapter
INNS, SIG Italy
Istituto Italiano per gli Studi Filosofici, Napoli
Provincia di Salerno

Acknowledgments

Special thanks go to Angelo Ciaramella and Antonino Staiano for editing and collecting the papers, and to Simone Bassis and Dario Malchiodi for their web management. Furthermore thanks go to Tina Nappi and Ornella De Pasquale for their secretarial work and to Michele Donnarumma for his technical work.

Table of Contents

I Eduardo R. Caianiello Lecture

II Models

III Architectures and Algorithms

IV Image and Signal Processing

V Applications

VI Special Session on "Bioinformatics and Statistics" Chaired by Francesco Masulli

VII Special Session on "Formats of Knowledge: words, images, narratives" Chaired by Maria Rita Ciceri

Nonparametric Hidden Markov Models: Principles and Applications to Speech Recognition

Edmondo Trentin

Dipartimento di Ingegneria dell'Informazione
Università di Siena, Via Roma, 56 - 53100 Siena, Italy
trentin@dii.unisi.it

Abstract. Continuous-density hidden Markov models (HMM) are a popular approach to the problem of modeling sequential data, e.g. in automatic speech recognition (ASR), off-line handwritten text recognition, and bioinformatics. HMMs rely on strong assumptions on their statistical properties, e.g. the arbitrary parametric assumption on the form of the emission probability density functions (pdfs). This chapter proposes a nonparametric HMM based on connectionist estimates of the emission pdfs, featuring a global gradient-ascent training algorithm over the maximum-likelihood criterion. Robustness to noise may be further increased relying on a soft parameter grouping technique, namely the introduction of adaptive amplitudes of activation functions. Applications to ASR tasks are presented and analyzed, evaluating the behavior of the proposed paradigm and allowing for a comparison with standard HMMs with Gaussian mixtures, as well as with other state-of-the-art neural net/HMM hybrids.

1 Introduction

The problem of sequence modeling (e.g. acoustic modeling in Automatic Speech Recognition (ASR), off-line handwritten text recognition, and several tasks in bioinformatics) is mostly approached relying on continuous-density hidden Markov models (HMMs) [8]. Yet effective to a significant extent, HMMs are inherently limited [13]. In particular, the assumption of a given parametric form for the probability density functions (pdfs) that represent the *emission* probabilities associated with HMM states is arbitrary and constraining. In addition, standard HMMs do not feature a discriminative training procedure [5], and they lack of any regularization techniques [4] that may improve their generalization capabilities. In this respect, the use of artificial neural networks (ANNs) appears promising. Unfortunately ANNs historically failed as a general framework for ASR [13] and for other long-sequence modeling applications, due to their limited ability to model long-term time dependencies, even when recurrent architectures are considered [3]. The failure led to the idea of combining HMMs and ANNs within *hybrid* ANN/HMM systems. A variety of hybrid paradigms

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 3–21, 2003.

were proposed in the literature [13]. This chapter introduces a novel approach to the combination of HMM and ANN, related to some extent to Bourlard and Morgans's architecture [5], as well as to Bengio's optimization scheme [2]. The proposed model [12] is a HMM in which the emission probabilities are estimated *via* a feed-forward ANN, i.e. the parametric (Gaussian) assumption is dropped and a (universal) nonparametric model is used. For this reason, in the following we use the term "nonparametric HMM".

This nonparametric HMM relies on an HMM topology, including standard initial probabilities and transition probabilities a_{ij} for each pair of states i, j. An output unit of the ANN holds for each of the states in the HMM, with the understanding that i-th output value $o_i(t)$ at time t represents the emission probability $b_{i,t}$ for the corresponding (i-th) state, evaluated over current observation $\mathbf{y}_t$. Recognition is accomplished applying the usual Viterbi algorithm [8], while a novel maximum-likelihood (ML) global training technique was introduced. The algorithm, reviewed in Section 2, relies on gradient-ascent to maximize the likelihood $L = P(Y \mid \mathcal{M})$ of the observation sequence Y given the model $\mathcal{M}$ under consideration. The other parameters of the underlying HMM, namely initial and transition probabilities, are estimated via the Baum-Welch algorithm [8]. Differences w.r.t. previous ANN/HMM hybrids were pointed out in [10,12].

A major question, addressed in Section 2.1, concerns the actual behavior of ANNs in modeling pdfs. Indeed, latter ones do satisfy the "integral equal to 1" constraint. Although ANNs are "universal approximators", such a constraint is not, in general, satisfied. Estimates for emission likelihoods tend to grow in an unbound fashion in order to increase the overall likelihood of the sample sequences at training time. This phenomenon yields degenerated models. We call it the "divergence problem". Specific techniques are proposed to tackle the problem, at both the architectural and algorithmic levels.

Finally, another relevant issue concerns noise-tolerance of the overall model. A regularization technique, based on a ML parameter grouping algorithm, is then explicitly given (Section 2.2) to increase the *generalization* capability of the model and, in turn, its noise-robustness. As a matter of fact, ANNs learning theory draws a relationship between "learning with noise" and the generalization capabilities of the learning machine [4]. Application of robust training techniques (e.g, regularization) during the learning process on clean (non-noisy) data is aimed at improving generalization, reducing overfitting, resulting in improved performance on noisy test data.

In [14] we proposed such an approach relying on the introduction of a parameter λ within the training scheme, assuming activation functions in the form $y = \lambda f(x)$. The gradient-ascent training algorithm to learn λ from the data according to the training criterion for the ANN/HMM hybrid, i.e. the likelihood L, is reviewed in Section 2.2. A "soft" parameter grouping [4] technique for the ANN/HMM is obtained, where all connection weights that start from a given unit (or set of units) subject to λ are *grouped* together. The *range* of their values is conditioned by λ, and the latter is learned from the data according to contributions from *all* the weights in the group. As a consequence, different search paths within the weight space are induced. In [9], a theoretical analysis is carried out to investigate the rationale behind the improvement yielded by the approach. The algorithm is applied simultaneously with the ML learning rule for the connection weights.

An experimental analysis of the proposed techniques in ASR tasks, along with a comparison w.r.t. continuous-density HMMs and Bourlard & Morgan's approach, is carried out in Section 3.

2 Connectionist Nonparametric Hidden Markov Models

In this Section, a sketch of the fundamental calculations for ML training of the nonparametric HMM are given. Along the line of standard Gaussian-density HMM parameter-estimation algorithms, the global criterion function C to be maximized by the model during training is the *likelihood* L of the observations given the model: $C = L$, where:

$$L = \sum_{i \in \mathcal{F}} \alpha_{i,T} \tag{1}$$

and the α's are the usual *forward* probabilities. The sum is extended to the set $\mathcal{F}$ of all possible *final* states within the HMM [1] that corresponds to the current data sequence (e.g., in ASR it is the HMM that represents the correct phonetic transcription of the acoustics under consideration). Such an HMM is supposed to involve Q states, and T is the length of the current observation sequence $Y = \mathbf{y}_1, \ldots, \mathbf{y}_T$.
Assuming that the ANN represents the emission pdfs in a proper manner, and given a generic weight w of the ANN, hill-climbing gradient-ascent over C prescribes a learning rule of the kind:

$$\Delta w = \eta \frac{\partial L}{\partial w} \tag{2}$$

where η is the *learning rate*. Let us observe (after [1]) that the following property can be easily shown to hold true:

$$\frac{\partial \alpha_{i,t}}{\partial b_{i,t}} = \frac{\alpha_{i,t}}{b_{i,t}}. \tag{3}$$

According to [6,1], the following theorem can be proved to hold true: $\frac{\partial L}{\partial \alpha_{i,t}} = \beta_{i,t}$, for each $i = 1, \ldots, Q$ and for each $t = 1, \ldots, T$. Given the theorem and Equation (3), repeatedly applying the chain rule we can expand $\frac{\partial L}{\partial w}$ by writing:

$$\begin{aligned} \frac{\partial L}{\partial w} &= \sum_i \sum_t \frac{\partial L}{\partial b_{i,t}} \frac{\partial b_{i,t}}{\partial w} \\ &= \sum_i \sum_t \beta_{i,t} \frac{\alpha_{i,t}}{b_{i,t}} \frac{\partial b_{i,t}}{\partial w}. \end{aligned} \tag{4}$$

where the sums are extended over all states $i = 1, \ldots, Q$ of the HMM corresponding to the correct transcription of the training utterance under consideration, and to all $t = 1, \ldots, T$, respectively. Let us consider a multilayer Perceptron (MLP), the j-th output of which, computed over t-th input observation $\mathbf{y}_t$, is interpreted as a non-parametric estimate of the emission probability $b_{j,t}$ associated with j-th state of the HMM at time t. An activation function $f_j(x_j(t))$ is associated with each unit j of the MLP, where $x_j(t)$ denotes input to the unit itself at time t. The corresponding output $o_j(t)$ is given by $o_j(t) = f_j(x_j(t))$. This ANN is assumed to have L layers $\mathcal{L}_0, \mathcal{L}_1, \ldots, \mathcal{L}_L$, where $\mathcal{L}_0$ is the input layer, and $\mathcal{L}_L$ is the output layer. For notational convenience we write $i \in \mathcal{L}_k$ to denote the index of i-th unit in layer $\mathcal{L}_k$.

Given a generic weight w_{jk} between k-th unit in layer $\mathcal{L}_{l-1}$ and j-th unit in layer $\mathcal{L}_l$, and defining the quantity

$$\delta_j(i,t) = \begin{cases} f_j'(x_j(t)) & \text{if } l = L, i = j \\ 0 & \text{if } l = L, i \neq j \\ f_j'(x_j(t)) \sum_{n \in \mathcal{L}_{l+1}} w_{ni} \delta_n(j,t) & \text{otherwise} \end{cases} \tag{5}$$

for each $i \in \mathcal{L}_n$, it is possible to show that [12]:

$$\frac{\partial b_{i,t}}{\partial w_{jk}} = \delta_j(i,t) o_k(t). \tag{6}$$

Using the above calculations to expand Equation (4) and substituting it into Equation (2), the latter can now be restated in the form of a *learning rule* for weight w_{jk}, by writing:

$$\Delta w_{jk} = \eta \sum_{i=1}^{Q} \sum_{t=1}^{T} \beta_{i,t} \frac{\alpha_{i,t}}{b_{i,t}} \delta_j(i,t) o_k(t) \tag{7}$$

where the term $\delta_j(i,t)$ is computed via Equation (5). In the following, Equation (7) will be referred to as the *Bare Maximum-Likelihood* (BML) on-line learning rule in the linear domain. Actually, *batch* versions (i.e., gradient computed over all sequences in the training set) as well as *logarithmic* versions (computations accomplished in the logarithmic domain, in order to avoid numerical stability problems over long sequences of probabilistic quantities) of the learning rules are presented in [10], and they are experimentally compared in Section 3.

2.1 Do ANNs Estimate Probability Density Functions?

The learning rule developed so far is expected to work given the fact that the ANN can be actually treated as a non-parametric estimate of the emission probabilities of the underlying HMM. Unfortunately, in order to satisfy the latter condition, the ANN itself should be a (general and non-parametric) model of a pdf, i.e. the integral of the function computed by the ANN over its whole domain (the feature space) should equal 1. While ANNs, being "universal approximators", *may* reasonably approximate any continuous and bound pdfs as close as desired, they completely lack of any constraints ensuring that, at any given time during training, they actually *are* pdfs. In practice, a direct application of the above training scheme may not lead to a correct solution, since the training criterion (ML) simply encourages the ANN to yield high output values, i.e. high emission probabilities, over all input patterns. In so doing, the overall likelihood of the training input sequences diverges toward infinity, since the ANNs do not estimate pdfs, but they simply develop larger values for their connection weights. The resulting model turns out to be meaningless (and useless). We refer to such a phenomenon as the "divergence problem". A strictly related scenario was pointed out by Bengio [1], where the ANNs for feature extraction might simply learn to concentrate all the inputs closely around the mean vectors of the Gaussian emission probabilities, the latter quickly becoming Dirac's deltas. While using Gaussian mixtures provided Bengio with the satisfaction of the "integral-equal-to-one" condition, in the present setup the problem is even harder.

Bengio states that, in practice, the problem is not crucial, since the ANN usually reaches "more interesting" extremes of the criterion function [1]. Similar reasoning could lead us to the expectation that a direct application of the algorithm described above could provide us with interesting solutions, maybe at the expense of a few training trials with different connectionist architectures and different starting points in the weight space. Although some experiments reported in Section 3 do confirm this expectation to some extent, the problem is definitely relevant, and further theoretical developments are sought in order to obtain a more reliable and mathematically motivated training scheme.

The present Section is devoted to the development of alternative solutions to the problem, *via* modification (or completion) of the proposed algorithm. Since explicit introduction of the probabilistic constraint, i.e. "integral equals 1", in the training criterion does not lead to any feasible analytical formulation, the following alternatives are considered herein:

1. introduction of architectural constraints on the ANN topology in order to ensure a suitable probability density estimation through a direct application of the above learning rules. As a first instance, Radial Basis Functions (RBF) networks may be interpreted as mixtures of Gaussian components. In particular, if the values of the connection weights that link the radial basis activation functions of the hidden layer with the linear output units sum to one, the model actually satisfies the requirements of a pdf. For this reason, RBFs may look promising in the present framework. In fact, there are two major drawbacks that severely limit the applicability of RBFs herein: (i) since they realize bare mixture of Gaussians, no significant gain in modeling capabilities may be expected over the standard (parametric) HMM; (ii) whenever trained via the gradient method, hidden-to-output weights may diverge in order to increase the overall likelihood (as discussed above) since no explicit constraint ensures that, for each output unit, they sum to one. At least, a normalization to one should be forced after each training iteration. It is evident that such a heuristic, while preserving the pdf requirements, would

have an impact on the training scheme. In spite of these problems, the RBF remains interesting - especially in a validation perspective - since it allows for an immediate numerical comparison with the standard HMM. For instance, the RBF may be initialized with the same parameters of a pre-trained HMM, and further trained using gradient-ascent. Section 3.1 ("Round 1") will take advantage of a similar initialization procedure over a slightly modified ANN architecture, namely a kind of log-RBF where the logarithms of Gaussian activation functions are used in the hidden layer.
An alternative is represented by a MLP having Gaussian output units. Presence of the latter ones could ensure the satisfaction of the pdf constraints, tackling the "divergence problem". Although such an architecture may appear unpromising, due to the fact that it basically replaces mixtures of Gaussians (as in standard HMMs) with individual Gaussians (one for each output unit, that is one for each state of the underlying HMM), it should be noticed that the MLP performs a non-linear transformation of the input features that is aimed at maximizing the likelihood. This means that the output Gaussians are defined over a new, transformed space which is expected to better suit the ML framework for the recognition problem at hand. This architecture, along with the BML algorithm, provides us with a scheme that strictly resembles Bengio's paradigm, i.e. the ANN extracts "optimal" features to be modeled with a Gaussian-based HMM. The "divergence problem" may thus rise again in the same form as in Bengio's case (as outlined in the previous Section): indeed, one possible way for the ANN to maximize the likelihood is simply to project all the input patterns onto a common internal representation - developed by the hidden layers - and then to map such a representation to a common centroid, shared among all output units, after driving all of the Gaussians toward the same mean vector;

2. application of a probabilistic constraint on the network weights. The criterion is no longer ML, but a joint probability that combines the likelihood of the observations *and* the probability of the ANN weights (considered as they were random variables) given the model. This is similar to some well-known *regularization* techniques [4], where "simpler" solutions, i.e. with small weight values, are privileged over complex solutions, e.g. with huge weights. A Gaussian distribution with zero mean is an effective choice for the ANN weights distribution. Under this assumption, a parameter σ^{-2} (the inverse of the variance of the Gaussian pdf) controls the weights distribution, and its impact on the training process is analyzed in Section 3.1. We refer to the present training scheme as the *Soft Weight Sharing* Maximum-Likelihood (SWS-ML) learning rule, as opposed to the basic BML. Batch, as well as logarithmic versions of SWS-ML are introduced in [10];
3. factorization of the emission probability using Bayes' theorem. Some terms of the factorization can be estimated separately from the training set, before ANN training. The ANN is then used to estimate the posterior probability of a HMM state given the input observation. This means that the emission probability is no longer expected to be the output of the ANN, but the latter is combined with other terms to obtain the emission probability itself.

Being the ANN output interpreted as a probability, the constraints to be satisfied are: (i) that each output unit is forced in the range $[0, 1]$, and (ii) that the sum over all output units be exactly 1. Such constraints can be explicitly introduced in the training scheme. The resulting learning rule is called the *Bayes* learning rule, featuring on-line and batch versions [10]. No logarithmic version of the Bayes learning rule is allowed, since the ANN outputs are expected to be probabilities, i.e. in the linear domain;

4. adoption of a "discriminative" training criterion, namely *Maximum A Posteriori*, instead of the bare Maximum Likelihood, in order to force probabilities along "incorrect" paths of the HMM to be discouraged. We formally define the MAP criterion $\mathcal{C}_{MAP}$ as follows:

$$\mathcal{C}_{MAP} = P(\mathcal{M} \mid Y) = \frac{P(Y \mid \mathcal{M})P(\mathcal{M})}{P(Y)} \tag{8}$$

where Y is the observation sequence and $\mathcal{M}$ is the specific model under consideration. Bayes' theorem was applied in order to obtain the right-end side of Equation (8): $P(\mathcal{M})$ is the *prior* probability of the model, and $P(Y)$ is the overall likelihood of the observation sequence Y, that does not depend on the specific choice for the model $\mathcal{M}$. $P(\mathcal{M})$ is independent of the ANN outputs (e.g., it depends on the *language model* only), and it can be computed separately. In particular, in the on-line learning case a suitable training algorithm is required to maximize the following, equivalent criterion:

$$\mathcal{C}'_{MAP} = \frac{P(Y \mid \mathcal{M})}{P(Y)}, \tag{9}$$

a formulation that emphasizes the strict relationship with the *Maximum Mutual Information* (MMI) criterion. A gradient-based learning rule for a generic connection weight w can be obtained by taking the partial derivative of $\mathcal{C}'_{MAP}$ with respect to w as follows:

$$\Delta w = \frac{\eta}{P(Y)} \left\{ \frac{\partial P(Y \mid \mathcal{M})}{\partial w} - \frac{P(Y \mid \mathcal{M})}{P(Y)} \frac{\partial P(Y)}{\partial w} \right\}. \tag{10}$$

The quantity $P(Y \mid \mathcal{M})$ and its derivative w.r.t. w are computed according to the calculations introduced above, since it is the usual likelihood of the observations given the "correct" model $\mathcal{M}$, i.e. the model *constrained* by the (known) transcription of the current observation sequence Y (e.g., in ASR it is the phonetic transcription of current training sequence). The quantity $P(Y)$ and its derivative w.r.t. w are computed by applying the same calculations to an *unconstrained*, or *recognition* model, as explained in [11]. The latter is the same model used for *recognition* with the Viterbi algorithm, where no prior knowledge on the correct transcription of current sequence is given. We refer to the present training scheme as the MAP learning rule (featuring on-line, batch and logarithmic versions [10]; see Section 3.1 for an experimental comparison).

We stress the fact that, in practice, the ANN is not necessarily required to realize a pdf, but *divergence* of its output values should rather be avoided. Some of the solutions that are outlined in the following are thus aimed at the definition of architectures/training algorithms that prevent the ANN to develop unbounded outputs while increasing the likelihood of the observations.

2.2 Increasing Noise-Tolerance via Soft Parameter Grouping

The parameter-grouping scheme is obtained by introducing *trainable* parameters $\lambda_{i,\ell}$ for each unit i in each layer $\mathcal{L}_\ell$, and by considering activation functions in the form

$$f_{i,\ell}(x_{i,\ell}(t)) = \lambda_{i,\ell}\tilde{f}_{i,\ell}(x_{i,\ell}(t)) \tag{11}$$

where dependence of the different quantities on the specific layer $\mathcal{L}_\ell$ was explicitly stated for notational convenience in the calculations. In the following, the symbol $\tilde{f}_{i,\ell}(x_{i,\ell}(t))$ will refer to a function of $x_{i,\ell}(t)$ which does not explicitly depend on $\lambda_{i,\ell}$.

Considering criterion (1), and relying on gradient ascent as in Eq. (4), we have:

$$\frac{\partial C}{\partial \lambda_{\iota,\ell}} = \sum_i \sum_t \beta_{i,t} \frac{\alpha_{i,t}}{b_{i,t}} \frac{\partial b_{i,t}}{\partial \lambda_{\iota,\ell}}. \tag{12}$$

Again, being $\frac{\partial b_{i,t}}{\partial \lambda_{\iota,\ell}} = \frac{\partial o_{i,L}(t)}{\partial \lambda_{\iota,\ell}}$, by defining the quantity $\delta_{\iota,\ell}(i,t)$ as

$$\begin{cases} 1 & \text{if } \ell = L, \iota = i \\ 0 & \text{if } \ell = L, \iota \neq i \\ \displaystyle\sum_{j \in \mathcal{L}_{\ell+1}} w_{j,\iota,\ell+1}\delta_{j,\ell+1}(i,t) f'_{j,\ell+1}(x_{j,\ell+1}(t)) & \text{otherwise} \end{cases} \tag{13}$$

it is possible to prove by induction [10] that:

$$\frac{\partial o_{i,\ell}(t)}{\partial \lambda_{\iota,\ell}} = \delta_{\iota,\ell}(i,t)\tilde{f}_{\iota,\ell}(x_{\iota,\ell}(t)). \tag{14}$$

In summary, Eq. (14) can be substituted into Eq. (12), obtaining an on-line learning rule in the form:

$$\Delta\lambda_{\iota,\ell} = \eta \sum_{i=1}^{Q} \sum_{t=1}^{T} \beta_{i,t} \frac{\alpha_{i,t}}{b_{i,t}} \delta_{\iota,\ell}(i,t)\tilde{f}_{\iota,\ell}(x_{\iota,\ell}(t)) \tag{15}$$

for each layer $\mathcal{L}_\ell = \mathcal{L}_1, \ldots, \mathcal{L}_L$ in the ANN, and for each unit ι in $\mathcal{L}_\ell$.

3 Applications to Automatic Speech Recognition

Since processing the whole speech databases for speaker independent, continuous speech recognition is computationally expensive, and evaluating the behavior of the proposed training schemes over different architectures is difficult when large amounts of data are used, we first introduce a simple case study (Section 3.1) drawn from the *SPK* corpus[1]. SPK was collected in laboratory conditions at ITC-irst (Trento, Italy). It consists of 1000 utterances of connected Italian digit strings having length 8 (for a total of 8000 words), acquired over 40 different speakers (21 male and 19 female). Spectral analysis of the speech signals (acquired at a sampling rate of 16kHz) was accomplished over 20ms Hamming windows having an overlap of 10ms, in order to extract 8 *Mel Frequency Scaled Cepstral Coefficients* (MFSCCs) [7] and the signal log-energy as acoustic features. The latter were normalized in order to have input values distributed in a uniform manner over the $[0, 1]$ interval, before feeding them into the connectionist models. Results obtained herein highlight the properties of the algorithms and provide us with significant cues on the most promising training schemes, parameters and architectures to be applied to the more complex ASR tasks. Latter ones are summarized, from literature, in Section 3.2.

3.1 Experimental Analysis: Reduced-Scale Task

The reduced-scale setup was built by choosing one individual speaker at random from SPK (speaker code: *gila0*), i.e. a *speaker dependent* recognizer is developed. A subset of the original feature space was considered, precisely a 3-dimensional space composed of the first two MFSCCs and the corresponding log-energy. The ASR task was the recognition of *isolated words*: 120 acoustic signals, corresponding to 12 different utterances of each Italian digit were used for training, while 80 isolated signals (8 instances for each digit) constituted the test set. Three different approaches to the problem are presented in the following sections, each corresponding to different ANN architectures and particular initialization schemes.

Round 1: ANN Initialized by Log-Gaussian Kernels. A Gaussian-based HMM was first trained and evaluated, providing a comparison baseline for the following algorithms. For ANN/HMM *initialization* purposes, the HMM was kept as simple as possible (in the present scenario, attention is paid to the behavior and relative performance of different models, not to the absolute suitability of the recognition rate to real-world applications). The topology was a set of 10 word (digit) models, each based on a 3-state left-to-right HMM, plus a 1-state model for the background noise (silence model), for a total of 31 states. Emission probabilities were modeled via individual Gaussian pdfs $G_i(\mathbf{x})$ associated with each state i of the HMM, where $G_i(\mathbf{x}) = \prod_{j=1}^{3} g_{ij}(x_j)$ and $g_{ij}(x_j)$ is a univariate Normal, defined over j-th dimension of the feature space. All such pdfs shared a common value θ for their variances θ_{ij}. The *segmental k-means* algorithm [8] was applied in order to initialize the HMM. Training was accomplished via the Baum-Welch algorithm, and the Viterbi decoding technique was applied over the test set.

[1] SPK is available from the European Language Resources Association (ELRA).

The connectionist model was then defined as a 2-layer feed-forward net, with 93 units in the hidden layer (one for each univariate Gaussian $g_{ij}(.)$ in the HMM, $i = 1, \dots, 31$, $j = 1, \dots, 3$), having the following form for the corresponding activation functions $f_{ij}(x)$ over a generic input x:

$$f_{ij}(x) = -\frac{1}{2}\left(\frac{x - \mu_{ij}}{\theta}\right)^2 = log(g_{ij}(x)) + c \tag{16}$$

where c is a constant (namely the logarithm of the usual normalization constant for Normal distributions having standard deviation θ). Input-to-hidden weights $w_{ij,k}$, connecting k-th input unit with ij-th hidden unit, were defined as

$$w_{ij,k} = \begin{cases} 1 \text{ if } j = k \\ 0 \text{ otherwise} \end{cases} \tag{17}$$

for each $k = 1, \dots, 3$. Hidden-to-output weights $w_{l,ij}$, connecting ij-th hidden unit with l-th output unit ($l = 1, \dots, 31$), were defined as

$$w_{l,ij} = \begin{cases} 1 \text{ if } i = l \\ 0 \text{ otherwise} \end{cases} \tag{18}$$

along with linear output units, one for each state l of the HMM, thus realizing the function

$$o_l = \sum_{j=1}^{3} log(g_{lj}(x_j)) = log(G_l(\mathbf{x})). \tag{19}$$

The resulting architecture is a kind of *log-RBF* network, that yields exactly the same emission probabilities as the standard HMM involved in its construction process. It is suitable for those algorithms of Section 2 which are defined in the logarithmic domain; it cannot be used for evaluating the *Bayes* learning rule, which assumes ANN linear outputs in the $[0, 1]$ interval.

It is worth mentioning that, although in principle this hybrid model is actually an HMM with Gaussian emissions, during training the values of the weights (and biases) are modified. This means that: (i) the log-Gaussian kernels of the hidden layer are progressively fed with linear combinations of the input features, learning to exploit correlations between the acoustic features to some extent; (ii) the output units learn to yield values that are no longer based on the evaluation of a single Gaussian *pdf* associated with the corresponding state; i.e., some Gaussian pdfs are shared among distinct states to better fit the corresponding emission distributions. These aspects are the rationale behind the improvement obtained in the experiments. Results yielded by the different training schemes are summarized in Table 1.

The BML learning rule was first applied. *Batch* learning turned out to be slightly advantageous, since in the on-line case all shorter training sequences (corresponding to utterances of shorter words) yielded higher likelihood values (being the latter the product of probabilities along shorter paths in the trellis [8]), thus inducing heavier changes in ANN weights that biased the training process

Table 1. Round 1, ANN with log-Gaussian kernels: Word recognition rate (WRR) on test set and final log-likelihood of the training acoustic observations given the model, yielded by the proposed algorithms applied to the SPK isolated digits, speaker-dependent (*gila0*) problem. 3-dim acoustic space: 2 MFSCCs and signal log-energy

Model/algorithm	WRR (%)	Final log-likelihood
HMM	70.00	-7.5054e+03
BML	68.75	-7.0964e+03
SWS-ML	71.25	-7.4166e+03
MAP	72.50	-7.7269e+03

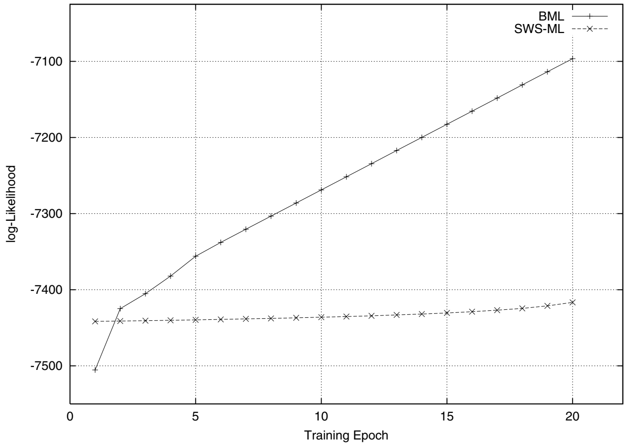


Fig. 1. Round 1: learning curves (log-likelihood of the acoustic observation sequences in the training set) as a function of the number of training epochs, for the BML and SWS-ML algorithms

toward those words. A kind of "normalization" with respect to the length of each training sequence tackles the problem. Figure 1 shows the evolution of the overall log-likelihood of the training set given the model for the BML (upper curve) as a function of the number of training epochs. As pointed out in Section 2.1, the divergence problem may arise in the BML case: in fact, the word recognition rate is decreased with respect to the HMM at the end of training.

The SWS-ML scheme allowed to overcome the problem, as expected. The evolution of the corresponding log-likelihood during the training process is represented by the lower curve in Figure 1. The final likelihood is lower than in the BML case, although increased with respect to the HMM. In particular, the likelihood increases very slowly (it may be even reduced at certain iterations). This is due to the fact that when the connection weights tend to become too big, the SWS-ML learning rule forces a reduction in their size (increasing their probability), even at the expense of the likelihood of the observations, avoiding divergence.

Finally, the MAP variant of the algorithm improved recognition performance over the HMM, as well as over the SWS-ML scheme, by increasing quantity (9), while preventing the bare likelihood of the acoustic observations from divergence. The likelihood (of the constrained model) may decrease at certain training epochs, whenever a broader reduction of the likelihood of the unconstrained model ensures an increment in the likelihood ratio. Figure 2 shows the log-likelihood of the training sequences given the constrained model (lower curve), and given the unconstrained model (upper curve), as functions of the number of training epochs. Evolution of the monotonically-increasing likelihood ratios, computed at each epoch over the training set, is plotted in Figure 3 (in the logarithmic domain).

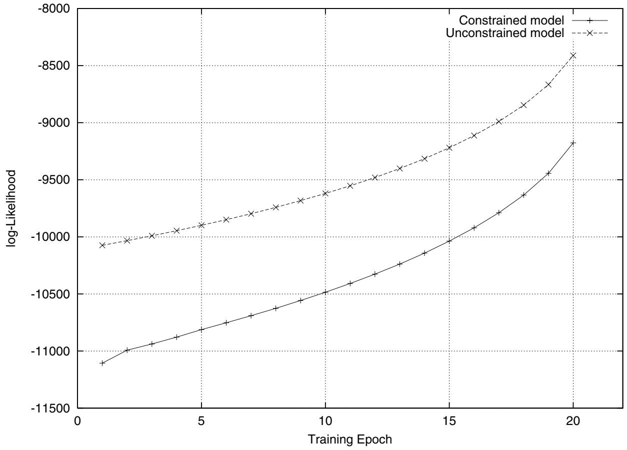


Fig. 2. Round 1: log-likelihood of the acoustic observation sequences in the training set, as a function of the number of training epochs, for the MAP algorithm. Upper curve: unconstrained model; lower curve: constrained model

Round 2: Generic MLP, Logarithmic Domain. In this section we apply the algorithms to the generic 2-layer MLP architecture with hidden sigmoid activation functions and linear output units. The topology is inherited from the previous experiments, i.e. 93 hidden units along with 31 outputs are used. Results are reported in Table 2. The first row of the table reports the baseline performance obtained with a standard HMM with Gaussian emission probabilities (a single Gaussian for each state) having variable means and covariances. A second HMM with mixtures of 8 Gaussian components was then trained and evaluated, for initialization and comparison purposes (second row of the table). Initialization of the MLP was accomplished according to the following algorithm:

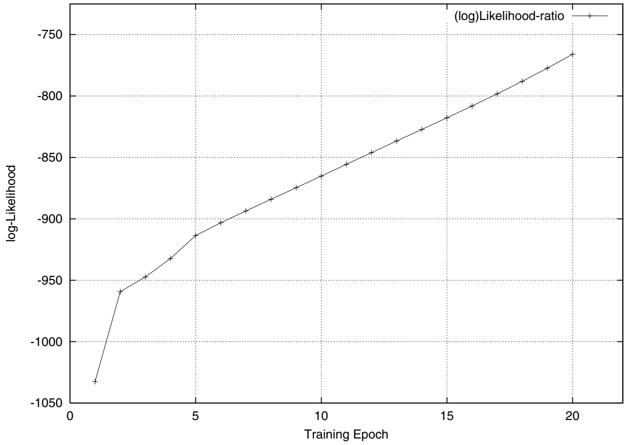


Fig. 3. Round 1: (log)likelihood ratios between the constrained and the unconstrained models, evaluated over the acoustic observation sequences in the training set as a function of the number of training epochs (MAP algorithm)

Table 2. Round 2, generic MLP with sigmoid hidden units and linear outputs: WRR on test set and final log-likelihood of the training acoustic observations given the model, yielded by the proposed algorithms applied to the SPK isolated digits, speaker-dependent problem. 3-dim acoustic space: 2 MFSCCs and signal log-energy

Model/algorithm	WRR (%)	Final log-likelihood
HMM with single Gaussians	77.50	1.1474e+04
HMM with 8-Gaussian mixtures	81.25	2.0520e+04
MLP initialized via BP	80.00	-9.1109e+04
SWS-ML	80.00	-9.1073e+04
MAP	85.00	-9.3754e+04

1. For each acoustic vector $\mathbf{x}$ in the training set, compute the value of the log-emission probabilities $y_i = log[b_i(\mathbf{x})]$, for $i = 1, \ldots, 31$, yielded by the HMM. Let $\mathbf{y} = (y_1, \ldots, y_{31})$.
2. Define a labeled training set $\mathcal{T} = \{(\mathbf{x}, \mathbf{y})\}$ including all such vector pairs.
3. Train the MLP with standard *backpropagation* (BP) to minimize the *Mean Squared Error* over $\mathcal{T}$.

The MLP is basically bootstrapped in order to “mimic” the behavior of the HMM, providing us with a reasonable approximation of log-emissions. This initialization scheme reminds us of Bourlard and Morgan’s training scheme. The results obtained applying Viterbi to the initialized net without any further hybrid-training procedure (third row of the table) do validate the initialization technique.

Given the nature of the output units and the fact that they are initialized to estimate logarithmic values, the present architecture (like that described in

Round 1) is not suitable for application of the *Bayes* learning rule. Instead, all the other training techniques (BML, SWS-ML and MAP), in their logarithmic variants, can be used herein.

As expected from the lack of any architectural constraints, training the MLP with the BML algorithm (not reported in the table) indefinitely increased the likelihood, lowering the WRR in a systematic manner. In the SWS-ML case, divergence in likelihood values was avoided only by choosing very small values for the parameter σ^{-2} that controls application of the SWS-ML case, not allowing for any tangible improvement over the initialized net.

Training in the logarithmic domain with the MAP criterion for 10 epochs (as suggested by preliminary cross-validation) in the on-line mode allowed for a significant WRR improvement over the standard HMM and the other hybrid-training techniques. Evolution of the log-likelihood ratio $\log P(\mathcal{Y} \mid \mathcal{M}) - \log \mathcal{P}(\mathcal{Y})$ between the constrained and the unconstrained models is shown in Figure 4. Application of the *batch* MAP learning rule was less effective in the present setup, yielding a 82.50% WRR.

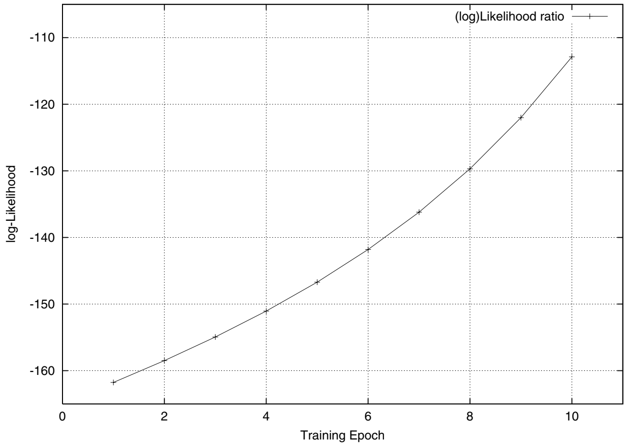


Fig. 4. Round 2: log-likelihood ratios between the constrained and the unconstrained models, evaluated over the training set as a function of the number of training epochs (MAP algorithm)

Round 3: Generic MLP, Linear Domain. Finally, we apply a generic MLP to estimate probabilistic quantities in the linear domain, in order to evaluate the linear version of the training algorithms, as well as to allow application of the *Bayes* training scheme, which requires ANN outputs to be probabilities, i.e. in the range $[0, 1]$. For this reason, sigmoids are introduced in the output layer. The initialization procedure is similar to that described above in the logarithmic domain, but it may be improved as follows:

1. an initial segmentation of the training set is obtained on the basis of prior knowledge on the correct HMM associated with each utterance and on the Viterbi hypothesis of segmentation provided by the pre-trained standard HMM;
2. target values are assigned according to this segmentation, namely a positive target[2] for unit i is assigned over all patterns belonging to state i given the segmentation, while a null target is assigned to all the other units (discriminative approach);
3. BP training of the MLP is accomplished over the labeled training set;
4. the trained MLP is used within the hybrid architecture along with the Viterbi algorithm to yield a novel segmentation of the training sequences;
5. the last two steps of the algorithm may be repeated for a certain number of iterations.

Note that this initialization technique basically coincides with Bourlard and Morgan's hybrid training scheme, once a probabilistic interpretation of ANN outputs is given in terms of "state posterior probabilities".

Table 3 shows the results obtained with the different techniques. The HMM with 8 Gaussians per state is the same used in round 2. The proposed initialization turns out to perform better than in the previous case, even improving over the standard HMM, i.e. Bourlard and Morgan's hybrid behaved better than the HMM, as expected. All the novel hybrid schemes, in turn, improved over the initialized MLP, with the exception of the BML that increased the likelihood but worsened the WRR (not reported in the table). In all cases, the on-line version of the algorithms yielded slightly better results than the batch case. A systematic improvement with respect to Round 2 is met.

The results do validate also the *Bayes* training scheme that could not be evaluated in the previous rounds. The state-priors Π_i to be used in the learning rule were estimated from the relative frequencies of the standard HMM states over the training set, during the initialization alignment procedure. No explicit estimation of the acoustic observations distribution $p(\mathbf{y})$ was needed to apply the *Bayes* algorithm herein: a uniform distribution of the features in the $[0, 1]$ interval was indeed guaranteed by the normalization of the input data. In Table 3, the term *MAP+Bayes* refers to the MAP scheme (constrained vs. unconstrained models) combined with the *Bayes* algorithm, i.e. MLP outputs are assumed to be the state posterior probabilities given the input acoustic observation.

Figures 5 and 6 show the log-likelihood of the training set given the model and the WRR, respectively, as functions of the parameter σ^{-2}. This parameter is the inverse of the variance of a zero-mean Gaussian that models the distribution of ANN weights (see Section 2.1). It is seen that a kind of "optimal" WRR is reached in correspondence with a value of σ^{-2} that represents a tradeoff between higher likelihood values (i.e. the model tends to degenerate when the likelihood diverges), corresponding to wide Gaussian pdfs for the ANN weights distribu-

[2] A target value equal to "1" is the default, but larger values may help improving convergence of the algorithm in practice.

Table 3. Round 3, generic MLP with sigmoid hidden and output units: Word recognition rate (WRR) on test set yielded by the proposed algorithms applied to the SPK isolated digits, speaker-dependent problem. 3-dim acoustic space: 2 MFSCCs and signal log-energy

Model/algorithm	WRR (%)
HMM with 8-Gaussian mixtures	81.25
MLP initialized via *Bourlard-like algorithm*	83.75
SWS-ML	90.00
Bayes	87.50
MAP	91.25
MAP+*Bayes*	87.50

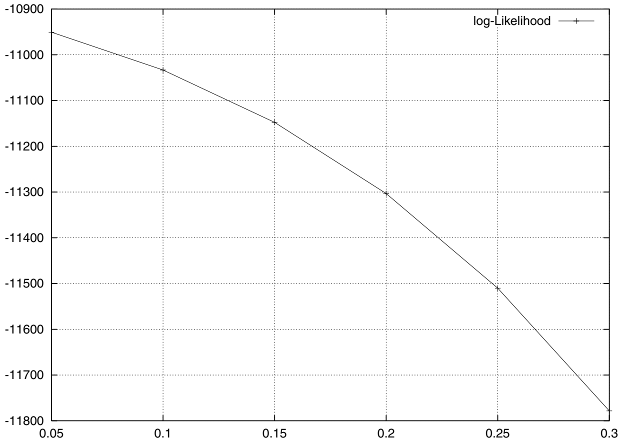


Fig. 5. Round 3, SWS-ML algorithm: log-likelihood evaluated over the training set as a function of the parameter σ^{-2} that controls the probability distribution of network weights

tion, and lower likelihood values due to too narrow Gaussians, i.e. oversimplified models.

To summarize, from the above experimental investigation it turns out that the BML algorithm increases the overall likelihood of the acoustic observations given the model, but an instance of the "divergence problem" systematically arises. The SWS-ML, *Bayes* and MAP algorithms allow for viable solutions, resulting in increased likelihood and improved recognition performance over the standard HMM. The MAP version of the algorithms results particularly effective. The on-line algorithms are usually better than the batch cases. This fact is aligned with the analogous phenomenon that occurs in standard BP for static ANNs. Finally, the MLP architectures outperform the log-RBF initialized via Gaussian-based HMM parameters; in particular, the MLP with output sigmoids jointly with the learning rules defined in the linear domain behaves better than

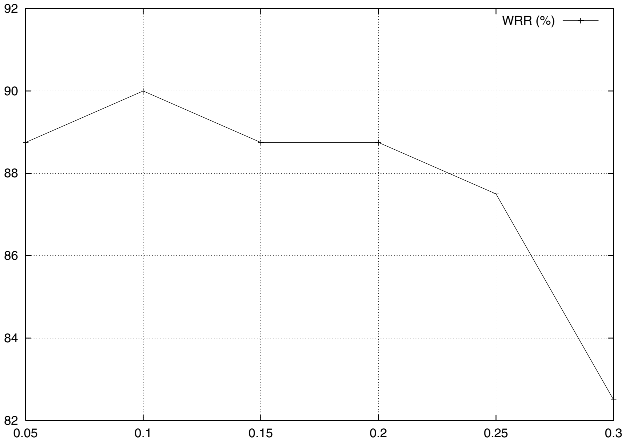


Fig. 6. Round 3, SWS-ML algorithm: WRR evaluated over the test set as a function of the parameter σ^{-2}

the corresponding MLP with linear outputs and the logarithmic version of the algorithms.

3.2 Experimental Analysis: Real-World Tasks

The nonparametric HMM was then successfully applied to several real-world continuous-speech, speaker independent ASR tasks, on both clean and noisy data. Most of the results are summarized in [10]. In [12] it is shown that the MAP version of the ANN/HMM allows for a 46.34% relative word error rate reduction with respect to standard HMMs in a task with a small vocabulary from the SPK database, using 8 MFSCCs and the log-energy. Application of the soft parameter grouping scheme yields a noise-tolerant model [14]: results on the *VODIS II/SpeechDatCar* database[3], collected in a real car environment, show a 64.58% relative word error rate reduction with respect to the standard HMM, as well as a dramatic gain over Bourlard and Morgan's hybrid.

4 Conclusive Remarks

This chapter discussed and analyzed a novel nonparametric HMM with connectionist estimates of the emission pdfs. Although the approach was evaluated on ASR tasks, it is suitable whenever sequence modeling with standard (parametric) HMMs may be accomplished. The theoretical dissertation and the experimental results in real-world tasks from literature highlights the potential advantages of

[3] Part of SpeechDatCar and Vodis-II projects was funded by the Commission of the EC, Telematics Applications Programme, Language Engineering, Contracts LE4-8334 and LE4-8336.

the proposed approach over HMMs and other state-of-the-art ANN/HMM hybrids, namely the well-known Bourlard & Morgan paradigm. We argue that such advantages may be explained on the basis of the nonparametric and "universal approximation" properties of ANNs, along with the joint optimization scheme of ANN and HMM parameters over the same criterion function, i.e., the likelihood of the observed pattern sequences given the model. We stressed the fact that explicit precautions are to be taken in order to overcome the "divergence problem", and we pointed out several effective solutions. Finally, an important remark concerns dealing with noisy data: robustness to noise is strictly related to the generalization capabilities of the model, and a suitable technique that improves learning and generalization was presented *via* the introduction of a ML trainable amplitude in the activation functions, resulting in a soft parameter grouping technique.

Acknowledgments

This chapter is based on my PhD Thesis, which is in memory of Bernhard Flury. Bernhard was Professor of Statistics at Indiana University and a jolly good fellow. He was a dear friend of mine and a constant reference to me; he continues to help me. somehow. Thank you, Bernhard.
The invaluable inspiration and contributions from Yoshua Bengio are gratefully acknowledged. Again, I wish to thank a lot my advisor & friend Marco Gori, who allowed the "whole thing" keep. Thanks to Marco Matassoni, Fabio Brugnara, Mauro Cettolo, Roberto Brunelli, and all the crew at DITH (Fabio and Titty). Last but not least, a big hug to Nadia Mana.

References

1. Y. Bengio. *Neural Networks for Speech and Sequence Recognition.* International Thomson Computer Press, London, UK, 1996.
2. Y. Bengio, R. De Mori, G. Flammia, and R. Kompe. Global optimization of a neural network-hidden Markov model hybrid. *IEEE Transactions on Neural Networks*, 3(2):252–259, 1992.
3. Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166, 1994. Special Issue on Recurrent Neural Networks, March 94.
4. C. M. Bishop. *Neural Networks for Pattern Recognition.* Oxford University Press, Oxford, 1995.
5. H. Bourlard and N. Morgan. *Connectionist Speech Recognition. A Hybrid Approach*, volume 247. Kluwer Academic Publishers, Boston, 1994.
6. J.S. Bridle. Alphanets: a recurrent 'neural' network architecture with a hidden Markov model interpretation. *Speech Communication*, 9(1):83–92, 1990.
7. S. B. Davis and P. Mermelstein. Comparison of parametric representations of monosyllabic word recognition in continuously spoken sentences. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 28(4):357–366, 1980.
8. La. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

9. E. Trentin. Networks with trainable amplitude of activation functions. *Neural Networks*, 14(4–5):471–493, May 2001.
10. E. Trentin. *Robust Combination of Neural Networks and Hidden Markov Models for Speech Recognition.* PhD thesis, DSI, Univ. di Firenze, 2001.
11. E. Trentin, Y. Bengio, C. Furlanello, and R. De Mori. Neural networks for speech recognition. In R. De Mori, editor, *Spoken Dialogues with Computers*, pages 311–361, London, UK, 1998. Academic Press.
12. E. Trentin and M. Gori. Continuous speech recognition with a robust connectionist/markovian hybrid model. In *Proceedings of ICANN 2001 (International Conference on Artificial Neural Networks)*, Vienna, Austria, August 2001.
13. E. Trentin and M. Gori. A survey of hybrid ANN/HMM models for automatic speech recognition. *Neurocomputing*, 37(1-4):91–126, March 2001.
14. E. Trentin and M. Gori. Toward noise-tolerant acoustic models. In *Proceedings of Eurospeech 2001*, Aalborg, Scandinavia, September 2001.

Cooperative Games in a Stochastic Environment

Bruno Apolloni, Simone Bassis, Sabrina Gaito, and Dario Malchiodi

Dipartimento di Scienze dell'Informazione, Università degli Studi di Milano
Via Comelico 39/41, 20135 Milano, Italy
{Apolloni,Bassis,Gaito,Malchiodi}@dsi.unimi.it

Abstract. We introduce a very complex game based on an approximate solution of a NP-hard problem, so that the probability of victory grows monotonically, but of an unknown amount, with the resources each player employs. We formulate this model in the computational learning framework and focus on the problem of computing a confidence interval for the losing probability. We deal with the problem of reducing the width of this interval under a given threshold in both batch and on-line modality. While the former leads to a feasible polynomial complexity, the on-line learning strategy may get stuck in an indeterminacy trap: the more we play the game the broader becomes the confidence interval. In order to avoid this indeterminacy we organise in a better way the knowledge, introducing the notion of *virtual game* to achieve the goal efficiently. Then we extend the one-player to a team mode game. Namely, we improve the success of a team by redistributing the resources among the players and exploiting their mutual cooperation to treat the indeterminacy phenomenon suitably.

1 Introduction

In this paper we face the task of learning to win a game characterized by highly complex aspects. Our player, Bob, has no information about the ability of his opponent, Nature; moreover the effects of his actions on the outcome of the single contest are neither known nor predictable, since the game mechanism is quite complex and uncertain. Bob can base his strategy only on a monotonicity property: the more Bob increases the value of a given parameter of the game (the one representing his strength), the more his winning ability improves. This class of games exploits the results of game theory [1] only partially since the payoffs associated to each strategy are random variables with unknown distribution. A similar lack of information can be overcome by considering this problem as a dynamic process in a computational learning context. In particular, we refer to the on-line learning paradigm [2,3] that specializes on a peculiar inference problem. In this framework we evaluate the actions through the construction of confidence intervals for the losing probability, thus abandoning the common goal of optimizing the expected value of a utility function [4].

This game model works well for a wide range of real life situations, from the sphere of economy and finance to the field of biology. We can consider, for example, two firms that produce the same kind of goods. The goal of each firm

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 25–34, 2003.

is to produce a final product qualitatively similar to the competitor's, but with a lesser expenditure of resources. We can also think in terms of an antiviral action against an unidentified virus: we do not know the effect of the application of a certain quantity of antibiotic, since the virus's characteristics are often unclear and the environment in which it survives very complex.

This paper is organized as follows. In Section 2 we describe the game. In Section 3 we solve our inference problem in both batch and on-line modality, analyzing an arising *indeterminacy phenomenon* that we relieve through a better organization of the available knowledge. In Section 4 we extend the problem to a team game framework and consider an approximately optimal distribution of resources among the players.

2 Statement of the Game

Bob and Nature play the following game.

Game 1 *The game consists in a series of contrasts between Bob (B) and Nature (N). In the single contest both players draw randomly an NP-hard problem's instance s from a set $\mathcal{S}$ (huge, but finite and discrete) and compute an approximate solution, whose accuracy depends on the amounts of resources γ_B and γ_N they employ (let us call them* strengths*), according to a monotone yet unknown relation. B wins, ties or loses if its solution is better, equal or worse than the opposite respectively. B's goal is to find (*learn*) the minimal strength that will let him lose with at most ε probability and good confidence level δ, rather than win in any future instance. The game is asymmetric: while B can arbitrarily modify his strength, N maintains her value fixed at an initial level.*

The complexity of the environment is modeled by choosing an NP-hard [5] optimization problem. In particular, we have adopted a *knapsack problem* [6] as a prototype of this complexity class that reads as follows: given (i) a set of objects, each characterized by a *weight* and a *comfort*, and (ii) a *capacity* to be filled by the weights, find within the subsets of objects fitting the capacity (the *feasible solutions*) the one maximizing the cumulative comfort. Its formal statement is the following:

Definition 1. *An instance for the 0-1 knapsack problem is a 4-ple $s = \langle n, W, Z, b\rangle \in \mathcal{S}$, where: $n \in \mathbb{N}$, $W = \{w_1, \ldots, w_n\} \subset \mathbb{N}^n$, $Z = \{z_1, \ldots, z_n\} \subset \mathbb{N}^n$, and $b \in \mathbb{N}$ is such that $w_i \leq b$ for each $i \in \{1, \ldots, n\}$ but $\sum_{i=1}^n w_i > b$.*

Given a knapsack instance s, an ordered n-ple $\mathbf{x} = \langle x_1, \ldots, x_n\rangle \in \{0,1\}^n$ is called a feasible solution *for s if $\sum_{i=1}^n w_i x_i \leq b$, and $g(x_1, \ldots, x_n) = \sum_{i=1}^n x_i z_i$ is called its* value*. Denoted with* Sol(*s*) *the set of all the feasible solutions for s, a solution of the knapsack problem on s maximizes g, i.e. is an n-ple*

$$\langle x_1^*, \ldots, x_n^*\rangle = \arg \max_{\langle x_1, \ldots, x_n\rangle \in \mathrm{Sol}(s)} g(x_1, \ldots, x_n)$$

where $\arg(g(x)) = x$.

Let us consider a sequence S [1] of instances that are submitted to the two competitors B and N. The challenge they face is to provide on each element s of the sequence an approximate solution better than the opponent's. To avoid operational traps due to exponentially long computations, both competitors adopt a polynomial-time approximation scheme (*PTAS*) [7]. In particular we assume that both B and N implement the Sahni algorithm [8], which on a given value γ_B of the strength of B, guarantees in time $O(n^{\gamma_B})$ a solution whose relative difference w.r.t. the value of the optimal one is no greater than $\eta = 1/(\gamma_B + 1)$, i.e. $\frac{g^* - \hat{g}}{g^*} \leq \eta$. It consists in two parts:

- k-PAS, that enumerates all the combinations of at most k objects, where k is the aforementioned parameter γ_B, corresponding to the strength of the player,
- GREEDY, that completes these partial solutions by filling the knapsack with the unused items considered in decreasing order of comfort-to-weight ratio, until no more insertions are allowed.

In the rest of the paper we will refer to the following inference framework that we called *algorithmic inference* elsewhere [9]. We consider a string of data (possibly of infinite length) that we partition into a prefix we assume to be known at present (and therefore call sample) and a suffix of unknown future data we call a population. All these data share the feature of being observations of the same phenomenon, which is exactly the object of our inference. In this framework *parameters concerning the population are random variables*, since they concern the unknown random suffix of an observed set of data. The basic inference tool is a *twisting* between properties we establish on the sample and random properties, such as the above parameter, we are questioning on about the population.

3 One-Player Game

Due to the game's monotonicity, we can always represent the instance space as a sequence of nested domains (Figure 1(a)): we call them $\gamma-$*not losing domains* (γ-*NL*), each characterized by a level γ of B strength. γ-NL contains all the instances on which Bob does not lose (not losing instances for short) with the strength γ. We learn a suitable strength for B in both batch and on-line modality.

3.1 Batch Mode

We assume in this section that whenever Bob suffers a defeat, increases his strength by one unit. In this case we determine the minimal number m of contests B must play in order to achieve a confidence $1 - \delta$ that the losing probability is less than or equal to ε. This number must satisfy the following inequality polynomial in $n, 1/\varepsilon, 1/\delta$, as stated in the following theorem [10]:

[1] By default capital letters will denote random variables and small letters their corresponding realizations; the sets the realizations will be denoted by gothic letters.

Theorem 1. *Having a knapsack problem on at most n objects as competition and using the Sahni approximately solving algorithm B learns a strength γ_B guaranteeing a losing probability less than or equal to ε with confidence $1-\delta$ after playing a number m of instances of the problem such that:*

$$\frac{1-\varepsilon}{\varepsilon}\log\frac{1}{\delta} \leq m \leq \max\left\{\frac{2}{\varepsilon}\log\frac{1}{\delta}, \frac{5.5(n-1)}{\varepsilon}\right\} \tag{1}$$

Proof. Let us assume that at the beginning of the game B strength equals 0 (for simplicity's sake) and that the result of our learning task is a strength γ_B inducing a not-losing region h. This means that during the game γ_B losing instances have been submitted to B. As an extreme case let us assume that all these instances are inside $\overline{h} = \mathcal{S} - h$ (actually, the true constraint is that the first losing instance s must lie outside the 0-NL domain, the second outside the 1-NL domain, etc.). In this case the following implication holds:

$$(U_{\overline{h}} < \varepsilon) \Leftarrow (K_\varepsilon \geq \gamma_B + 1) \tag{2}$$

where $U_{\overline{h}}$ is the measure of $\overline{h}$ (a random measure according to the algorithmic inference framework) and K_ε accounts the number of losing instances if the defeat's probability would be equal to ε.

Consider a nested sequence $\mathbf{B}$ of domains pivoted around $\overline{h}$, i.e. each domain in the sequence includes the previous ones and $\overline{h}$ is an element of the sequence. In this respect relation (2) reads: "Given that the game ended with strength γ_B, we acknowledge that $U_{\overline{h}}$ is less than ε because a domain B_ε of measure ε exists in $\mathbf{B}$ containing from the sample points both the above γ_B points and a further one outside h witnessing the inclusion of h in B_ε". Since having γ_B points all inside h is less probable than the necessary condition of having the first losing instance s outside the 0-NL domain, the second outside the 1-NL domain, etc., we can assume the probability of the rightmost event in (2) as a lower bound to $\mathrm{P}\left(U_{\overline{h}} < \varepsilon\right)$. Moreover, since we do not know what the learnt strength γ_B will be, we further bound from below this probability by substituting γ_B with its maximum value $n-1$ in the above probability:

$$\mathrm{P}\left(U_{\overline{h}} < \varepsilon\right) \geq \mathrm{P}\left(K_\varepsilon \geq n\right). \tag{3}$$

K_ε, as the number of points in a sample falling in a region of measure ε, follows a binomial distribution law, and thus by inverting inequality (3) we obtain the upper bound in the claim on the number of contests needed to learn γ_B [11].

The lower bound comes strictly from the fact that at least one instance s must lie in the first enlargement of $\overline{h}$ generated by our learning algorithm. Namely, suppose that all contended losing instances fall inside γ_B-NL domain. At least one however must fall outside $(\gamma_B - 1)$-NL domain, exactly the instance that required the γ_B^{th} increment. Thus at least one instance must fall in a NL domain complement, e.g. of strength γ_ε, of measure $\varepsilon > u_{\overline{h}}$. Thus

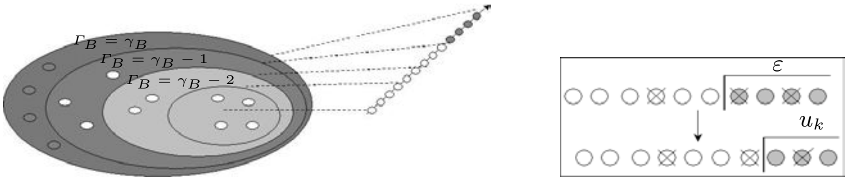
$$(u_{\overline{h}} < \varepsilon) \Rightarrow (k_\varepsilon \geq 1) \tag{4}$$

and, inverting the corresponding probability inequalities, we obtain the lower bound in the claim.

We note that the learning task computational complexity is polinomial too.

3.2 On-Line Mode

This modality consists in following the evolution of the game step by step. We discover some histories exist such that the confidence interval we can compute for Bob's losing probability has the disappointing feature of increasing its width as long as the game proceeds.



(a) (b)

Fig. 1. Partition of the instance space with respect to the strength values (a), and equivalent game (b).

We will refer to an equivalent game. Having introduced the partial order relation among the instances through the nested regions as in Figure 1(a), we can project all the space on the real line in such a way that the order relation is preserved. In the equivalent game the sequence of instances is divided in two regions by a cursor, as shown in the Figure 1(b): the right region contains all the losing instances (gray balls) while the left all the not losing ones (white balls).

Each iteration of the game consists in these steps: an instance is drawn (marked balls in the equivalent game), both players generate a solution and Bob changes his strength depending on his strategy. As a first strategy we adopt the strength updating of section 3.1 which corresponds to shifting the cursor right by an unknown number of instances. To estimate Bob's defeat probability U_k, which is the measure of the losing region in the instance space, we have to use two statistics: K, or the number of losing instances at run-time; and $\widetilde{K}$, or the number of instances which were losing when extracted and remain losing even with the current level of strength. The monotonicity of the game guarantees the validity of these two implications:

$$(U_k \leq \varepsilon) \Rightarrow (K_\varepsilon \leq k) \tag{5}$$

$$\left(\widetilde{K}_\varepsilon \geq \widetilde{k}\right) \Rightarrow (U_k \leq \varepsilon) \tag{6}$$

where K_ε is a random variable counting the number of losing instances at run-time if the defeat's probability would be equal to ε, and $\widetilde{K}_\varepsilon$ corresponds to $\widetilde{K}$ when the losing probability is equal to ε. Equation (5) comes from the fact that if the number of losing instances at run-time increases, then the number of increments in Bob's strength cannot decrease. This means that the cursor position cannot shift left and, finally, the losing probability cannot grow. Implication (6) explains the fact that, for whatever strength, an expansion of the losing region at the basis of an increase of the defeat probability enlarges or at least does not diminish the set of sampled losing instances inside it.

The probabilistic companions of (5) and (6) give rise to the following bounds for the cumulative distribution of the losing probability:

$$I_\varepsilon\left(\widetilde{k}, m-\widetilde{k}+1\right) \geq \mathrm{P}\left(U_k \leq \varepsilon\right) = F_{U_k}(\varepsilon) \geq I_\varepsilon\left(\widetilde{k}+1, k-\widetilde{k}\right) \tag{7}$$

where $I_\beta(h,r) = 1 - \sum_{i=0}^{h-1} \binom{h+r-1}{i} \beta^i (1-\beta)^{h+r-i-1}$ is the Incomplete Beta function.

The upper bound derives directly from the second implication, while to compute the lower bound we have to apply some results on *order statistics theory* [12]. Namely, having ordered the losing instances according to the straight line in Figure 1(a), we realize that the losing region after k updates is constituted by the $\widetilde{k}$ rightmost *equivalent blocks* identified by the position of the instances still remaining loser after these updates, plus at most one more block to manage the gap between the surely losing and surely not losing regions. We obtain from equation (7) a $1-\delta$ confidence interval (l_i, l_s) for the losing probability U_k depending on the number of iterations m and on the values of the two statistics K and $\widetilde{K}$, where l_i is the $\delta/2$ quantile of the Beta distribution of parameters $\widetilde{k}$ and $m-\widetilde{k}+1$, and l_s is the analogous $1-\delta/2$ quantile for parameters $\widetilde{k}+1$ and $k-\widetilde{k}$. We can see this from the following equations:

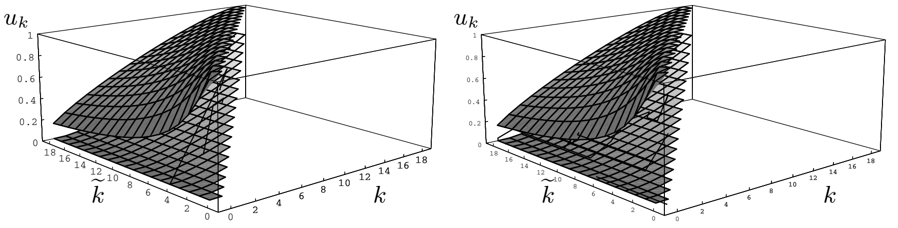
$$I_{l_s}\left(\widetilde{k}+1, k-\widetilde{k}\right) = 1-\delta/2, \qquad I_{l_i}\left(\widetilde{k}, m-\widetilde{k}+1\right) = \delta/2$$

The plot of these intervals with k and $\widetilde{k}$ for fixed m in Figure 2(a) shows an *indeterminacy phenomenon*, which manifests in two aspects:

(i) for some game's histories we cannot simultaneously obtain a low defeat probability and a high accuracy in its estimation; some trajectories remain trapped in the indeterminacy zone (corresponding to the central section of the graph); they cannot reach the target region, at the extreme left of the chart, characterized by a low defeat probability and a high accuracy in its estimation; in this case

(ii) by increasing the number of iterations, the lower bound diminishes and the confidence interval width grows.

However the two trajectories of game histories represented in Figure 2(a) show that the spread of the confidence interval is needed to contain the losing probability.



(a) (b)

Fig. 2. 0.9 confidence regions for the losing probability with m = 20. The trajectories are obtained (a) updating Bob strength whenever he suffers a defeat, and (b) implementing the strategy related to the virtual game.

Solving problems related to the indeterminacy phenomenon is intriguing: we can either organize the available knowledge in a better way or introduce a team game. Now we analyze the first method. We can see from the three-dimensional graph that if we want to reach the target zone we have to establish a meaningful difference between k and $\widetilde{k}$; we obtain it playing two games simultaneously: an *effective game*, where Bob uses the current strength to compute $\widetilde{k}$; and a *dummy game*, featuring a strength equal zero, which lets him accumulate k. These statistics are the same as a *virtual game* which starting from zero, reaches the actual value of Bob' strength. Figure 2(b) shows that with this game all the trajectories reach the target zone in few steps.

4 Team Game

Now we move to a more complex game where we are interested not only in the individual behavior of the players but also in the global losing probability of one team versus another (each consisting in a fixed number ν of players). We will see that the introduction of a *team game* is a good antidote for the indeterminacy phenomenon, as it exploits the synergy derived from the players' mutual collaboration. The game is similar to the previous one: each player of Bob's team plays a single contest with his corresponding antagonist in Nature's team; and this happens simultaneously for all the players in each contest. While Nature maintains her strength at a fixed level, Bob can modify his strength in such a way that the total resources used in a given contest cannot exceed a fixed team budget b_B. Bob's aim is not only to have a low losing probability, but also to use the minimal amount of resources.

We have merged all the individual instance spaces in a global one and have obtained the following confidence interval for the team losing probability (i.e. probability that a player, uniformly extracted, will suffer a defeat in the next contest) based on the statistics $S_B = \sum_{i=1}^{\nu} k_i$ and $\widetilde{S}_B = \sum_{i=1}^{\nu} \widetilde{k}_i$:

$$I_\varepsilon\left(\widetilde{S}_B, m\nu - \widetilde{S}_B + 1\right) \geq \mathrm{P}\left(U_{S_B} \leq \varepsilon\right) = F_{U_{S_B}}(\varepsilon) \geq I_\varepsilon\left(\widetilde{S}_B + 1, S_B - \widetilde{S}_B\right) \quad (8)$$

We must underline that, unlike the single statistics K and $\widetilde{K}$, the sums S_B and $\widetilde{S}_B$ follow no simple binomial distribution $\mathcal{B}(n, p)$, since we sum random variables following binomial distribution laws with different values of the parameter p. The use of the sum of these variables improves the accuracy of the confidence interval because we can increase the value of the statistics.

Now the virtual game introduced in the previous section is extended. Each competitor plays several games: an effective one and as many dummy games as strength levels between 0 and $\gamma_{\max}$, where $\gamma_{\max}$ is the strength of the most powerful player in Bob's team. Each player stores an outcome table (Table 1(a)) in which he notes down at different time steps the outcomes of all the games (1 in case of defeat and 0 otherwise) and a defeat table (Table 1(b)) containing the number of defeats suffered with different strengths.

Table 1. Outcome table for a game concerning a knapsack problem of 10 objects (a) and associated Defeat table (b).

Step	1	2	3	4	5	6	7	8
$\gamma = 0$	1	1	1	0	1	1	1	1
$\gamma = 1$	1	1	1	0	1	1	1	1
$\gamma = 2$	1	1	0	0	1	1	0	1
$\gamma = 3$	1	1	0	0	0	1	0	1
$\gamma = 4$	0	1	0	0	0	1	0	1
$\gamma = 5$	0	1	0	0	0	1	0	0
$\gamma = 6$	0	0	0	0	0	1	0	0
$\gamma = \gamma_{B_i}$	0	0	0	0	0	1	0	0
$\gamma = 8$	0	0	0	0	0	0	0	0
$\gamma = \gamma_{\max}$	0	0	0	0	0	0	0	0

(a)

Step	1	2	3	4	5	6	7	8
$\gamma = 0$	1	2	3	3	4	5	6	7
$\gamma = 1$	1	2	3	3	4	5	6	7
$\gamma = 2$	1	2	2	2	3	4	4	5
$\gamma = 3$	1	2	2	2	2	3	3	4
$\gamma = 4$	0	1	1	1	1	2	2	3
$\gamma = 5$	0	1	1	1	1	2	2	2
$\gamma = 6$	0	0	0	0	0	1	1	1
$\gamma = \gamma_{B_i}$	0	0	0	0	0	1	1	1
$\gamma = 8$	0	0	0	0	0	0	0	0
$\gamma = \gamma_{\max}$	0	0	0	0	0	0	0	0

(b)

With this information each player observes the statistics k and $\widetilde{k}$ for different strength levels and estimates the losing probability in each situation. Generalizing this mechanism, we can compute the sums S_B and $\widetilde{S}_B$ for different admissible *strengths' configurations*[2] estimating the corresponding team losing probability. Thus we find another optimization problem to solve: among the sets of all considered configurations we have to choose the one minimizing the team losing probability U_k. Due to the expensive computational cost of analyzing all configurations, we have suitably restricted this domain by introducing the notion of neighborhood:

Definition 2. *We call B_i^r the neighborhood of i of radius r. It consists of the set of at most $(2r+1)$ strength integer values ranging from $\gamma_{B_i} - r$, to $\min\{\gamma_{B_i} + r, \gamma_{\max}\}$.*

The aforementioned optimization problem has the following form:

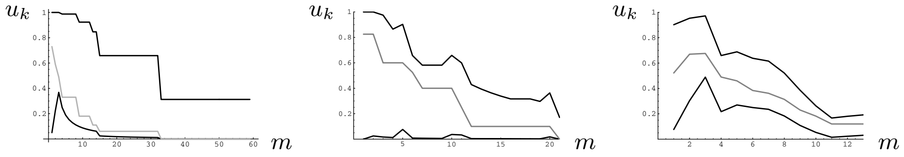
$$\boldsymbol{\gamma}_{\boldsymbol{B}}^{*} = \arg\min_{\boldsymbol{\gamma}_{\mathbf{B}} \in \mathcal{B}} \left\{ P_{\gamma_B}(U_k) : S_B = \sum_{i=1}^{\nu} k_{\gamma_{B_i}}, \widetilde{S}_B = \sum_{i=1}^{\nu} \widetilde{k}_{\gamma_{B_i}}, \sum_{i=1}^{\nu} \gamma_{B_i} \leq b_B \right\}$$

[2] An admissible strengths' configuration is a ν-uple $(\gamma_1, \ldots, \gamma_\nu)$, where each γ_i represents the strength employed by the i^{th} player in the contest and $\sum \gamma_i \leq b_B$.

where $P_{\gamma_B}(U_k)$ represents the losing probability when the strength of the i^{th} player is γ_{B_i}, $k_{\gamma_{B_i}}$ and $\widetilde{k}_{\gamma_{B_i}}$ are the statistics corresponding to the i^{th} player, $\mathcal{B} = B_1^r \times \ldots \times B_\nu^r$ and $\boldsymbol{\gamma_B} = (\gamma_{B_1}, \ldots, \gamma_{B_\nu})$. In order to ensure a high plasticity in the modification of γ_{max} we have adopted some heuristics.

An open problem of this strategy is the optimality guarantee. In other words we are not sure that the final configuration of resources is effectively the minimal one guaranteeing a team losing probability at most equal to the fixed threshold ε. We may reformulate this problem as a new knapsack problem to be solved by a hypothetical general whose constraint is the team budget and whose objects are the redistribution of resources among the players. In order to solve his knapsack problem the general needs a certain quantity of resources and uses a fraction of the team budget b_B (say $b_B p$), so that the remaining part of the budget ($b_B(1-p)$) — distributed appropriately among the players — is optimal. This can lead to a self referential problem, which could be overcome by the intervention of a figure in a further level of the hierarchy.

Figure 3 summarizes simulation results, showing the behavior of losing probability u_k (gray line) versus the number of iterations m, and the corresponding confidence intervals (black lines). In the first graph of Figure 3(a) Bob increases the strength any time he suffers a defeat. Due to indeterminacy, we cannot obtain simultaneously a low defeat probability and a high accuracy in its estimation. We obtain a better confidence interval in Figure 3(b), introducing the virtual game. We note that the convergence rate is better, too. Finally in Figure 3(c) we show the graph of team losing probability: we can see a fast convergence (we obtain a losing probability not greater than $\varepsilon = 0.2$ in only 12 steps) and an even better accuracy in the confidence intervals.



(a) First strategy (b) Virtual game (c) Team game

Fig. 3. Simulations of game based on a knapsack problem with 12 objects related to three different strategies. In the last game the teams are composed of 5 players. Light grey ↦ true value of the losing probability; black ↦ confidence region.

5 Conclusions

In this work we have introduced a new kind of stochastic game, characterized by its high difficulty coming both from the complexity of the involved basic problems and from the high uncertainty of its boundary conditions. Using stringent logical statements that we reverse in probabilistic inequalities within *algorithmic inference* we were able to solve it and to treat the uncertainty of the model; in particular, we have discovered an indeterminacy phenomenon that we relieved through a skill exploitation of available informations. This is obtained by introducing both a virtual game to enhance the effects of the current solution and a team strategy to profitably accumulate the information gatered by each agent within the team.

References

1. Nash, J.: Non-cooperative games. Annals of Mathematics **54** (1951) 286–295
2. Angluin, D.: Queries and concept learning. Machine Learning (1988) 319–342
3. Ben-David, S., Kushilevitz, E., Mansour, Y.: Online learning versus offline learning. Machine Learning **29** (1997) 45–63
4. Blackwell, D., Girshick, M.A.: Theory of Games and Statistical Decisions. Dover Publications, Inc., New York (1979)
5. Sahni, S.: Some related problems from network flows, game theory, and integer programming. In: Proceedings of the 13th Annual IEEE Symposium of Switching and Automata Theory. (1972) 130–138
6. Martello, S., Toth, P.: The 0–1 knapsack problem. In: Combinatorial Optimization. Wiley (1979) 237–279
7. Papadimitriou, C.H.: Computational Complexity. Addison-Wesley, Reading, Massachusetts (1994)
8. Sahni, S.: Approximate algorithms for the 0/1 knapsack problem. Journal of the Association of Computing Machinery **22** (1975) 115–124
9. Apolloni, B., Bassis, S., Malchiodi, D., Gaito, S.: The statistical bases of learning. *In* From Synapses to Rules: Disovering Symbolic Rules from Neural Processed Data, International School on Neural Nets "E.R. Caianiello" **5th course** (2002)
10. Apolloni, B., Ferretti, C., Mauri, G.: Approximation of optimization problems and learnability. *In L. Di Pace, editor*, Atti del Terzo Workshop del Gruppo AI*IA di Interesse Speciale su Apprendimento Automatico (1992)
11. Apolloni, B., Chiaravalli, S.: Pac learning of concept classes through the boundaries of their items. Theoretical Computer Science **172** (1997) 91–120
12. Tukey, J.: Nonparametric estimation II. Statistically equivalent blocks and multivariate tolerance regions. the continuous case. Annals of Mathematical Statistics **18** (1947) 529–539

Lattice Models for Context-Driven Regularization in Motion Perception

Silvio P. Sabatini, Fabio Solari, and Giacomo M. Bisio

Department of Biophysical and Electronic Engineering
University of Genova - Via Opera Pia 11/a
16145 Genova - ITALY
pspc@dibe.unige.it
http://www.pspc.dibe.unige.it

Abstract. Real-world motion field patterns contain intrinsic statistic properties that allow to define Gestalts as groups of pixels sharing the same motion property. By checking the presence of such Gestalts in optic flow fields we can make their interpretation more confident. We propose a context-sensitive recurrent filter capable of evidencing motion Gestalts corresponding to 1st-order elementary flow components (EFCs). A Gestalt emerges from a noisy flow as a solution of an iterative process of spatially interacting nodes that correlates the properties of the visual context with that of a structural model of the Gestalt. By proper specification of the interconnection scheme, the approach can be straightforwardly extended to model any type of multimodal spatio-temporal relationships (i.e., multimodal spatiotemporal context).

1 Introduction

Perception can be viewed as an inference process to gather properties of real-world, or *distal*, stimuli (e.g., an object in space) given the observations of *proximal* stimuli (e.g., the object's retinal image). The distinction between proximal stimulus and distal stimulus touches on something fundamental to sensory processes and perception. The proximal stimulus, not the distal stimulus, actually sets the receptors' responses in motion. Considering the ill posedness of such inverse problem, one should include *a priori* constraints to reduce the dimension of the allowable solutions, or, in other terms, to reduce the uncertainty on visual measures. These considerations apply both if one tackles the problem of perceptual interpretation as a whole, and if one considers the confidence on single feature measurements. Each measure of an observable property of the stimulus is indeed affected by an uncertainty that can be removed, or, better, reduced by making use of context information. *Early cognitive vision* can be related to that segment of perceptual vision that takes care of reducing the uncertainty on visual measures by capturing coherent properties (Gestalts) over large, overlapping, retinal locations, a step that precedes the true understanding of the scene.

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 35–42, 2003.

In this perspective, we formulate a probabilistic, model-based approach to image motion analysis, which capture, in each local neighborhood, coherent motion properties to obtain context-based regularized patch motion estimation. Specifically, given motion information represented by an optic flow field, we want to recognize if a group of velocity vectors belongs to a specific pattern, on the basis of their relationships in a spatial neighborhood. Casting the problem as a generalized Kalman filter (KF)[1], the detection occurs through a spatial recurrent filter that checks the consistency between the spatial structural properties of the input flow field pattern and a structural rule expressed by the process equation of the KF. Due to its recurrent formulation, KF appears particularly promising to design *context-sensitive filters* (CSFs) that mimic recurrent cortical-like interconnection architectures.

2 Kalman-Based Perceptual Inference

In general, KF represents a recursive solution to an inverse problem of determing the distal stimulus based on the proximal stimulus, in case we assume: (1) a stochastic version of the regularization theory involving Bayes' rule, (2) Markovianity, and (3) linearity and Gaussian normal densities. The first condition can be motivated by the fact that the a priori contraints necessary to regularize the solution can be described in probabilistic terms. Bayes' rule allows the computation of the *a posteriory* probability as $p(\boldsymbol{x}|\boldsymbol{y}) = p(\boldsymbol{y}|\boldsymbol{x})p(\boldsymbol{x})/p(\boldsymbol{y})$, where $p(\boldsymbol{x})$ is the *a priori* probability densities for the distal stimulus and represents *a priori* knowledge about the visual scene; $p(\boldsymbol{y}|\boldsymbol{x})$ is the likelihood function for $\boldsymbol{x}$. This function represents the transformation from the distal to proximal stimulus and includes information about noise in the proximal stimulus. Finally, $p(\boldsymbol{y})$ is the probability of obtaining the proximal stimulus. The inverse problem of determining the distal stimulus can be solved by finding $\hat{\boldsymbol{x}}$ that maximizes the *a posteriori* probability, $p(\boldsymbol{x}|\boldsymbol{y})$. Such $\hat{\boldsymbol{x}}$ is called a maximum a posteriori (MAP) estimator. Although the Bayesian framework is more general than the standard regularization, there exist a relationship between the deterministic and stochastic methods of solving inverse problems. Under the assumption of normal probability densities, maximizing the *a posteriory* probability $p(\boldsymbol{x}|\boldsymbol{y})$ is, indeed, equivalent to minimizing the Tikhonov functional. The second concept, the Markovianity, captures the step-by-step local nature of the interactions in a cooperative system, and makes possible Kalman recursion, by allowing to express *global* properties of the state in terms of its *local* properties. Under these hypotheses the conditional probability that the system is in a particular state at any time is determined by the distribution of states at its immediately preceding time. That is, the conditional distribution of that states of a system given the present and past distributions depends only upon the present. Specifically, considering the visual signal as a random field, the Markovianity hypothesis implies that the joint probability distribution of that random field has associated positive-definite, translational invariant conditional probabilities that are spatially Markovian (cf. Markov Random Fields). The third assumption represents the necessary conditions to achieve the exact, analytical solution of the KF.

3 Local Motion Gestalts

Local spatial features around a given location of a flow field, can be of two types: (1) the average flow velocity at that location, and (2) the structure of the local variation in a the neighborhood of that locality [2]. The former relates to the *smoothness constraint* or *structural uniformity*. The latter relates to *linearity constraint* or *structural gradients* (linear deformations). Velocity gradients provide important cues about the 3-D layout of the visual scene. On a local scale, velocity gradients caused by the motion of objects provide perception of their 3-D structure (structure from motion and motion segmentation), whereas, on a global scale, they specify the observer's position in the world, and his/her heading.

Formally, first-order deformations can be described by a 2×2 velocity gradient tensor

$$\mathbf{T} = \begin{bmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{bmatrix} = \begin{bmatrix} \partial v_x/\partial x & \partial v_x/\partial y \\ \partial v_y/\partial x & \partial v_y/\partial y \end{bmatrix} . \tag{1}$$

Hence, if $\boldsymbol{x} = (x, y)$ is a point in a spatial image domain, the linear properties of a motion field $\boldsymbol{v}(x, y) = (v_x, v_y)$ around the point $\boldsymbol{x}_0 = (x_0, y_0)$ can be characterized by a Taylor expansion, truncated at the first order:

$$\boldsymbol{v} = \bar{\boldsymbol{v}} + \bar{\mathbf{T}}\boldsymbol{x} \tag{2}$$

where $\bar{\boldsymbol{v}} = \boldsymbol{v}(x_0, y_0) = (\bar{v}_x, \bar{v}_y)$ and $\bar{\mathbf{T}} = \mathbf{T}|_{\boldsymbol{x}_0}$. By breaking down the tensor in its dyadic components, the motion field can be locally described through 2-D maps representing *cardinal* EFCs:

$$\boldsymbol{v} = \boldsymbol{\alpha}^x \bar{v}_x + \boldsymbol{\alpha}^y \bar{v}_y + \boldsymbol{d}_x^x \left.\frac{\partial v_x}{\partial x}\right|_{\boldsymbol{x}_0} + \boldsymbol{d}_y^x \left.\frac{\partial v_x}{\partial y}\right|_{\boldsymbol{x}_0} + \boldsymbol{d}_x^y \left.\frac{\partial v_y}{\partial x}\right|_{\boldsymbol{x}_0} + \boldsymbol{d}_y^y \left.\frac{\partial v_y}{\partial y}\right|_{\boldsymbol{x}_0} \tag{3}$$

where $\boldsymbol{\alpha}^x : (x, y) \mapsto (1, 0)$, $\boldsymbol{\alpha}^y : (x, y) \mapsto (0, 1)$ are pure translations and $\boldsymbol{d}_x^x : (x, y) \mapsto (x, 0)$, $\boldsymbol{d}_y^x : (x, y) \mapsto (y, 0)$, $\boldsymbol{d}_x^y : (x, y) \mapsto (0, x)$, $\boldsymbol{d}_y^y : (x, y) \mapsto (0, y)$ represent cardinal deformations, basis of the linear deformation space. In this work, we consider two different classes of deformation templates (opponent and non-opponent), each characterized by two gradient types (stretching and shearing), see Fig. 1. More complex local flow descriptors such as the divergence, the curl and the two components of shear, can be straightforwardly obtained by linear combination of such basic templates.

4 The Context Sensitive Filter

For each spatial position (i, j) and at time step k, let us assume the optic flow $\tilde{\boldsymbol{v}}(i, j)[k]$ as the corrupted measure of the actual velocity field $\boldsymbol{v}(i, j)[k]$. For

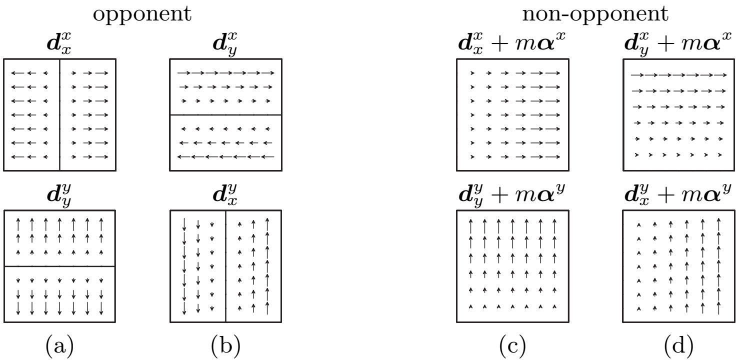


Fig. 1. Basic gradient type Gestalts considered. In stretching-type components (a,c) velocity varies *along* the direction of motion; in shearing-type components (b,d) velocity gradient is oriented *perpendicularly* to the direction of motion. Non-opponent patterns are obtained from the opponent ones by a linear combination of pure tranlations and cardinal deformations: $\boldsymbol{d}_j^i + m\boldsymbol{\alpha}^i$, where m is a proper positive scalar constant.

the sake of notation, we drop the spatial indices (i, j) to indicate the vector that represents the whole spatial distribution of a given variable. The difference between these two variables can be represented as a noise term $\boldsymbol{\varepsilon}(i, j)[k]$:

$$\tilde{\boldsymbol{v}}[k] = \boldsymbol{v}[k] + \boldsymbol{\varepsilon}[k] \ . \tag{4}$$

Due to the intrinsic noise of the nervous system, the neural representation of the optic flow $\mathbf{v}[k]$ can be expressed by a *measurement equation*:

$$\mathbf{v}[k] = \tilde{\boldsymbol{v}}[k] + \boldsymbol{n}_1[k] = \boldsymbol{v}[k] + \boldsymbol{\varepsilon}[k] + \boldsymbol{n}_1[k] \tag{5}$$

where $\boldsymbol{n}_1$ represents the uncertainty associated with a neuron's response. The Gestalt is formalized through a *process equation*:

$$\boldsymbol{v}[k] = \boldsymbol{\Phi}[k, k-1]\boldsymbol{v}[k-1] + \boldsymbol{n}_2[k-1] + \boldsymbol{s} \ . \tag{6}$$

The state transition matrix $\boldsymbol{\Phi}$ is *de facto* a spatial interconnection matrix that implements a specific Gestalt rule (i.e., a specific EFC); $\boldsymbol{s}$ is a constant driving input; $\boldsymbol{n}_2$ represents the process uncertainty. The space spanned by the observations $\mathbf{v}[1]$, $\mathbf{v}[2]$,..., $\mathbf{v}[k-1]$ is denoted by $\boldsymbol{\mathcal{V}}_{k-1}$ and represents the internal noisy representation of the optic flow. We assume that both $\boldsymbol{n}_1$ and $\boldsymbol{n}_2$ are independent, zero-mean and normally distributed: $\boldsymbol{n}_1[k] = N(0, \boldsymbol{\Lambda}_1)$ and $\boldsymbol{n}_2[k] = N(0, \boldsymbol{\Lambda}_2)$. More precisely, $\boldsymbol{\Phi}$ models space-invariant nearest-neighbor interactions within a finite region Ω in the (i, j) plane that is bounded by a piece-wise smooth contour. Interactions occur, separately for each component of the velocity vectors (v_x, v_y), through anisotropic interconnection schemes:

$$\begin{aligned} v_{x/y}(i,j)[k] = {} & w_N^{x/y} v_{x/y}(i,j-1)[k-1] + w_S^{x/y} v_{x/y}(i,j+1)[k-1] + \\ & w_W^{x/y} v_{x/y}(i-1,j)[k-1] + w_E^{x/y} v_{x/y}(i+1,j)[k-1] + \\ & w_T^{x/y} v_{x/y}(i,j)[k-1] + n_2^{x/y}(i,j)[k-1] + s_{x/y}(i,j) \end{aligned} \tag{7}$$

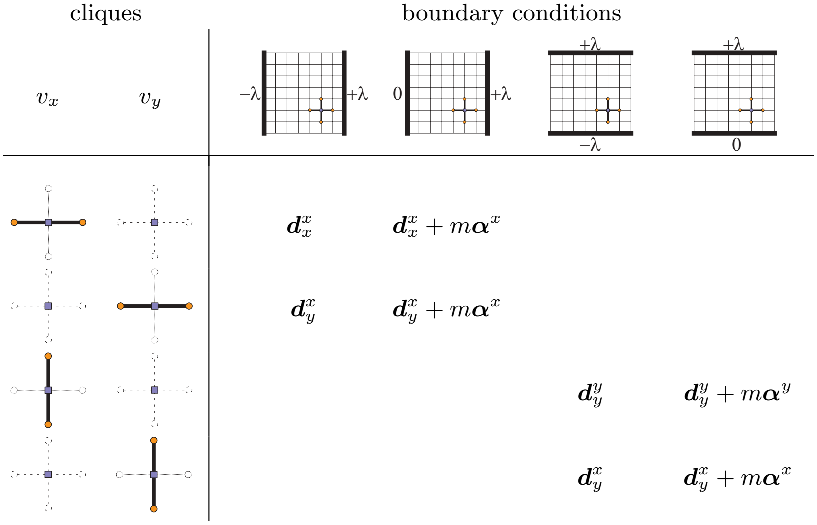

cliques
boundary conditions
v_x
v_y
−λ
+λ
0
+λ
+λ
−λ
+λ
0
$\boldsymbol{d}_x^x$
$\boldsymbol{d}_x^x + m\boldsymbol{\alpha}^x$
$\boldsymbol{d}_y^x$
$\boldsymbol{d}_y^x + m\boldsymbol{\alpha}^x$
$\boldsymbol{d}_y^y$
$\boldsymbol{d}_y^y + m\boldsymbol{\alpha}^y$
$\boldsymbol{d}_y^x$
$\boldsymbol{d}_y^x + m\boldsymbol{\alpha}^x$

Fig. 2. Basic lattice interconnection schemes for the linear deformation templates considered. The boundary value λ controls the gradient slope.

where (s_x, s_y) is a steady additional control input, which models the boundary conditions. In this way, the structural constraints necessary to model cardinal deformations are embedded in the lattice interconnection scheme of the process equation. The resulting lattice network has a *structuring effect* constrained by the boundary conditions that yields to structural equilibrium configurations, characterized by specific first-order EFCs. The resulting pattern depends on the anisotropy of the interaction scheme and on the boundary conditions (see Fig. 2). Given Eqs. (5) and (6), we may write the optimal filter for optic flow Gestalts. The filter allows to detect, in noisy flows, intrinsic correlations, as those related to EFCs, by checking, through spatial recurrent interactions, that the spatial context of the observed velocities conform to the Gestalt rules, embedded in $\boldsymbol{\Phi}$.

5 Results

To understand how the CSF works, we define the *a priori* state estimate at step k given knowledge of the process at step $k-1$, $\hat{\boldsymbol{v}}[k|\boldsymbol{\mathcal{V}}_{k-1}]$, and the *a posteriori* state estimate at step k given the measurement at the step k, $\hat{\boldsymbol{v}}[k|\boldsymbol{\mathcal{V}}_k]$. The aim of the CSF is to compute an *a posteriori* estimate by using an *a priori* estimate and a weighted difference between the current and the predicted measurement:

$$\hat{\boldsymbol{v}}[k|\boldsymbol{\mathcal{V}}_k] = \hat{\boldsymbol{v}}[k|\boldsymbol{\mathcal{V}}_{k-1}] + \boldsymbol{G}[k]\ \left(\mathbf{v}[k] - \hat{\mathbf{v}}[k|\boldsymbol{\mathcal{V}}_{k-1}]\right) \tag{8}$$

The difference term in Eq. (8) is the *innovation* $\boldsymbol{\alpha}[k]$ that takes into account the discrepancy between the current measurement $\mathbf{v}[k]$ and the predicted measurement $\hat{\mathbf{v}}[k|\boldsymbol{\mathcal{V}}_{k-1}]$. The matrix $\boldsymbol{G}[k]$ is the Kalman gain that minimizes the *a posteriori* error covariance:

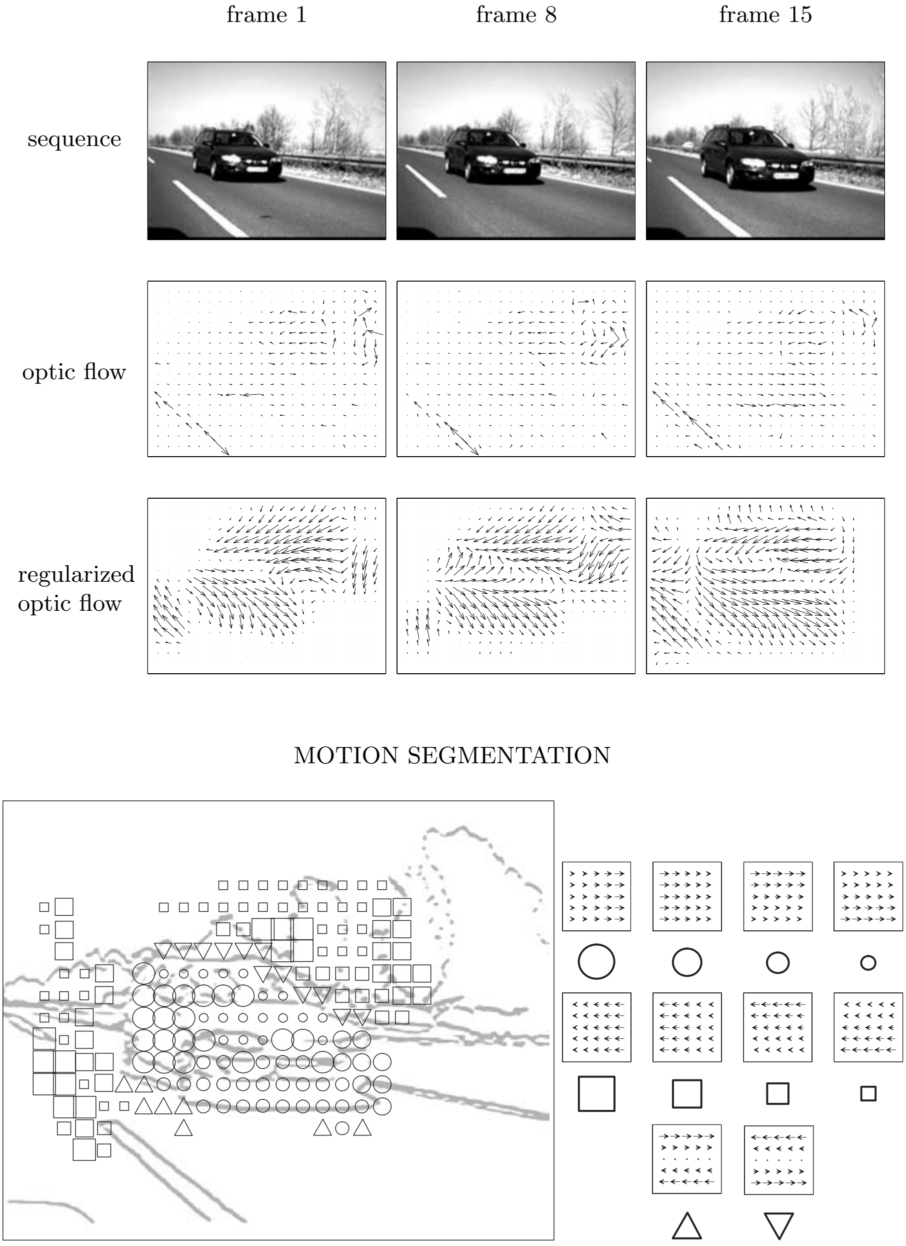


Fig. 3. Results on a driving sequence showing a road scene taken by a rear-view mirror of a moving car under an overtaking situations: Gestalt detection in noisy flows and the resulting motion segmentation (context information reduces the uncertainty on the measured velocities). Each symbol indicates a kind of EFC and its size represents the probability of the given EFC. The absence of symbols indicates that, for the considered region, the reliability of the segmentation is below a given threshold.

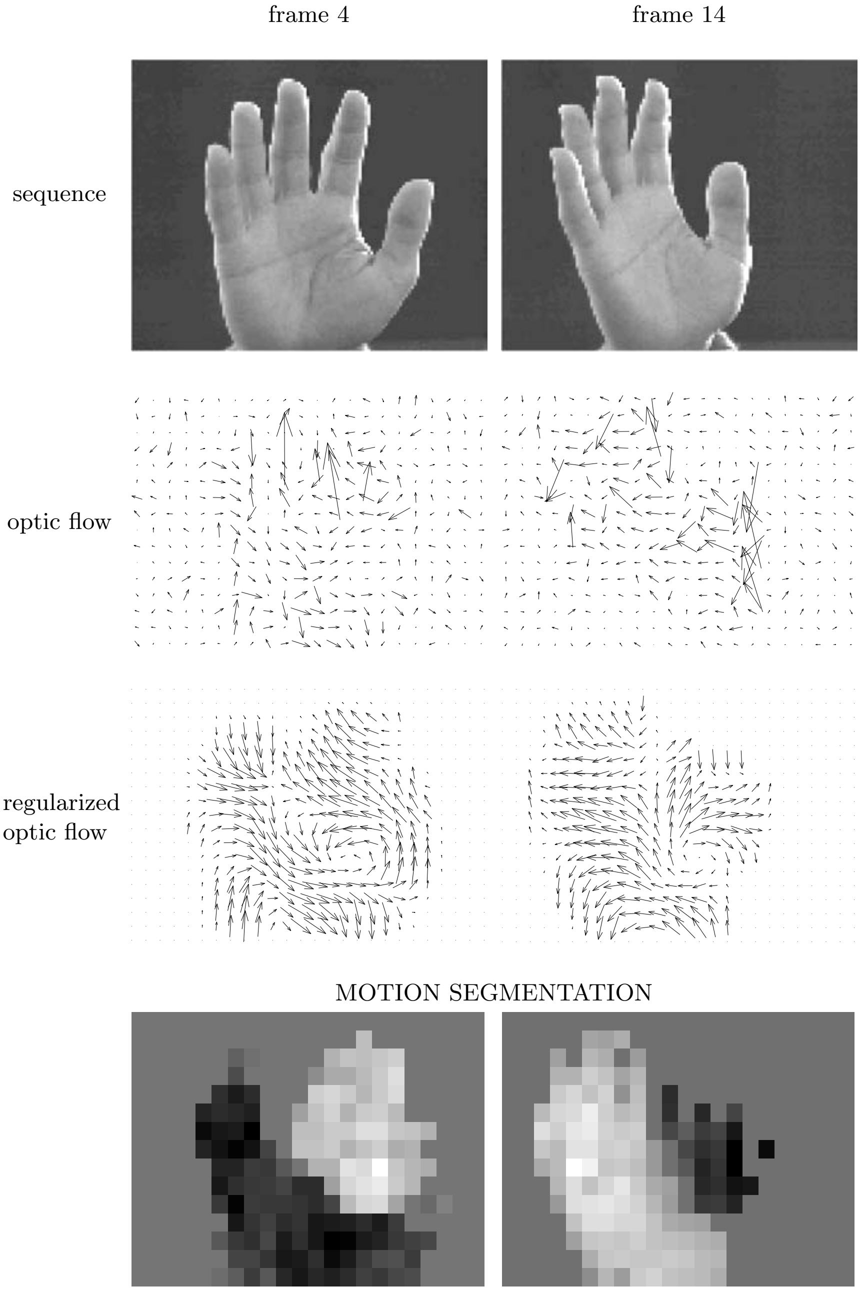


Fig. 4. Context-based patch motion estimation on a sequence showing a hand rotating around its vertical axis. The outputs of the CSFs can be used for motion segmentation evidencing segregation of different motions of each part of the hand: lighter grays indicate leftward motion, whereas darker grays indicate rightward motion.

$$\boldsymbol{K}[k] = E\left\{(\boldsymbol{v}[k] - \hat{\boldsymbol{v}}[k|\boldsymbol{\mathcal{V}}_k])(\boldsymbol{v}[k] - \hat{\boldsymbol{v}}[k|\boldsymbol{\mathcal{V}}_k])^T\right\} \ . \tag{9}$$

Eqs. 8 and 9 represent the mean and covariance expressions of the CSF output.

The covariance matrix $\boldsymbol{K}[k]$ provides us only information about the properties of convergence of the KF and not whether it converges to the correct values. Hence, we have to check the consistency between the innovation and the model (i.e., between observed and predicted values) in statistical terms. A measure of the reliability of the KF output is the Normalized Innovation Squared (NIS):

$$NIS_k = \boldsymbol{\alpha}^T[k]\ \boldsymbol{\Sigma}^{-1}[k]\ \boldsymbol{\alpha}[k] \tag{10}$$

where $\boldsymbol{\Sigma}$ is the covariance of the innovation. It is possible to exploit Eq. (10) to detect if the current observations are an instance of the model embedded in the KF [3].

To assess the performances of the CSFs, we applied them to real world optic flows. A “classical” algorithm [4] has been used to extract the optic flow. Regularized motion estimation has been performed on overlapping local regions of the optic flow on the basis of twenty-four elementary flow components. In this way, we can compute a dense distribution of the local Gestalt probabilities for the overall optic flow. Thence, we obtain, according to the NIS criterion, the most reliable (i.e. regularized) local velocity patterns, e.g., the patterns of local Gestalts that characterize the sequence (see Figs. 3 and 4).

Acknowledgments

We wish to thank Martin Mülenberg of Hella for having provided the driving sequences. This work was partially supported by EU Project IST-2001-32114 *ECOVISION “Artificial vision systems based on early cognitive cortical processing”*.

References

1. S. Haykin. *Adaptive Filter Theory.* Prentice-Hall International Editions, 1991.
2. J.J. Koenderink. Optic flow. *Vision Res.*, 26(1):161–179, 1986.
3. Y. Bar-Shalom and X.R. Li. *Estimation and Tracking, Principles, Techniques, and Software.* Artech House, 1993.
4. B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. *Proc. DARPA Image Understanding Workshop*, pages 121–130, 1981.

On the Dynamics of Scale-Free Boolean Networks

Roberto Serra, Marco Villani, and Luca Agostini

Centro Ricerche Ambientali Montecatini via Ciro Menotti 48, I-48023 Marina di Ravenna
rserra@cramont.it

Abstract. The dynamical features of Random Boolean Networks (RBN) are examined, in the case where a scale-free distribution of outgoing connectivities is introduced. RBN are known to display two major dynamical behaviours, depending upon the value of some model parameters, In the "ordered" regime the number of attractors is a growing polynomial function of the number of nodes N, while in the "chaotic" regime the growth is exponential. We present here a modification of the classical way of building a RBN, which maintains the property that all the nodes have the same number of incoming links, but which gives rise to a scale-free distribution of outgoing connectivities. Because of this modification, the dynamical properties are deeply modified: the number of attractors is much smaller than in classical RBN, their length and the duration of the transients are shorter. Perhaps more surprising, the number of different attractors is almost independent of the network size, over almost three order of magnitudes. Besides pertaining to the study of the dynamics of nonlinear networks, these results may have interesting biological implications.

1 Introduction

In this paper we study the dynamics of random boolean networks, and we show that the modification of some topological properties (in particular, the change from the usual Poissonian distribution of outgoing connectivities to a scale-free one) deeply affects the set of attractors and their basins of attraction.

Random boolean networks (RBN), which have been originally devised as models of gene regulatory networks, share a number of interesting features with boolean neural networks (like e.g. the well known Hopfield model) including massively parallel information processing, computing with dynamical attractors and intriguing robustness properties. A synthetic discussion of the relationship between RBN and neural models is presented in section 2.

In RBN, each node can either take the value 1 (active) or 0 (inactive); all the states of the N nodes are synchronously updated at discrete time steps, and the value at time t+1 is determined by a (constant) boolean function, associated to each node, of the values at time t of k_{in} other nodes [1].

If the state of node i depends upon that of node j, then there is a directed link from j to i. There are therefore k_{in} incoming links for every node; the other terminal of each link is chosen at random, with uniform probability, among the N-1 remaining genes. This is the classical way of building the network, and it gives rise to a probability distribution of outgoing links, $p(k_{out})$, which is approximately Poissonian for large N. For each node, the boolean function is chosen at random (with uniform probability) among all the possible k-input boolean functions, or among properly chosen subsets.

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 43–49, 2003.

The dynamics of RBN plays a very important role, and extensive studies have been performed on the number of attractors and their robustness (for finite N, all the attractors are cycles, a fixed point being a cycle of period 1). A general finding is that, for different parameter values, these networks may present either an "ordered" behaviour or a "disordered" (also termed "chaotic") one [1]. The border between ordered and disordered states depends upon the number of incoming links k_{in}, and the set of allowed boolean functions; if all the boolean functions are allowed, for k_{in} =2 the system is in an ordered state, and for k_{in} >2 in a disordered one. In both regimes the number of attractors N_A and the average period of the attractors T_A are growing functions of the network size; in the ordered regime, N_A and T_A grow slowly (according to a power law) with N, while in the disordered regime they grow exponentially with N.

There is however growing evidence [2] that many natural and artificial networks, including metabolic [3] [4] and protein [5] networks, as well as the Internet and the World Wide Web [6][7], are endowed with a scale-free topology, where the nodes may have different connectivities, with a power law probability distribution $p(k)\sim k^{-\gamma}$. Topology may affect the dynamical properties of the networks [8]: it has been shown that oscillator synchronization is easier to achieve with scale-free networks with respect to random networks [9], and that the attractors of boolean automata evolving according to the majority rule are modified by changing the topology from regular to random [10]. It has also been shown that the dynamical properties of boolean networks with a scale-free distribution of incoming links differ from those of RBN [11] [12] .

However, since most previous studies of RBN concern the case where k_{in} is the same for all the nodes, a proper comparison of scale-free vs. poissonian random networks could be better performed by fixing the value of k_{in} equal for all the nodes and introducing a scale-free distribution of the outgoing links. We study therefore how the dynamical properties of boolean networks are modified, with respect to RBN, if the distribution of the outgoing connections is scale free, instead of Poissonian as is the case in RBN.

In the model, which will be called here SFRBN (scale-free RBN), there are k_{in} incoming links for each node, exactly like in the original RBN, but the distribution of the other terminals is scale-free. The boolean function for each node is chosen at random, as in RBN. The constraint that k_{in} be the same for each node allows a careful comparison with the RBN model, the only difference between RBN and SFRBN being the probability distribution of outgoing links.

In section 3 we present the algorithm for generating SFRBN, while in section 4 we describe the dynamical behaviour of these networks in the case where k_{in}=2. In section 5 these results are discussed and indications for further work are provided.

2 Random Booelan Networks and Neural Networks

Boolean descriptions, where a node can take either the value 1 (usually associated to "firing") or 0 ("quiescent"), have been widely used as models of neurons, starting from the pioneering work of McCulloch and Pitts [13] and including the much studied Hopfield model [14], which played a major role in the renewal of interest in neural network research in the 80's [15]. Most models of boolean neurons are of the threshold type, where the node takes value 1 (at time t+1) if the weighted sum of inputs (at time t) is higher than a threshold and 0 otherwise

A different updating rule is used in random boolean networks, where the new state of a node is computed as a boolean function of its inputs. In spite of this difference, for any finite RBN it is possible to build an equivalent neural network and, conversely, for any booelan neural network it is possible to build an equivalent RBN. Here two networks are considered as equivalent if it is possible to find two subnetworks, one from each parent network, and a 1-1 correspondence among the nodes belonging to these subnetworks, such that all the pairs of corresponding nodes always take the same value.

The demonstration that it is possible to build a neural network equivalent to a given RBN is a straightforward consequence of the well known fact that it is possible to build a boolean neural network which realizes an arbitrary finite logical function of its inputs. The reverse (i.e. that, given a neural network, it is always possible to build an equivalent RBN), stems from the trivial observation that, for any threshold function with fixed weights, it is always possible to define a boolean function which, for any combination of its inputs, takes exactly the same value as the threshold function.

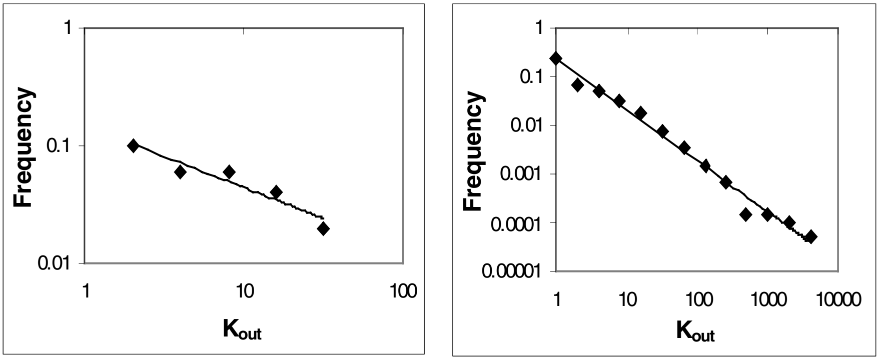
RBN have been used mostly as models of genetic regulatory networks, with fixed input weights. However, learning algorithms have also been proposed and it has bee shown that it is possible to achieve learning by examples in these networks: in particular, a RBN has been trained to perform arithmetic operations, by modifying the connection topology and the boolean functions, using a simulated annealing optimization method [16]. Boolean networks have also been evolved to play against each other an imitation game [1]. It should however be recalled that RBN have not been as effective as neural nets in learning and generalization.

For reasons which are largely due to the original biological inspiration, i.e. genomic regulatory networks, the empahsis of research in RBN has been mainly concerned with the study of their dynamical properties, and in particular of the way how the number and length of attractor cycles are affected by changing the system parameters.

3 Scale-Free Random Boolean Networks

A network with a scale-free distribution of outgoing connectivities can be built using an algorithm similar to the one introduced by Barabasi and Albert for undirected networks [17]: we start with k_{in} initial nodes, without any interconnecting link. Then, one at a time, we add a new node with exactly k_{in} incoming links from the already existing ones. The node which is connected to the new one is chosen among those which already belong to the network with a probability which is proportional to the "conventional" degree of outgoing connectivity of that node (which is equal to the degree of outgoing connectivity k_{out} if $k_{out}>0$, and is equal to 1 if $k_{out}=0$; this peculiar choice is necessary to avoid that a node which has 0 outgoing links at the time of its creation will never be connected). Multiple links from node j to node i are prohibited. The procedure is iterated until there are N nodes in the network. In the end, each of the initial k_{in} nodes, which have so far no incoming link, receives k_{in} incoming links from other nodes chosen at random, without preferential attachment.

We considered networks with different number of genes N ranging between 50 and 20000, and verified that the distribution of outgoing connectivities follows a power law (see fig.1) whose exponent ranges between –1.5 for N=50 and –2 for N=20000.



(a) (b)

Fig. 1. Distribution of typical outgoing connectivities (shown with logarithmic binning) of the network obtained with the modified Barabasi-Albert procedure presented in the text: (a) a network with N=50; (b) a network with N=20000. On the x-axis, number of outgoing links; on the y-axis, normalized frequency. A best fit of the power law exponent yields respectively the values γ=-1.52 and γ=-2.04.

4 The Dynamics of SFRBN

We performed a series of simulations of the SFRBN model, for the value k_{in} =2 which has been extensively studied in RBN. In this case, it has been shown in several papers that the number of attractors is a growing function of the number N of nodes. In several simulations a polynomial dependency has been observed (i.e. the number of attractors scales as N^a): earlier simulations indicated a small exponent a=1/2, while more recent work indicates a linear dependency [18] [19]. According to Bastolla and Parisi [20], in the case of large systems close to the critical zone, there are two different scales, one increasing as $N^{1/2}$ and another one “for rare but not vanishingly rare networks” increasing as an exponential of N. Therefore, the typical scale of the number and of the length of the attractors increase faster than $N^{1/2}$. Our own data (not shown), in the range of values considered, are consistent with an approximately linear (or slightly superlinear) growth with N.

The behaviour is very different in the case of SFRBN, as it is shown in fig. 2. The most interesting observations are that i) the number of attractors in SFRBN is much smaller than that of the corresponding RBN, ii) the number of attractors is almost independent of the network size N (for values of N between 50 and 20000, iii) the average period of attractor cycles is considerably smaller in SFRBN than in RBN and iv) the duration of transients is also much shorter in the scale-free case (the duration of transients grows very slowly with the network size N).

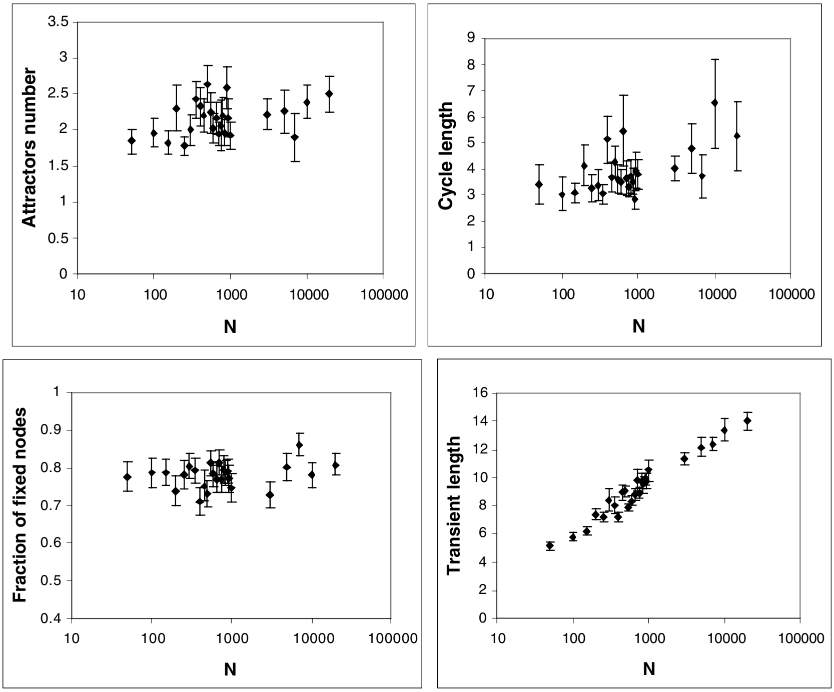


Fig. 2. Average values of the number of attractors (upper left), attractor cycle length (upper right), fraction of non-oscillating nodes (lower left) and transient duration (lower right) as a function of the number of nodes N of the network. The results are obtained averaging 50 networks for each point, each network tested with 200 initial conditions; the error bars correspond to one standard deviation.

5 Comments and Conclusions

The data of fig.2 refer to simulations with 200 different initial conditions per network, which is certainly a very strong undersampling of the set of initial conditions. Undersampling is unavoidable in simulations of such large networks (there are $2^{50} \cong 10^{15}$ different initial conditions for a network with 50 genes, and $2^{20000} \cong 10^{6000}$ for a network with 20000 genes) so, in order to test the robustness of our results, we checked the value of the number of attractors using 2000 different initial conditions per network, for values of N up to 5000, and we found no significant difference with respect to those of fig.2. We also performed simulations with 20000 different initial conditions of networks with N=3000, and we again found no significant difference. These results strongly support the conclusion that the small number of attractors, and its independence from the number of nodes, are true properties of SFRBN and are not artifacts due to undersampling of the set of initial conditions.

Note that the shortening of asymptotic cycles cannot be attributed to a growth of the set of non oscillating nodes: the average number of nodes which never change state in a single network is indeed slightly larger in the case of RBN (data not shown).

The fact that a scale-free boolean network seems "more ordered" than the corresponding RBN has been noticed also by Fox and Hill [12], although with a different model.

It has been shown here that, in the case where the number of incoming connections is equal for every node, the change in the topology of the outgoing connections suffices to cause profound modifications in the phase portrait of the system: the number of attractors is much smaller, and is almost independent of the network size. Moreover, the period of asymptotic cycles and the duration of transients are also shorter than in RBN. Further investigations should be performed to confirm these findings for different values of k_{in}. It should be interesting to analyze also the effects of the introduction of a cutoff in the maximum number of allowed links per node, and to assess the role of scale-free topology of ingoing links. Despite these limitations, this work already provides a clear indication concerning the existence of a mechanism potentially able to control the behavior of a growing and sparsely interconnected system.

As far as the biological interpretation is concerned, one should observe that the present results demonstrate that the scaling laws concerning the dependence of the number of attractors from the number of nodes of the network, which has been observed and discussed by Kauffman [1], is not robust with respect to the topological changes which have been investigated here.

It would be interesting to assess these scaling properties in the case where both incoming and outgoing connections are scale-free (thus removing the somewhat unnatural constraint of equal k_{in} for every node) and to compare the results with the case of purely random graphs (where, for any pair of nodes A and B, there is a fixed probability of drawing a link from node A to node B, and from node B to node A, independent from all the other existing links). The issue whether real genetic networks are closer to the scale-free or to the random graph model remains open, at the present level of our knowledge of real genetic control circuits.

Acknowledgments

We gratefully acknowledge interesting and fruitful discussions with David Lane and Alessandro Semeria, and an important comment by Stuart Kauffman.

References

1. Kauffman, S. A.: The origins of order. Oxford University Press (1993)
2. Amaral, L.A.N., Scala, A., Barthelemy, M, Stanley, H.E.: Classes of Small-World Networks. PNAS 97 (2000) 11149-11152
3. Jeong, H., Tombor, B., Albert, R., Oltvai, Z.N, Barabasi, A.L.: The large-scale organization of metabolic networks. Nature 407 (2000) 651-654
4. Wagner, A. and Fell, S.: The small world inside large metabolic networks. Santa Fe Institute Working Paper 00-07-041 (2000)
5. Jeong, H., Mason, S.P., Barabasi, A.L. & Oltvai, Z.N.: Lethality and centrality in protein networks. Nature 411 (2001) 41
6. Albert, R., Jeong H. & Barabasi, A.L.: Error and attack tolerance of complex networks. Nature 406, (2000) 378-382

7. Huberman B.A. & Adamic L.A.: Growth dynamics of the World-Wide Web. Nature 401 (1999) 131
8. Strogatz, S.H.: Exploring complex networks Nature 410 (2001) 268-276
9. Wang, X.F., Chen, G..; IEEE Trans. Circuits and Systems 49 (2002) 54-62
10. Serra, R. & Villani, M.: Perturbing the regular topology of cellular automata: implications for the dynamics. Lecture Notes in Computer Science 2493 (2002) 168-177
11. Aldana, M.: Dynamics of Boolean Networks with Scale-Free Topology. in press (available at: http://arXiv:cond-mat/0209571 v1)
12. Fox, J.J. & Hill, C.C.: From Tolology to Dynamics in Biochemical Networks. Chaos 11 (2001) 809-815
13. McCulloch W.S. and Pitts W.: A logical calculus of the ideas immanent in nervous activity. Bulletin. Math. Biophys. (1943) 115-137
14. Hopfield J.J.: Neural networks and physical systems with emergent collectivecomputational abilities. Proc. Natl. Acad. USA 79 (1982) 2554-2558
15. Serra R. & Zanarini G.: Complex Systems and Cognitive Processes. Springer Verlag (1990)
16. Patarnello A. & Carnevali P.: Boolean Networks which Learn to Compute. Europhysics Letters, 4(4): (1987) 503-508
17. Barabasi, A.L. & Albert, R.: Emergence of scaling in random networks. Science 286 (1999) 509-512
18. Socolar J.E.S. & Kauffman S.A.: Scaling in ordered and critical random Boolean networks. (2002) (available at: http://arXiv:cond-mat/0212306 v1)
19. Aldana, M., Coppersmith, S. & Kadanoff, L.P.: Boolean Dynamics with Random Couplings. (2002) (available at: http://arXiv:nlin.AO/0204062 v2)
20. Bastolla U., Parisi G.: The Modular Structure of Kauffman Networks. J. Phys. D 115 (1998) 219-233

A Deterministic Learning Approach Based on Discrepancy

Cristiano Cervellera[1] and Marco Muselli[2]

[1] Istituto di Studi sui Sistemi Intelligenti per l'Automazione - CNR
via De Marini, 6 - 16149 Genova, Italy
cervellera@ge.issia.cnr.it

[2] Istituto di Elettronica e di Ingegneria dell'Informazione
e delle Telecomunicazioni - CNR
via De Marini, 6 - 16149 Genova, Italy
marco.muselli@ieiit.cnr.it

Abstract. The general problem of reconstructing an unknown function from a finite collection of samples is considered, in case the position of each input vector in the training set is not fixed beforehand, but is part of the learning process. In particular, the consistency of the Empirical Risk Minimization (ERM) principle is analyzed, when the points in the input space are generated by employing a purely deterministic algorithm (*deterministic learning*). When the output generation is not subject to noise, classical number-theoretic results, involving discrepancy and variation, allow to establish a sufficient condition for the consistency of the ERM principle. In addition, the adoption of low-discrepancy sequences permits to achieve a learning rate of $O(1/L)$, being L the size of the training set. An extension to the noisy case is discussed.

1 Introduction

Neural networks are recognized to be universal approximators, i.e., structures that can approximate arbitrarily well general classes of functions [1,2].

Once the set of neural networks for the approximation of a given unknown function is chosen, the problem of estimating the best network inside such set (i.e., the network that is "closer" to the true function) from the available data (the *training set*) can be effectively analyzed in the context of *Statistical Learning Theory (SLT)*.

Consistency of classical Empirical Risk Minimization (ERM) method is guaranteed by proper theorems [3], whose hypotheses generally include that the samples are generated by i.i.d. realizations of a random variable having an unknown density p.

When the target function f represents a relation between an input space X and an output space Y, like in classification and regression problems, training samples are formed by input-output pairs (x_l, y_l), for $l = 0, \ldots, L-1$ where $x_l \in X$ and y_l is a possibly noisy evaluation of $f(x_l)$. If the location of the input patterns x_l is not fixed beforehand, but is part of the learning process, the term *active learning* or *query learning* is used in the literature.

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 53–60, 2003.

Most existing active learning methods use an optimization procedure for the generation of the input sample x_{l+1} on the basis of the information contained in previous training points [4], which possibly leads to a heavy computational burden. In addition, some strong assumptions on the *observation noise* $y - f(x)$ (typically, noise with normal density [5]) or on the class of learning models are introduced.

In the present paper a deterministic learning framework is discussed, which ensures consistency for the worst-case error under very general conditions. In particular, the observation noise can be described by any probability distribution, provided that it does not depend on x and its mean is zero, and the true function and the model can have any form (it is sufficient that they satisfy mild regularity conditions).

Number theoretic results on integral approximation can be employed to obtain upper bounds for the generalization error, which depends on the size of the training set. In the case where the output is unaffected by noise, the corresponding learning rates can be almost linear, which are better than those obtained with the passive learning approach [3]. Furthermore, they are intrinsically deterministic and do not imply a confidence level.

In case of noisy output, the consistency of the method can still be proven, though the stochastic nature of the noise spoils the advantages of a deterministic design, thus resulting in a final quadratic rate of convergence of the estimation error (which, however, is not worse than classical SLT bounds [3]).

Since the generation of samples for learning and the entire training process is intrinsically deterministic, the formal approach introduced in this paper will be named *deterministic learning* to distinguish it from the widely accepted *statistical learning.*

2 The Deterministic Learning Problem

Inside a family of neural networks $\Gamma = \left\{\psi(x,\alpha) : \alpha \in \Lambda \subset \mathbb{R}^k\right\}$ we want to estimate the device (i.e., the parameter vector α) that best approximates a given functional dependence of the form $y = g(x)$, where $x \in X \subset \mathbb{R}^d$ and $y \in Y \subset \mathbb{R}$, starting from a set of samples $(x^L, y^L) \in (X^L \times Y^L)$. Suppose that the output for a given input is observed without noise; the extension to the "noisy" case is discussed in Section 5.

In the following we will assume that X is the d-dimensional semi-closed unit cube $[0,1)^d$. Suitable transformations can be employed in order to extend the results to other intervals of $\mathbb{R}^d$ or more complex input spaces.

A proper deterministic algorithm will be considered to generate the sample of points $x^L \in X^L$, $x^L = \{x_0, \ldots, x_{L-1}\}$; since its behavior is fixed a priori, the obtained sequence x^L is uniquely determined and is not the realization of some random variable.

The goodness of the approximation is evaluated at any point of X by a *loss function* $\ell : (Y \times Y) \mapsto \mathbb{R}$ that measures the difference between the function g and the output of the network. The *risk functional* $R(\alpha)$, which measures the difference between the true function and the model over X, is defined as

$$R(\alpha) = \int_X \ell(g(x), \psi(x, \alpha))dx \tag{1}$$

Then, the estimation problem can be stated as

Problem E
Find $\alpha^* \in \Lambda$ *such that* $R(\alpha^*) = \min_{\alpha \in \Lambda} R(\alpha)$

The choice of the Lebesgue measure in (1) for evaluating the estimation performance is not a limitation: in fact, under mild hypotheses, the consistency of learning for the risk computed with the Lebesgue measure implies consistency also for the risk computed with any other measure which is absolutely continuous with respect to the uniform one [6].

Since we know g only in correspondance of the points of the sequence x^L, according to classical SLT literature, we consider the minimization of R on the basis of the *empirical risk* given L observation samples

$$R_{emp}(\alpha, L) = \frac{1}{L} \sum_{l=0}^{L-1} \ell(y_l, \psi(x_l, \alpha))$$

In order to obtain a minimum point for the actual risk $R(\alpha)$, we adopt an optimization algorithm Π_L aimed at finding $R^*_{emp}(L) = \min_{\alpha \in \Lambda} R_{emp}(\alpha, L)$. α_L is the parameter vector obtained after L samples have been extracted and the minimization has been performed.

$r(\alpha_L)$ will denote the difference between the actual value of the risk and the best achievable risk: $r(\alpha_L) = R(\alpha_L) - R(\alpha^*)$.

Definition 1 *We say that the learning procedure is* deterministically consistent *if* $r(\alpha_L) \to 0$ *as* $L \to \infty$.

The next theorem gives sufficient conditions for a sequence x^L to be deterministically consistent. It can be viewed as the corresponding result in deterministic learning of the *key theorem* [3] for classical SLT. The proof can be found in [6].

Theorem 1 *Suppose the following two conditions are verified:*

1. *The deterministic sequence* x^L *is such that*

$$\sup_{\alpha \in \Lambda} |R_{emp}(\alpha, L) - R(\alpha)| \to 0 \ \textit{as} \ L \to \infty. \tag{2}$$

2. *The learning algorithm* Π_L *is deterministically convergent, i.e., for all* $\epsilon > 0$ *and all* L*, it is possible to obtain* $R_{emp}(\alpha_L, L) - R^*_{emp}(L) < \epsilon$ *after a "sufficient" training.*

Then the learning procedure is deterministically consistent (i.e., $r(\alpha_L) \to 0$ *as* $L \to \infty$*).*

3 Discrepancy-Based Learning Rates

Since we are using the Lebesgue measure, which corresponds to a uniform distribution, to weight the loss function on the input space X, we must ensure a good spread of the points of the deterministic sequence x^L over X.

A measure of the spread of points of x^L is given by the *star discrepancy*, widely employed in numerical analysis [7] and probability [8].

Consider a multisample $x^L \in X^L$. If c_B is the characteristic function of a subset $B \subset X$ (i.e., $c_B(x) = 1$ if $x \in B$, $c_B(x) = 0$ otherwise), we define $C(B, x^L) = \sum_{l=0}^{L-1} c_B(x_l)$.

Definition 2 *If β is the family of all closed subintervals of X of the form $\prod_{i=1}^{d} [0, b_i]$, the* star discrepancy *$D_L^*(x^L)$ is defined as*

$$D_L^*(x^L) = \sup_{B \in \beta} \left| \frac{C(B, x^L)}{L} - \lambda(B) \right| \tag{3}$$

where λ indicates the Lebesgue measure.

The smaller is the star discrepancy, the more uniformly distributed is the sequence of points in the space.

By employing the Koksma-Hlawka inequality [9], it is possible to prove that the rate of (uniform) convergence of the empirical risk to the true risk is closely related to the rate of convergence of the star discrepancy of x^L.

In particular, it is possible to prove [6] that

$$\sup_{\alpha \in \Lambda} |R_{emp}(\alpha, L) - R(\alpha)| \leq V D_L^*(x^L) \tag{4}$$

where V is a bound on the *variation in the sense of Hardy and Krause* [10] of the loss function. Such parameter, which takes into account the regularity of the involved functions, plays the role that VC-dimension has in SLT literature. It is interesting to note that the structure of (4) permits to deal separately with the issues of *model complexity* and *sample complexity.*

3.1 Low-Discrepancy Sequences

Since the rate of convergence of the learning procedure can be controlled by the rate of convergence of the star discrepancy of the sequence x^L, in this section we present a special family of deterministic sequences which turns out to yield an almost linear convergence of the estimation error. Such sequences are usually referred to as *low-discrepancy sequences*, and are commonly employed in *quasi-random* (or *quasi-Montecarlo*) integration methods (see [7] for a survey).

An *elementary interval in base b* (where $b \geq 2$ is an integer) is a subinterval E of X of the form $E = \prod_{i=1}^{d} [a_i b^{-p_i}, (a_i + 1) b^{-p_i})$, where $a_i, p_i \in \mathbb{Z}$, $p_i > 0$, $0 \leq a_i \leq b^{p_i}$ for $1 \leq i \leq d$.

Definition 3 *Let t, m be two integers verifying $0 \leq t \leq m$. A* (t,m,d)-net in base b *is a set P of b^m points in X such that $C(E; P) = b^t$ for every elementary interval E in base b with $\lambda(E) = b^{t-m}$.*

If the multisample x^L is a (t, m, d)-net in base b, every elementary interval in which we divide X must contain b^t points of x^L. It is clear that this is a property of good "uniform spread" for our multisample in X.

Definition 4 *Let $t \geq 0$ be an integer. A sequence $\{x_0, \dots, x_{L-1}\}$ of points in X is a* (t,d)-sequence in base b *if, for all integers $k \geq 0$ and $m \geq t$ (with $(k+1)b^m \leq L-1$), the point set consisting of $\left\{x_{kb^m}, \dots, x_{(k+1)b^m}\right\}$ is a (t, m, d)-net in base b.*

Explicit non-asympthotic bounds for the star discrepancy of a (t, d)-sequence in base b can be given. For what concerns the asymptotic behaviour, special optimized sequences can be obtained that yield a convergence rate for $D^*_L(x^L)$ of order $O\left((\log L)^{d-1}/L\right)$

Consequently, for what concerns the learning problem, by employing (t, d)-sequences we obtain

$$\sup_{\alpha \in \Lambda} |R_{emp}(\alpha, L) - R(\alpha)| \leq O\left(\frac{V(\log L)^{d-1}}{L}\right) \tag{5}$$

If we compare the bound in (5) with classic results from SLT [3], we can see that, for a fixed dimension d, the use of deterministic low-discrepancy sequences permits a faster asymptotic convergence. Specifically, if we ignore logarithmic factors, we have a rate of $O(1/L)$ for a $(T(v, d), d)$-sequence, and a rate of $O\left(1/L^{1/2}\right)$ for a random extraction of points.

4 Experimental Results

In order to present experimental results on the use of low-discrepancy sequences, three different test functions taken from [11] have been considered and approximated by means of neural networks of the form of one hidden-layer feedforward networks with sigmoidal activation function.

For each function, low-discrepancy sequences (LDS) and training sets formed by i.i.d. samples randomly extracted with uniform probability (URS) have been compared. The LDS are based on *Niederreiter sequences* with different prime power bases, which benefit from the almost linear convergence property described in Subsection 3.1.

The empirical risk R_{emp}, computed by using a quadratic loss function, has been minimized up to the same level of accuracy for each function.

The generalization error has been estimated by computing the square Root of the Mean Square error (RMS) over a set of points obtained by a uniform discretization of the components of the input space.

Function 1) (Highly oscillatory behavior)

$$g(x) = \sin(a \cdot b), \quad \text{where } a = \mathrm{e}^{2x_1 \cdot \sin(\pi x_4)},\ b = \mathrm{e}^{2x_2 \cdot \sin(\pi x_3)},\ x \in [0,1]^4.$$

Six training sets with length $L = 3000$, three of which random and three based on Niederreiter's low-discrepancy sequences, have been used to build a neural network with $\nu = 50$ hidden units. In order to construct a learning curve which shows the improvement achieved by increasing the number of training samples, the results obtained with subsets containing $L = 500, 1000, 1500, 2000$ and 2500 points of the basic sequences are also presented.

Table 1 contains the average RMS for the two kinds of sequences, computed over a fixed uniform grid of $15^4 = 50625$ points.

Table 1. Function 1: Average RMS for random and low-discrepancy sequences.

Training	Sample size L					
Set	500	1000	1500	2000	2500	3000
URS	0.366	0.302	0.256	0.235	0.220	0.199
LDS	0.323	0.291	0.251	0.228	0.209	0.193

Function 2) (Multiplicative)

$$g(x) = 4\left(x_1 - \tfrac{1}{2}\right)\left(x_4 - \tfrac{1}{2}\right)\sin\left(2\pi\sqrt{(x_2^2 + x_3^2)}\right), \quad x \in [0,1]^4$$

For this function, the same network with $\nu = 50$ hidden units and the same input vectors contained in the 36 training sets used for function 1 were employed.

Table 2 contains the average RMS (in 10^{-3} units) for the different sequences, computed over the same uniform grid of $15^4 = 50625$ points used for function 1.

Table 2. Function 2: Average RMS for random and low-discrepancy sequences.

Training	Sample size L					
Set	500	1000	1500	2000	2500	3000
URS	0.657	0.493	0.271	0.213	0.179	0.157
LDS	0.402	0.336	0.221	0.157	0.136	0.124

Function 3) (Additive)

$$g(x) = 10\sin(\pi x_1 x_2) + 20\left(x_3 - \tfrac{1}{2}\right)^2 + 10x_4 + 5x_5 + x_6, \quad x \in [0,1]^6$$

36 new training sets with $L = 3000$ points were employed for this six-dimensional function. Again, 18 of them are based on a random extraction with uniform probability and the others 18 are based on low-discrepancy sequences.

For each set, the same network with $\nu = 40$ hidden units was trained by minimizing the empirical risk The RMS are computed over a uniform grid of $6^6 = 46656$ points.

Table 3 contains the average values of the RMS (in 10^{-3} units) for the two kinds of sampling.

Table 3. Function 3: Average RMS for random and low-discrepancy sequences.

Training	Sample size L					
Set	500	1000	1500	2000	2500	3000
URS	0.140	0.108	0.093	0.087	0.086	0.083
LDS	0.130	0.112	0.088	0.080	0.077	0.073

5 Comments

About the Variation. In order for the bound in (5) to hold, the loss function ℓ must satisfy suitable regularity conditions. In particular, its variation in the sense of Hardy and Krause must be finite. It can be shown [6] that this implies finiteness of the variation for each element of the family of neural networks Γ and the unknown function g.

In general, computing the variation of a function can be a very difficult task [10]. However, in case of "well-behaved" functions (having continuous derivatives), the computation of upper bounds for the variation can be much simpler. For this reason, it is possible to prove [6] that commonly used neural networks, such as feedforward neural networks and radial basis functions, satisfy the required regularity conditions.

Consistency in Case of Noisy Output. Suppose that the value of the output y, for a given input x, is affected by a random noise $\epsilon \in E \subset \Re$: $y = g(x) + \epsilon$. Then, a random term ϵ_l is given in correspondance of any sample input x_l.

We make the hypotheses that (i) the vectors ϵ_l are i.i.d. according to a probability measure P_ϵ with density p_ϵ and have zero mean; (ii) the vectors ϵ_l are independent from x_l for $l = 1, \ldots, L$; (iii) the loss function ℓ is quadratic (a typical choice in regression problems).

In this case, it can be shown [6] that the consistency of the learning algorithm is preserved. Anyway, the presence of the noise spoils the linear rate of estimation for the "deterministic" part of the output, resulting in a global quadratic rate of convergence (which is not worse than classic SLT rates). This is reasonable, since we can expect to fully exploit the advantageous properties of the quasi-random approach in a purely deterministic context. Nevertheless, if the output error is small, we can also expect that, for a finite number L of samples, the corresponding term in the estimation error is negligible.

Experimental Evidence. The experimental results obtained for the three test functions, each having different behavior and complexity, show that LDS outperform URS. In fact, in all the cases but one (function 3, $L = 1000$) the average RMS given by LDS are smaller. Finally, the better performance of the LDS with respect to URS becomes more evident when the size L increases, thus confirming the good asymptotic properties of this particular kind of deterministic sequences.

References

1. G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of Control, Signals, and Systems*, vol. 2, pp. 303–314, 1989.
2. F. Girosi, M. Jones, and T. Poggio, "Regularization theory and neural networks architectures," *Neural Computation*, vol. 7, pp. 219–269, 1995.
3. V. N. Vapnik, *Statistical Learning Theory.* New York: Wiley, 1995.
4. D. MacKay, "Information-based objective functions for active data selection," *Neural Computation*, vol. 4, pp. 305–318, 1992.
5. K. Fukumizu, "Statistical active learning in multilayer perceptrons," *IEEE Transactions on Neural Networks*, vol. 11, pp. 17–26, 2000.
6. C. Cervellera and M. Muselli, "Deterministic design for neural network learning: an approach based on discrepancy," To appear in the *IEEE Trans. on Neural Networks*, 2003.
7. H. Niederreiter, *Random Number Generation and Quasi-Monte Carlo Methods.* Philadelphia: SIAM, 1992.
8. N. Alon and J. H. Spencer, *The Probabilistic Method.* New York: John Wiley & Sons, 2000.
9. E. Hlawka, "Funktionen von Beschränkter Variation in der Theorie der Gleichverteilung," *Ann Mat. Pura Appl.*, vol. 54, pp. 325–333, 1961.
10. M. Blumlinger and R. F. Tichy, "Bemerkungen zu einigen Anwendungen gleichverteilter Folgen," *Sitzungsber. Österr. Akad. Wiss. Math.-Natur. Kl. II*, vol. 195, pp. 253–265, 1986.
11. V. Cherkassky and F. Mulier, *Learning from Data: Concepts, Theory, and Methods.* New York: John Wiley & Sons, 1998.

An Hybrid Neural/Genetic Approach to Continuous Multi-objective Optimization Problems

Mario Costa[1], Edmondo Minisci[2], and Eros Pasero[1]

[1] Politecnico di Torino – Dept. of Electronics
Corso Duca degli Abruzzi 24, 10129 Turin, Italy
{mario.costa,eros.pasero}@polito.it
[2] Politecnico di Torino – Dept. of Aerospace Engineering
Corso Duca degli Abruzzi 24, 10129 Turin, Italy
edmondo.minisci@polito.it

Abstract. Evolutionary algorithms perform optimization using the information derived from a population of sample solution points. Recent developments in this field regard optimization as the evolutionary process of an explicit, probabilistic model of the search space. The algorithms derived on the basis of this new philosophy maintain every feature of the classic evolutionary algorithms, but are able to overcome some drawbacks. In this paper an evolutionary multi-objective optimization tool based on an estimation of distribution algorithm is proposed. It uses the ranking method of non-dominated sorting genetic algorithm-II and the Parzen estimator to approximate the probability density of solutions lying on the Pareto front. The proposed algorithm has been applied to different types of test case problems and results show good performance of the overall optimization procedure in terms of the number of function evaluations.

1 Introduction

The extensive use of evolutionary algorithms in the last decade demonstrated that an optimization process can be obtained by combining effects of interactive operators such as selection - whose task is mainly to identify the best individuals in the current population - and crossover and mutation, which try to generate new and better solutions starting from the selected ones. But, if the mimicking of natural evolution in living species has been a source of inspiration of new strategies, the attempt to copy natural techniques as they are sometimes introduces a great complexity without a corresponding improvement of algorithms performance. Moreover standard evolutionary algorithms can be ineffective when problems exhibit a high level of interaction among variables. This is mainly due to the fact that recombination operators are likely to disrupt promising sub-structures of optimal solutions.

Alternatively, in order to make a rational use of the evolutionary metaphor and/or to create optimization tools that are able to handle very hard problems (with several parameters, with difficulties in linkage learning, deceptive), some algorithms have been proposed that automatically learn the structure of the search space. Following this way, several works, based on explicit probabilistic-statistic tools, have been carried out.

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 61–69, 2003.

Generally, these methods, starting from results of current populations, try to identify a probabilistic model of the search space, and crossover and mutation operators are replaced with sampling. Those methods have been named Estimation of Distribution Algorithms (EDAs).

Most EDAs have been developed to manage optimization processes for mono-objective, combinatorial problems, but several works regarding problems in continuous domains have been proposed.

We can distinguish three types of EDAs depending on the way the probabilistic model is built: a) without dependences among variables [1]; with bivariate dependences among variables [2]; c) with multivariate dependences ([3], [4]).

Recently, EDAs handling multi-objective optimizations have been proposed. References [5] and [6] respectively extend the mono-objective version in [4] and [3]. They describe the algorithms and present some results when applied to well known test problems.

In this paper we propose a multi-objective optimization algorithm for continuous problems that uses the Parzen method to build a probabilistic representation of Pareto solutions, with multivariate dependences among variables.

Similarly to what was done in [6] for multi-objective Bayesian Optimization Algorithm (BOA), the already known and implemented techniques of Non Dominated Sorting Genetic Algorithm II (NSGA-II) [7] are used to classify promising solutions, while new individuals are obtained by sampling from the Parzen model.

The Parzen method, as introduced in the next section, can appear analogous to the normal kernel method described and used in [4]. Actually, the two methods are different and, even if both put kernels on each sampled point, our method uses classical Parzen dictates to set terms of the covariance matrix (non-diagonal) of kernels in order to directly approximate the joint Probability Density Function (PDF).

A brief introduction on the general problem of building probabilistic models is followed by a description of the main characteristics of the Parzen method. In section 3 the structure of the algorithm and the practical implementation are discussed; results of application to test cases are detailed. A final section of concluding remarks summarizes the present work and indicates future developments.

2 Parzen Method

When dealing with continuous-valued random variables, most statistical inferences rely on the estimation of PDFs and/or associated functionals from a finite-sized sample. Whenever something is known in advance about the PDF to be estimated, it is worth exploiting that knowledge as much as we can in order to shape a special-purpose estimator. In fact any additional information we are able to implement in the estimator as a built-in feature is equivalent to some effective increase in the sample size. Otherwise stated, in so doing we improve the estimator's efficiency.

In the statistician's wildest dream some prime principles emerge and dictate that the true PDF must belong to a certain parametric family of model PDFs. This restricts the set of admissible solutions to a finite-dimensional space, and cuts the problem down to the identification of the parameters thereby introduced. In fact parametric estimation is so appealing that few popular families of model PDFs are applied almost everywhere even in lack of any guiding principle, and often little effort is made to check their actual faithfulness. On the other hand, a serious check has to rely on composite hypothesis tests that are like to be computationally very expensive.

While designing an EDA for general-purpose multi-objective optimization there is really no hint on how the true PDF should look like. For instance, that PDF could well have several modes, whereas most popular models are uni-modal. The possible occurrence of multiple modes is usually handled through *mixtures* of uni-modal kernel PDFs. Since the "correct" number of kernels is not known in advance, the size of the mixture is optimized (e.g. by data clustering) just like any other parameter: that is, the weight and the inner parameters of each kernel.

The usage of mixtures does however not alleviate us from worrying about faithfulness. Otherwise stated, the choice of the parametric family the kernels belong to still matters. In fact the overall number of parameters (and therefore the number of kernels) must grow sub-linearly with the sample size n, or else the variance of the resulting estimator would not vanish everywhere as $n \to \infty$, thus precluding ubiquitous converge to the true PDF in the mean square sense. But if that condition is met, then even a single "wrong" kernel can spoil convergence wherever it injects some bias. This is nothing but another form of the well-known bias-variance dilemma.

The Parzen method [8] pursues a non-parametric approach to kernel density estimation. It gives rise to an estimator that converges everywhere to the true PDF in the mean square sense. Should the true PDF be uniformly continuous, the Parzen estimator can also be made uniformly consistent. In short, the method allocates exactly n identical kernels, each one "centered" on a different element of the sample. In contrast with parametric mixtures, here no experimental evidence is spent to identify parameters. This is the reason why the presence of so many kernels does not inflate the asymptotic variance of the estimator. As a consequence, the detailed shape of the kernels is irrelevant, and the faithfulness problem is successfully circumvented. Of course some restrictions are in order: here is a brief explanation.

Let z be a real-valued random variable. Let $p^z(\cdot):\Re \to \Re_+ \cup \{0\}$ be the associated PDF. Let $\boldsymbol{D}_n=\{z_1,..,z_n\}$ be a collection of n independent replicas of z. The empirical estimator $\hat{\boldsymbol{p}}_n^E(\cdot)$ of $p^z(\cdot)$ based on $\boldsymbol{D}_n$ is defined as follows:

$$\forall z \in \Re \quad \hat{\boldsymbol{p}}_n^E(z) = \frac{1}{n}\sum_{i=1}^{n} \delta(z - z_i) \cdot \tag{1}$$

The estimator just defined is unbiased everywhere but it converges nowhere to $p^z(\cdot)$ in the mean square sense because $Var\left[\hat{\boldsymbol{p}}_n^E(z)\right] = \infty$ irrespective of both n and z. This last result is not surprising, since the Dirac's delta is not squared integrable.

The Parzen estimator $\hat{\boldsymbol{p}}_n^S(\cdot)$ of $p^z(\cdot)$ based on $\boldsymbol{D}_n$ is obtained by convolving the empirical estimator with some squared integrable kernel PDF $g_S(\cdot)$:

$$\forall z \in \Re \quad \hat{\boldsymbol{p}}_n^S(z) = \int_{-\infty}^{\infty} \hat{\boldsymbol{p}}_n^E(x) \frac{1}{h_n} g_S\left(\frac{z - x}{h_n}\right) dx = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{h_n} g_S\left(\frac{z - z_i}{h_n}\right) \cdot \tag{2}$$

The kernel acts as a low-pass filter whose "bandwidth" is regulated by the scale factor $h_n \in \Re_+$. It exerts a "smoothing" action that lowers the sensitivity of $\hat{\boldsymbol{p}}_n^S(z)$ w.r.t. $\boldsymbol{D}_n$ so as to make $Var\left[\hat{\boldsymbol{p}}_n^S(z)\right] < \infty$ $\forall z \in \Re$. Thus for any given sample size the larger is the scale factor, the smaller is the variance of the estimator. But the converse is also true: since $\hat{\boldsymbol{p}}_n^S(z)$ is nothing but a mean, then for any given scale factor the larger is the sample size, the smaller is the variance of the estimator (indeed it is inversely proportional to the sample size). Both statements are in fact special cases of the following property:

$$\forall z \in \Re \quad \lim_{n\to\infty} nh_n = \infty \Rightarrow \lim_{n\to\infty} Var\left[\hat{\boldsymbol{p}}_n^S(z)\right] = 0 \ . \tag{3}$$

On the other hand, the same smoothing action produces an unwanted "blurring" effect that limits the resolution of the approximation. Intuitively the scale factor should therefore vanish as $n\to\infty$ in order to let the estimator closely follow finer and finer details of the true PDF. Also this last remark finds a precise mathematical rendering in the following property:

$$\forall z \in \Re \quad \lim_{n\to\infty} h_n = 0 \Rightarrow \lim_{n\to\infty} E\left[\hat{\boldsymbol{p}}_n^S(z)\right] = p^z(z) \ . \tag{4}$$

To summarize, the conflicting constraints dictated by the bias-variance dilemma can still be jointly satisfied by letting the scale factor decrease slowly enough as the sample size grows. The resulting estimator converges everywhere to the true PDF in the mean square sense irrespective of the kernel employed, provided that it is squared integrable.

The above results were later extended to the multi-variate case by Cacoullos [9].

3 Parzen EDA

The main idea of the work is the use of the Parzen method to build a probabilistic model and to sample from the estimated PDF in order to obtain new promising solutions. A detailed description of the Multi-Objective Parzen EDa (MOPED algorithm) follows, and some results are presented in order to show capabilities and potentialities of the algorithm.

Moreover, an extensive use of the Parzen method could lead to simplify the overall optimization procedure towards a parameter-less tool. As a first step in this direction, at the end of section we introduce a different spreading technique for solutions in the Pareto front.

3.1 General Algorithm

As summarized in figure 1, the general optimization procedure can be described as follows:

1. Starting: N_{ind} individuals are sampled from a uniform m-dimensional PDF.
2. Classification & Fitness evaluation: by using NSGA-II techniques [7], individuals of current population are ranked and ordered in terms of dominance criterion and crowding distance in the objective function. A fitness value, linearly varying from 2-α (best individual) to α (worst individual), with $0 < \alpha < 1$, is assigned to each individual.

3. Building model & sampling: on the basis of information given by N_{ind} individuals, by means of the Parzen method a probabilistic model of promising search space portion is built. For generic processes can be useful adopting different kernels alternatively from a generation to the other in order to obtain an effective exploration. In this work Gauss and Cauchy distributions are used. Actually, these types of kernel, for their intrinsic characteristics, are complementary and results will show that the use of only one of them could be inefficient for some problems.

   From the probabilistic model so determined, τN_{ind} new individuals are sampled. Fitness values are used to calculate variance of kernels (the fitness values are related to the scale factors introduced in section 2) and to favor sampling from most important kernels.
4. Evaluation: New τN_{ind} individuals are evaluated in terms of objective functions.
5. Classification & Fitness evaluation: following NSGA-II criteria, individuals of intermediate population, of which dimension is $(1+\tau)\ N_{ind}$, are ordered. A fitness value, linearly varying from $2-\alpha$ (best individual) to α (worst individual), with $0 < \alpha < 1$, is assigned to each individual.
6. New population: best N_{ind} individuals are selected to be next generation.
7. EXIT or NEXT ITER: if convergence criteria are achieved the algorithm stops, otherwise it restarts from point 3.

The algorithm presented above demonstrated satisfactory performance in solving several test cases, when performance is measured in terms of objective function evaluations to obtain a good approximation of the Pareto front. Some results will be shown in the next paragraph.

The still open question is finding an efficient convergence criterion that could be adopted for a generic optimization. That is finding a convergence criterion that guaranties an optimal approximation of Pareto front (efficacy) and requires a number of objective function evaluations as low as possible (efficiency).

Following results show that neither the maximum generation number nor all of the individuals in first class are without gaps. The former because of an extremely low efficiency if a too high maximum number of generations is used, the latter because of premature convergence on a local, non-optimal, front.

Consequently, the maximum generation number is always used. An upper limit for iteration is imposed as suggested from literature results, even if this kind of stopping criterion makes the algorithm inefficient.

3.2 Test Cases Results

In order to have some ideas regarding effectiveness and efficiency of the method, the proposed algorithm has been applied to some well-known test problems taken from literature [10].

For all of test cases 10 independent runs have been performed and results in terms of number of function evaluations are given as average values.

As said in the previous description of the algorithm, in absence of an effective and efficient criterion a maximum number generation criterion has been adopted. In order to allow comparison with obtained results in literature, our results are presented in terms of effective number of iteration, or better, in terms of number of functions evaluations required to obtain the approximation of the optimal front as well.

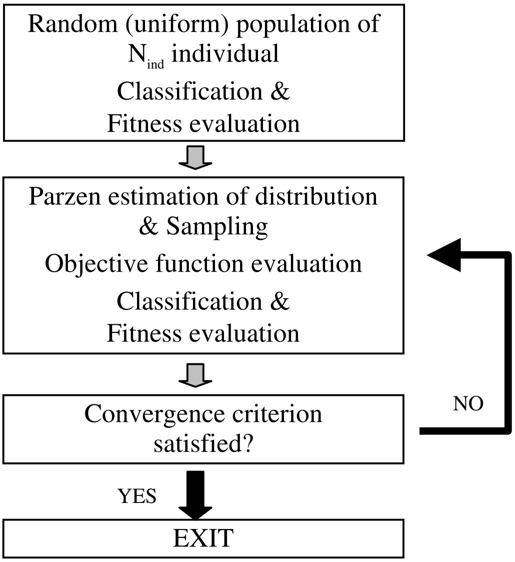

Random (uniform) population of N_{ind} individual
Classification &
Fitness evaluation
Parzen estimation of distribution & Sampling
Objective function evaluation
Classification &
Fitness evaluation
Convergence criterion satisfied?
NO
YES
EXIT

Fig. 1. General structure of the algorithm

All tests have been run with the same values of the following parameters: a) the number of individuals ($N_{ind} = 100$), the sampling parameter ($\tau = 2$), and the fitness parameter ($\alpha = 0.2$).

In figure 2 one of the fronts obtained for the MOP4 problem is shown in the upper left corner. The other three parts of the figure represent the marginal bivariate PDFs of variables when normal kernels are used. The triple structure of the approximated front can be identified from every marginal PDF, even if for this run it is more evident in the x_1 - x_2 PDF.

For this problem a maximum number of iterations is set equal to 55 (11,100 function evaluations), but still in this case in order to have a stable configuration of the solutions a less number of iterations is needed, which is 44.9 (9,090 function evaluations).

Problems EC4 and EC6 are more complex and a presentation of relative results allows a deeper discussion of advantages and gaps of the proposed algorithm.

EC6 is presented as a problem that tests the ability of algorithms to spread solutions on the whole front. MOPED demonstrates to be able to cover the entire optimal range, even if most of runs produce one or two sub-optimal solutions on the left part of the Pareto front (figure 3.a shows one of the fronts). What happens is similar to the results of the Strength Pareto Evolutionary Algorithm (SPEA) when applied to the same problem as reported in [7].

For both EC4 and EC6 we know that achievement of optimal front corresponds to $g(x)$=1. Therefore, for these problems we adopted the following exit criterion: when the $g(x)$ value averaged on the whole population is ≤ 1.01, this allows to have an error less that 1%.

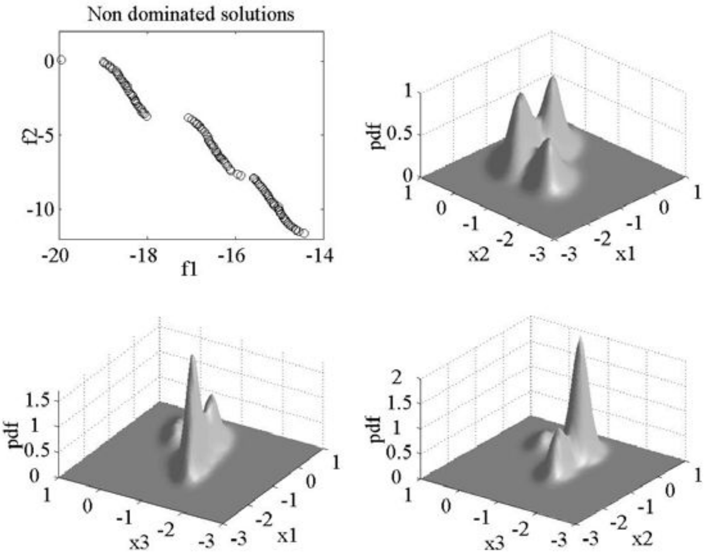


Fig. 2. MOP4 problem. In the left upper corner non-dominated solutions in the objective plain are shown. The other three parts of the figure show the marginal bivariate PDFs of problem's variables when normal kernels are used

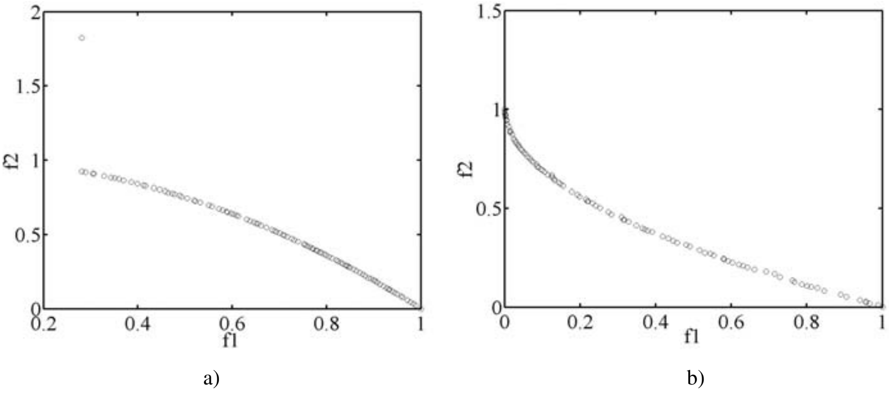


Fig. 3. a) Obtained non-dominated solutions on EC6 problem. Most of obtained fronts display some sub-optimal solutions. - b) Obtained non-dominated solutions on EC4 problem

For EC6 problem we imposed 15,000 maximum function evaluations, but the convergence criterion related to *g(x)* function has been reached after approximately 8,300 evaluations.

Problem EC4 is the hardest in terms of number of function evaluations needed to reach the true optimal front. Because of the form of the functions to optimize, process tends to get stuck on sub-optimal fronts.

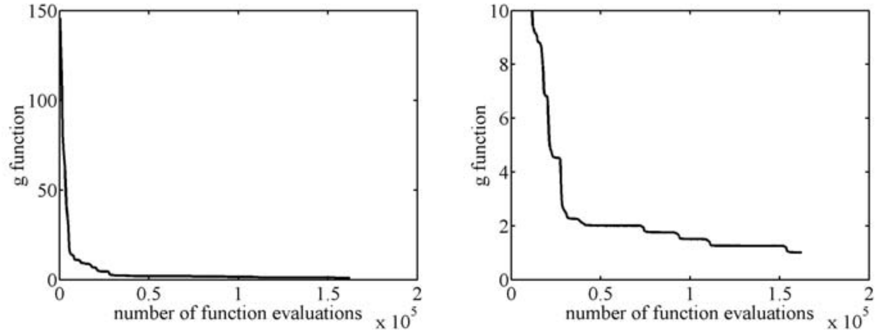


Fig. 4. General trend of *g(x)* function for EC4 problem. On the left the trend during the whole process is shown. On the right side, last part of the process allows a deeper comprehension of difficulties in terms of function evaluation to jump from a local front to a better one

Results demonstrate that the optimal solution (figure 3.b shows one of the fronts) can be obtained after 153,710 function evaluations, with a minimum value of 66,100 in one of the ten runs, and a maximum of 244,100 , when the upper limit of function evaluations is set to 300,000.

From figure 4, which shows a general trend of *g(x)* function, it is possible to see how the process goes on. Ranges with null slope mean a transitorily convergence on a local Pareto-optimal front. In order to allow some comparison we monitored the *g(x)* function and it reaches the value of 3 after 40,862 objective function evaluations.

4 Conclusions

Here we have presented a new estimation of distribution algorithm that is able to manage multi-objective problems following Pareto criterion. The algorithm uses the Parzen method in order to create a non-parametric, model-independent probabilistic representation of promising solutions in the search space. Results obtained when the algorithm is applied to well-known test cases show good performance of the optimization process in terms of the number of objective function evaluations and in the spreading of solutions on the whole front.

Contrary to previous works, in this paper we do not attempt to identify a conditionally independent structure in the genome. We know that this may increase the efficiency of the Parzen estimator. It is our intention to address this important point in the future along a frequentist approach with minor changes in the underlying philosophy. In fact the hypothesis testing inherent in the frequentist approach allows the user to impose a known error of the first kind.

Acknowledgment

This work was partly funded by the MIUR National Program “Sistemi elettronici dedicati basati su algoritmi di apprendimento statistico per l'analisi di dati sperimentali in applicazioni scientifiche ed industriali”.

References

1. Mühlenbein, H., The equation for the response to selection and its use for prediction, Evolutionary Computation 5(3), pp. 303-346, 1998.
2. Pelikan, M., Mühlenbein, H., The bivariate marginal distribution algorithm. In Roy, R., Furuhashi, T., & Chawdhry, P. K. (Eds.), Advances in Soft Computing Engineering Design and Manufacturing, pp. 521-535, London: Springer-Verlag, 1999.
3. Pelikan, M., Goldberg, D. E., and Cant'u-Paz, E. (1999). BOA: The Bayesian optimization algorithm. In Banzhaf, W., Daida, J., Eiben, A. E., Garzon, M. H., Honavar, V., Jakiela, M., & Smith, R. E. (Eds.), Proceedings of the Genetic and Evolutionary Computation Conference GECCO-99, Vol. I, pp. 525-532. Orlando, FL, Morgan Kaufmann Publishers, San Fransisco, CA, 1999.
4. Bosman, P.A.N., Thierens, D., Expanding from discrete to continuous estimation of distribution algorithms: The IDEA, in M. Schoenauer, K. Deb, G. Rudolph, X. Yao, E. Lutton, J.J. Merelo, and H.-P. Schwefel, eds., Parallel Problem Solving from Nature, pp 767-776, Springer, 2000.
5. Thierens, D., Bosman, P.A.N., Multi-Objective Mixture-based Iterated Density Estimation Evolutionary Algorithms L. Spector, E.D. Goodman, A. Wu, W.B. Langdon, H.-M. Voigt, M. Gen, S. Sen, M. Dorigo, S. Pezeshk, M.H. Garzon and E. Burke, editors, Proceedings of the Genetic and Evolutionary Computation Conference - GECCO-2001, pages 663-670, Morgan Kaufmann Publishers, 2001.
6. Khan, N., Goldberg, D.E., and Pelikan, M., Multi-Objective Bayesian Optimization Algorithm, IlliGAL Report No. 2002009, March 2002.
7. Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T., A fast and elitist Multiobjective Genetic Algorithm: NSGA-II, IEEE Transactions on Evolutionary Computation, Vol. 6, No. 2, April 2002.
8. Parzen, E., On Estimation of a Probability Density Function and Mode, Ann. Math. Stat., Vol. 33, pp. 1065-1076, 1962.
9. Cacoullos, T., Estimation of a Multivariate Density, Ann. Inst. Stat. Math., Vol. 18, pp. 179-189, 1966.
10. Deb, K., Multi-Objective Genetic Algorithm: Problem Difficulties and Construction of Test Problems, Evolutionary Computation, Vol. 7, No. 3, pp. 205-230, The MIT Press, 1999.

An Algorithm to Model Paradigm Shifting in Fuzzy Clustering

Francesco Masulli[1,2] and Stefano Rovetta[1,3]

[1] INFM, Istituto Nazionale per la Fisica della Materia, 16146 Genova, Italy
[2] Dipartimento di Informatica, Università di Pisa, 56125 Pisa, Italy
masulli@di.unipi.it
[3] DISI, Università di Genova, 16146 Genova, Italy
rovetta@disi.unige.it

Abstract. The *graded possibilistic clustering paradigm* includes as the two extreme cases the "probabilistic" assumption and the "possibilistic" assumption adopted by many clustering algorithms. We propose an implementation of a graded possibilistic clustering algorithm based on an interval equality constraint enforcing both the normality condition and the required graded possibilistic condition. Experimental results highlight the different properties attainable through appropriate implementation of a suitable graded possibilistic model.

1 Introduction

In the clustering problem setting, on the basis of a finite set of unlabeled samples $X = \{\mathbf{x}_k|\ k = 1, ..., n\}$, we must estimate the cluster centers (or prototypes) $Y = \{\mathbf{y}_j | j = 1, ..., c\}$, and the matrix of the memberships of samples to clusters $U = [u_{jk}]$.

Many clustering algorithms, such as C-Means (CM) [3], Fuzzy C-Means (FCM) [2], and Deterministic Annealing (DA) [9,1], follow a *probabilistic approach*, according to which the sum of the membership values of a point to all the clusters must be equal to one, i.e., the "probabilistic constraint" $\sum_{j=1}^{c} u_{jk} = 1$. In this paradigm, each membership is therefore formally equivalent to the probability that an experimental outcome coincides with one of c mutually exclusive events.

In [5,6], Krishnapuram and Keller showed the limits of the probabilistic approach to clustering and proposed a *possibilistic approach* to it. Their approach assumes the membership function of a point in a *fuzzy* set (or cluster) is absolute, i.e. it is an evaluation of a *degree of typicality* not depending on the membership values of the same point in other clusters.

Krishnapuram and Keller [5,6] presented two version of a Possibilistic C-Means algorithm (PCM) that relax the probabilistic constraint, in order to allow a *possibilistic* interpretation of the membership function as a *degree of typicality.* In PCM, the elements of U fulfill the following conditions:

$$u_{jk} \in [0, 1] \quad \forall \quad j, k; \tag{1}$$

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 70–76, 2003.

$$0 < \sum_{k=1}^{n} u_{jk} < n \ \ \forall \ \ j; \tag{2}$$

$$\max_{j} \quad u_{jk} > 0 \ \ \forall \ \ k. \tag{3}$$

Then, the possibilistic approach implies that each membership is formally equivalent to the probability that an experimental outcome coincides with one of c mutually *independent* events. This is due to the complete absence of a constraint on the set of membership values ($\psi \equiv 0$).

Note that, due to lack of competitiveness among clusters, clustering algorithms based on the possibilistic approach, need of an initial distribution of prototypes in the feature space and the estimation of some parameters, that can be obtained using a probabilistic clustering methods. E.g., in [5,6], a Fuzzy C-Means initialization has been applied, while Masulli and Schenone [8] used a prototypes initialization based on the Capture Effect Neural Network (CENN) [4].

However, it is possible (and in practice it is frequent) that pairs of events are not mutually independent, but are not completely mutually exclusive either. Instead, events can provide *partial information* about other events. Of course, this is a problem-dependent situation and accounting for it may or may not be appropriate.

An interesting case of partial information, in the context of the present research, is the concept of *graded possibility.* The standard possibilistic approach to clustering implies that all membership values are independent. In contrast, the graded possibilistic model assumes that, when one of the c membership values is fixed, the other $c - 1$ values are constrained into a subset of the interval $[0, 1]$.

Clearly, this situation includes the possibilistic model, and also encompasses the standard ("probabilistic") approach.

An example of such graded possibility is given by a glass and by the fuzzy concepts of "full" and "empty". If the glass is full or almost full, its membership to the concept "empty" should clearly be around zero, and similarly for the empty or almost empty case. However, if the glass is half filled, it is much more difficult to assess the membership in the concept "empty" with similar confidence. The profile of the membership functions in this case should be decided according to further considerations.

In short, in these intermediate cases the membership function should not be constrained by the cost function, but should be arbitrary to a certain degree.

2 Modeling Graded Possibility

A class of constraints $\psi(U) = 0$, which includes the probabilistic and the possibilistic cases, can be expressed by the following unified formulation:

$$\psi(U) = \sum_{j=1}^{c} u_{jk}^{[\xi]} - 1, \tag{4}$$

where $[\xi]$ is an interval variable representing an arbitrary real number included in the range $[\,\underline{\xi}, \overline{\xi}\,]$. This interval equality should be interpreted as follows: there must exist a scalar exponent $\xi^* \in [\,\underline{\xi}, \overline{\xi}\,]$ such that the equality $\psi = 0$ holds.

This constraint enforces both the normality condition and the required probabilistic or possibilistic constraints; in addition, for nontrivial finite intervals $[\xi]$, it implements the required graded possibilistic condition.

The constraint presented above can be implemented in various ways. A particular implementation is as follows: the extrema of the interval are written as a function of a running parameter α, where

$$\underline{\xi} = \alpha \qquad \overline{\xi} = \frac{1}{\alpha} \tag{5}$$

and

$$\alpha \in [0, 1] \tag{6}$$

This formulation includes as the two extreme cases:

- The "probabilistic" assumption:

$$\alpha = 1$$
$$[\xi] = [1, 1] = 1$$
$$\sum_{j=1}^{c} u_{jk} = 1$$

- The "possibilistic" assumption:

$$\alpha = 0$$
$$[\xi] = [0, \infty]$$
$$\sum_{j=1}^{c} u_{jk}^{0} \geq 1 \qquad \sum_{j=1}^{c} u_{jk}^{\infty} \leq 1$$

The latter case can be better understood as the limit of the process of bringing $\alpha \to 0$. The interval exponent $[\xi]$ expands, so that the actual value can be any arbitrary number between α and $1/\alpha$. Therefore, each equation containing an interval is equivalent to a set of two inequalities:

$$\sum_{j=1}^{c} u_{jk}^{\alpha} \geq 1 \qquad \sum_{j=1}^{c} u_{jk}^{1/\alpha} \leq 1.$$

This is graphically depicted in Figure 1, where the bounds of the feasible regions are plotted, for $c = 2$, for values of α which decrease in the direction of the arrows.

In the first limit case, the feasible values for u_{jk} must lie on a one-dimensional set (a line segment). In the second limit case, the feasible values for u_{jk} are in the unity square, a two-dimensional set. In intermediate cases, the feasible values are on two-dimensional sets which however do not fill the whole square, but are limited to an eye-shaped area around the line segment.

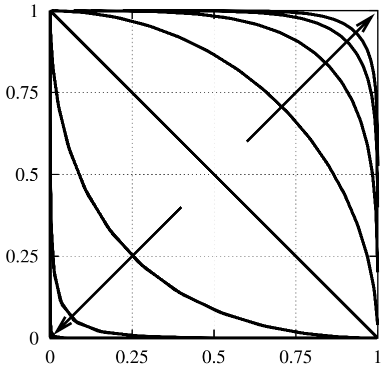


Fig. 1. Bounds of the feasible region for u_{jk} for different values of α (decreasing from 1 to 0 along the direction of the arrows)

3 The Graded Possibilistic Clustering Algorithm

In Tab 1, we depict an elementary example of the graded possibilistic clustering algorithm, based on the application of the ideas in the previous section. Note that, the stopping criterion can be selected as usual in C-Means clustering family of algorithms, e.g,, when after the iteration no centroids move more than an assigned threshold.

The proposed example is an application of the ideas in the previous section. However, it is possible to apply many variations to this algorithm, so that appropriate properties can be obtained.

For the proposed algorithm implementations, the free membership function has been selected as in the DA and PCM-II algorithms:

$$v_{jk} = e^{-d_{jk}/\beta_j} . \tag{7}$$

The generalized partition function can be defined as follows:

$$Z_k = \sum_{j=1}^{c} v_{jk}^{\kappa} \tag{8}$$

where:

$$\begin{array}{lll} \kappa = 1/\alpha & \text{if} & \sum_{j=1}^{c} v_{jk}^{1/\alpha} > 1 \\ \kappa = \alpha & \text{if} & \sum_{j=1}^{c} v_{jk}^{\alpha} < 1 \\ \kappa = 1 & & \text{else.} \end{array}$$

These definitions ensure that, for $\alpha = 1$, the algorithm reduces to standard DA, whereas in the limit case for $\alpha = 0$, the algorithm is equivalent to PCM-II. Note that in the implementation of the algorithm in Tab 1 the variation of α from 1 to 0 allows to obtain a probabilistic initialization of prototypes and a following refinement in a possibilistic sense.

Table 1. Basic Graded possibilistic clustering algorithm

```
select c
select alphastep $\in \mathbb{R}$
select stopping criterion (see text)
randomly initialize $y_j$
for $\alpha = 1$ down to $0$ by alphastep do
begin
    compute $v_{jk}$ using (7)
    compute $Z_k$ using (8)
    compute $u_{jk} = v_{jk}/Z_k$
    if stopping criterion satisfied  then  stop
    else compute the centroids $y_j$

end
```

The required value for the β_j can be assessed from previous experiments, possibly in an independent way for each cluster (as done in PCM), or gradually lowered in an iterated application of the algorithm (as done in DA).

4 Experimental Analysis

In [7] we report some results aimed to highlighting the properties attainable through appropriate implementation of a suitable graded possibilistic model. The showed results demonstrated that:

1. the proposed implementation of the graded possibilistic model (Tab. 1) is able to correctly model the membership functions of data point without need of long experimental work, as necessary with the PCM, and
2. a very high outliers rejection is attainable, by setting the upper extremum of $[\xi]$ to 1 and the lower extremum to α.

In this section we illustrate a case of a-priori knowledge usage. We propose an experimental demonstration where we make use of a suitable value for α to improve the results with respect to the extreme cases (probabilistic and pure possibilistic). In this case the optimum value is inferred from the results but not used (for lack of a test set); in real applications it can be estimated on the training set prior to use on new data.

We show sample results from the following unsupervised classification experiment. First, the graded possibilistic clustering procedure was applied to the Iris data set. Only one cluster center per class was used ($c = 3$). Then the cluster memberships were "defuzzified" by setting the maximum to 1 and the other two to 0. Subsequently, the hard memberships were used to associate class labels to each cluster (by majority). Finally, the classification error was evaluated. The classification error percentages as a function of α are shown in Figure 2.

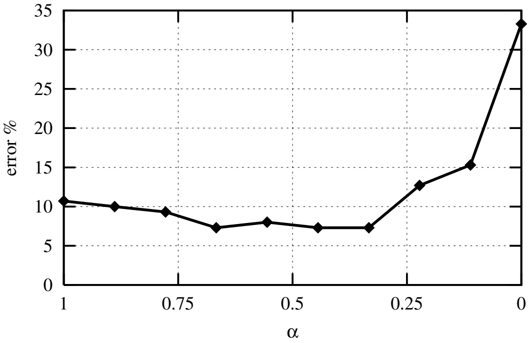


Fig. 2. Error percentage plot for the unsupervised Iris classification

Although these are only a sample of the results, which may have been different in other runs, the profile of the graph was qualitatively almost constant in all trials. The best classification performance with $c = 3$ was 7.3% error, which means 11 mistaken points.

In all experiments this value was obtained for *intermediate* values of α, between 0.3 and 0.7. In other words, the graded possibilistic model was able to catch the true distributions of data better than either the probabilistic or the possibilistic approaches. The pure possibilistic case gave rise (as in the results presented in the figure) to a percentage of cases with overlapping cluster centers, in accordance with previous experimental observations [6].

The error levels can be categorized into three classes. The first is around the optimum (11 or 12 or occasionally 13 wrong classifications). The second, sometimes observed in the pure possibilistic case, is the case of overlapping clusters, with about 33% error rate. The third, above 10%, is typical of the probabilistic case, where competition among clusters does not allows optimal placement of the cluster centers.

5 Conclusions

The concept of graded possibility applied to clustering, which has been presented in this paper, is a flexible tool for knowledge representation. By tuning the level of possibility it is possible to represent overlapped clusters, as in standard possibilistic clustering, with the added capability to adapt the level of overlap to the problem at hand. This results in interesting rejection capabilities and in an adaptable trade-off between the mode-seeking and the partitioning behaviors of its two special cases – possibilistic and standard (probabilistic) fuzzy clustering.

Our current activities involve the application of this flexible behavior in the areas of Web content analysis, document data mining, DNA microarray data analysis. Deeper theoretical investigations are planned as well.

References

1. G. Beni and X. Liu. A least biased fuzzy clustering method. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 16:954–960, 1994.
2. James C. Bezdek. *Pattern recognition with fuzzy objective function algorithms.* Plenum, New York, 1981.
3. Richard O. Duda and Peter E. Hart. *Pattern Classification and Scene Analysis.* John Wiley and Sons, 1973.
4. F. Firenze and P. Morasso. The capture effect model: a new approach to self-organized clustering. In *Sixth International Conference. Neural Networks and their Industrial and Cognitive Applications. NEURO-NIMES 93 Conference Proceedings and Exhibition Catalog*, pages 65–54, Nimes, France, 1993.
5. Raghu Krishnapuram and James M. Keller. A possibilistic approach to clustering. *IEEE Transactions on Fuzzy Systems*, 1(2):98–110, May 1993.
6. Raghu Krishnapuram and James M. Keller. The possibilistic C-Means algorithm: insights and recommendations. *IEEE Transactions on Fuzzy Systems*, 4(3):385–393, August 1996.
7. F. Masulli and S. Rovetta. Soft transition from probabilistic to possibilistic fuzzy clustering. Technical Report DISI-TR-02-03, Dipartimento di Informatica e Scienze dell'Informazione, Università di Genova (Italy), Italy, 2002.
(*http://www.disi.unige.it/person/MasulliF/papers/DISI-TR-02-03-masulli.pdf*).
8. F. Masulli and A. Schenone. A fuzzy clustering based segmentation system as support to diagnosis in medical imaging. *Artificial Intelligence in Medicine*, 16:129–147, 1999.
9. Kenneth Rose, Eitan Gurewitz, and Geoffrey Fox. Statistical mechanics and phase transitions in clustering. *Physical Review Letters*, 65:945–948, 1990.

ANFIS Synthesis by Hyperplane Clustering for Time Series Prediction

Massimo Panella, Fabio Massimo Frattale Mascioli,
Antonello Rizzi, and Giuseppe Martinelli

University of Rome "La Sapienza", Via Eudossiana 18, I-00184 Rome, Italy
panella@infocom.uniroma1.it
http://infocom.uniroma1.it/~panella

Abstract. Time series prediction can be considered as a function approximation problem whose inputs are determined by using past samples of the sequence to be predicted. ANFIS networks are neural models particularly suited to the solution of such problems, which usually require a data driven estimation technique. In this context, clustering procedures represent a straightforward approach to the synthesis of ANFIS networks. A novel use of clustering, working in the joint input-output data space, is proposed in the paper. It is intended to improve the ANFIS approximation accuracy by directly estimating the hyperplanes associated with the consequent parts of Sugeno first-order rules. Simulation tests and comparisons with other prediction techniques are discussed to validate the proposed synthesis approach. In particular, we consider the prediction of environmental data sequences, which are often characterized by a chaotic behaviour.

1 Introduction

The prediction of future values of real-world data sequences is often mandatory to the cost-effective management of available resources. Consequently, many worldwide research activities are intended to improve the accuracy of well-known prediction models. Among them, an important role can be played by Adaptive Neuro-Fuzzy Inference Systems (ANFIS) [1]. In fact, as it will be pointed out in the paper, such computational models can be used as predictors by means of a suitable transformation of the prediction problem into a function approximation one.

An ANFIS network performs the approximation of an unknown mapping $y = f(\underline{x})$, $f : \mathrm{I\!R}^N \to \mathrm{I\!R}$, by implementing a fuzzy inference system constituted by M rules of Sugeno first-order type. The k-th rule, $k = 1 \ldots M$, has the form:

$$\text{If } x_1 \text{ is } B_1^{(k)}, \ldots, \text{ and } x_N \text{ is } B_N^{(k)} \text{ then } y^{(k)} = \sum_{j=1}^{N} a_j^{(k)} x_j + a_0^{(k)} \ , \qquad (1)$$

where $\underline{x} = \left[x_1 \ x_2 \cdots x_N\right]$ is the input pattern and $y^{(k)}$ is the output associated with the rule. The antecedent part of the rule is characterized by the membership

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 77–84, 2003.

functions (MFs) $\mu_{B_j^{(k)}}(x_j)$ of the fuzzy input variables $B_j^{(k)}$, $j = 1 \ldots N$; the consequent part is characterized by the coefficients $a_j^{(k)}$, $j = 0 \ldots N$, of the crisp output $y^{(k)}$. Several alternatives are possible for choosing the fuzzification type of crisp inputs, the composition of input MFs, and the way rule outputs are combined [1]. By using the commonly adopted options in this regard, the overall ANFIS output will be obtained by the following approximation model:

$$\widetilde{y} = \frac{\sum\limits_{k=1}^{M} \mu_{\underline{B}^{(k)}}(\underline{x})\ y^{(k)}}{\sum\limits_{k=1}^{M} \mu_{\underline{B}^{(k)}}(\underline{x})} \,, \tag{2}$$

where $\widetilde{y}$ is the estimated output of the actual value $y = f(\underline{x})$, and $\mu_{\underline{B}^{(k)}}(\underline{x})$ is the overall input MF of the k-th rule, which can be obtained either by a direct estimation procedure or by the composition of the corresponding input MFs, i.e. $\mu_{B_j^{(k)}}(x_j)$, $j = 1 \ldots N$.

When dealing with data driven estimation procedures, the mapping $y = f(\underline{x})$ is known by means of numerical examples, i.e. by a training set of P input-output pairs $\{\underline{x}_i, y_i\}$, $i = 1 \ldots P$. In this case, a useful approach to the synthesis of ANFIS networks is based on clustering the training set. Different types of clustering approaches can be used in this regard [2,3]. For instance, clustering in the joint input-output space can overcome some drawbacks pertaining most of traditional approaches, where clustering is used only to determine the rule antecedents in the input space [4].

We present in Sect. 2 an ANFIS synthesis procedure that is based on the joint input-output space approach. Firstly, we determine the coefficients of the Sugeno rule consequent by using a suitable clustering procedure. Then, we use a fuzzy classifier in order to determine the fuzzy variables of Sugeno rule antecedents. Fuzzy Min-Max classifiers represent a valid solution in this regard, because of their computational effectiveness [5]. In particular, the use of Min-Max classifiers based on adaptive resolution mechanisms [6] will be adopted.

The generalization capability of an ANFIS network can be satisfactory, provided that its architecture consists of a suitable number of rules. This is a crucial problem since, in general, neuro-fuzzy networks can be easily overfitted in case of noisy or ill-conditioned data. In order to determine automatically the optimal number of rules, we also suggest in Sect. 2 an optimisation of the proposed synthesis procedure based on well-known concepts of learning theory.

As previously mentioned, the use of ANFIS networks can result very effective also in several prediction problems. In fact, the latter can be considered as function approximation problems whose inputs are suitably determined by using past samples of the sequence to be predicted. We illustrate in Sect. 3 a standard technique, which is particularly suited to the prediction of real-world data sequences that often manifest chaotic behaviours.

The validity of the proposed ANFIS synthesis procedure is ascertained in Sect. 4, where several benchmark results, obtained by using environmental data sequences, are presented.

2 The Hyperplane Clustering Synthesis Procedure

The use of clustering in ANFIS synthesis reduces the redundancy in the data space(s), allowing the determination of significant rules directly from the clusters that have been discovered in the available data set. Each rule corresponds to a structured set of points in these spaces. Three different types of data spaces can be used for clustering: input, output, and joint input-output [3,4].

Traditional techniques, either based on input or output space, critically depend on the regularity properties of the mapping to be approximated. In fact, such techniques implicitly assume that close points in the input space (belonging to the same rule) are mapped into close points in the output space. Unfortunately, this not a usual situation, especially in forecasting real-world data sequences where mappings to be approximated might present several irregularities and sharp behaviours.

2.1 Hyperplane Clustering in the Joint Input-Output Space

In order to overcome the problems of the previous approaches, we illustrate in the following the use of a clustering strategy based on a joint input-output space [4]. Such a clustering procedure is oriented to the direct determination of the structure of the unknown mapping $f(\underline{x})$. Namely, the ANFIS architecture can be considered as a piecewise linear regression model, where $f(\underline{x})$ is approximated by a suitable set of M hyperplanes, each related to an input-output cluster. Therefore, the prototype of the k-th cluster, $k = 1 \ldots M$, in the joint input-output space will be represented by the coefficients $a_j^{(k)}$, $j = 0 \ldots N$, which determine the linear consequent of the corresponding k-th rule. We propose an alternating optimisation technique, i.e. a C-Means clustering in the 'hyperplane space', in order to determine such prototypes:

- <u>*Initialisation*</u>. Given a value of M, the coefficients of each hyperplane are initialised by following a suitable criterion. In this case, we choose a simple random initialisation of them. Successively, each pair $\{\underline{x}_i, y_i\}$, $i = 1 \ldots P$, of the training set is assigned to the hyperplane A_q, $1 \leq q \leq M$, based on the procedure discussed in Step 2.
- <u>*Step 1*</u>. The coefficients of each hyperplane are updated by using the pairs assigned to it (either in the successive step 2 or in initialisation). For the k-th hyperplane, $k = 1 \ldots M$, a set of linear equations has to be solved:

$$y_t = \sum_{j=1}^{N} a_j^{(k)} x_{tj} + a_0^{(k)} \ , \tag{3}$$

  where index 't' spans all pairs assigned to the k-th hyperplane. Any least-squares technique can be used to solve the previous set of linear equations.

- *Step 2.* Each pair $\{\underline{x}_i, y_i\}$ of the training set is assigned to the updated hyperplane A_q, with q such that:

$$e_i = \left| y_i - \left(\sum_{j=1}^{N} a_j^{(q)} x_{ij} + a_0^{(q)} \right) \right| = \min_{k=1\ldots M} \left| y_i - \left(\sum_{j=1}^{N} a_j^{(k)} x_{ij} + a_0^{(k)} \right) \right| . \quad (4)$$

- *Stop criterion.* If the overall error defined by

$$E = \frac{1}{P} \sum_{i=1}^{P} e_i \quad (5)$$

has converged then stop, else go to step 1.

2.2 Determination of the Antecedent Part of Sugeno Fuzzy Rules

The previous procedure only yields the linear consequent of Sugeno rules. However, the set of patterns associated with a hyperplane may overlap in the input space with the set of patterns associated with a different hyperplane. This means that each hyperplane may correspond to several clusters in the input space, i.e. to well separated sets of input patterns $\underline{x}_t$ associated with it. Consequently, the determination of the input MFs is not straightforward. This problem can be solved by considering a suitable classification problem, where each pattern is labelled with an integer q, $1 \leq q \leq M$, representing the hyperplane to which it has been previously assigned. Any fuzzy classification algorithm can be used in this regard. The use of the Adaptive Resolution Classifier (ARC), belonging to the class of well-known Simpons's Min-Max models, is herein proposed.

Min-Max classification technique was proposed by Simpson in [5]; it consists in covering the patterns of the training set with hyperboxes (HBs). It is possible to establish size and position of each HB by two extreme points: the 'Min' and 'Max' vertices. The hyperbox can be considered as a crisp frame on which different types of membership functions can be adapted. In the following, we will adopt the original Simpson's membership function, in which the slope outside the hyperbox is established by the value of a fuzziness parameter γ. We will use in the following the default value $\gamma = 1$.

As shown in [6], ARC technique outperforms the original Min-Max algorithm, and some optimised versions of it, both in the generalization capability and in the training time. Moreover, Simpson's classifiers trained by ARC are independent of the pattern presentation order. Therefore, on the basis of these results, we will adopt the ARC algorithm in order to find the input MFs of the ANFIS network.

As previously stated, several HBs of the ARC classifier can be associated with the same hyperplane (rule) determined by the hyperplane clustering. In this case, the overall input MF $\mu_{\underline{B}^{(k)}}(\underline{x})$, $k = 1 \ldots M$, of each rule can be determined on the basis of the composition operators usually adopted for fuzzy Min-Max neural networks [5]. For instance, if $H_1^{(q)}$, $H_2^{(q)}$,..., $H_R^{(q)}$ are the HBs associated with the class label q, and $\mu_1^{(q)}(\underline{x})$, $\mu_2^{(q)}(\underline{x})$,..., $\mu_R^{(q)}(\underline{x})$ the corresponding MFs, then we will have:

$$\mu_{\underline{B}^{(q)}}(\underline{x}) = \max\left\{\mu_1^{(q)}(\underline{x}),\, \mu_2^{(q)}(\underline{x}),\, \dots,\, \mu_R^{(q)}(\underline{x})\right\} . \tag{6}$$

The combination of both the hyperplane clustering and the ARC Min-Max classification in the input space, for a given value of M, allows the determination of the ANFIS network. This procedure will be denoted in the following as Hyperplane Clustering Synthesis (HCS) algorithm.

2.3 Structural Optimisation of ANFIS Networks

The HCS algorithm, and the resulting ANFIS network, depends upon a given value of M. The optimal value of M, yielding the best performing network in terms of generalization capability (i.e. lowest error on a test set), should be accomplished during training without any knowledge about the test set. We proposed in [4] a constructive technique, denoted as Optimised HCS (OHCS), which is the most general optimisation technique that can be pursued in this context. The value of M is progressively increased and several ANFIS networks are generated, by using the HCS algorithm, in correspondence to any value of M. In fact, HCS uses a random initialisation of the hyperplanes coefficients, hence different initialisations can yield different networks for the same value of M.

Successively, the optimal value of M is chosen by relying on basic concepts of learning theory [7], i.e. by finding the minimum value of the following cost functional:

$$F(M, \underline{A}_{\mathrm{o}}) = (1-\lambda)\frac{E(M, \underline{A}_{\mathrm{o}}) - E_{min}}{E_{max} - E_{min}} + \lambda\frac{M}{P} , \tag{7}$$

where $E(M, \underline{A}_{\mathrm{o}})$ is the training set performance (i.e. the approximation error on the training set) obtained for a given M and for a given initialisation $\underline{A}_{\mathrm{o}}$; E_{min} and E_{max} are, respectively, the minimum and maximum value of E, obtained during the investigation of the different M and $\underline{A}_{\mathrm{o}}$; λ is a weight in the range $[0, 1]$. This weight is not critical since the results are slightly affected by the variation of it in a large interval centred in 0.5.

3 The Function Approximation Approach to Time Series Prediction

Due to the actual importance of prediction, the technical literature is plenty of proposed methods for implementing a predictor, especially in the field of neural networks [8]. The general approach to solve a prediction problem is based on the solution of a suitable function approximation problem. Let us consider a general predictor, which is used to predict a sampled sequence $S(t)$ and is obtained as the solution of a function approximation problem $y = f(\underline{x})$, $f : \mathrm{I\!R}^N \rightarrow \mathrm{I\!R}$. For example, the simplest approach is based on linear models: each input vector $\underline{x}_t$

is constituted by N consecutive samples of $S(t)$ and the target output y_t is the sample to be predicted at distance m; i.e.

$$\underline{x}_t = \left[S(t)\ S(t-1) \cdots S(t-N+1)\right] , \ \ y_t = S(t+m) ,$$
$$f_{lin}(\underline{x}_t) = -\sum_{j=1}^{N} \lambda_j x_{tj} \quad \Rightarrow \quad \widetilde{S}(t+m) = -\sum_{j=1}^{N} \lambda_j S(t-j+1) , \tag{8}$$

where $\widetilde{S}(t+m)$ denotes the estimate of the actual value $S(t+m)$.

Usually, the way to determine the input vectors $\underline{x}_t$, based on past samples of $S(t)$, is called 'embedding technique'. The function $f_{lin}(\cdot)$, i.e. the coefficients λ_j, $j = 1 \ldots N$, can be determined in this case by relying on global statistical properties of the sequence $S(t)$, i.e. on its autocorrelation function.

Real-world data sequences often posses a chaotic behaviour that is typical for almost all real-world systems. The performance of a predictor depends on how accurate it models the unknown context delivering the sequence to be predicted. Unfortunately, when dealing with chaotic sequences, the previous linear predictor would hardly fail since it is based on a linear approximation model and on trivial embedding technique. Consequently, more care should be taken on choosing both the approximation model and the embedding parameters [9].

In the case of a chaotic sequence $S(t)$, the latter can be considered as the output of a chaotic system that is observable only through $S(t)$. Consequently, the sequence $S(t)$ should be embedded in order to reconstruct the state-space evolution of this system, where $f(\cdot)$ is the function that approximates the relationship between the reconstructed state (i.e. $\underline{x}_t$) and its corresponding output (i.e. the value y_t to be predicted) [10]. Because of the intrinsic non-linearity and non-stationarity of a chaotic system, $f(\cdot)$ should be a non-linear function, which can be determined only by using data driven techniques.

The usual embedding technique, which is useful for chaotic sequences, is based on the determination of both the embedding dimension D of the reconstructed state-space attractor and the time lag T between the embedded past samples of $S(t)$; i.e.:

$$\underline{x}_t = \left[S(t)\ S(t-T)\ S(t-2T) \cdots S(t-(D-1)T)\right] . \tag{9}$$

Both the values of D and T will be determined in the following by applying the methods suggested in [10]. In particular, D will be obtained by using the False Nearest Neighbours (FNN) method, whereas T will be obtained by using the Average Mutual Information (AMI) method.

From the above discussion, it is evident how the implementation of a predictor will coincide with the data driven determination of a non-linear function approximation model. For this purpose, we propose in this paper the use of ANFIS networks trained by the OHCS algorithm. In fact, we will prove in the next section how the robustness of fuzzy logic, together with the structural regularization of the OHCS procedure, provides to ANFIS networks the accuracy and the flexibility in solving those prediction problems where the mappings to be approximated are often characterized by very irregular behaviours.

4 Illustrative Tests

The prediction performances of ANFIS networks, resulting from the proposed OHCS procedure, have been carefully investigated by the several simulation tests we carried out in this regard. Two different predictors are also considered: the linear predictor introduced in (8) and determined by a standard least-squares technique; another ANFIS predictor, whose network is generated in this case by applying the 'subtractive clustering' method for rule extraction [11] and then a least-squares method together with the back-propagation gradient [1].

All the previous computational models are trained on the first 2000 samples of the time series. These samples are also applied to both the AMI and FNN methods, in order to compute the embedding dimension and the time lag. The OHCS procedure is run from $M = 1$ to $M = 200$, i.e. about 10% of the training set cardinality. This value is considered as the maximum complexity allowed to the network. Ten different initialisations are carried out for each value of M. The performances of the resulting predictors are tested on the successive 600 samples of the sequence (slight changes may occur in these numbers because of the different embedding quantities used for each sequence). The performance is measured by the Normalized Mean Squared Error (NMSE) defined as the ratio between the mean squared prediction error and the variance of the sequence to be predicted.

We illustrate in the following the results concerning well-known chaotic prediction benchmarks as the 'Ikeda', 'Henon' and 'Mackey-Glass' time series. Furthermore, we consider three environmental data sequences, which have been obtained from the observation of some pollution indicators of the downtown of Rome (Italy) and are characterized by a chaotic behaviour. The first two sequences are relevant to the level of Ozone (in μg) and of acoustic noise (in Db), both obtained by using a sampling rate of 5 minutes. The third sequence is the level of the electric power consumption in (MW); its values are obtained by using a sampling rate of 1 hour. The results are summarized in Tab. 1: the first three columns indicate, respectively, the time lag (T), the embedding dimension (D), and the number of ANFIS rules (M) determined by the OHCS procedure; the successive three columns show the NMSE of the test set, obtained by using the predictors under investigation.

Table 1. Prediction results (NMSE) on chaotic benchmarks and real data sequences

Test	T	D	M	OHCS	Linear	ANFIS
Ikeda	5	3	10	$7.01 \cdot 10^{-1}$	$1.50 \cdot 10^{0}$	$7.28 \cdot 10^{-1}$
Henon	17	1	11	$7.65 \cdot 10^{-2}$	$2.30 \cdot 10^{0}$	$7.80 \cdot 10^{-2}$
Mackey-Glass	11	3	15	$2.98 \cdot 10^{-3}$	$1.32 \cdot 10^{-2}$	$3.24 \cdot 10^{-3}$
Ozone	3	5	16	$1.72 \cdot 10^{-1}$	$2.43 \cdot 10^{-1}$	$2.19 \cdot 10^{-1}$
Acoustic noise	4	14	9	$3.47 \cdot 10^{-1}$	$4.39 \cdot 10^{-1}$	$3.04 \cdot 10^{-1}$
Electric load	7	5	13	$9.93 \cdot 10^{-3}$	$4.95 \cdot 10^{-2}$	$3.45 \cdot 10^{-2}$

5 Conclusions

In this paper, we propose the use of ANFIS networks to solve prediction problems in connection with well-known techniques used for chaotic system modelling. The ANFIS training algorithm is based on the HCS procedure, which uses a joint input-output data clustering and a fuzzy Min-Max classification on the input data space. Successively, the ANFIS network is determined by using the OHCS optimisation procedure, so that the optimal number of rules, and therefore the best generalization capability of the network, is automatically achieved.

As evidenced by the prediction performances on real-world data sequences, the OHCS procedure is particularly suited to critical forecasting applications (industrial and environmental tasks, for example) where even slight improvements of the prediction accuracy might result in a more effective management of the human and economic available resources.

References

1. Jang, J.S., Sun, C.T., Mizutani, E.: Neuro-Fuzzy and Soft Computing: a Computational Approach to Learning and Machine Intelligence. Prentice-Hall, Upper Saddle River, NJ (1997)
2. Frattale Mascioli, F.M., Mancini, A., Rizzi, A., Panella, M., Martinelli, G.: Neuro-fuzzy Approximator based on Mamdani's Model. Proc. of WIRN Vietri–01, Vietri Sul Mare, Salerno, Italy (2001) 23–59
3. Guillaume, S.: Designing Fuzzy Inference Systems from Data: an Interpretability Oriented Review. IEEE Transactions on Fuzzy Systems, **9** (2001) 426–443
4. Panella, M., Rizzi, A., Frattale Mascioli, F.M., Martinelli, G.: ANFIS Synthesis by Hyperplane Clustering. Proc. of IFSA/NAFIPS 2001, Vancouver, Canada (2001) 340–345
5. Simpson, P.K.: Fuzzy Min-Max Neural Networks Part 1: Classification. IEEE Transactions on Neural Networks **3** (1992) 776–786
6. Rizzi, A., Panella, M., Frattale Mascioli, F.M.: Adaptive Resolution Min-Max Classifiers. IEEE Transactions on Neural Networks **13** (2002) 402–414
7. Haykin, S.: Neural Networks, A Comprehensive Foundation, 2nd Edition. Prentice-Hall, Englewood Cliffs, NJ (1999)
8. Masulli, F., Studer, L.: Time Series Forecasting and Neural Networks. Proc. of IJCNN'99, Washington D.C., USA (1999)
9. Panella, M., Rizzi, A., Frattale Mascioli, F.M., Martinelli, G.: Constructive MoG Neural Networks for Pollution Data Forecasting. Proc. of IJCNN 2002, Honolulu, Hawaii, U.S.A. (2002) 417–422.
10. Abarbanel, H.D.I.: Analysis of Observed Chaotic Data. Springer, New York (1996)
11. Chiu, S.: Fuzzy Model Identification Based on Cluster Estimation. Journal of Intelligent & Fuzzy Systems **2** (1994) 267–278

Generalized Splitting 2D Flexible Activation Function

Francesca Vitagliano, Raffaele Parisi, and Aurelio Uncini

INFOCOM Dept. – University of Rome “La Sapienza”
Via Eudossiana 18, I-00184 Rome, Italy
aurel@ieee.org
http://infocom.uniroma1.it/aurel

Abstract. It is well known that in problems where both amplitude and phase recovery is essential - like in signal processing for communications, or in problems of nonlinear signal distortions, like control, signal processing and imaging applications - it is important to consider the complex nature (and thus the intimate relation between real and imaginary part) of the data.
One of the main problem to design complex neural networks (CpxNN) consists in the definition of the complex Activation Functions (AF): to ensure the universal approximation network capabilities, the AFs should be bounded and differentiable. In the complex domain these characteristics are in contrast with Louiville’s theorem, which asserts that the only bounded and differentiable (analytic) function is the constant function. In this paper we investigate the use of 2D spline to define a new class of flexible activation functions, which are bounded and (locally) analytic suitable to define a new class of complex domain neural networks (CpxNN).

1 Introduction

The classical neuron computes the weighted sum of its inputs and feeds it into a nonlinear function called activation function (AF) [1]. The behavior of a neural network (NN) built with such neurons, as in the multi-layer perceptron (MLP), thus depends on the chosen AFs. Sigmoids are commonly used for this purpose. Different classes of nonlinear AFs, depending on some free parameters, have also been widely studied and applied (see for example [2]).

Although the main theoretical developments are defined for real valued NNs, it is well known that complex domain NNs (CpxNN) are suitable for many signal processing applications.

It is well known that using CpxNN is more advantageous than using a real-valued NN fed with a pair of real numbers [3] - [5].

In CpxNNs, one of the main problem is related to the complex domain AF. Let $f(z)$ be the complex AF with $z \in \mathrm{C}$ (z is the complex linear combiner output); the main properties that $f(z)$ should satisfy are:

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 85–95, 2003.

1) $f(z) = u(x, y) + jv(x, y)$;
2) $f(z)$ should be non linear and bounded;
3) In order to derive the backpropagation (BP) algorithm the partial derivatives of $f(z)$ should exist and be bounded.

Unfortunately the main difficulty is the lack of bounded and at the same time analytic complex nonlinear AFs in the complex plane C. In fact Liouville's theorem (see [10] for more details) states that: *'The only bounded differentiable (analytic) functions defined for the entire complex domain are constant functions'*.

Clearly the properties of boundedness and differentiability in all the complex domain are contrasting requirements, if we want to use a complex AF defined in C. In other words $f(z)$ should be defined as a nonlinear complex function that is bounded almost everywhere in the complex domain C.

Recently, a complex-valued adaptive spline neural network has been presented [12]. The author used the splitting method, where the two real functions are substituted by two flexible spline curves controlled by a small number of parameters. It is shown that this architecture is well suited for supervised signal processing applications, because it is based on an efficient Catmull-Rom spline AF whose regularization properties are described in [13].

In this paper we introduce the use of bi-dimensional (2D) spline in order to define a new class of flexible AF, called generalized split AF (GSAF), which are bounded and (locally) analytic functions, suitable to define a new class of complex domain neural networks. In order to demonstrate the effectiveness of the proposed model, experiments on channel equalizations are described in the last Section.

2 On the Complex Activation Functions

In the last years several studies have been carried out in order to develop a complex domain learning algorithm using suitable AFs. Kim and Guest [3], proposed a complex domain modification of the BP algorithm by an extension of the real valued sigmoidal activation function to the complex plane. However, this function has a periodic singularity:

$$\frac{1}{1+e^{-z}} \to \infty \qquad \text{for} z = \pm j(2k+1)\pi \qquad \forall k \in N \tag{1}$$

In [4] Clarke proposed to use a hyperbolic function ($\tanh z$) in the complex domain as a generalization of real valued activation functions.

In order to have a bounded and differentiable AF, Benvenuto *et al.* and Leung and Haykin [5]-[7], proposed an *ad-hoc* extension of the real valued backpropagation. According to properties 2) and 3) previously stated, they proposed a *split* activation function consisting of the superposition of a real and an imaginary (I and Q signal components) part AF

$$f(z) = f_I(\Re e(z)) + jf_Q(\Im m(z)) \ ; \tag{2}$$

where functions $f_I(.)$ and $f_Q(.)$, can be simple real-valued sigmoids or more sophisticated adaptive functions. However such split activation function is not analytic and the BP from the output layer takes split paths through disjoint real-valued gradients.

In order to develop a fully complex domain BP Georgiou and Koutsougeras in [8] proposed the AF defined as

$$f(z) = \frac{z}{c + \frac{1}{r}|z|} ; \tag{3}$$

where c and r are suitable constants. This function maps the complex signals into an open circle of radius r. By this way the activation function takes only values belonging to the interval $(0, r]$.

Hirose in [9] proposed the use of a fully complex AF defined as

$$f(s_k e^{j\beta_k}) = \tanh(s_k/m)e^{j\beta_k} ; \tag{4}$$

where s_k and β_k are the norm and the argument of the summation of the input vector fed from the previous hidden layer ($j\beta_k = \Sigma_j w_{kj} y_j$), and m is a constant which is inversely related to the gradient of the absolute function value $|f|$ along the radius direction around the origin of the complex coordinate.

The *suitable constants* of the AFs (3) and (4) are chosen in order to normalize or to scale the amplitude of the complex input signal. However, these functions preserve the phase, thus being incapable of learning the phase variation between the input and the target in NN without delay lines at the input layer. Moreover due to their radial mapping, as stated in [15], in the case of time-delayed NN they perform poorly in restoring nonlinear amplitude and phase distortion of non constant modulus signals.

A. *Elementary Transcendental Functions*

More recently Kim and Adali, in order to define a family of useful fully complex AFs, proposed the use of the so called elementary transcendental functions (ETF) [15].

They reduced the 'desirable properties' of a complex AF to the unique condition:

$$f'(z) = f_x = -if_y \tag{5}$$

that is the Cauchy-Riemann equation.

They classified the ETFs in two categories of unbounded functions, depending on which kind of singularities they possessed. A singularity is a point in which a function is not analytic (and thus not differentiable): if $\lim_{z\to z_0} f(z) \to \infty$ but the function is analytic in a deleted neighborhood of z_0 (that is a pole), the singularity is said to be *isolated*; if $\lim_{z\to z_0} f(z)$ exists it is isolated but *removable*; if none of these cases are met, the function has an *isolated essential singularity.*

The proposed elementary functions including $tan(z)$, $atan(z)$, $sin(z)$, $asin(z)$, $acos(z)$, $tanh(z)$, $atanh(z)$, etc. are identified to provide adequate nonlinear discriminant capabilities as required for an AF.

These transcendental functions are entire (analytic) and bounded *almost everywhere*, i.e. they are unbounded only on a set of points having zero measure. If used as AFs in neural networks they assure convergence almost everywhere (with probability 1).

Moreover, in signal processing applications where the domain of interest is a bounded neighborhood of the unit circle these singular points scarcely pose a problem.

B. *Universal Approximation Property*

Recently in [16] the universal approximation theorem for fully CpxNNs has been demonstrated.

Let I_n denote the n-dimensional unit cube $[0,1]^n$ and $C(I_n) \subset M(I_n)$ the space of all continuous complex functions on I_n. The finite sums of the form

$$G(z) = \sum_{k=1}^{N} \beta_k f(w_k^T z + \theta_k) \; ; \tag{6}$$

where $\beta_k, \theta_k \in C \; w_k, z \in C^n$, N is the number of hidden units and $f(z): \; C^n \rightarrow C$ are dense in $C_1(I_n)$ if $f(z)$ is a complex bounded measurable discriminatory function i.e.

$$\forall g \in C(I_n), \forall \varepsilon > 0, \forall z \in C(I_n) \Rightarrow \exists \; G(z) \; |G(z) - g(z)| < \varepsilon \tag{7}$$

where $f(z)$ is an activation function of the hidden layer and $G(z)$ is an output of the net.

In [16] they also demonstrated the theorem for any complex bounded measurable discriminatory function and further for functions having isolated or essential singularities, but in compact subsets of the deleted neighborhood of the singularity.

All these functions are bounded almost everywhere and have countable singular points; some of them can be handled separately during the learning process (removable singularities) or can be placed outside of the working domain. In this way the conflicting requirements between boundedness and differentiability can be relaxed by paying attention to the domain of operation, that can be easily identified for almost all applications.

3 Generalized Split AF by 2D Spline

This section is dedicated to the description of the new AF starting from the theories on fully CpxNNs: starting from previously reported considerations we tried to employ a generalization of the split method proposed in [12].

If we consider the expression of a fully complex function in relation to the real and imaginary part we may write:

$$f(z) = f(x,y) = u(x,y) + jv(x,y) \tag{8}$$

in which, as we have shown in the introduction, the two functions must be bounded and differentiable at least in the whole domain of the problem at hand.

If we consider each part as a function of two variables, we can perform each of them with a bi-dimensional spline; one plays the role of the real part and one of the imaginary part of the complex activation function. With regard to the 'desired properties' stated for the fully complex AFs we can note that 2D splines:

- They are non linear functions with respect to the coordinates; thus $f(z)$ is a nonlinear function with respect to x and y;
- They haven't singularities and they are bounded for each $z = x + jy$;
- The partial derivatives u_x, u_y, v_x, v_y are continuous and bounded;
- The condition $u_x v_y \neq v_x u_y$ is verified;

If we consider Adali's conditions, the Cauchy-Riemann equations are not satisfied by the complex 2D spline AF itself, but we tried to impose them by an algorithm constraint during the learning process:

$$u_x = v_y = \frac{(u_x + v_y)}{2}, \qquad u_y = -v_x = \frac{(v_x + u_y)}{2}; \tag{9}$$

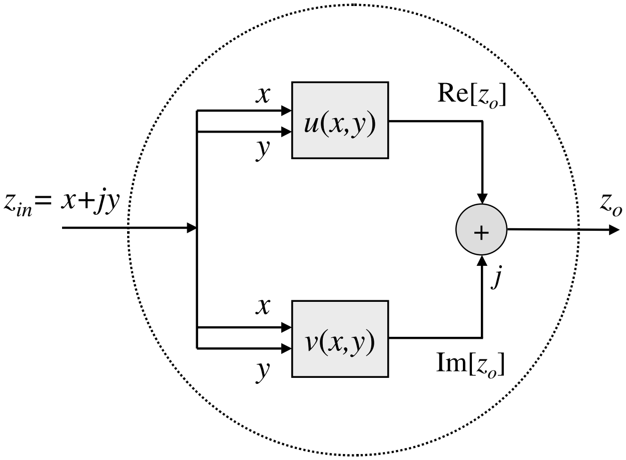


Fig. 1. Generalized splitting activation function. The functions $u(x, y)$ and $v(x, y)$ are built using flexible bi-dimensional splines.

The neural architecture apparently is the same as a traditional MLP, but the difference lays inside each neuron.

◇ *Backpropagation for the generalized splitting AF*

Clearly the input to the neuron is the complex weighted sum of the outputs of the previous layer $(l - 1)$ of the net:

$$z_{in} = x + jy = \sum_{k=0}^{N_{l-1}} w_k x_k = \sum_{k=0}^{N_{l-1}} \Big(w_{I,k} + jw_{Q,k}\Big)\Big(x_{I,k} + jx_{Q,k}\Big) \tag{10}$$

When a signal z_{in} comes to the input of the neuron, both real and imaginary part are sent to the two functions, which are each a bi-dimensional spline. The output of each function is a real value; we impose these two outputs to be respectively the real and imaginary part of the output of the whole activation function. This pair is met at the output of a neuron, is weighted with complex weights and summed with all the other weighted outputs, ready to be sent to the next layer. Consequently the BP algorithm is fully complex.

Let $e(n) = d(n) - y(n)$, during the backward pass the δ's are calculated as follows:

$$J = \frac{1}{2}\sum_{j=0}^{N_M} |e_j|^2; \tag{11}$$

$$\delta = -\frac{\partial J}{\partial u} - j\frac{\partial J}{\partial v}; \tag{12}$$

where for a neuron of the output layer $(M-1)$ it follows that

$$\begin{aligned}
\frac{\partial J}{\partial u_k} &= \frac{1}{2}\frac{\partial}{\partial u_k}\sum_{j=0}^{N_M}|e_j|^2 = e_{I,k}\frac{\partial}{\partial u_k}\sum_{j=0}^{N_M}\Big(d_j - f_j(z)\Big) = -e_{I,k} \\
\frac{\partial J}{\partial v_k} &= \frac{1}{2}\frac{\partial}{\partial v_k}\sum_{j=0}^{N_M}|e_j|^2 = e_{Q,k}\frac{\partial}{\partial v_k}\sum_{j=0}^{N_M}\Big(d_j - f_j(z)\Big) = -e_{Q,k}
\end{aligned} \tag{13}$$

and for a hidden layer (l):

$$\begin{aligned}
\frac{\partial J}{\partial u_m} &= \sum_{i=0}^{l+1}\frac{\partial J}{\partial u_i}\Big(\frac{\partial u_i}{\partial x_i}\frac{\partial x_i}{\partial u_m} + \frac{\partial u_i}{\partial y_i}\frac{\partial y_i}{\partial u_m}\Big) \\
\frac{\partial J}{\partial v_m} &= \sum_{i=0}^{l+1}\frac{\partial J}{\partial v_i}\Big(\frac{\partial v_i}{\partial x_i}\frac{\partial x_i}{\partial v_m} + \frac{\partial v_i}{\partial y_i}\frac{\partial y_i}{\partial v_m}\Big)
\end{aligned} \tag{14}$$

We called $x_i + jy_i$ the generic input to layer $(l+1)$, meaning by that, as before, the weighted sum of the outputs of the previous layer.

This new architecture differs from the network with real 2D splines in two main aspects:

- When used to process complex signals, the real 2D spline considers the real and imaginary part of an input as two distinct signals; they are fed two times to each input neuron (a neuron j in the input layer has two inputs $s_{1,j}$ and $s_{2,j}$) but with different weights; as a consequence the number of free parameters is higher than in a complex S.N.
- The output of an activation function of the real 2D spline is single and real; no more trace of the complex nature of the data is held, thus forcing the use of a pseudo-complex backpropagation.

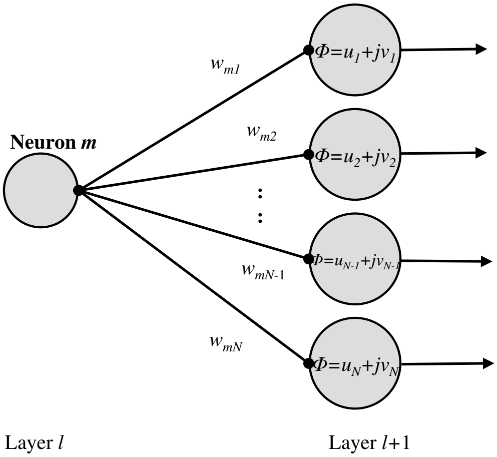


Fig. 2. Connection between a neuron of a layer and all the neurons in the following layer.

4 Experimental Results

In order to proof the effectiveness of the generalized splitting AF CpxNN and of its adaptation scheme, several experiments, consisting in communication channel equalization using different channels model have been carried out.

The first experiment consists in the equalization of a non-minimum-phase linear channel modelled by an impulse response h=(0.3482, 0.8704, 0.3482) [17]. At the filter's output is applied a polynomial memoryless nonlinear function f, as shown in Figure 3.

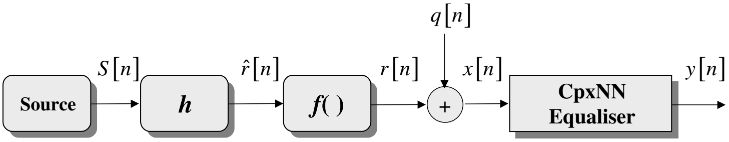


Fig. 3. Discrete time model of a digital communication system.

The output of the channel can be simply computed as

$$\widehat{r}[n] = \sum_{k=0}^{L-1} h[k]s[n-k] \tag{15}$$

where the memoryless nonlinear function is quadratic polynomial

$$r[n] = \widehat{r}[n] + 0.2\widehat{r}^2[n] \tag{16}$$

and finally, by adding Gaussian white noise $q[n]$ with variance σ_N^2

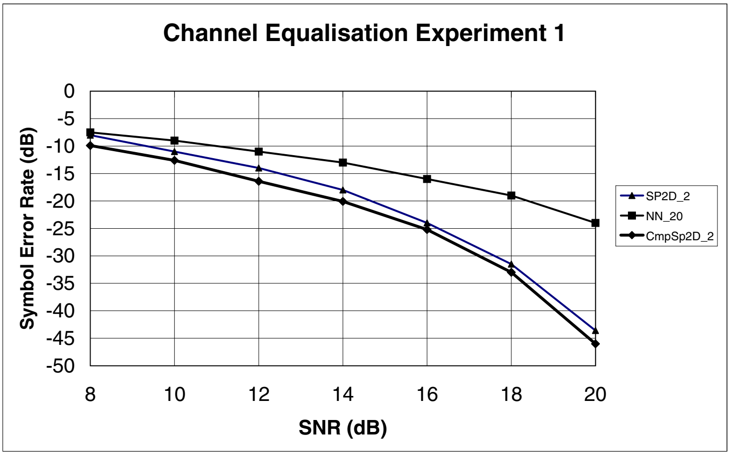


Fig. 4. Symbol error rate vs signal to noise ratio for the nonlinear channel $H(z)$ with a 4-QAM signal.

$$x[n] = r[n] + q[n] \tag{17}$$

such that the signal to noise ratio (SNR) is defined as

$$SNR = \frac{\lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} E\left[|r[n]|^2\right]}{\sigma_N^2} \tag{18}$$

where $E[.]$ represents the expectation operator.

The waveform is a $4-QAM$ (quadrature amplitude modulation) whose alphabet is the set: $\frac{1}{\sqrt{2}}\{(1,j);(1,-j);(-1,-j);(-1,j)\}$.

The network is composed of two layers: the hidden has two neurons with complex multidimensional spline activation functions, while the output is linear.

In figure 4 is shown the symbol error rate (SER), in relation to the SNR; both are expressed in dB.

The network was compared with a real 2D spline with two neurons in the hidden layer and linear output, and with a MLP network with 20 hidden neurons and linear output.

The number of free parameters for NN_20 is 302, for Sp_2D_2 is 88 and for CpxSp2D_2 is 98.

As we can see from the table, the performances of the complex network are better than that of the other two networks.

In the second experiment the complex spline network is compared with a fully complex network using $tanh(z)$ as AF and with a split (real) network using $tanh(x)$. The problem is still channel equalization, but with a different channel model and a QPSK signal of alphabet $\{(1,0);(0,j);(-1,0);(0,-j)\}$.

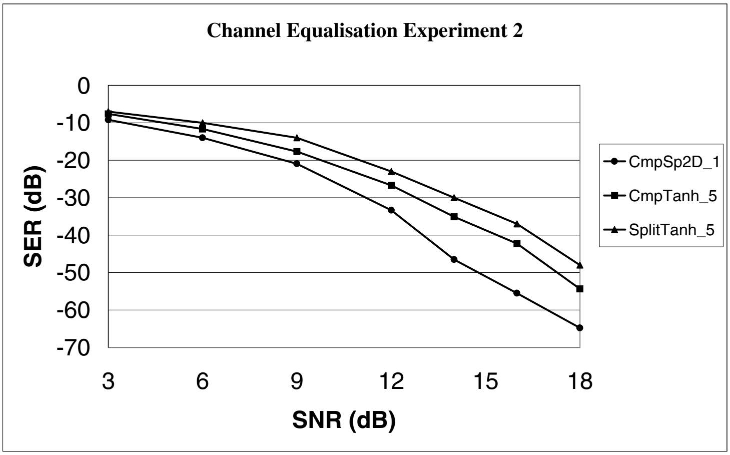


Fig. 5. Symbol error rate vs signal to noise ratio for the nonlinear channel $H(z)$ with a QPSK signal.

The channel is modelled by a complex FIR filter defined by the following transfer function

$$H(z) = \frac{\sqrt{2}}{2}(1+j) + 0.1z^{-1} \tag{19}$$

while the nonlinear part is a third order component providing a nonlinear rotation of the constellation points

$$r[n] = \widehat{r}[n] + 0.2\widehat{r}^3[n] \tag{20}$$

White Gaussian noise, with independent real and imaginary parts, with zero mean and variance σ_N^2 is added at the channel output.

The complex spline network is composed of two layers: the hidden has only one spline neuron and the output is linear, for a total of 50 free parameters. The other two networks have both 5 neurons in the hidden layer and linear output, for a total of 50 free parameters. The networks are trained with 5x10^3 symbols (independent and uniformly distributed), while the test set is composed of 3x10^6 symbols. For the spline network the learning rate was fixed to $\mu = 0.001$ and the spline rate was $\eta_{sp} = 0.01$ in all the trials. The tests were repeated for ten distinct realizations, each with different initialization (the free parameters were initialized randomly as in the previous experiments.

Figure 5 reports the results of the performed experiment. As we can note the spline complex network reaches better performances with respect to tanh networks.

It must be pointed out that in simulations with complex tanh as AF, if no limitation is imposed to the range swept away by the function, the learning is discontinuous and for low levels of SNR the algorithm may not converge at all. Moreover, the weights are constrained to be initialized to small values, otherwise the updating is oscillating. The learning process is very sensitive to the size of the learning rate, constraining it to very small values and it must be changed during the training process (by trials). As a consequence, for low levels of SNR, the convergence is very slow and very often the target MSE (prefixed for each SNR test and reached by the spline network) is not reached (3 or 4 dB less).

5 Conclusions

In this work we introduce a new complex activation function for neural networks, based on spline interpolation and in order to process complex data. Only in recent time efforts were made to find a way of handle these kind of signals: in fact previously the traditional approach considered complex signals just like a pair of real numbers and real and imaginary parts were elaborated separately as independent from each other. This approach doesn't take advantage of the deep correlation between real and imaginary part of a complex signal and represents a compromise to avoid the difficulty to find nonlinear complex activation functions. In fact the main obstacle to their use is the conflict between the boundedness and the differentiability of complex functions in the whole complex plane.

Another approach is that of using fully complex neural networks with activation function that are differentiable and bounded almost everywhere. We think that fully complex neural networks are best suited to deal with complex data because they are defined in the same domain (the complex plane) and provide substantial advantages in learning complex nonlinear mappings because of their efficiency. The experimental results confirm the validity of this approach. In particular complex bidimensional spline networks outrun the performances of other complex networks in both generalization capability and speed of convergence. This is due to the desirable properties owned by spline functions, that demonstrated to be a very powerful tool.

However this work is only a preliminary step in the field of complex networks and further investigation may lead to new discoveries and better improvements.

References

1. Haykin S. "Neural Networks, a comprehensive foundation" second edition, Prentice Hall International Inc., New Jersey, 1999.
2. K. Hornik, M. Stinchombe, H. White, "Universal Approximation of an Unknown Mapping and Its Derivatives Using Multilayer Feedforward Networks", Neural Networks, Vol. 3, pp. 551-560, 1990.
3. M.S. Kim and C.C. Guest, "Modification of Backpropagation Networks for Complex-Valued Signal Processing in Frequency Domain", Proc. of Int. Joint Conference Neural Networks, San Diego (CA), pp. II 27-31, June 1990.
4. T.L. Clarke, "Generalization of Neural Networks in the Complex Plane", Proc. of Int. Joint Conference Neural Networks, San Diego (CA), pp. II 435-440, June 1990.
5. N. Benvenuto, M. Marchesi, F. Piazza, A. Uncini, "A Comparision Between Real and Complex Valued Neural Networks in Communication Application" Proc. of INNC 91, European Int. Neural Network Conference , Helsinki (Finland), June 1991.

6. N. Benvenuto, F. Piazza, “On the Complex Backpropagation Algorithm”, IEEE Trans Signal Processing, Vol.40, pp.967-969, Apr. 1992.
7. H. Leung, S. Haykin, “The Complex Backpropagation Algorithm”, IEEE Trans Acoust. Speech and Signal Process., Vol.ASSP-39, pp.2101-2104, Sept. 1991.
8. G. Georgiou and C. Koutsougeras, “Complex Backpropagation”, IEEE Trans. On Circuits and Systems II, Vol. 39, No. 5, pp. 330-334, May 1992.
9. A. Hirose, “Continuous Complex-Valued Back-propgagation Learning”, Electronics Letter, Vol. 28, No. 20, pp. 1854-1855, September 1992.
10. W. Rudin, “Real and Complex Analysis”, McGraw Hill, 1974.
11. N. Benvenuto, M. Marchesi, F. Piazza, A. Uncini, “Non Linear Satellite Radio Links Equalized Using Blind Neural Networks”, IEEE Int. Conference on Acoustic Speech and Signal Processing, Toronto, Canada, pp. 1521-1524, May 1991.
12. A. Uncini, L. Vecci, P. Campolucci, F. Piazza, “Complex-Valued Neural Networks with Adaptive Spline Activation Function for Digital Radio Links Nonlinear Equalisation”, IEEE Transaction On Signal Processing, Vol.47, No 2., 1999.
13. L. Vecci, F Piazza, A. Uncini, “Learning and Approximation Capabilities of Adaptive Spline Activation Function Neural Networks”, Neural Networks, Vol. XI, No.2, pp. 259-279, 1998.
14. Solazzi M. Uncini A. “Regularising Neural Networks Using Flexible Multivariate Activation Function”, submitted to Neural Networks, 2003.
15. T. Kim, T. Adali, “Complex Backpropagation Neural Networks Using Elementary Transcendental Activation Functions”, Proc. of IEEE ICASSP, Vol. II, pp.1281-1284, 2001.
16. T. Kim, T. Adali, “Universal Approximation of Fully Complex Feed-Forward Neural Networks”, Proc. of the IEEE ICASSP, Vol. I, pp.973-976, 2002.
17. Proakis J.G. “Digital communications” second edition, McGraw-Hill International, New York. [1989]

Face Localization with Recursive Neural Networks

Monica Bianchini, Marco Gori,
Paolo Mazzoni, Lorenzo Sarti, and Franco Scarselli

Dipartimento di Ingegneria dell'Informazione
Università degli Studi di Siena
Via Roma, 56 — 53100 Siena, Italy
{monica,marco,sarti,franco}@ing.unisi.it
djpable@libero.it

Abstract. Recognizing a particular face in a complex image or in a video sequence, which the humans can simply accomplish using contextual information, is a difficult task for an automatic recognizer. Moreover, the face recognition problem is usually solved having assumed that the face was previously localized, often via heuristics based on prototypes of the whole face or significant details. In this paper, we propose a novel approach to the solution of the face localization problem using recursive neural networks. In particular, the proposed approach assumes a graph–based representation of images that combines structural and sub–symbolic visual features. Such graphs are then processed by recursive neural networks, in order to establish the eventual presence and the position of the faces inside the image. Some preliminary experiments on snapshots from video sequences are reported, showing very promising results.

1 Introduction

In several applications, ranging from static matching of controlled photographs to video surveillance, the problem of face recognition plays a crucial role. Face recognition compares a given image, with one or more faces, against a database, and reports an eventual match. The localization of the face is a preliminary step, which is required in order to be able to recognize the face. Given an arbitrary image, the goal of face localization consists of spotting whether or not a face is present and, if it is so, of returning the location of each face.

Frequently, to solve the face recognition problem, the positions of the faces are supposed to be known, since even the face localization problem is a very challenging task. In fact, the appearance of a face in an image is unsettled with respect to scale, location, orientation and pose [1]. Face localization methods can be classified into four main categories: knowledge–based, feature invariant, template matching, and appearance–based methods. In knowledge–based methods, human knowledge about faces is encoded by some rules. The localized faces must respect the predefined rules [2]. The aim of feature invariant methods is to

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 99–105, 2003.

discover a set of features that are invariant with respect to pose, viewpoint, lighting conditions. Those features are then used to localize the faces [3]. Template matching methods store several patterns of a face and describe each pattern by facial features. The correlation among an input image and the stored patterns is computed for detecting new faces [4]. Finally, in appearance–based methods, the templates describing the faces are learned by examples. Several machine learning techniques are used, from static neural networks [5] and Hidden Markov Models, to Support Vector Machines.

In this paper, we present a new appearance–based method that exploits recursive neural networks to learn templates. The novelty of the approach consists of using graphs to represent images. The graphs are processed by a recursive network, a particular model that extends neural network processing to structured data [6]. Since, the graphical representation is invariant under image translations, rotations and scaling, the method does not suffer from those image modifications. Moreover, due to the intrinsic parallelism of recursive networks, the technique can be implemented very efficiently.

The paper is organized as follows. In the next section, the graph–based representation of images and the recursive neural network model are described. In Section 3, some preliminary experimental results are reported and, finally, in Section 4 some conclusions are drawn.

2 Classifying Images with Recursive Networks

In order to describe how a face can be localized in a complex image, we have to detail the two fundamental preprocessing phases which allow us, first, to extract an informative graph–based representation of the image, and then to transform the Region Adjacency Graph (RAG) representing the image into a recursive–equivalent tree, with which the recursive neural network can be fed. Finally, the recursive neural model is briefly sketched.

From Images to Graphs

In our approach, images are encoded using the HSV (Hue, Saturation, Value) color space. Then, each image is binarized (see Fig. 1) by a filter on hue and saturation. In fact, the saturation and the hue of human skin color belong to a well known range [7]. Even if it is well known that the unstructured information given in the HSV space is not sufficient to identify faces in images, the topological relationships among different parts of the images, belonging or not to a face, help in localizing them. In fact, subsequently, the images are split into homogeneous regions (black or white). Very small regions are removed and their pixels are inserted into the adjacent regions. Images used in our experiments contain up to one hundred regions.

The structural information related to the spatial relationships between pairs of regions can be coded by a *Region Adjacency Graph* (RAG), using the method proposed in [8]. Two connected regions R_1, R_2 are *adjacent* if, for each pixel

$a \in R_1$ and $b \in R_2$, there exists a path connecting a and b, entirely lying into $R_1 \cup R_2$. The RAG is extracted from the segmented image by:

- Associating a node to each region. Each node is labeled with a real vector of features extracted from the original image (area, average color, barycentre coordinates, minimum bounding box).
- Linking the nodes associated to adjacent regions. The edges are not oriented.

Therefore, a RAG takes into account both the topological arrangement of the regions and the sub–symbolic visual information. Notice also that the RAG connectivity is invariant under translations, rotations and scaling, which is a useful property for the goal of locating faces. Moreover, in order to set up a learning environment, also a binary target in $\{0, 1\}$ is attached to each node of RAGs. In fact, in a RAG, nodes represent the objects contained in the image. Thus, the target of a node states whether it represents or not a part of a face. For example, in Fig. 1, nodes with a cross inside denotes part of the face and have target 1, whereas the other nodes have target 0.

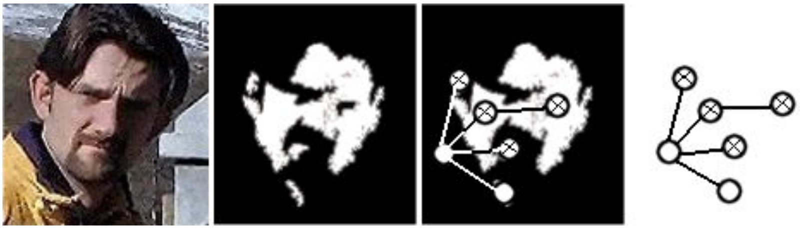


Fig. 1. The original image, the segmented image, and the extracted RAG.

From RAGs to Trees

Since recursive neural networks can process only Directed Acyclic Graphs (DAGs), each RAG must be transformed into some directed structure. In the following, we describe a method to carry out this task. The procedure takes in input a RAG R along with a node n and produces a tree T having n as its root. The method must be repeated for each node n of the RAG. It can be proved that the forest of trees built from R contains the same information as R [9].

The first step of the procedure is a preprocessing phase that transforms R into a directed graph G, by assuming that a couple of edges is attached to each undirected one, thus preserving the duplex information exchange. Then, G is unfolded by the following algorithm:

1. Insert a copy of n into T
2. Visit G, starting from n, using a breadth–first strategy: for each visited node v, insert a copy of v into T and link v to its parents
3. Repeat step 2 until all arcs have been visited

According to [9], the above unfolding strategy produces the minimal tree that holds the same information contained in R (Minimal Unfolding, Fig. 2(b)). However, other strategies are possible. For example, we can continue to repeat step 2 for a random number of times after that all arcs have been visited (Random Unfolding, Fig. 2(c))) or we can replace the breadth–first visit strategy of step 2 with a random visit of the graph (Medium Unfolding, Fig. 2(d)).

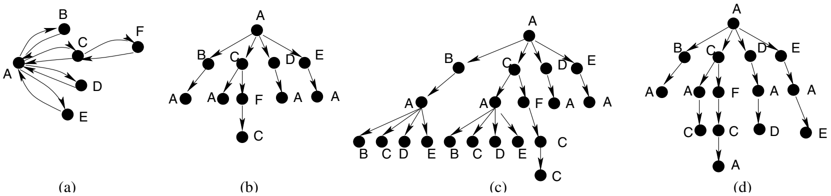


Fig. 2. (a) The graph G. Trees obtained from G using (b) Minimal, (c) Random, and (d) Medium Unfolding, respectively.

The Recursive Neural Network Model

Recursive neural networks are a generalization of recurrent networks particularly suited to learn directed positional acyclic graphs (DPAGs) and have been already used in some applications [6]. In order to process a graph G, the recursive network is unfolded through the graph structure, producing the *encoding* network (see Fig. 3). The graphs we consider have only one root node (supersource), i.e. a node with no ancestors. This does not limit the computational model since any graph can be always transformed into a graph which satisfies this property [10]. At each node v of the graph, the *state* $\mathbf{X}_v$ is computed by a feedforward network as a function of the input label $\mathbf{U}_v$ and the state of its children:

$$\mathbf{X}_v = f(\mathbf{X}_{\mathrm{ch}[v]}, \mathbf{U}_v, \theta_f),$$

where $\mathbf{X}_{\mathrm{ch}[v]}$, that collects the state of the children of v, is equal to the *frontier state* $\mathbf{X}_0$, if node v lacks of its i–th child. At the supersource, also an output function is evaluated by a feedforward network, called the *output* network:

$$\mathbf{Y}_s = g(\mathbf{X}_s, \theta_g).$$

The parameters θ_f and θ_g are connection weights, being θ_f independent of node v [1]. The parametric representations f and g can be implemented by a variety of neural network models. In the experiments, carried out to evaluate the proposed method, f and g were implemented both with one and two–layer perceptrons.

[1] In this case, we say that the recursive neural network is *stationary*.

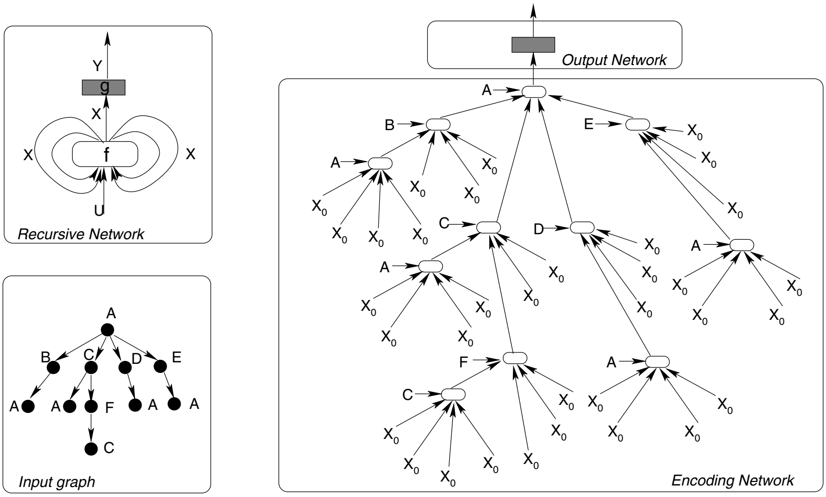


Fig. 3. The recursive neural network, unfolded in the encoding and the output network, w.r.t. an input tree.

3 Experimental Results

In order to evaluate the effectiveness of our approach a preliminary experimentation was performed. Faces were searched in a dataset containing 201 images, acquired by TV video sequences, and 238 faces (each image contains at least one face). The appearance of faces in the images is unsettled with respect to orientation, light conditions, dimensions, etc. (see Fig. 4). The images were divided in three sets: training set, cross–validation set, and test set. Both the training and the cross–validation set contain 50 images, whereas 101 images constitute the test set. Each image belonging to the three sets was segmented obtaining a RAG. Subsequently each RAG was unfolded using the three methods described in Section 2. Finally, the recursive neural network was trained to predict whether the nodes in the graphs belong to faces or not The obtained results are evaluated

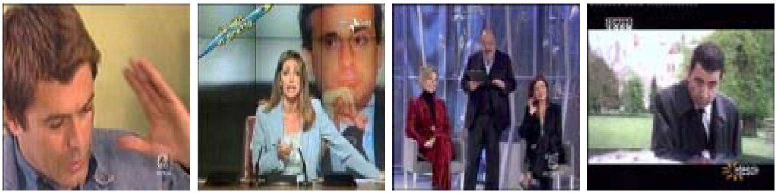


Fig. 4. Variability of faces appearance in the images chosen.

Table 1. Results obtained varying the unfolding method and the recursive neural network architecture.

Unfolding method	**RNN architecture**	**Accuracy rate**	**Detection rate**
Minimal	one layer - 2 state neurons	85.54%	82.64%
Medium	one layer - 2 state neurons	86.78%	90.08%
Random	one layer - 2 state neurons	83.88%	81.82%
Medium	one layer - 10 state neurons	80.58%	90.08%
Medium	two layer - 10 state, 5 hidden neurons	84.71%	83.47%

and compared using accuracy and detection rates. The accuracy rate is obtained dividing the number of nodes correctly classified by the number of nodes in the graphs. Instead, the detection rate is obtained dividing the number of faces correctly localized by the number of faces in the images. Several recursive neural networks were trained with the aim of determining the best network architecture (number of state and hidden neurons, number of layers). The results obtained are reported in Table 1.

4 Conclusions

In this paper we have proposed a new method to localize faces in images using recursive neural networks. Due to the graphical representation of images, the approach is invariant under image translations, rotations, and scaling. Moreover, the technique can be implemented very efficiently exploiting the intrinsic parallelism of neural networks. Finally, the preliminary experimental results are very promising. Future matters of research include experimentation on more sophisticated clustering techniques and extensive comparisons with established face detection methods.

References

1. M.-H. Yang, J. Kriegman, and N. Ahuja, "Detecting faces in images: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 34–58, January 2002.
2. G. Yang and T. Huang, "Human face detection in complex background," *Pattern Recognition*, vol. 27, no. 1, pp. 53–63, 1994.
3. R. Kjeldsen and J. Kender, "Finding skin in color images," in *2^{nd} Int. Conf. on Automatic Face and Gesture Recognition*, pp. 312–317, 1996.
4. I. Craw, D. Tock, and A. Bennett, "Finding face features," in *2^{nd} European Conference on Computer Vision*, pp. 92–96, 1992.
5. H. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, pp. 23–38, January 1998.
6. P. Frasconi, M. Gori, and A. Sperduti, "A general framework for adaptive processing of data structures," *IEEE Transactions on Neural Networks*, vol. 9, pp. 768–786, September 1998.

7. K. Sobottka and I. Pitas, "Looking for faces and facial features in color images," *Pattern Recognition and Image Analysis: Advances in Mathematicl Theory and Applications, Russian Accademy of Sciences*, vol. 7(1), 1997.
8. C. de Mauro, M. Diligenti, M. Gori, and M. Maggini, "Similarity learning for graph–based image representations," in *GBR 2001*, (Ischia (Naples)), pp. 250–259, May 23–25, 2001.
9. M. Bianchini, M. Gori, and F. Scarselli, "Theoretical properties of recursive networks with linear neurons," *IEEE Transactions on Neural Networks*, vol. 12, no. 5, pp. 953–967, 2001.
10. A. Sperduti and A. Starita, "Supervised neural networks for the classification of structures," *IEEE Transactions on Neural Networks*, vol. 8, pp. 429–459, 1997.

Multi-class Image Coding via EM-KLT Algorithm

Alessandra Budillon[1] and Francesco Palmieri[1]

Dip. di Ingegneria dell'Informazione
Seconda Università di Napoli
via Roma, 29 81031 Aversa (CE) - Italy
{frapalmi,alebudil}@unina.it

Abstract. This paper presents a new image coding method in which the image blocks are assigned to different classes learned by the EM algorithm. Each class is matched to a multidimensional Gaussian density function and the Karhunen-Loeve Transform (KLT), followed by optimal quantization and coding, is applied to each one of them. The performance of this Class-KLT coder is compared to the classical KLT coder (one class) showing appreciable improvement in image quality.

1 Introduction

Among the "lossy" techniques for image compression [3,4], coding schemes based on the discrete Karhunen-Loeve transform are considered the best ones since they realize compression with optimal reconstruction property. Furthermore, if the input random variables are Gaussian, KL transform achieves the lowest overall distortion of any orthogonal transform [4].

In this work the image coding problem is broken down into two steps, as previously proposed in [6,7]:

1. estimation of the probability density function (pdf) of the random source, associated to the image blocks;
2. efficient quantization of the estimated pdf, based on optimal bit allocation.

As already pointed out in [7], separating the density estimation from the quantization maybe useful in the design of a compressor, since allows to change only the quantization design if the bit rates vary. A Gaussian mixture model is assumed for the pdf since it can provide good fits to smooth densities and can be well and simply estimated by the EM algorithm [1]. In [5] it has also been underlined the role of the Gaussian mixture as a "worst case" model in compression problems. Therefore it promises to be a robust approach to classification and compression also for non Gaussian sources with similar local second order properties.

In section 2 is briefly presented the proposed coding scheme and in section 3 are reported some simulation results.

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 106–112, 2003.

2 Guassian Mixture Optimized Coding

The coding algorithm is shown in Fig. 1. The input image is first used to train the EM algorithm for the Gaussian mixture parameters estimation. Then, after classification and projection on the subspaces, the coefficients are quantized. More specifically the $nr \times nc$ image is first subdivided into $K = \frac{nr}{L} \times \frac{nc}{L}$ blocks of $L \times L$ pixels (in our simulations $L = 8$), each block is vectorized so that the image becomes a sequence of N-dimensional vectors $\mathbf{x}$, with $N = L^2$. Each vector $\mathbf{x}$ is assumed to be a sample of an N-dimensional random variable that is distributed according to a Gaussian mixture of C clusters

$$p(\mathbf{x}|\boldsymbol{\Phi}) = \sum_{j=1}^{C} \alpha_j \mathcal{N}_j(\mathbf{x}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j), \tag{1}$$

where $\mathbf{x} \in R^N$ is the observation space, $\{\boldsymbol{\Phi}\}$ is the parameters set, i.e. $\boldsymbol{\alpha} = \{\alpha_1, \ldots, \alpha_C\}$ are the a priori probabilities, $\boldsymbol{\mu} = \{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_C\}$ are the class means, $\boldsymbol{\Sigma} = \{\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_C\}$ are the covariance matrices. $\mathcal{N}_j(\mathbf{x}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$ is the Gaussian density corresponding to class j. The posterior probability, i.e. the probability that $\mathbf{x}$ belongs to class j, given the observation $\mathbf{x}$, is written then as

$$p(j|\mathbf{x}, \boldsymbol{\Phi}) = \frac{\alpha_j \mathcal{N}_j(\mathbf{x}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}{p(\mathbf{x}|\boldsymbol{\Phi})}. \tag{2}$$

According to this model from a set of observed data $\{\mathbf{x}_k\}$ with $k = 1, \ldots, K$, the EM algorithm [1] finds an estimate $\{\hat{\boldsymbol{\Phi}}\}$ of the set of parameters $\{\boldsymbol{\Phi}\}$. The number C of Gaussian clusters is defined heuristically and it is in general kept small to avoid over-fitting problems. The EM algorithm is a well-known iterative approach to maximum likelihood estimation , where at each iteration the likelihood is increased. For a given dimension N, the number of mixture model parameters scales as $O(N^2)$ and, since the training data size scales linearly with the number of parameters [2], the training sample size scales quadratically with the dimension. This means that, for example, an image of 512×512 can be used for estimating the parameters of a Gaussian mixture model where the observation are N-dimensional vectors $\mathbf{x}$, with $N = 64$. Once the pdf has been estimated an optimal quantizer can be built. Each gaussian cluster is quantized separately. Within each cluster, a KL-transform is applied and a dynamic bit allocation algorithm is used. A given observation is firstly associated to an appropriate cluster among the C Gaussian clusters by a Maximum A Posteriori criterion, then it is quantized according to the quantizer design of that cluster. The quantizer is not optimal for the mixture model but only for the Gaussian distribution generating that cluster. We have used a cluster bit allocation criterion, derived in [7] and used for speech coding, that minimizes the total average quantization error under the constraint that the average amount of bits is fixed

$$\bar{b} = \sum_{i=1}^{C} \alpha_i b_i, \tag{3}$$

where b_i is the number of bits allocated to the i^{th} cluster. The total average distortion is

$$D_{tot} = \sum_{i=1}^{C} \alpha_i D_i(b_i), \tag{4}$$

where $D(b_i)$ represent the mean square distortion of an optimal b_i transform coder of cluster i. Using high resolution expression [4]

$$D_i(b_i) = \frac{\sqrt{3}\pi}{2} N c_i 2^{-2b_i/N}, \tag{5}$$

where

$$\begin{aligned} c_i &= (\prod_{k=1}^{N} \lambda_{i,k})^{1/N}, \; 1 \leq i \leq C, && (6) \\ \mathbf{\Lambda}_i &= \mathrm{diag}(\lambda_{i,1}, \ldots, \lambda_{i,N}), && (7) \\ \boldsymbol{\Sigma}_i &= \mathbf{Q}_i \mathbf{\Lambda}_i \mathbf{Q}_i^T, && (8) \end{aligned}$$

the optimal bit allocation scheme that minimizes (4) subject to the constraint (3) is

$$b_i = \bar{b} + \frac{N}{2}(\log_2 c_i - \sum_{j=1}^{C} \alpha_j \log_2 c_i), \;\; 1 \leq i \leq C. \tag{9}$$

Once an observation $\mathbf{x}$ is assigned to the i^{th} cluster, the mean μ_i is subtracted and is KL transformed using the matrix $\mathbf{Q}_i^T$. Scalar non uniform quantization is adopted for each coefficient with a standard Gaussian compressor followed by a uniform quantizer [3]. In our algorithm only the m Principal Components for each class are used. The decoder transforms the data using the correspondent expander, the correlator $\mathbf{Q}_i$ and adds the mean μ_i.

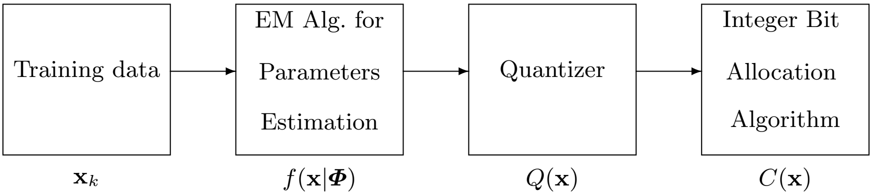


Fig. 1. Coding Scheme

3 Experimental Results

We tested the proposed compression algorithm on several images from the USC-SIPI Image Database, we report some reconstructed images obtained for different number of gaussian clusters, for different bit rates and number of principal components obtained on the 512 × 512 “Lenna” image showed in Fig. 2.

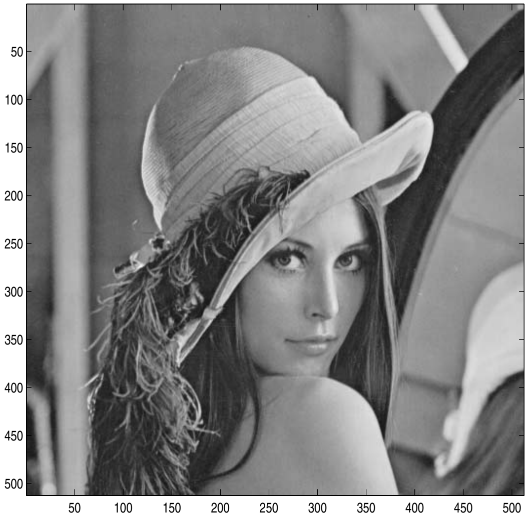


Fig. 2. Reference image “Lenna” (512 × 512)

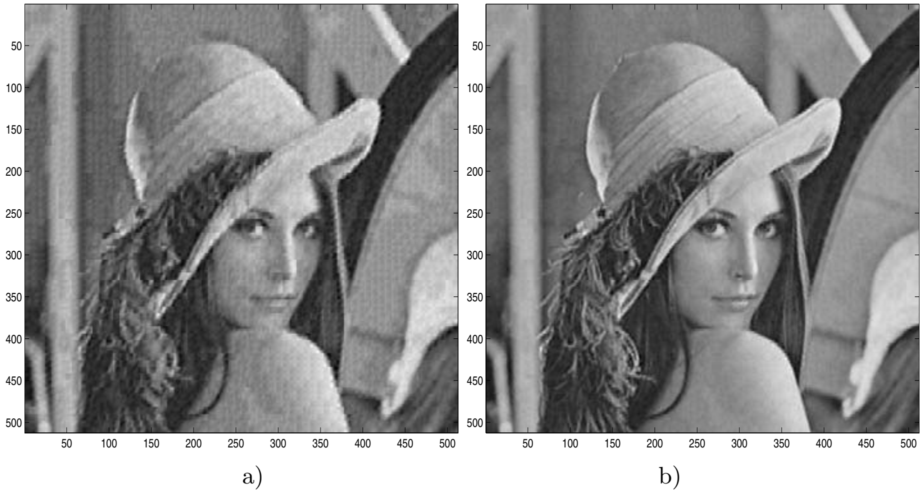
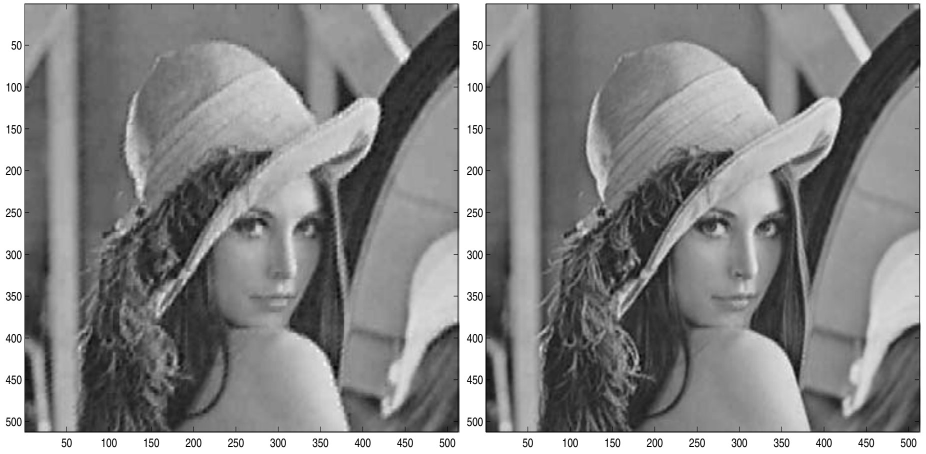


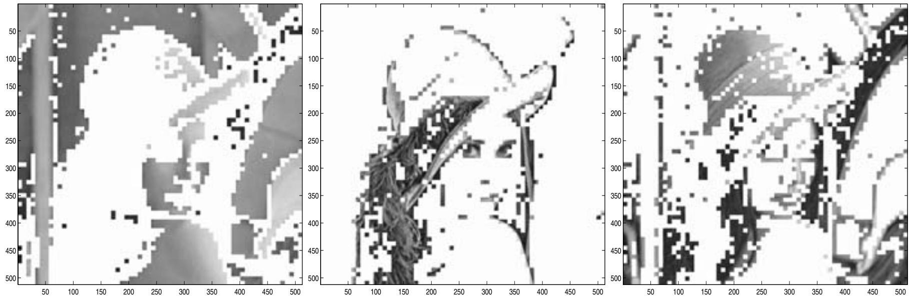
Fig. 3. a) SNR=21.29dB, BR=0.5bpp, # classes=1, # PCs=4, b) SNR=25.06dB, BR=1bpp, # classes=1, # PCs=8

As a measure of the reconstructed image quality, signal-to-noise ratio SNR = $10 \log_{10} \frac{\sum_{k=1}^{K} \|\mathbf{x}_k\|^2}{\sum_{k=1}^{K} \|\mathbf{x}_k - \hat{\mathbf{x}}_k\|^2}$ ($\hat{x}_k$ is a block belonging to the reconstructed image), is employed. As index of the compression we compute for each case the average bit rate BR = $\frac{\sum_{j=1}^{C} \alpha_j b_j}{64}$. Figs. 3 a) and b) are the results obtained when only one



a) b)

Fig. 4. a) SNR=22.58dB, BR=0.5bpp, # classes=3, # PCs=4, b) SNR=25.79dB, BR=1bpp, # classes=3, # PCs=8



a) b) c)

Fig. 5. a) First of the 3 Gaussian clusters corresponding to a priori probability $\alpha_1 = 0.48$, BR=0.875bpp, b) Second of the 3 Gaussian clusters corresponding to a priori probability $\alpha_2 = 0.19$, BR=1.218bpp, # PCs=8, c) Third of the 3 Gaussian clusters corresponding to a priori probability $\alpha_3 = 0.33$, BR=1.078bpp, # PCs=8

Gaussian cluster is used for the whole image (standard KLT). For a bit rate of 0.5bpp we have considered only 4 Principal Components (PCs), while for 1bpp we have considered 8PCs. Figs. 4 a) and b) can be compared, respectively, with Figs. 3 a) and b) having same bit rate and number of PCs, but both are referred to a 3 Gaussian clusters model. These two reconstructed images confirm a clear improvement both in visual quality and in SNRs with respect to the previous ones. Figs. 5 a), b) and c) show the 3 extracted classes, in the 3-class model. The algorithm associates the first class to the background. This class has the highest a priori probability but ends up being coded with the lowest bit rate. The other

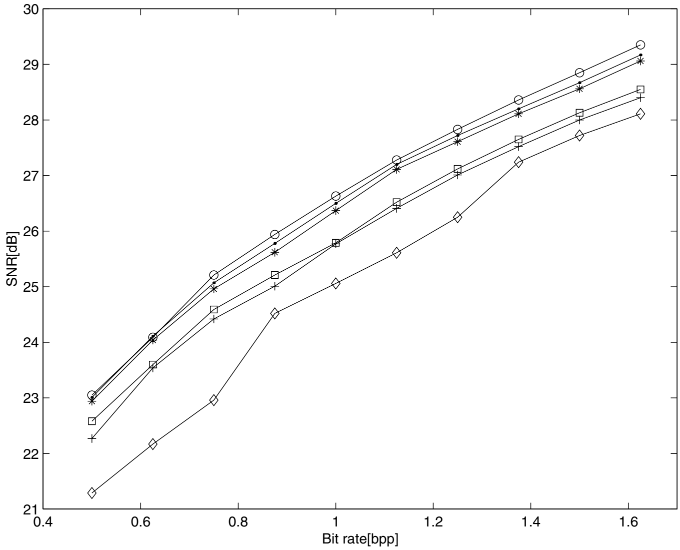


Fig. 6. SNR value versus the number of bits per pixel, each curve corresponds to a different number of gaussian clusters. Legend: 1 class=diamond, 2 classes=plus, 3 classes=square, 4 classes=star, 5 classes =point, 6 classes=circle

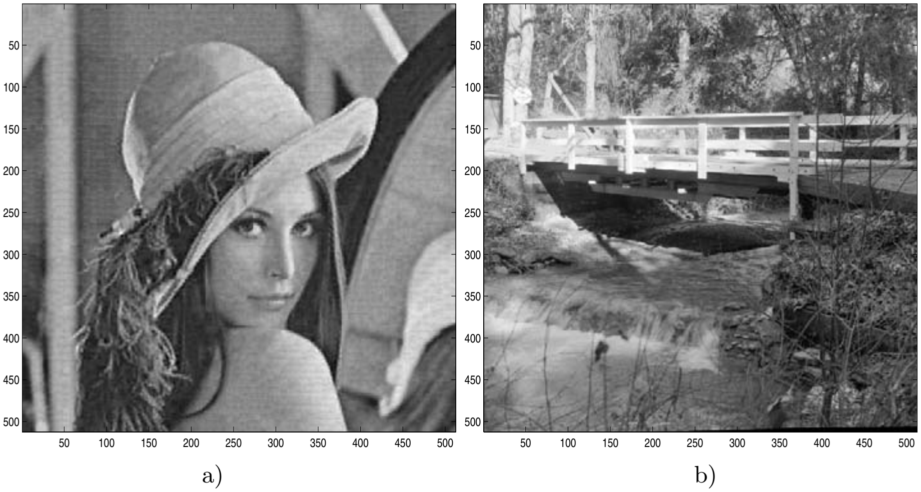


Fig. 7. a) SNR=23.69dB, BR=1bpp, # classes=3, # PCs=8, b) "Prototype image"

two are related to image details and have smaller a priori probabilities but need higher bit rates. The class partition allows to achieve small total average bit rate with good quality. Fig. 6 shows the SNR value versus the number of bits per pixel, for different number of Gaussian clusters. The strategy of considering more then one class is effective in giving always higher signal-to-noise in comparison to one class. Note that the curves do not change much if more then 4 classes are used. Therefore assuming a 3 Gaussian clusters model resulted to be a good compromise between quality and computational complexity in most of the tested images. It is also worth noticing that visually the reconstructed images, when assuming more than 3 classes look similar in spite of greater values exhibited in SNR. The overall computational cost is quite high due to the EM algorithm but it can be reduced thanks to the generalization ability of the algorithm. In fact it has been observed experimentally that different images belonging to the same category (e.g. natural scenes) can be compressed by exploiting EM estimates and PCs of a “prototype image”. Figs. 7 a) and b) show the reconstructed image for a 3 Gaussian cluster model, with the parameters estimated from the “prototype image” b) , with 1bpp and 8PCs. The image , as expected, exhibits a smaller value in SNR with respect to the case in Fig. 4 b) but the quality is acceptable.

References

1. A. Dempster, N. Laird and D. Rubin, “Maximum likelihood from incomplete data via the EM algorithm,” *Journal of the Royal Statistical Society*, vol. B39, pp. 1-38, 1977.
2. R.O. Duda, P.E. Hart and D.G. Stork, *Pattern Classification*, Sec. Ed., Wiley, 2001.
3. A.K. Jain, *Fundamentals of Digital Image Processing*, Prentice Hall, 1989.
4. A. Gersho and R.M. Gray, *Vector Quantization and Signal Compression*, Wiley, 1994.
5. R.M. Gray,“Gauss mixture vector quantization”, *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2001.
6. A. Ortega and M. Vetterli, “Adaptive Scalar quantization without side information”, *IEEE Trans. on Image Processing*, vol. 6, pp. 665-676, 1997.
7. A.D. Subramaniam and B.D. Rao, “PDF optimized parametric vector quantization of speech line spectral frequencies”, *Proc. IEEE Workshop on Speech Coding*, 2000.

A Face Detection System Based on Color and Support Vector Machines

Elena Casiraghi, Raffaella Lanzarotti, and Giuseppe Lipori

Dipartimento di Scienze dell'Informazione
Università degli Studi di Milano
Via Comelico, 39/41 20135 Milano, Italy
{casiraghi,lanzarotti}@dsi.unimi.it
gl588430@silab.dsi.unimi.it

Abstract. We describe a face detection algorithm, which characterizes and localizes skin regions and eyes in 2D images using color information and Support Vector Machine. The method is scale-independent, works on images of either frontal, rotated faces, with a single person or group of people, and does not require any manual setting or operator intervention. The algorithm can be used in face image database management systems both as a first step of a person identification, and to discriminate the images on the basis of the number of faces in them.

1 Introduction

In this paper we propose an automatic method for face detection in still color images. Such task is the first fundamental step for many applications such as face recognition and 3D face reconstruction.

Due to the high inter-personal variability (e.g. gender and race), the intra-personal changes (e.g. pose and expression), and the acquisition conditions (e.g. lighting and image resolution) face detection is not a trivial task and it is still an open problem.

In the last decade, several works have been presented [7]; more recently color has been considered a useful information to identify skin regions, since it supplies powerful methods in terms of scale and pose independence, as well as robustness to occlusions. One of the most effective method in this direction is described in [4]; the algorithm works well in most of the situations, even if it still fails in the challenging ones, where it produces a high percentage of false positive. Such wrong segmentation requires to deal with great uncertainty in the next steps, and it is the cause of errors.

Moreover, even supposing to develop a perfect skin detector, it remains the necessity to have a validation step which discriminates the skin regions in two classes: faces and non-faces [3].

In this paper we propose a method which identifies the skin regions (skin map) very precisely, and then verifies whether they are faces or not, searching for the eyes within them. Such objective is achieved localizing the potential eyes and then validating them with a classifier.

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 113–120, 2003.

We have experimented both Multi-Layer Perceptron and the Support Vector Machines, and we have developed a method based on the SVM, since its behavior has confirmed to be better than the Neural Network one. The method is able to detect a face either in the foreground or in groups, independently of scale, head rotation and partial occlusions, providing that at least one eye is visible. It works with very different illumination and background conditions.

The paper is organized as follows: in sections 2 and 3 the skin map and eye characterization algorithms are described; in section 4 the MLP and the SVM classifiers are compared; in section 5 the face validator based on the eye recognition is reported; finally in 7 results and further developments are discussed.

2 Skin Map Determination

The first step of the algorithm allows to localize the skin regions by means of a Gaussian mixture model which characterizes the peculiar skin colors, and a region growing algorithm that, starting from the pixels with the highest probability of being skin, dilates the corresponding regions taking also into account the presence of borders, whose function is to brake the dilation [2].

Regarding the skin color Gaussian mixture model, we adopt a simple and statistically well justified model made of two bidimensional Gaussians each one parameterized by $(\mu_i, \sigma_i^2 I)$ [6].

The color sample has been constructed paying attention to gather samples of different people whose pictures have been taken in very different illumination conditions. In particular, we have collected four million samples taking pictures of 8 caucasian people under 10 illumination conditions.

We have built the Gaussian mixture models of the gathered samples transforming them in six color spaces: CIE-Luv, CIE-Lab, xyY, NCC-RGB, YPbPr, TSL. The models are built on the sample chrominance components. In order to choose one space, we have experimentally verified the results shown in [5]: the segmentation works better in spaces where the luminance is well separated from the chrominance and where the skin colors are compactly clusterized. According to that, we adopt the YPbPr color space.

Once the skin color model is built, in order to obtain a robust skin map, we identify the pixels in the image which have a high probability of being skin and iteratively grow the corresponding regions, making the expansion stop at the face borders. Of course we do not know where the face borders are, thus we look for the most significant edge pixels in the image and modify their colors so that they will have a very low probability of being skin (this is obtained emphasizing the green components).

At each iteration the colors of the found skin map pixels are changed into the color with the highest probability and the image is then low pass filtered. This process has the effect of 'absorbing' the weakest edges, usually due to shadows, while stopping the skin map expansion at the strongest borders.

In figure 1 we show the output of the most significant steps of the algorithm on an image with two people in foregrounds. It can be seen that all the skin

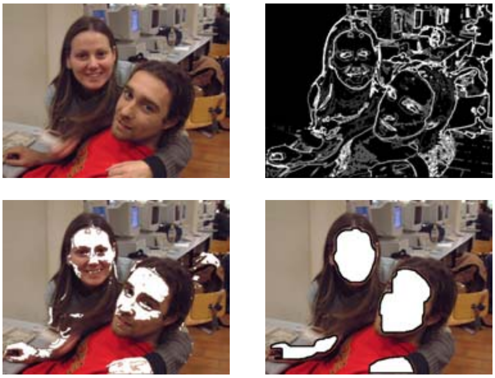


Fig. 1. Face localization: Original image; *B*: Border image; Initial skin map; Final skin map.

regions are localized very precisely and that to discriminate between faces and not faces a validation criterion is necessary (see next paragraphs).

3 Eye Characterization

To determine if a skin region corresponds to a face or not, different criteria could be adopted [4], [7]. The method we propose verifies whether at least one eye is present in the skin region or not; to this end, rather than examine the whole skin region, we first restrict the research area localizing the potential eyes, and then we validate them by means of a classifier.

As observed in [4], the eyes are characterized in the *PrPb* planes by a low red component and a high blue one; on the basis of that, the authors defined the following *EyeMap* transformation:

$$EyeMap = \frac{1}{3}\{(P_b^2) + (\hat{P}_r)^2 + (P_b/P_r)\}$$

where P_b^2, $(\hat{P}_r)^2$, and P_b/P_r all are normalized to the range $[0, 255]$ and $\hat{P}_r$ is the negative of P_r (i.e $255 - P_r$).

Since it can happen that the *EyeMap* has high values in correspondence to the mouth too, we calculate also the *MouthMap* with the transformation:

$$MouthMap = (255 - (P_r - P_b)) \cdot P_r^2$$

we binarize it maintaining the 4% highest values, and put to 0 in the *EyeMap* the pixels corresponding to the mouth. At this stage the *EyeMap* image is thresholded, maintaining the 20% of the highest values.

Moreover, examining the eye luminance values, we observe that they are always among the darkest within the skin region. Thus, to strengthen the eye localization, we threshold the normalized gray level portion of image corresponding to the considered skin region, maintaining the pixels with a gray level lower than 0.3, and we multiply it with the binary *EyeMap*.

We finally need a classifier to distinguish between eyes and all other possible regions identified by this module. To this end in the next paragraph the Multi Layer Perceptron and the Support Vector Machine are compared.

4 Classifier Choice

We have experimented both Multi Layer Perceptrons and a Support Vector Machines [1].

The experiments have been carried out using 944 images of the FERET database [12] for the training phase and 289 images from our database for the test one. For each image the centers of eyes, nose, and lips are given, and they have been used to determine 13 sub-images representing both eyes and non-eyes. The sub-images have been extracted at different positions and scales, since in the automatic eye detection we want to deal with faces at different scales and eyes not necessary centered in the sub-image.

The training and test sets consist respectively of 7439 and 2245 images, and in both cases about one sixth of the sets corresponds to eyes. All the images have been down sampled to 20×20 pixels. Such dimension has been experimentally found as trade off between the necessity to maintain low the computational costs, and to have sufficient details to learn.

Regarding the neural network, several experiments have been carried out, obtaining the best results with a neural network architecture consisting in 3 layers, respectively with 400, 20, and 2 neurons. With such setting we obtained an error of the 7.7%.

Concerning the Support Vector Machine, we have trained a C-SVM with RBF parametrized by $\gamma = 0.005$, and $C = 3.5$. With this setting we have obtained an error of 3.9%, confirming the power of such classifier.

In the following we describe the eye detection method referring to the support vector machine classifier.

5 Eye Detection

The eye detection module allows to determine whether a skin region corresponds to a face or not. Starting from the *EyeMap* blobs b highlighted in section 3, we try to localize the eyes within them, if present.

In order to speed up the procedure, we first extract a sub-set of pixels from each blob b: we consider only the positions corresponding to the vertices of a grid overlapped to b, whose spacing is proportional to the dimension and regularity of b.

More formally, being A_b the area of b, and $A_{BB(b)}$ the area of the bounding box strictly enclosing b, we define the grid spacing C_b as a *measure of regularity* of the blobs:

$$C_b = \begin{cases} \left\lceil \left(\sqrt{\frac{A_b}{\pi}} \cdot \frac{A_b}{A_{BB(b)}} \right) / 1.5 \right\rceil & \text{if } A_b \geq 30 \\ 1 & \text{otherwise} \end{cases}$$

For each visited point p, we extract 3 candidates sub-images $\mathbf{x}_p^{scale}$ which vary in scale, we give them as input to the C-SVM classifier, and we develop a cumulative image $\rho_{SVM}(p)$

$$\rho_{SVM}(p) = f_{SVM}(\mathbf{x}_p^{opt/2}) + f_{SVM}(\mathbf{x}_p^{opt}) + f_{SVM}(\mathbf{x}_p^{opt \times 2})$$

where to each point corresponds the sum of the output obtained at the different scales. The idea of cumulating the outputs of the SVM classifier is justified by the observation that such an output represents the estimated *margin* for the input example. Therefore the bigger is the output, the safer is the classification.

For each region r, we extract separately the positions $\overline{p}_i$ in which the $\rho_{SVM}(\overline{p}_i)$ values are greater than $T_r = 0.7 \times max_{p \in r}(\rho_{SVM}(p))$, and we aggregate them according to their mutual distance, determining the centroids c_j.

Among the positions in $\rho_{SVM}(p)$ which are 'close' to the found centroids, we determine the ones whose values are greater than $T_r' = 0.1 \times max_{p \in r}(\rho_{SVM}(p))$ (discarding the $\overline{p}_i$ already selected), obtaining the points $a_k(c_j)$.

We name *cluster* each set made by a c_j and all the $a_k(c_j)$ associated to it. Consequently we calculate the *cluster centroid* cl_j as

$$cl_j = \frac{\sum_{i=1}^{s} \overline{p}_i + \sum_{k=1}^{t} a_k(c_j)}{s+t} = \frac{s \times c_j + \sum_{k=1}^{t} a_k(c_j)}{s+t}.$$

Such cluster centroids represent the potential eyes positions.

Finally we calculate for each cluster centroid cl_j a vote v_{cl_j}, according to the following rule:

$$v_{cl_j} = \frac{s \sum_{i=1}^{s} \rho_{SMV}(\overline{p}_i) + t \sum_{k=1}^{t} \rho_{SMV}(a_k(c_j))}{n}$$

where n is the number of blob's candidates visited in a square window centered in cl_j and with side equal to $8C_b + 1$ (in order to normalize the vote to the number of voting pixels).

A position cl_j is classified as 'eye', if the corresponding v_{cl_j} is greater than the threshold $T'' = 0.5$.

In order to determine T'', we have examined the distributions of the v_{cl_j} corresponding to eye and not-eye regions extracted from 100 face foreground images. Regarding the eyes, we obtained a mean value equal to 2.16 and a standard deviation equal to 1, while the corresponding values for the not-eyes are 0.58 and 0.65. Moreover we observe that 162 eye regions have arrived to this validation step, against the 26 corresponding to not-eyes. This consideration highlights the fact that most of the not-eye regions have already been discarded in the previous steps, allowing at this stage to adopt a threshold which captures also the queues of the eye distribution.

6 Results

We have experimented the face detection algorithm both on images representing a single face, and on images representing groups of people.

For the test on single face images we have chosen the XM2VTS Database [13].

For the test on groups of people we have taken a set of 20 images acquired in our laboratory.

The XM2VTS Database consists of true color images of face foregrounds. The people differ for the race, the expression and the age. The images have been taken with a blue background, good illumination conditions, frontal pose. The experiment has been carried out on a set of 750 images, that is all the images in XM2VTS that do not contain people wearing glasses.

For each image we expect to find two and only two eyes. We report the obtained results organized according to the number of correct eye found, and the false positives:

eyes classified correctly	no. of false positives	no. of images	% of the total
2	0	472	62.9%
2	1	18	2.4%
2	2	2	0.26%
1	0	171	22.8%
1	1	18	2.4%
1	2	2	0.26%
percentage of images with at least 1 correct eye			**91.1%**
0	0	48	6.4%
0	1	15	2%
0	2	4	0.5%

The obtained results are encouraging, considering that with the objective of validating a face we arrive to a success in 91% of the cases, that is when at least one eye is detected.

The other test has been carried out on a set of 20 images representing 2, 3 or 4 people, for a total of 60 people. The algorithm has detected correctly 55 people, and has given only 3 false-positives.

7 Discussion

There have been in the recent years many works on face detection that have used the Support Vector Machines, due to the good experimental results that they exhibit. Even if there exists a recent paper that classifies faces generated from synthetic databases (refer to [11]), the key point of a face detection technic is the way in which it is able to identify a face standing in the complex background of a real world image. In [8] and [10] the SVMs have been used to accomplish such a task; a SVM is trained on a sample of faces and non-faces examples and then used in a multi-scale exhaustive search on the grey scale transformation of the input image. The first work on face detection that has applied the SVMs is [8], it has been computationally improved by [10] but thereafter this approach has been discarded because it seems clear that global technics fail in front of feature-based ones at the experimental proof (see [9]). Moreover it is desirable not to scan all the image but to restrict the areas of interest by means of some face characterization.

In this respect, we have made a big effort to identify faces relying only on color considerations (following [6]), thus developing an approach independent from geometric properties (works at different scales and irrespective of roto-translations). Consequently it becomes much easier to implement a classification step based on SVM in order to extract the desired features, like the eyes in our case.

Regarding future work, during the development of the application it has become clear that the measure of regularity C_b is not sufficient to deal with eyes blobs that are often irregular in shape and dimension. Therefore it is necessary a pre-processing step to discard big portions of blobs that do not contain useful information, by means of a pre-scan with the SVM classifier. What's more, we plan to extend the classifier to recognize eyes of people who wear glasses.

Acknowledgements

The authors are grateful to Paola Campadelli for her invaluable comments and suggestions.

References

1. Vladimir N. Vapnik, The Nature of Statistical Learning theory, Springer, New York, 1995.
2. P. Campadelli, F. Cusmai, and R. Lanzarotti. A color based method for face detection. *Submitted*, 2003.
3. F. Fleuret and D. Geman. Coarse-to-fine face detection. *International Journal of computer vision*, 41(1/2):85–107, 2001.
4. R. Hsu, M. Abdek-Mottaleb, and A.K. Jain. Face detection in color images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):696–706, May 2002.
5. J.C. Terrillon, M.N. Shirazi, H. Fukamachi, and S. Akamatsu. Comparative performance of different skin chrominance models and chrominance for the automatic detection of human faces in color images. *Proceedings of the IEEE International conference of Face and Gesture Recognition*, pages 54–61, 2000.
6. M. Yang and Narendra Ahuja. Gaussian mixture model for human skin color and its applications in image and video databases. *SPIE Proceedings Storage and Retrieval for Image and Video Databases VII, 01/23 - 01/29/1999, San Jose, CA, USA*, pages 458–466, 1999.

7. M.H. Yang, D. Kriegman, and N. Ahuja. Detecting face in images: a survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(1):34–58, 2002.
8. E. Osuna, R. Freund, F. Girosi, Training Support Vector Machines: an Application to Face Detection, *in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Juan, Puerto Rico*, pages 130–136, 1997.
9. Bernd Heisele, Purdy Ho, Tomaso Poggio, *Face Recognition with Support Vector Machines: Global versus Component-based Approach (ICCV)*, pages 688–694, 2001.
10. S.Romdhani, P. Torr, B. Schoelkopf and A. Blake, Computationally efficient face detection, *in the Proc. Int. Conf. on Computer Vision (ICCV)*, II:695–700, 2001.
11. J. Huang, V. Blanz, Bernd Heisele, Face Recognition Using Component-Based SVM Classification and Morphable Models, *in SVM 2002, edited by S.W. Lee and A. Verri, Springer-Verlag, Berlin*, pages 334–341, 2002.
12. Feret database. *Web address: http://www.itl.nist.gov/iad/humanid/feret*, 2001.
13. University of Surrey. The XM2VTS database, frontal views. 2000.

A Neural Architecture for 3D Segmentation

Antonio Chella[1,2], Umberto Maniscalco[2], and Roberto Pirrone[1,2]

[1] DINFO - University of Palermo
Viale delle Scienze 90128 Palermo, Italy
{chella,pirrone}@unipa.it

[2] ICAR - Italian National Research Council
Viale delle Scienze 90128 Palermo, Italy
maniscalco@icar.pa.cnr.it

Abstract. An original neural scheme for segmentation of range data is presented, which is part of a more general 3D vision system for robotic applications. The entire process relies on a neural architecture aimed to perform first order image irradiance analysis, that is local estimation of magnitude and orientation of the image irradiance gradient.

In the case of dense 3D data, irradiance is replaced by depth information so irradiance analysis of these pseudo-images provides knowledge about the actual curvature of the acquired surfaces. In particular, boundaries and contours due to mutual occlusions can be detected very well while there are no false contours due to rapid changing in brightness or color. To this aim, after a noise reduction step, both magnitude and phase distributions of the gradient are analysed to perform complete contour detection, and all continuous surfaces are segmented.

Theoretical foundations of the work are reported, along with the description of the architecture and the first experimental results.

1 Introduction

An original technique to perform range data segmentation is presented, that is based on the use of a one-layer neural network in order to group points belonging to the same surface. To this aim, first order irradiance analysis is performed on data as if they were a grey-level image, and all closed contours are detected to separate surfaces.

The present work is part of a wider research activity carried on by some of the authors, in the field of cognitive systems for mobile robotics [1]. In this general framework, vision is used to acquire dense 3D data, even from stereo cameras, and to model them in terms of geometric primitives that are the used as the basic description of objects in the scene, allowing high level components of the system to guide the robot in motion and/or manipulation tasks. In our approach superquadrics are used as primitives, so a non trivial segmentation problem is posed.

The segmentation topic has been treated by several authors in recent years. In general, this process is decoupled from reconstruction. Gupta and Bajcsy [2]

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 121–128, 2003.

use a surface-based segmentation technique, and surface labeling supports decision making in deriving 3D segments. Snakes fitting concavity points are used by Ferrie, Lagarde and Whaite [3] to isolate 3D convex regions: concavities are detected using differential geometry considerations. Some of the authors proposed the Moving Target (MT) approach [4]. Here, concavities are computed projecting data onto a sphere from their center of mass, and analysing their motion towards the original configuration: concavity points have the highest velocity. This process can be iterated to refine segmentation. Conversely from the previous approaches, Leonardis, Jalick and Solina [5] start with some model seeds placed inside the range points envelop, and alternate a fitting and a model selection phase. Models with the highest goodness-of-fit are selected to grow and fit again data, while the others are discarded. Segmentation is thus achieved automatically.

All the presented techniques suffer for high computational load, while the proposed system does not because it is a trained neural network, and it is more suited to the real time constraints imposed by robot vision. The key idea in this work is that a pseudo image representing the depths of a set of dense range points appears exactly as the image of a matte surface lit by a unique point source placed at infinity. As a consequence, deriving local irradiance gradient in this kind of images is exactly the same as the estimation of surface gradient, and allows the extraction of contours like edges, boundaries, or concavities. The proposed network implements the first order irradiance analysis reported in [6] [7] which starts from the work of Grossberg and Mingolla [8] and derives a computational framework to extract a map of local estimates of the brightness gradient, where all contours are present.

The rest of the paper is arranged as follows. In section 2 some remarks are given about image irradiance analysis, and the neural implementation. Section 3 explains in detail the whole segmentation process, while experimental results are reported in section 4. Finally conclusions are drawn, and future developments are discussed in section 5.

2 Theoretical Remarks

In order to analyse the first order differential structure of image irradiance we use a general class of isotropic compass operators formally defined in the continuous space by rotating a Kernel $k^{'}_{R}(\rho, \phi)$ such that:

$$k^{'}_{R}(\rho, \phi) = \begin{cases} k^{'}(\rho)\ , \rho \leqslant R, \quad 0 \leqslant \phi < \pi \\ -k^{'}(\rho)\ , \rho \leqslant R, -\pi \leqslant \phi < 0 \\ 0\ , \rho > R \end{cases} \tag{1}$$

where the coordinate system is shown in fig. 1.

These operators comprises many common gradient edge detection filters, like Kirsh's and assimilated [9] [10], witch estimate the edge direction by thresholding the output of increasingly oriented operators, and circular "difference of gaussian" (DOG) [11] or "blurred derivative" filters [12], which estimated the

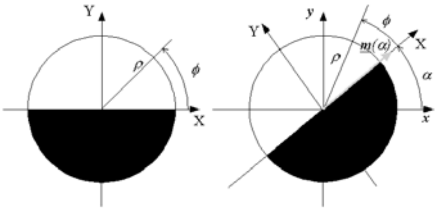


Fig. 1. A pictorial illustration of the generalized compass operator. The rectangular and polar coordinate systems used in the analysis are shown superimposed.

gradient by composing the output of two operators with orthogonal contrast axes. The operators can be thought (and are computationally implemented) as masks with odd symmetry with respect to a "contrast axis" (the X axis in fig. 1), and even symmetry with respect to a "gradient axis" orthogonal to the previous one (the Y axis in fig. 1). For the purpose of the following discussion it is convenient to consider the operator as affected by a unit vector $\underline{m}$ oriented as the contrast axis. The result of masking an image with irradiance density distribution $i(\rho, \phi)$ (intensity per unit area) by one of the above operators with generic inclination α (fig. 1) obviously depends on the image contrast direction, i.e. on the irradiance gradient ∇i, and can be regarded as weighting the mask unit vector as follows

$$\underline{M}^{'}(\alpha, \nabla i) = M^{'}(\nabla i)\underline{m}(\alpha) = \underline{m}(\alpha)\int_0^R \rho d\rho \int_{-\pi}^{\pi} i(\rho,\phi)k_R(\rho,\phi)d\phi \qquad (2)$$

Due to various sources of error and noise the measured mask output is generally an erroneous $\underline{M}^{'}_{\varepsilon}(\alpha) = \underline{M}^{'}(\alpha, \nabla i) + \varepsilon$, which may be considered depending only on the mask orientation α, since this is the only parameter experimentally prescribed (it identifies one particular operator). In a previous work, some of the authors proved that if the images irradiance is locally smooth the sum of the $\underline{M}^{'}_{\varepsilon}$ vectors minimizes the integral of squared errors [6]:

$$E^{'}(\nabla i) = \int_0^{\pi} \left[M^{'}_{\varepsilon}(\alpha) - M^{'}(\alpha, \nabla i)\right]^2 d\alpha \qquad (3)$$

and the optimal linear estimates for the irradiance gradient $\underline{g}$ in the given point is:

$$\underline{g} = \left[-\frac{2}{\pi K'}\int_0^{\pi} M^{'}_{\varepsilon}(\alpha)\sin(\alpha)d\alpha, \frac{2}{\pi K'}\int_0^{\pi} M^{'}_{\varepsilon}(\alpha)\cos(\alpha)d\alpha\right] \qquad (4)$$

where $K^{'}$ is a constant depending only on the particular kernel function $k^{'}_R(\cdot)$. It should be noticed that the same optimality of estimate is not achieved by taking as direction of $\underline{g}$ the axis of the mask with the largest output, as for the usual compass operators. Composing the output of two orthogonal DOG filters is not guaranteed to yield an optimal estimate either, since this amounts to approximate the above integrals by two values of the integrands only. The proposed method gives instead the best (in the linear domain) possible result for each chosen operator kernel, essentially by averaging away the errors through the vectorial composition.

All the computations previously outlined can be performed by a simple parallel distributed architecture. We detail in this section the implementation for the first order case but the extension to the second order analysis is straightforward. Drawing upon the proportionality of gradient components to the integrals in equation (4), a single layer linear neural network whose input is the set op image pixels suits the task (fig. 2). The input image is decomposed in a set of squared tiles, and all the pixels falling into one region are collected by a layer of N compass filters whose kernel can be chosen accordingly to equation (1). Image tiles may overlap accordingly to the filtering step one has chosen. In out implementation a step function kernel has been chosen, taking only the values -1 and 1 while crossing the symmetry axis of the compass mask.

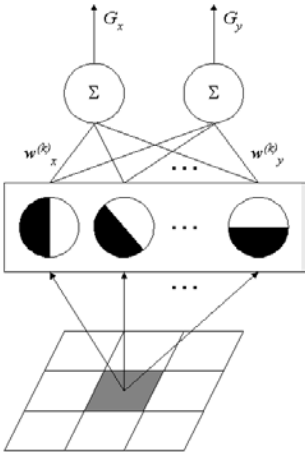


Fig. 2. The scheme of the neural architecture implementing irradiance analysis.

Each compass cell is responsible for a single orientation k; in particular, being $I(l,m)$ the generic intensity value in position (l,m) expressed with respect to the center of the image cell, the output F_k of the k-th filter with radius R can be computed as:

$$F_k = \sum_{l,m} f_k(\rho_{l,m}, \varphi_{l,m}) I(l,m)$$

where $\rho_{l,m} \triangleq \sqrt{l^2+m^2}$, $\varphi_{l,m} \triangleq \arctan(m/l)$, and

$$f_k(\rho_{l,m}, \varphi_{l,m}) = \begin{cases} 1 \ , \rho \leqslant R, & \frac{k}{N}\pi \leqslant \varphi_{l,m} < (1+\frac{k}{N})\pi \\ -1 \ , \rho \leqslant R, & (1+\frac{k}{N})\pi \leqslant \varphi_{l,m} < (2+\frac{k}{N})\pi \\ 0 \ , \ \rho > R & \end{cases} \tag{5}$$

Finally, the outputs of the compass cells are collected by two linear cells whose activations represent the discrete form of the integrals expressed in equation (4).

$$\begin{array}{ll} G_x = \sum_{k=1}^{N} w_x^{(k)} F_k, & w_x^{(k)} \triangleq -\sin(\frac{k}{N}\pi) \\ G_y = \sum_{k=1}^{N} w_y^{(k)} F_k, & w_y^{(k)} \triangleq \quad \cos(\frac{k}{N}\pi) \end{array} \tag{6}$$

The whole network is built by repeating the two layers already described in order to cover all the image tiles. Because of the units linearity, a learning process is guaranteed to converge to the correct weights. However in our implementation all the weights are pre-computed, so no learning process is used. Outputs are normalized in a global way, that is all vectors are scaled so that the maximum gradient is unitary.

3 The Segmentation Process

The segmentation starts with a preprocessing step where the typical artifacts in range images like isolated points, and small holes are removed. In particular, holes are detected as the 4-connected black regions with the smallest area, while filling occurs by means of a bilinear interpolation inwards form the boundary points of each hole. We preferred to use interpolation instead of a generic blurring filter in order to preserve the image dynamics that, in the case of range images, represents depth information.

After preprocessing, a first stage of irradiance analysis is performed applying the compass operators with unitary filtering step. The output of the net is arranged in order to obtain two images related respectively to the gradient magnitudes and phases. The magnitude distribution image allows to detect the boundaries of the object, but it fails in finding those contours arising form curvature changes, like concavities or sharp edges, where there is a strong variation of surface orientation. The phase distribution image is able to locate these features, but it is still too irregular due to local surface noise deriving from the accuracy in the depth estimate that, for the projection-based range sensors, is computed geometrically from the readings of a calibrated camera.

A second stage of irradiance analysis is performed on the phase distribution to achieve regularization. This analysis uses again compass filters with step 1, but greater in size with respect to the first stage analysis. Wider masks weight the contribution of more inputs, thus obtaining smoothed estimate of gradient and removing noise in the phase image. Again the phase distribution is taken form the irradiance analysis, it is thresholded, and composed with the magnitude image taking the maximum between them at each pixel (fig. 3).

The image thus obtained has no sharp contours neither closed ones. Contour closing is achieved by applying a morphological dilation followed by a skeletonization. Contours are then used a stop condition for a 8-connected region growing process.

In general, segmented images suffer from the presence of several small regions that are completely inside wider ones: this is the obvious consequence of some

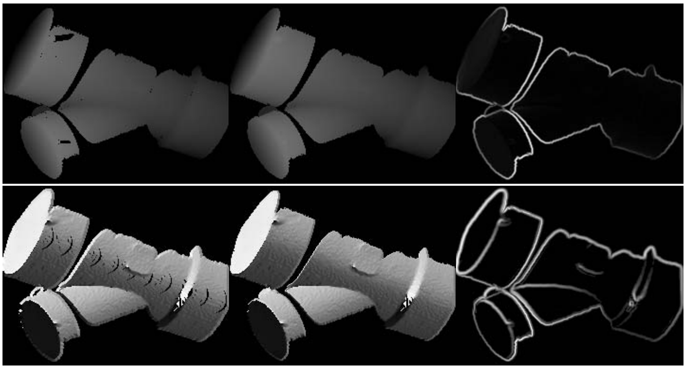


Fig. 3. From upper left to bottom right: original image, noise reduction, magnitude, and phase distribution, phase distribution after second irradiance analysis, contour image.

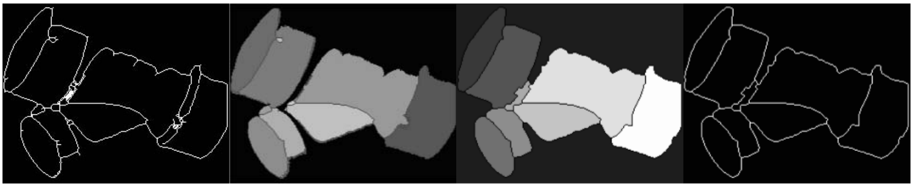


Fig. 4. From left to right: processed contour image, regions, aggregated regions, final contours.

imperfection in the range image which has not been corrected by the regularization process, and generates false contours during the morphological processing. To solve this problem, a simple aggregation process is performed merging regions having a very small area with the nearest big one. The new contour image is obtained following the borders of aggregated regions (fig. 4). Still the new regions have some imperfections: in general some input points near the borders are missed, and some spurious regions are present in corresponding to background points. Missing points derive from morphological operations, but they are of no importance to the extent of recovering 3D models of data. Spurious regions can be removed combining the regions image and the input one in a logical AND operation.

4 Experimental Setup

All the experiments have been performed on the range images that are distributed with the SEGMENTOR package, developed by Solina and his colleagues [5]. All these images are acquired projecting some geometrical patterns onto the object, and reading the deformation induced by the surface with a calibrated camera.

The whole segmentation set up has been implemented as a non-optimized MATLAB code, and runs on a Pentium III at 1000Mhz with 256MB RAM in about 16sec, without any I/O operation. After some trials, irradiance analysis parameters have been chosen to obtain the better compromise between fastness and accuracy: the number of orientations N=12 in both stages, while the compass masks were 4x4 pixels wide in the first stage, and 6x6 pixels wide in the second one. In figure 5 a couple of experiments are reported.

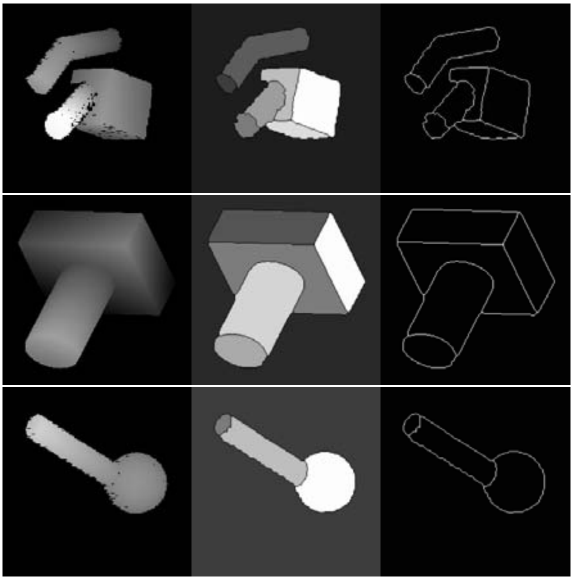


Fig. 5. In each row from left to right: original image, regions, and contour image.

5 Conclusions

An original scheme for surface segmentation of range images has been presented that relies on a very simple neural architecture aimed to perform irradiance analysis. The key idea is that range images appear the same as a pure diffusive surface illuminated from a point light source placed at the infinity, so the estimated brightness gradient is actually a surface gradient.

Experimental results are satisfactory, and future work will be oriented to a complete 3D segmentation of data as a collection of convex regions. If we extend the theory of compass operators to the second order irradiance analysis, they allow us to obtain a local curvature estimate. Starting from this consideration, in a 3D segmentation scheme each region obtained from the approach presented in this work can be labelled with a curvature type and a grouping strategy can be set up to merge those surfaces belonging to the same convex region like a cylinder, a box or an ellipsoid. Moreover, this classification can result very useful in the initialization of the model parameters when starting the reconstruction process.

References

1. Chella, A., Gaglio, S., Pirrone, R.: Conceptual Representations of Actions for Autonomous Robots. Robotics and Autonomous Systems **34** (2001) 251–263
2. Gupta, A., Bajcsy, R.: Volumetric Segmentation of Range Images of 3-D Objects Using Superquadric Models. Computer Vision, Graphics and Image Processing: Image Understanding **58** (1993) 302–326
3. Ferrie, F., Lagarde, J., Whaite, P.: Darboux Frames, Snakes, and Super-Quadrics: Geometry From the Bottom Up. IEEE Trans. on Pattern Analysis and Machine Intelligence **15** (1993) 771–784
4. Pirrone, R.: Part based Segmentation and Modeling of Range Data by Moving Target. Journal of Intelligent Systems **11** (2001) 217–247
5. Leonardis, A., Jaklic, A., Solina, F.: Superquadrics for Segmenting and Modeling Range Data. IEEE Trans. on Pattern Analysis and Machine Intelligence **19** (1997) 1289–1295
6. Callari, F., Maniscalco, U.: New Robust Approach to Image and 3-D Shape Reconstruction. In: Proc. of International Conference on Computer Vision and Pattern recognition, Jerusalem, Israel (1994) 103–107
7. Callari, F., Maniscalco, U., Storniolo, P.: Hybrid Methods for Robust Irradiance Analysis and 3-D Shape Reconstruction from Images. In: Proc. of International Conference on Artificial Neural Networks, Sorrento, Italy (1994)
8. Grossberg, S., Mingolla, E.: Neural Dynamics of Perceptual Grouping: Textures, Boundaries and Emergent Segmentation. Perception and Psychophysics **38** (1985) 141–171
9. Jain, A.: Foundamentals of Digital Images Processing. Prentice-Hall, Eglewood Cliff, NJ (1988)
10. Kirsh, R.: Computer Determination of the Constituent Structure of Biological Images. Comput. Biomed. Res. **4** (1971) 315–328
11. Marr, D., Hildreth, E.: Theory of edge detection. Proc. Royal Society **207** (1980) 187–217
12. Koenderink, J., Van Doorn, A.: Representation of Local Geometry in the Visual System. Biological Cybernetics **55** (1987) 367–385

Automatic Polyphonic Piano Music Transcription by a Multi-classification Discriminative-Learning

Stefano D'Urso and Aurelio Uncini

INFOCOM Dept. - University of Rome "La Sapienza"
Via Eudossiana 18, 00184 Rome - Italy
aurel@ieee.org
http://infocom.uniroma1.it/aurel

Abstract. In this paper we investigate on the use locally recurrent neural networks (LRNN), trained by a discriminative learning approach, for automatic polyphonic piano music transcription. Due to polyphonic characteristic of the input signal standard discriminative learning (DL) is not adequate and a suitable modification, called multi-classification discriminative learning (MCDL), is introduced. The automatic music transcription architecture presented in the paper is composed by a pre-processing unit which performs a constant Q Fourier transform such that the signal is represented in both time and frequency domain, followed by a peak-peaking and decision blocks: the last built with a LRNN. In order to demonstrate the effectiveness of the proposed MCDL for LRNN several experiments have been carried out.

1 Introduction

Music Transcription is formally defined as the process of finding the score of a musical piece, that is, finding a parametric representation of an acoustic waveform (notes, intensity, starting times, durations, instruments and other sound features).

The 1^{st} attempt in Automatic Music Transcription (AMT) has been made by Moorer [1] in 1975. He was able to identify some of the problems that persist to this day as monophonic-polyphonic classification, onset-offset detection, octave errors, ghost notes, repeated notes, notes' length and reverberation. Several transcription systems have been developed after Moorer's one: although different, all of them follow a three steps procedure.

The 1^{st} phase is known as signal's pre-processing, and its aim is to obtain a time-frequency representation of the musical signal. The simplest one is the spectrogram, but it fails because of its incoherence with the logarithmic human's way of hearing sounds. In 1988, [2] proposed the Constant Q Transform (CQT). It emulates human ears, modelling the critical band scale and avoiding the fixed bandwidth (a Fourier Transform's feature): the constant factor Q, represents the ratio of frequency resolution. In 1996, Guillermain and Kronland-Martinet [3]

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 129–138, 2003.

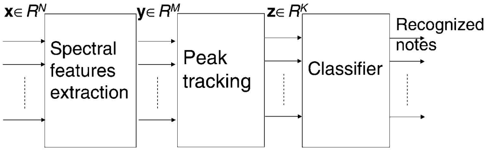


Fig. 1. General scheme of automatic music transcription system. The vector x represent the input time windowed signal; y is the array of its time-frequency representation; z is the array containing the principal spectral peaks.

proposed the Wavelet Transform (WT) (also known as scalogram). It decomposes the signal in terms of shifts and dilations of an elementary function known as the mother wavelet. The result is then interpreted in the time-scale domain.

The 2^{nd} step is known as tracking phase and its aim is to track partials (locating sinusoids in the note). It is possible to find different approaches: Klapuri's [4] sinusoid tracks, Sterian and Wakefield's [5] Kalman filter algorithm or blackboard architectures [6]. The interpretation of tracking results is know as recognition phase.

In this work, we decide to pay attention on Automatic Transcription of Polyphonic Piano Music. We exploited most related works [7] [8], in particular, Matija Marolt's SONIC [8]. We introduced a novel neural network model made up of Locally Recurrent Neural Network (LRNN) trained by a discriminative learning algorithm opportunely modified for the task of multi classification of polyphonic music recognition.

This paper is organized as follows: in section 2 we illustrate the neural network model made up of LRNNs for AMT; in section 3 we consider the discriminative learning approach and its modified version, the MCDL; in section 4 we combine the LRNNs model with the MCDL technique. We finally related about experiments results.

2 A Neural Networks Approach for AMT

The AMT should be considered a typical problem of dynamic pattern association, since we have to associate sounds to notes.

The increasing popularity of neural network models to solve pattern recognition problems has been primarily due to their low dependence on domain-specific knowledge the availability of efficient learning algorithms for practitioners to use: they provide a new suite of non-linear algorithms for feature extraction (using hidden layers) and classification. In addition, existing feature extraction and classification algorithms can also be mapped on neural network architectures for efficient (hardware) implementation.

Referring to Figure 1, for the feature extractions we used constant Q transform CQT [2] as signal's pre-processing phase. More details on its fast implementation are in [14].

In [8] Marolt tested several neural networks models (MLP, RBF, time-delay, Elmann, Fuzzy ARTMAP) and the one that best fitted with the SONIC architecture was the TDNN. Our motivation for RNNs relies on the fact that dynamic recurrent neural networks have proved to be really useful in many temporal processing applications as DSP and temporal pattern recognition. In particular, we'll make use of Locally Recurrent neural Network (LRNNs): in [9] it is possible to find the major advantages of LRNNs with respect to other models of RNNs as buffered MLP or fully RNNs.

We used Causal Recursive BackPropagation (CRBP) [9] as gradient-based learning: it is an on-line algorithm that implements and combines together the BackPropagation Through Time (BPTT) and the Real-time Recurrent Learning (RTRL); it can be efficiently implemented (respect to truncated BPTT) and has the advantage of being local in space and time (respect to RTRL that is not local in space).

In order to obtain improvements in terms of generalization capability and of learning speed we'll make use of the flexible spline activation function [10].

When working with IIR synapses it is important to assure stability: it is known that a static causal filter is asymptotically stable if and only if (iff) the poles of its transfer function lie inside the unit circle of the complex plane. We implemented the Intrinsically Stable Adaptation (ISA) [11] that makes possible to continually adjust the coefficients with no need of stability test or poles projection: the coefficients are adapted in a way that intrinsically assures the poles to be inside the unit circle.

The neural network model is depicted in Figure 2:

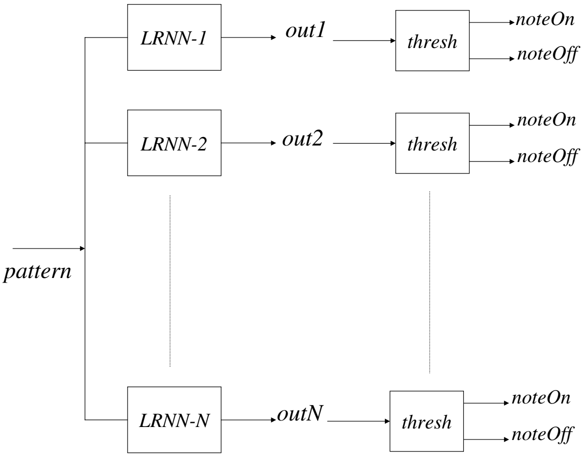


Fig. 2. Neural Network model for Automatic Music Transcription.

Each LRNN is trained to recognize a note among the N chosen; the input pattern (the output of the signal's pre-processing phase), is FW through each net: if the out of the single net is above a threshold, this means that the note is ON, OFF otherwise.

3 The Discriminative Learning for Multi-classification Problems

3.1 The Discriminative Learning

One drawback of traditional approaches to pattern classification is that the estimation error does not immediately translate into correct recognition: the standard MLP uses a minimum mean square error (MMSE) criterion that doesn't necessarily minimize the classification error rates. In [12], Juang and Katagiri introduced a new formulation for the minimum error classification (MEC) problem called discriminative learning.

Let's consider a k-dimensional feature vector **x**; a linear discriminant function could be defined as:

Each LRNN is trained to recognize a note among the N chosen; the input pattern (the output of the signal's pre-processing phase), is FW through each net: if the out of the single net is above a threshold, this means that the note is ON, OFF otherwise.

4 The Discriminative Learning for Multi-classification Problems

4.1 The Discriminative Learning

One drawback of traditional approaches to pattern classification is that the estimation error does not immediately translate into correct recognition: the standard MLP uses a minimum mean square error (MMSE) criterion that doesn't necessarily minimize the classification error rates. In [12], Juang and Katagiri introduced a new formulation for the minimum error classification (MEC) problem called discriminative learning.

Let's consider a k-dimensional feature vector **x**; a linear discriminant function could be defined as:

$$g(\mathbf{x}) = \mathbf{w}\mathbf{x}^T + w_0 \tag{1}$$

where **w** represent the weight vector and w_0 the threshold.

If we have a pattern recognition problem with M classes, we'll have M discriminant function, and so a classifier parameter set Λ:

$$\Lambda = \{\boldsymbol{\lambda_1}, \boldsymbol{\lambda_2}, \ldots, \boldsymbol{\lambda_M}\} = \{\mathbf{w_1}, w_{01}, \mathbf{w_2}, w_{02}, \ldots, \mathbf{w_M}, w_{0M}\} \tag{2}$$

where $\boldsymbol{\lambda_i} = \{\mathbf{w_i}, w_{0i}\}$

Let the feature vector element $x_0 = 1$, each discriminant function could be written as:

$$g(\mathbf{x}) = \mathbf{w}\mathbf{x}^T + w_0 x_0 = \boldsymbol{\lambda_i}\mathbf{x}^T \tag{3}$$

This classifier uses the following decision rule:

$$C(\mathbf{x}) \in C_i \qquad \text{if } g_i(\mathbf{x}, \Lambda) = \max_j g_j(\mathbf{x}, \Lambda) \tag{4}$$

Thus, a feature vector $\mathbf{x}$ belongs to the class C_i that has the maximum value of the discriminant function; having linear discriminant functions brings hyperplan decision boundaries.

Despite the fact that learning with MMSE criterion does not necessarily lead to MEC, due to the computational efficiency, the determination of the classifier parameters set is usually formulated as a MMSE procedure using an objective function weighted by a discriminate function.

In order to derive the new objective criterion, the traditional discriminant formulation, have to be replaced with the following three-step procedure:

1. Determination of the form of the discriminant functions $g_i(\mathbf{x}, \Lambda)$
2. Determination of a quantity that indicates whether an input token x of the k-th class is to be misclassified according to the design rule of [7], implemented by the classifier parameter set Λ. This quantity is known as misclassification measure and one reasonable possibility is:

$$d_k(\mathbf{x}) = -g_k(\mathbf{x}, \Lambda) + \left[\frac{1}{M-1} \sum_{i,j \neq k} g_j(\mathbf{x}, \Lambda)^\eta \right]^{\frac{1}{\eta}} \tag{5}$$

3. Formulation of the minimum error objective. A general form of the cost function can be defined as:

$$\ell_k(d_k(\mathbf{x}) = d_k(\mathbf{x}, \Lambda) \tag{6}$$

Note that the cost function ℓ_k and the misclassification measure d_k can be defined individually for each class k. Two of the several possibilities for the cost function are the exponential or the translated sigmoid: both of them are smoothed zero-one cost functions suitable for gradients algorithms. Clearly, a correct classification have no costs, instead a misclassification leads to a penalty that becomes a count of the classification error determined by the loss defined above.

Finally, for any unknown $\mathbf{x}$, it is possible to define an empirical average cost as:

$$J(\mathbf{x}, \Lambda) = \sum_{k=1}^{M} \ell_k(\mathbf{x}, \Lambda) a \tag{7}$$

where $a = \begin{cases} 1 & \text{if } \mathbf{x} \in C_k \\ 0 & \text{otherwise} \end{cases}$

This classifier performance function is the basis of the objective that we shall optimize with descent methods as the MMSE: $\Lambda_{t+1} = \Lambda_t + \epsilon \nabla J(\Lambda_t)$. The Probabilistic Descent Theorem [13] assures the convergence of Λ_t to a locally optimum solution Λ^*.

In [12] it is possible to find the complete MEC formulation for MLP. The MLP structure for an M classes classification problem is depicted in Figure 3: the network has M output neurons, each one of which models the discriminant function of the DL formulation. Since an input x belongs to the class with the highest discriminant function value among the M discriminant functions values, we expect that the correspondent output neuron of the MLP has the highest value among the M output neurons values.

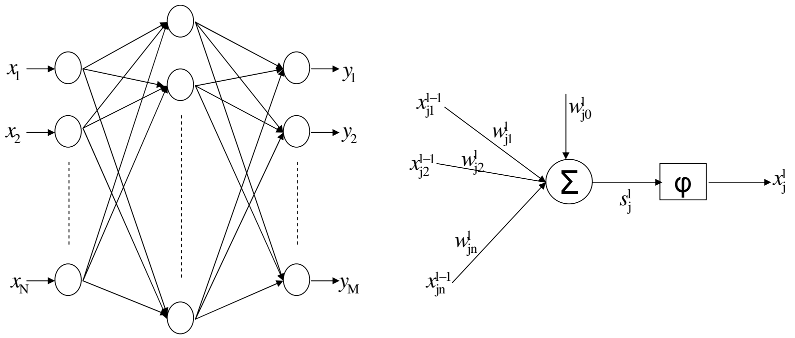


Fig. 3. MLP for M classes classification problem and neuron-j, level-l, n-input.

4.2 Multi-classification Discriminative Learning Algorithm

The DL model is attractive but it can't be applied directly to our architecture, in fact it is required that each input pattern belongs to a single class; instead, our model can contain input patterns that belong to different classes, or in other words, sounds made up of different notes (polyphony). We need to re-consider the basic idea of discrimination: we need a procedure that takes into account the fact that there's the possibility of a not complete separation between classes and that some input could lie in that common territory; we need to extend the MEC for MLP to MCDL for LRNN.

Intuitively, we could apply a sort of superimposition of effects procedure in which we consider a multiple classes input $\mathbf{x} \in C_p \subseteq \bigcup_{k=1}^{M} C_k$ as belonging separately to each one of the right p classes, and then combine the results. The application of this procedure to the standard weight adjustment rule for neuron in Figure 3:

$$\varDelta w_{ji}^{l} = -\eta \frac{\partial \ell}{\partial s_j^l} \frac{\partial s_j^l}{\partial w_{ji}^l} = \delta_j^l x_i^{l-1} \tag{8}$$

leads to the following modification:

$$\varDelta w_{ji}^{l} = \sum_{a \in \{C_p\}} \varDelta w_{ji}^{l}(a) = x_i^{l-1} \sum_{a \in \{C_p\}} \delta_j^l(a) \tag{9}$$

where $\varDelta w_{ji}^{l}(a)$ and $\delta_j^l(a)$ are respectively the weight adjustment rule and the delta rule for an input $\mathbf{x}$ that belongs to the a-th right class among the p, according to the superimposition of effects rule.
Let's consider the delta rule for the last layer:

$$\begin{cases} \delta_j(a) = -\eta \frac{\partial \ell(a)}{\partial s_j} = -\eta \frac{\partial \ell(a)}{\partial d_j}\frac{\partial d_j}{\partial y_j}\frac{\partial y_j}{\partial s_j} = -\eta \ell'(\partial d_a)\frac{\partial d_j}{\partial y_j} \\ \frac{\partial y_j}{\partial s_j^l} = \varphi'(s_j) = 1 \qquad \text{because of the final linearity} \end{cases} \tag{10}$$

$$d_a = -y_a + \left[\frac{1}{M-p}\sum_{k \notin \{C_p\}} y_k^{\eta}\right] \tag{11}$$

$$\frac{\partial d_a}{\partial y_k} = \begin{cases} -1 & k \in \{C_p\} \\ \frac{y_k^{\eta-1}}{M-p}\left[\frac{1}{M-p}\sum_{k' \notin \{C_p\}} y_{k'}^{\eta}\right] & k \notin \{C_p\} \end{cases}$$

Combining the results above:

$$\mathit{\Delta} w_{ji}^l = \sum_{a \in \{C_p\}} \delta_j(a) = \begin{cases} -\eta \mathbf{x}_i^{l-1}\frac{\partial d_a}{y_j}\ell'(d_a) & j \in \{C_p\} \\ -\eta \mathbf{x}_i^{l-1}\frac{\partial d_a}{y_j}\sum_{k \in \{C_p\}} \ell'(d_a) & j \notin \{C_p\} \end{cases}$$

where $\frac{\partial d_a}{y_j}$ is replaced by its right expression. If an input $\mathbf{x}$ belongs to no classes, no adjustment is made ($\mathit{\Delta} w_{ji}^l = 0$).

We also have to redefine the classifier decision rule: an input $\mathbf{x}$ belongs to a set of classes $\{C_p\}$, for each one of which, the discriminant function satisfies a quantification belonging condition rule as for example a thresh function or a domain integrity test.

5 The MCDL LRNN for ATM

The final model is depicted in Figure 4:

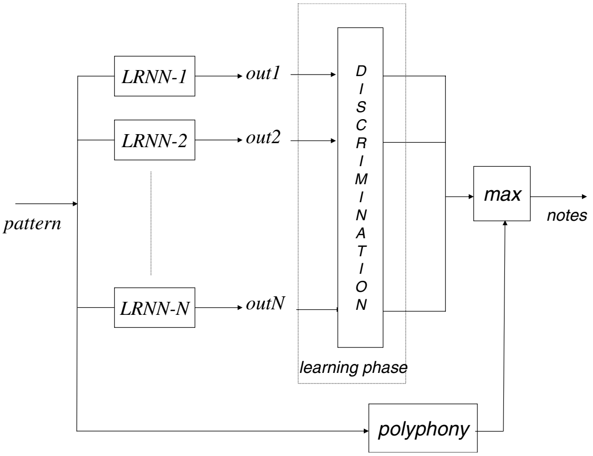


Fig. 4. MCDL model for Automatic Music Transcription.

After a pre-training phase, in which each LRNN is trained singularly to recognize its target note, a discriminative over-training is made in order to assure generalization and relationship between nets (that is: which is other nets' behaviour when an input pattern is presented to the single net).

There's the necessity to introduce a polyphony net in order to establish how many outputs to take, that is, how many notes are playing (are ON) at a particular instant: if the polyphony net outputs m, the highest m outs are taken. We choose as polyphony net a LRNN that has to be trained independently from other LRNNs, since it has a different task from the others.

6 Experimental Results

The experimental results reported here are of two kinds: the first is a comparison between the model depicted in Figure 2 and a TDNNs model (similar to SONIC), in order to demonstrate the advantages in using LRNNs and the drawbacks of such approaches respect to discriminative ones; the second is mainly the description of the growing learning process of the MCDL model and its strength respect to the model in Figure.

All the networks used in these experiments had two layers, three hidden neurons with hyperbolic activation spline function, one output linear neuron and a variable number of input (generally 200) depending on the parameters chosen for the CQT analysis (frequency range and the resolution). We used MA-AR:2-1 for the first layer in LRNN and MA:2 for the first layer in TDNN.

All the wave samples used for training/testing are random pieces (16 bit - Mono - 11025Hz) generated using the Roland Virtual Sound Canvas VSC-88: these pieces has a maximum polyphony level of 5 and a minimum note length of 50ms.

6.1 TDNN Model vs. LRNN Model

We used nearly 1000 wave samples in the range [C3; B5] as training and testing sets. The LRNN model reveals its efficacy respect to the TDNN both on training and testing sets, in particular, the LRNN model is able to identify correctly notes that are misclassified by the TDNN with octave errors.

One drawback with this scheme is the redundancy of training, in fact each network has to be trained independently with a specific training set. Moreover, there's the need to construct larger training sets in order to minimize the classification error on testing sets that are substantially different from the training sets (that is: testing sets that contain different music styles respect to the corresponding training sets): however, very large training sets makes the learning a real difficult task.

That's why there's the necessity to introduce a new model that is able to overcome these difficulties.

6.2 MCDL for ATM

We used single note wave files (with different lengths) as training set for all of the LRNNs in the scheme. After an individual pre-training phase in which every LRNN learns its target note from its single note wave files, a discriminative over-training is made (with the whole training set), in order to relate each LRNN to the other.

The model is able to classify exactly single notes and monophonic sequences, without the need to enlarge the training set. That's an interesting result, and reveals the strength of the DL respect to a standard LRNN model: it's impossible for a standard LRNN model to obtain this result, conditions being equal.

If we over-train again the monophonic-model obtained before, with growing levels of polyphony wave files, using the MCDL algorithm, we are able to obtain again correct classification, without wasting the preceding results.

Obviously, when working with high levels of polyphony, there's the necessity to reconsider the preceding over-trainings steps and vary the η parameter of discrimination: generally, a short value for η is used when polyphony grows.

7 Conclusions

In this paper, we presented a new approach to Automatic Music Transcription of Polyphonic Piano Music: the MCDL LRNN model. This model reveals its efficacy in most of typical Music Transcription problems combining the advantages of LRNNs (respect to other dynamic pattern recognition techniques) and discriminative learning (respect to other classification techniques).

The multi-classification technique is the real novelty of this approach: its usage makes the learning simpler and faster, making useless the build of separated training sets for each notes, giving coherence and generalization to the whole model.

Because of the novelty of this approach, several tests there have to be made besides Music Transcription context, in order to understand the exact of efficacy of this technique. Extension of this system to higher levels of polyphony, shorter note's length and different pre-processing techniques have to be considered. Additionally, a post-processing block (a neural network or an hidden Markov model) may be considered to correct errors.

References

1. Moorer, J.A. (1975), On the Segmentation and Analysis of Continuous Sound by Digital Computer, Ph.D.Thesis, Department of Computer Science, Stanford University, Stanford
2. Brown J. C. (1988), Calculation of a Constant Q Spectral Transform, J. Acoust. Soc. Am., 1991
3. Guillermain P., Kronland-Martinet R. (1996), Characterization of Acoustic Signal through Continuous Linear Time-Frequency Representations, Proc. IEEE, vol.84, no.4, pp.561-585

4. Klapuri A. (1997), Automatic Transcription of Music, Master of Science Thesis, Tampere University of Technology
5. Sterian A., Wakefield G. H. (1998), A model-base approach to partial tracking for musical transcription
6. Kashino K., Nakadai K., Kinoshita T., Tanaka H. (1995), Application of Bayesian probability network to music scene analysis, Proceedings of International Joint Conference in AI, Workshop on Computational Auditory Scene Analysis, Montreal, Canada
7. Dixon S. (2000), On the Computer Recognition of Solo Piano Music, Proceedings of Australian Computer Music Conference, Brisbane, Australia
8. Marolt M., SONIC: Transcription of Polyphonic Piano Music with Neural Networks, Ph.D. Thesis
9. Campolucci P., Piazza F. Uncini A. (1999), On-line Learning Algorithm for Locally Recurrent Neural Networks, IEEE Trans. On Neural Networks, vol. 10, no. 2
10. L. Vecci, F Piazza, A. Uncini, "Learning and Approximation Capabilities of Adaptive Spline Activation Function Neural Networks", Neural Networks, Vol. XI, No.2, pp. 259-279, 1998
11. Campolucci P., Piazza F. (2000), Intrinsic Stability-Control Method for Recursive Filters and Neural Networks, IEEE Trans. On Circuits and Systems - II: Analog and Digital Signal Processing, vol.47, no.8
12. Juang B. H., Katagiri S. (1992), Discriminative Learning for Minimum Error Classification, IEEE Trans. On Signal Processing, vol. 40, no. 12
13. Katagiri S., Lee C. H., Juang B. H. (1991), New discriminative algorithm based on the generalized probabilistic descent method, Proc. 1991 IEEE Workshop Neural Netoworks for Signal Processing, Piscataway, NJ, pp. 299-308
14. J. C., Puckette M. S. (1992), An efficient algorithm for the calculation of a constant Q transform, J. Acoust. Soc. Am., 92(5):2698-2701

An Adaptive Learning Algorithm for ECG Noise and Baseline Drift Removal

Anna Esposito[1,2] and Pierluigi D'Andria[2]

[1] Dept. of Psychology, Second University of Naples, Via Vivaldi 13, (CA), Italy
{iiass.anna}@tin.it
[2] International Institute for Advanced Scientific Studies, Via Pellegrino 19
Vietri - INFM, Salerno, Italy
{p.dandria}@inwind.it

Abstract. Electrical noise and power line interference may alter ECG morphology. Noise reduction in ECG is accomplished applying filtering techniques. However, such filtering may mutate the original wave making difficult the interpretation of pathologies. To overcome this problem an adaptive neural method able to filter ECGs without causing the loss of important information is proposed. The method has been tested on a set of 110 ECGs segments from the European ST-T database and compared with a recent morphological filtering technique. Results showed that morphological filters cause inversions and alterations of the original signal in 65 over 110 ECGs, while the neural method does not. In 96% of the cases the signal processed by the network is coherent with the original one within a coherence value of 0.92, whereas this values for the morphological filter is 0.70. Moreover, the adaptability of the neural method does not require estimating appropriate filter parameters for each ECG segments.

1 Introduction

Electrocardiogram (ECG) refers to the graph that results from plotting time versus voltage in a patient chest. Voltage sensors are located on the chest and the signal they pick up is directed to an electrocardiograph, an apparatus that plots such a signal. Electrocardiograms (ECGs) represent the recording of the heart's electrical potential, and therefore ECG's reading and interpretation is a useful tool for physicians to diagnose heart's diseases. One of the major problems when recording signals so weak as the ECG is noise. There are several kinds of noise that can affect ECGs but the strongest are the 50-60 Hz interference from the main power distribution, and the 1Hz power line interference due to the movement of patients, the erroneous electrode's fixing and the amplifier's drift [1]. An ECG with such noise is shown in Fig. 1. Its temporal representation clearly shows a baseline drift and a displacement from the 0 reference value (the signal minimum value is around 100) of the amplitude values. During the last twenty years, the literature has proposed several solutions to remove the power line interference from ECG signals. Most of these solutions are based on ECG filtering. Filters solutions fall into four common categories [2]:

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 139–147, 2003.

1) Low pass filters that also severely attenuate important frequency components of the ECG signal;
2) General notch-rejection filters which can be grouped in two categories: IIR and FIR filters. FIR filters behave similarly to low pass filters, attenuating the entire signal content together [3]. IIR filters instead causes undesirable distortions especially after the QRS complex [4];
3) Adaptive filters remove interferences from ECG using as reference input a pure power line noise [5] [6]. This filtering method leave the source signal undistorted, but it cannot follow fast changes in the interference amplitude producing an undesired ringing effect [7];
4) Global filters have the main drawback to produce an inter-beat average difference and to be not very suitable for real-time implementations [8].

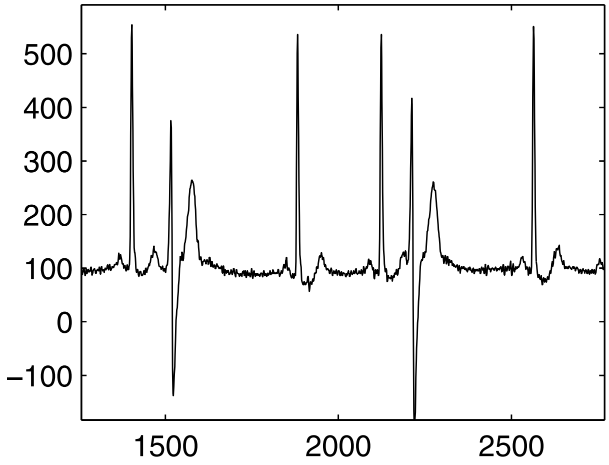


Fig. 1. ECG with noise and shift of the amplitude values from the zero reference value.

A detailed characterization of digital filtering methods for ECG noise removal can be found in [2] and [7], which also proposed an incremental adaptive digital filtering method for the removal of power line interferences. It appears clear from the above discussion that it is impossible to filter ECG at a specific or over a given frequency range without losing important information that can produce an erroneous interpretation of the signal. It is also important to underline that none of the above filtering techniques was able to remove the baseline drift and to rectify the signal around the 0 reference value. A better approach to overcome the limitations discussed above is to use a filtering techniques based on morphological filters [9], which has been showed to work better than the traditional filtering techniques. Sedaaghi (1998) claims that morphological filters are both able to remove the baseline drift interferences and rectify the signal amplitude around zero. Neural Networks were mostly used for ECG classification

[10]. Neural network for signal processing have suggested by several authors [11], [12], [13]. However, in ECG processing, their use was limited to classification tasks (see [14] and [15] among others). The present work proposes a filtering techniques based on an adaptive learning algorithm. The performance of the proposed method are assessed through a comparison with morphological filters [9]. It will be showed, in the following paragraphs, that the proposed algorithm performs better than morphological filters, eliminating either the baseline drift and rectifying the amplitude values around zero without altering the original ECG signal.

2 Materials and Procedures

The ECG waves were extracted from the European ST-T database. The database represents the results of an international effort for defining a standard in the analysis of ST- T changes in the ECG's waves from ambulatory recordings (AECG). The project was founded by the European Community and by the European Society of Cardiology [14], [15]. Thirteen research groups from eight European countries (coordinated by the Institute of Clinical Physiology (National Research Council) in Pisa (Italy) and by the Thorax Center of Erasmus University in Rotterdam (Netherlands)) provided the AECG tapes and annotated beat by beat the selected 2-channel recordings, each of 2 hours in durations. The annotation scheme was revised in cooperation with the Biomedical Engineering Centre of MIT such that it was consistent with both the MIT-BIH database format (MIT-BIH distribution). The European ST-T database consists of 90 annotated excerpts of AECG recordings from 78 subjects (70 males aged from 30 to 84, and 8 females aged from 55 to 71), each 2 hours in duration, sampled at 250 Hz and encoded with a 12-bit resolution over a nominal $\pm 10mV$ range. Pathologies are annotated in each recording. From this database we extracted 110 pathologic and non pathologic ECG segments from the leads V2, V3, V4, V5 and MLIII, 100 segments (20 from each lead) of 40s in duration and 10 segments (2 from each lead) of 160s in duration. On these data we performed our noise filtering experiments.

3 The Adaptive NeuralNetwork

The adaptive neural network was a two layer MLP with fifteen input nodes and three output nodes. The net is adaptive in the sense that it never stop to learn, adapting its behaviour to the noise's trajectory. The number of hidden nodes depends on the lead from which the ECG was recorded. There were 15 hidden nodes for the MLIII, V2 and V5 leads, and 5, and 10 hidden nodes respectively for the V3 and V4 leads. The learning rate was fixed to $\eta = 0,4$. The activation functions were hyperbolic tangents in the first hidden layer, and linear functions in the output layer. To remove the noise, the idea is let the network to predict it from the original ECG. To this aim, the network, at each learning step, takes as input 15 ECG samples and estimates from them the noisy part of the

ECG. This is done along all the ECG length. In particular, at the presentation of each input pattern, the network predicts three noisy samples reconstructing along the time the entire noisy wave that is then subtracted from the original ECG in order to obtain the filtered one. This process is implemented minimizing the error function: $E(i) = E(t(i), y(i)) = (t(i) - y(i))^2$ between the target $t(i)$ for the noisy sample i and the network output $y(i)$ through the updating of the weights. However, if the learning procedure is left without control, the net prediction will also include the prediction of the QRS complex and the P and T waves which are not noise but crucial parts of the ECG signal. In order to avoid that, the network learning is stopped in proximity of the QRScomplex and the P and T waves. This is done monitoring the value of the prediction error during the learning procedure, through an *adaptive threshold.* Currently, the net must commit the maximum prediction error when it reach a signal segment corresponding to an ECG peaks. If this error is greater than the adaptive threshold (which is computed along the signal prediction) the learning is stopped and the prediction of the following noisy samples is done simulating the prediction of the signal previously learned. This process is kept as long as the network error value for the successive ECG sample is above the threshold. The learning start again after this phase. The threshold is dynamically computed during the learning as the absolute value of the error between the *ith* ECG sample and the *ith* network output, averaged over the number N of ECG samples processed until the current sample n, i.e.:

$$Threshold(n) = abs(\frac{1}{N}\sum_{i=1}^{N}(t(i) - (y(i)) \ . \tag{1}$$

However, the predicted noise estimated during the learning has a zigzag trajectory due to the ECG over-sampling. The zigzag can cause the lost of important information when subtracted from the original signal. Therefore, as a final prediction step, a smoothing procedure that uses a median filter [16] is applied. The behaviour of the net on a given ECG signal is depicted in Fig.2, 3, 4. Fig. 2 shows the predicted noisy wave (dashed line) on the ECG signal (solid line) reported in Fig.1 without the application of the median filter. Fig.3 shows the effect of the median filter (solid line) on the predicted noisy wave (dashed line). Fig.4 shows the filtered and rectified ECG.

4 Comparison with Morphological Filters

To assess the performance of our procedure we compared the network predictions with those obtained applying the morphological filters [9]. As an example, Fig.5 reports both the morphological filter performance (on the top) and the net performance (on the bottom) on the ECG signal in Fig.1. From a visual inspection of the signal it is clear that the morphological filters modify the form of the original signal. In particular, in the range of 2500-3000 samples there is an inversion of the ECG wave (top of Fig.5) with respect to the original signal (see Fig.1) that does not appears when the filtering is made with the neural network

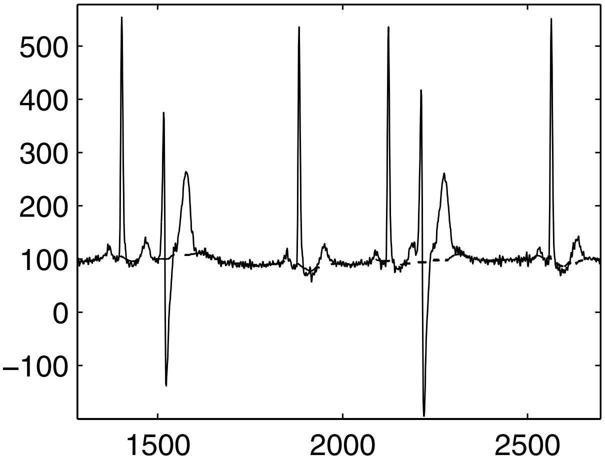


Fig. 2. The same ECG as in Fig.1 (solid line) and the network output (dashed line).

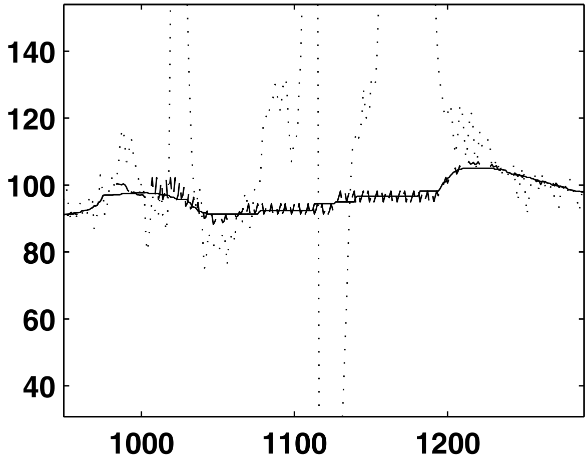


Fig. 3. The effect of the median filter (solid line) on the network output (dashed line) over the original ECG (dotted line).

(bottom of Fig.5). A visual analysis performed over all the 110 ECGs constituting our database shows that the morphological filters mutate the original signal in 65 over 110 cases, whereas the neural net does not [17]. A quantitative estimate of the network performance was made using the *coherence function* in [18]. The coherence $C_{xy}(\omega)$ is a function of the frequency ω with values between 0 and 1 that indicate how well the input signal x corresponds to the output signal y. The mathematical formulation is reported below:

$$C_{xy}(\omega) = \frac{|P_{xy}(\omega)|^2}{P_{xx}(\omega)P_{yy}(\omega)} \ . \tag{2}$$

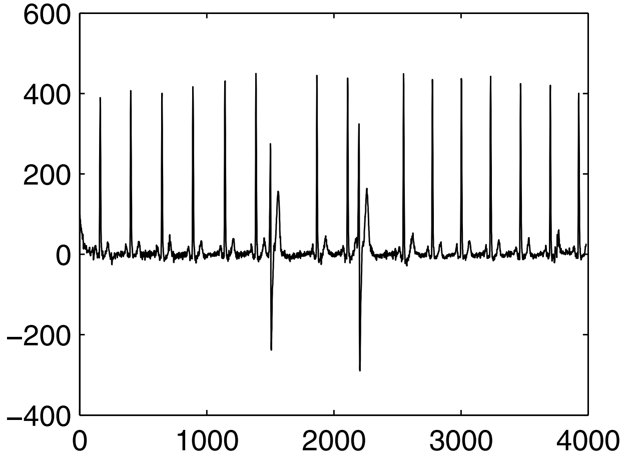


Fig. 4. The ECG filtered and rectified.

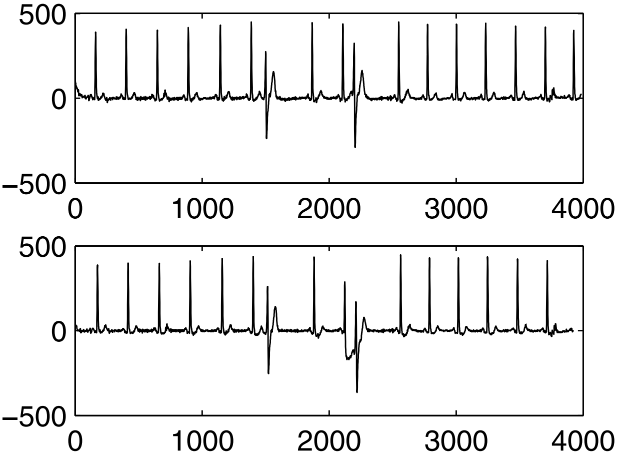


Fig. 5. The ECG in Fig.1 filtered by the morphological filters (top) and the neural network (bottom). The morphological filter produce an inversion of the ECG wave during the a pathological event (sample range [3000,3050]).

were $P_{xy}(\omega)$ is the cross spectral density as defined in [18]. As an example, the coherence between an original ECG and its filtered version by the net (dashed lines) and by the morphological filters (solid lines) is reported in Fig.6. It appears clear from Fig.6 that either the net and the filter are able to cancel the 1 Hz component of the baseline drift. However, as we move along the signal, the net prediction is more close than the filter prediction to the original signal. Detailed results on the coherence values are given in Table 1 for both the morphological

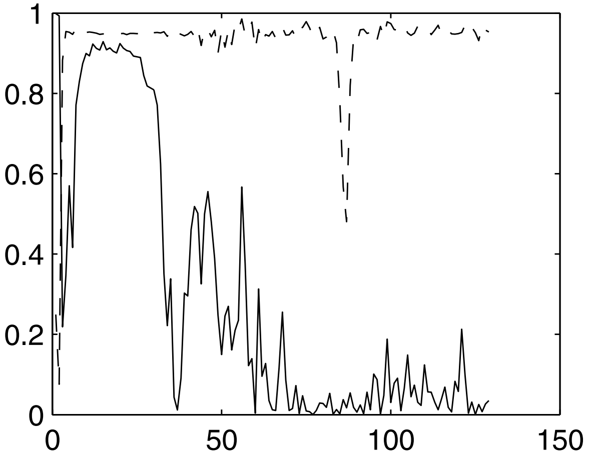


Fig. 6. Coherence values among a non-filtered ECG and the same ECG filtered by the neural network (dashed line) and the morphological filter (solid line).

filter and the adaptive neural network. As showed in Table 1, the net coherence value is always more close to 1 than the filter coherence value. This is true, in general for all the database under examination. The coherence values for all the ECG of the present work can be find in [17].

5 Conclusions

This work suggests a novel procedure for the baseline drift removal in ECG that employs an adaptive neural network. The assessment of the network performance was made on a set of 110 ECG segments extracted from the European ST-T database. Results show that the net is able to filter and rectify the signal without mutate its original form thus preserving important information that can be lost with standard filtering techniques. It was also showed that the net performance are qualitatively and quantitatively better than morphological filters that were shown to be the best filtering technique proposed up to now. In 96% of the cases the average coherence value between the original ECG and that filtered by the neural network is close to 1, whereas coherence values obtained using morphological filters were lower. Moreover, morphological filters cause inversions and ECG wave alterations in 65 over 110 (59%) cases, whereas the neural method does not. Furthermore, the adaptive neural network does not require the estimation of appropriate filter parameters for each ECG segments.

Acknowledgements

We would like to thanks the Guido Aversano, Adele Bolino, Antonietta Esposito, Maria Marinaro and A. Taddei for their useful suggestions and comments in developing this work.

Table 1. Averaged coherence values computed between non-filtered ECGs (from the lead V4) and those filtered by the neural network (first column) and by the morphological filter (second column).

Learnig Algorithm	Morphological Filters
0.9430	0.3790
0.9391	0.4240
0.9490	0.8679
0.9318	0.7651
0.9510	0.8540
0.9411	0.5237
0.9390	0.3218
0.9507	0.7656
0.9421	0.6908
0.9496	0.9350
0.9422	0.6808
0.6773	0.0354
0.9396	0.5232
0.9439	0.6659
0.9246	0.3288
0.9158	0.2974
0.9911	0.3203
0.9291	0.6167
0.9444	0.8931
0.9276	0.5499
0.9103	0.5207
0.9269	0.5341

References

1. Hutha, J.C., Webster, J.G.: 60Hz interferences in electrocardiography. IEEE Trans. On Biomed. Eng., 2, BME-20 (1973) 91-101.
2. McManus, D., Neubert, K.D., Cramer, E.: Characterization and elimination of AC noise in electrocardiograms: a comparison of digital filtering methods. Computers and Biomedical Research, 26 (1993) 48-67.
3. Van Alste, J.A., Schilder, T.S.: Removal of base line wander and power-line interference from ECG by an efficient FIR filter with a reduced number of taps. IEEE Trans. on Biomed. Eng., 12, BME-32, (1985) 1052-1060.
4. Herrera-Bendez-, L.G.: Real time digital filters for ECG signals: evaluation and new designs. IEEE Computers in Cardiology, 2(12) (1992) 133-136.
5. Ferrara, E.R., Widrow, B.: Fetal electro-cardiogram enhancement by time sequenced adaptive filtering. IEEE Trans. on Biomed. Eng., BME-29 (1982) 458-469.
6. Almenar, V., Albiol, A.: A new adaptive scheme for ECG enhancement. Signal Processing, 75, (1999) 253-263.
7. Limacher, R.: Removal of power line interference from the ECG signal by an adaptive digital filter. In Proc. of European Telemetry Conference (ETC-1996), Gramish-Part, May 21-23 (1996) 300-309.

8. Barr, R.E., Fendjallah, M. Frequency domain digital filtering techniques for selective noise elimination in biomedical signal processing. IEEE Computers in Cardiology, 2(12) (1990) 813-814.
9. Sedaaghi, M.H.: Wave detection using morphological filters. Applied Signal Processing. 5 (1998) 182-194.
10. Papaloukas, C., Fotiadis, D.I., Likasb, A., Michalisc, L.K.: An ischemia detection method based on artificial neural networks. Artificial Intelligence in Medicine, 24 (2002) 167-178.
11. Piazza, F., Uncini, A., Zenobi, M.: Neural Network with digital LUT activation functions. In Proceedings of IJCNN, 2, Nagoya, Japan (1993) 1401-1404.
12. Guarnieri, S., Piazza, F., Uncini, A.: Multilayer neural networks with adaptive spline based activation functions. In Proceedings of WCNN, Vol. 1, Washington (1995) 739-742.
13. Chen, T., Chang, W.D.: A feedforward neural netvork with function shape autotuning, Neural Networks, Vol. 6(9) (1996) 627-641.
14. Taddei, A., Biagini, A., Distante G.,et al.: An annotated database aimed at performance evaluation of algorithms for ST-T change analysis. Computer in Cardiology, Vol.16 (1990) 117-120.
15. Taddei, A., Distante, G., Emdin, M., Pisani, P., Moody, G.B., Zeelemberg, C., Marchesi C.: The European ST-T database: standard for evaluating systems for the analysis of ST-T changes in ambulatory electrocardiography. European Heart Journal, 13 (1992) 1164-1172.
16. Pratt, W.K.: Digital image processing. Wileyi & Sons New York (1978) 330-333.
17. D'Andria, P.: Neural Techni ques for baseline drift removal from ECG signals (In Italian). Laurea Thesis, Salerno University, Department of Physics "E. R. Caianiello" (September 2001).
18. Lynn, P.: Analysis and processing of signals. Chapter 5, The MacMillan Press LTD (1973).

Towards an Automatic Lesion Segmentation Method for Dual Echo Magnetic Resonance Images Using an Ensemble of Neural Networks

Andreas Hadjiprocopis* and Paul Tofts

Institute of Neurology
University College London
Queen Square, LONDON, WC1N 3BG, UK
A.Hadjiprocopis@ion.ucl.ac.uk

Abstract. There is a well recognised need for a robust, accurate and reproducible automatic method for identifying multiple sclerosis (MS) lesions on proton density (PD-weighted) and T_2-weighted magnetic resonance images (MRI).

Feed-forward neural networks (FFNN) are computational techniques inspired by the physiology of the brain and used in the approximation of general mappings from one finite dimensional space to another. They present a practical application of the theoretical resolution of *Hilbert's* 13^{th} *problem* by Kolmogorov and Lorenz, and have been used with success in a variety of applications.

We present a method for automatic lesion segmentation of fast spin echo (FSE) images (PD-weighted & T_2-weighted) based on an*ensemble* of feed-forward neural networks. The FFNN of the input layer of the ensemble are trained with different portions of example lesion and non-lesion data which have previously been hand-segmented by a clinician. The final output of the ensemble is determined by a gate FFNN which is trained to weigh the response of the input layer to unseen training data. The scheme was trained and tested with data extracted from brains suffering from MS. The results are presented.

1 Introduction

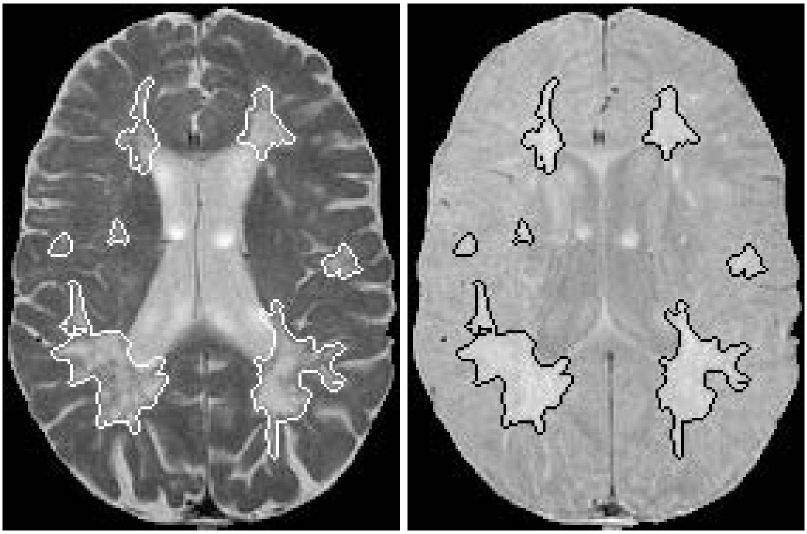
1.1 Multiple Sclerosis and MRI

Multiple sclerosis is a chronic, often disabling disease that attacks the central nervous system. It involves an immune-system attack against the insulating myelin coating of nerve fibers in the brain and spinal cord. The attack can strip myelin from the nerve fiber and leave a scarred lesion that can be seen on magnetic resonance imaging (MRI) scans. In MS, myelin is lost in multiple areas, leaving scar tissue called sclerosis.

Magnetic resonance imaging has contributed to the understanding of multiple sclerosis in several ways. It is valuable in the diagnosis of MS, in understanding

* Supported by a grant from the MS Society, UK.

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 148–157, 2003.



(a) T_2-weighted (b) PD-weighted

Fig. 1. Example hand-segmented images. Lesions are outlined

the nature of the disease process, and as a tool in clinical trials to provide a rapid and objective means of assessing the influence of treatment on the course of MS.

Three types of MRI scans are commonly used to investigate MS, all of which are sensitive to the higher than normal water content found in MS lesions and the increase in relaxation times T_1 and T_2. They are obtained by manipulating the radiowave pulses used in MRI in different ways, and are called T_1-weighted, T_2-weighted, and proton density (PD-weighted) scans.

In practice, patients are scanned with a small subset of the whole range of MRI protocols because of time constraints and other considerations. In our case, the available data set comprises of T_2-weighted and PD-weighted scans.

Figure 1 shows a slice of a T_2-weighted (**a**) and PD-weighted (**b**) MRI scan of the brain of a patient with MS; lesions outlined by hand, are shown. The scanner settings used to obtain these images are given is section 2.1.

1.2 Lesion Segmentation

Lesion segmentation in MRI brain scans is usually a problem due to inaccuracies because of various image inhomogeneities and artifacts, relatively limited MR scanner resolution, inherent tissue heterogeneity, signal overlap from the different tissue types and the effect of noise and partial volume averaging.

Existing manual techniques for identifying and segmenting multiple sclerosis brain lesions are time consuming and have limited reproducibility, especially between human raters. Given this, there is a clear need for a robust, accurate and reproducible automatic technique for identifying such lesions on T_2-weighted and PD-weighted images. Further, there are two additional reasons to develop such a method:

1. although MS high-signal lesions are pathologically heterogeneous their identification and quantification in itself is a useful objective measure of disease (for example, in the context of trials of new therapeutic agents),
2. there is an increasing number of newer analysis techniques that rely on segmentation of brain images into grey matter, white matter and cerebrospinal fluid (CSF) fractions. The application of these techniques is limited by their inability to deal with lesions.

1.3 Previous Studies

Previous studies have used automatic or semi-automatic methods utilising various techniques based on thresholding, region growing, interslice connectivity criteria, [1,2,3], knowledge-based systems, [4], and statistical models of tissue distribution in healthy brains, [5,6].

A combination of thresholding and artificial neural networks including a post-processing phase requiring manual input was used in [7] to identify MS lesions enhanced by Gadolinium (Gd), a paramagnetic medium injected to the patient before scanning. The use of a contrast agent, such as Gd, makes *active* lesions look brighter but it is relatively expensive and the injection might be uncomfortable for the patient. Additionally, older lesions might not be enhanced by Gd. This poses a problem as the lesion detection algorithm will have to account for both enhanced and non-enhanced lesions.

Lately, the use of fuzzy clustering methods has been applied in general MRI segmentation problems (e.g. segmentation of the various tissue types rather than lesions and other artifacts), [8]. Also, [9,10] provide a useful review on the application of supervised and unsupervised pattern recognition techniques in general MRI segmentation problems.

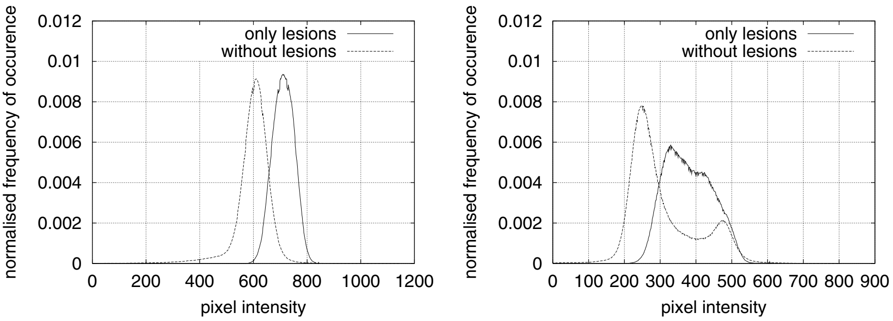
1.4 Feed-Forward Neural Networks

Feed-forward neural networks are computational techniques inspired by the physiology of the brain and used in the approximation of general mappings from one finite dimensional space to another.

From a mathematical point of view, FFNN are *universal function approximators*. Based on the resolution of Hilbert's 13^{th} problem by Kolmogorov and Lorenz, it was shown that a FFNN with a single hidden layer, employing as many non-linear units as required can approximate any continuous function arbitrarily well.

In the past several years research in exploring the ability of feed-forward neural networks to generalise on example data and predict on previously unseen data has yielded many successes in a wide variety of applications, [11].

The training of FFNN is exercised through a supervised process by which the network is presented with a sequence of inputs and their respective desired



(a) T$_2$-weighted **(b)** PD-weighted

Fig. 2. Pixel value histograms of lesion and non-lesion areas

outputs. "Learning" occurs by adjusting the free parameters of the model (the weights) in a way that the discrepancy between the actual and desired responses is minimised, for all the training data. The most common training algorithm is *error back-propagation* used in conjuction with the *steepest gradient descent* optimization method, [11].

2 Methods

2.1 MRI Protocol

Fast spin echo images were acquired with an echotrain length of 8 and two echoes on a SIGNA 1.5T system. The proton density echo had a TE= $15ms$ and the T$_2$-weighted echo had a TE= $90ms$. Both echoes had TR= $3s$, slice thickness of $3mm$, image matrix of 256×256 pixels and pixel size of $0.9375mm \times 0.9375mm$. All images were corrected for RF coil inhomogeneity by rescaling their intensity values, [12].

Additionally, the images were realigned and mapped into some standard anatomical space (e.g. a *stereotactic space*), [13] using the package SPM, [14]. The realignment procedure aims at correcting for the patient's movements during scanning. The mapping procedure estimates the warping parameters and maps each scan onto a template that already conforms to some standard anatomical space.

2.2 Input Images Data

The images of figure 1 show the same slice of the same brain for two different sets of MRI parameters. Fig. 1(**a**) corresponds to the T$_2$-weighted modality and fig. 1(**b**) corresponds to the PD-weighted. The outlined areas show lesions hand-segmented by a clinician.

The difficulty of segmenting lesions solely on the basis of their intensity is demonstrated by the histogram plots of figure 2. In fig. 2(**a**), the left curve corresponds to the histogram of the pixel intensities of a T$_2$-weighted volume (i.e. 44 slices from the base to the top of the head, 3 *mm* apart) with no lesions

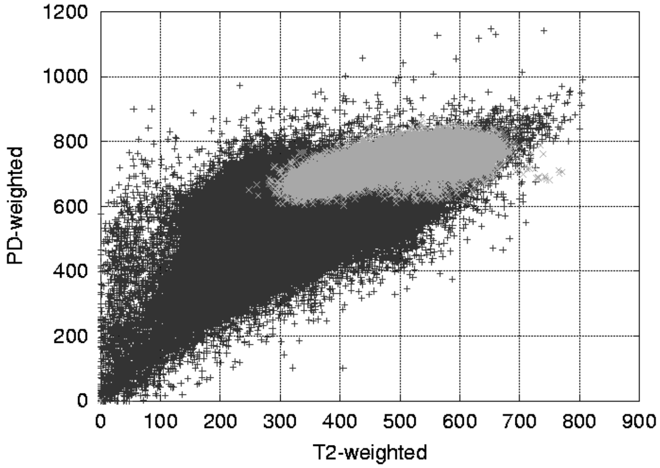


Fig. 3. A scatter plot of the pixel intensities of the images. The dark area corresponds to the whole image and the brighter area to the lesion areas

present. In the same figure, the right curve shows the histogram of the intensities of all the lesion voxels of the same volume (no normal voxels were present). The same applies for the plot of fig. 2(**b**) which refers to the PD-weighted volume.

Fig. 3 shows the scatter plot of pixel intensities of T_2-weighted versus PD-weighted volumes. The dark area shows the intensities of normal pixels from the volumes and the light area shows the intensities of lesion pixels. In both the histograms and the scatter plot an overlap between lesion and non-lesion pixel values is apparent.

2.3 Input Fields

Because the pixel intensities alone are not enough to separate lesions from normal appearing tissue, it is necessary to increase the dimensionality of the training data by deriving more input fields. This is done as follows:

1. *spatial information*:
   (a) 2D or 3D *absolute position* of each voxel in cartesian ($SA_{c,2D}$ or $SA_{c,3D}$), (x, y, z), and/or polar coordinates ($SA_{p,2D}$ or $SA_{p,3D}$), (r, θ, ϕ).
   (b) 2D or 3D polar coordinates (RD^{L}_{2D} or RD^{L}_{3D}) relative to one or more robust anatomical landmarks (L) previously selected by a clinician. There exist some structures in the brain which are consistent across patients and should be interesting to see the relative position of lesions with respect to these landmarks. For example, it is common to observe lesions around the central ventricle (this is clearer in fig. 1(**a**); the central ventricle is the bright area in the middle of the scan).
2. *Statistical indicators* of the pixel intensity of a *region* or *neighbourhood* ($\mathbf{W}$) around the voxel under consideration, such as *mean* ($SI^{\mathbf{W}}_{\mu}$) and *standard deviation* ($SI^{\mathbf{W}}_{\sigma}$).

3. *Absolute pixel values*[1] of the voxel under consideration (PAV_v) and/or absolute pixel values of a region ($\mathbf{W}$) around it ($PAV_r^{\mathbf{W}}$).

By using information other than absolute pixel values and by considering brain landmarks and regions around each voxel, a number of extra input fields are introduced. This additional information, it is hoped, will increase the degree of orthogonality between lesion and non-lesion voxels and will have a positive effect to the ability of the neural networks to differentiate between the lesion and non-lesion classes.

2.4 Creating the Training and Test Sets

Each voxel of the available image volumes can be described uniquely in terms of its spatial position (x, y, z) and the patient's unique ID, a. If all the available images form an array v, then the pixel value of each voxel is given by $v[x][y][z][a]$ as a set of two values, one for the $\mathrm{T_2}$-weighted and one for the PD-weighted modality.

A region around a voxel at (x, y, z, a) can be selected by specifying a few parameters. For example, shape (rectangular, elliptical), size, *etc.* Let all the parameters required to specify a region centred around a voxel be described by the set $\mathbf{S}^{x,y,z,a}$ and let all the voxels in that region be given by the set $\mathbf{W}^{x,y,z,a,\mathbf{S}}$.

A line of data to feed a neural network (call it F_1) can be formed once the data creation criteria enumerated in the previous section are decided. The set of these criteria (call them $\mathbf{C}^{F_1}$) are specific to the given neural network and will be the same for all lines of input requested by it. For example, the set of criteria: $\mathbf{C}^{F_1} = \{SA_{c,2D}, RD_{3D}^{L_1}, SI_{\mu}^{\mathbf{W}_1}, SI_{\sigma}^{\mathbf{W}_1}, PAV_v^{\mathbf{W}_1}, PAV_r^{\mathbf{W}_1}\}$, will create lines of input to the neural network F_1 which, for each voxel in the image data, will contain the voxel's cartesian 2D coordinates, 3D polar coordinates of that voxel relative to the brain landmark L_1, mean and standard deviation of the pixel values of a region ($\mathbf{W}_1$) around the voxel, the absolute pixel values of the given voxel and the values of the voxels in the region $\mathbf{W}_1$.

Depending on whether the specified voxel is a lesion or a non-lesion, the output class of the created line of data will be 0 or 1.

In summary, the procedure for constructing a data set to be used in conjuction with any FFNN (call it F_i) is as follows:

1. choose the number of lines in the data set, N,
2. choose the set of criteria, $\mathbf{C}^{F_i}$. For this, it may be necessary to:
   (a) define one or more brain landmarks (optional),
   (b) define one or more regions (optional),
3. draw N voxels from the images by randomly choosing the voxel's location (x, y, z, a),

[1] Because we are dealing with multi-spectral images,the term 'pixel intensity' refers to the set of the voxel intensities in each image modality. In our case these are two: $\mathrm{T_2}$-weighted and PD-weighted.

4. for each of the selected N voxels construct a line of input $L_i^{x,y,z,a,\mathbf{C}^{F_i}}$ and label it with 0 if it is a normal voxel or 1 if it is a lesion voxel,

The above procedure along with region, brain landmark and input field criteria specifications must be associated with F_i. Every time more training or test data is needed for the given FFNN, the same procedure must be followed exactly.

2.5 The Neural Networks and the Ensemble

The number of available images hand-segmented by human raters is very large, with the potential of creating millions of training examples. A monolithic FFNN implementation utilising all available training data would not be practical while the risk of over-fitting would be high.

For this reason, an ensemble of smaller FFNN trained with different subsets of the available data and employing different criteria for selecting input fields is more likely to be a better choice. How many neural networks, how wide the selection of these subsets and how large the number of training examples will be, all depend upon the computational resources at hand.

The ensemble of FFNN has two layers; the *input layer* and the *gate* FFNN. Each of the FFNN of the input layer will be trained with a different data set which is created by combining a variety of regions, brain landmarks and criteria for selecting the input fields. In this way, it is possible to utilise a number of different combinations of input fields as well as covering more effectively and uniformly the large number of available images.

The architecture of each FFNN of the input layer depends upon the number of lines and input fields of its training set.

The gate FFNN will have as many inputs as the number of FFNN of the input layer of the ensemble, each of which will provide one input for the gate. The gate provides the final output of the ensemble.

2.6 Training the Ensemble

Once the regions, brain landmarks and input field criteria are specified, the size of each FFNN of the input layer can be decided and training and test sets are created.

The training of the ensemble is done in three phases. During phase (1), all input layer FFNN are trained with their assigned data sets. In phase (2), the, now trained, input layer FFNN are fed with unseen data. This data is part of the large set of hand-segmented images but was not used in phase (1). The response of each of input layer FFNN will form the training set for the gate FFNN of the ensemble. In phase (3), the gate FFNN will be trained with the data produced during phase (2).

The classification of an unknown voxel by the ensemble is also done in three phases. In phase (1), a line of input corresponding to the voxel to be classified has to be formed for each input layer FFNN following the same procedures as those used for the construction of training sets. In phase (2), each input layer

FFNN is fed with its corresponding line and an output value is produced. In phase (3), the output of each input layer FFNN is put together and presented to the gate FFNN which produces the final output of the ensemble.

3 Results and Discussion

An ensemble of 48 input layer FFNN and a single gate FFNN were trained following the procedures below:

1. The criteria for determining the input fields of each of the input layer FFNN were constructed by using a combination of 2D and 3D voxel absolute position and position relative to a single brain landmark (the *caudate nucleus* of the *basal ganglia*). Pixel values of the voxel and a region around it and mean and standard deviation of pixel values of voxels in region were also used. Four different 2D region sizes were used with 10, 25, 50 and 75 voxels. The regions were constructed using a "*spiral*" centred around the given voxel. With this method the exact number of voxels can be included in the region (unlike rectangular, elliptical or circular regions) while a certain symmetry around the central voxel is retained.
2. The architecture of the FFNN were calculate using the following rules of thumb: (a) first hidden layer of 1.2 × number of inputs, and (b) second hidden layer of 0.2 × number of inputs. All FFNN, including the gate, employed sigmoidal activations at their output.
3. The training set of each input layer FFNN contained $20,000$ lines, half of those came from lesion voxels and half came from normal voxels (some voxels were present in training sets of more than one FFNN).
4. The set of hand-segmented images contained data from 14 MS patients[2]. The number of normal appearing voxels was $4,748,701$ and the number of lesion voxels was $63,216$ (about 1.3% of normal voxels).
5. The *gate* FFNN had 2 hidden layers. The first hidden layer had 190 neurons and the second 65. Its training set contained $40,000$ lines.
6. The ensemble was trained using simple back propagation, minimising the discrepancy between actual and desired outputs using steepest gradient descent. It took 5 days to train sequentially on a SunBlade 100 computer.
7. The performance of the classifier was tested with data from 6 MS patients (in addition to the 14 above) which contained $2,335,277$ normal voxels and $29,189$ lesion voxels. The results were as follows: **true positive**: $25,452$ (87.2%), **true negative**: $1,910,256$ (81.8%), **false positive**: $425,021$ (18.2%), **false negative**: $3,737$ (12.8%), **sensitivity** $= \frac{TP}{TP+FN} = 0.87$, **specificity** $= \frac{TN}{TN+FP} = 0.82$.

Fig. 4 shows a schematic of the ensemble. The two left-most images show the response of two FFNN (from a total of 48) of the input layer, while the right-most image shows the final ensemble response provided by the gate FFNN. The

[2] Informed consent was obtained from all subjects and the study had approval by the ethics committee of the Institute of Neurology, Queen Square, London.

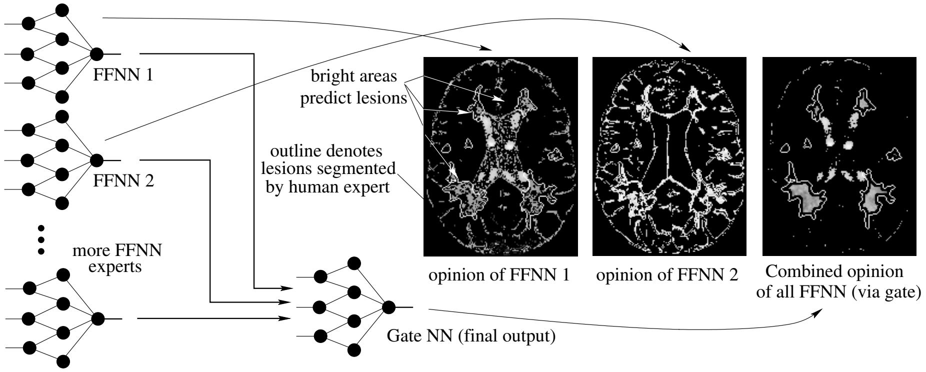

FFNN 1
FFNN 2
more FFNN experts
bright areas predict lesions
outline denotes lesions segmented by human expert
opinion of FFNN 1
opinion of FFNN 2
Combined opinion of all FFNN (via gate)
Gate NN (final output)

Fig. 4. Schematic of the ensemble of neural networks used in lesion segmentation. The first layer of FFNN feeds the **gate** which weighs their response and yields the final output. The first 2 images (left-most) show the output of only 2 of the FFNN (out of 48) when presented with unseen data. Pixel brightness shows the probability of belonging to a lesion. The first 2 classifications (two left-most images) differ but are not totally inconsistent. The third image (right-most) shows the final output of the ensemble. Outlines show the lesions hand-segmented by the clinician

outlined areas are lesions identified by a human rater. In all three images pixel brightness shows the probability of belonging to a lesion.

Most false positive voxels were part of cerebrospinal fluid (the bright ventricle in the middle of fig. 1(**a**)). Additional post-processing may improve the accuracy of the classification by segmenting out this ventricle.

With advancing scanner technology, it is now possible to reduce the slice thickness (in our case it was $3mm$) to $1mm$. In this case it will make sense to use interslice connectivity information in the training set but also as part of a separate post-processing stage where suspected lesion voxels are checked as to whether they are part of a 3D structure resembling a lesion or just isolated false positives.

Future work will concentrate on using other methods for combining neural networks together, for example [15,16,17].

References

1. Arridge, S.R., Grindrod, S.R., Linney, A.D., Tofts, P.S., Wicks, D.: Use of greyscale voxel databases for improved shading and segmentation. Medical Informatics **14** (1989) 157–171
2. Tofts, P.S., Wicks, D.A.G., Barker, G.J.: The MRI measurement of NMR and physiological parameters in tissue to study disease process. Progress in Clinical and Biological Research **363** (1991) 313–326
3. Wicks, D.A.G., Tofts, P.S., Miller, D.H., du Boulay, G.H., Feinstein, A., Sacares, R.P., Harvey, I., Brenner, R., MacDonald, W.I.: Volume measurement of multiple

sclerosis lesions with magnetic resonance images. a preliminary study. Neuroradiology **34** (1992) 475–479

4. Kamber, M., Shinghal, R., Evans, A.C., Collins, D.C., Francis, G.S.: Knowledge-based interpretation of magnetic resonance images: Detecting multiple sclerosis lesions. Artificial Intelligence in Medicine **10** (1993) 32–43
5. Yu, S., Pham, D., Shen, D., Herskovits, E.H., Resnick, S.M., Davatzikos, C.: Automatic segmentation of white matter lesions in T_1-weighted brain MR images. IEEE International Symposium on Biomedical Imaging (2002) 253–256
6. Moon, N., Bullitt, E., van Leemput, K., Gerig, G.: Automatic brain and tumor segmentation. Proc. Medical Image Computing and Computer-Assisted Intervention MICCAI 2002 **2488** (2002) 372–379
7. Goldberg-Zimring, D., Achiron, A., Miron, S., Faibel, M., Azhari, H.: Automated detection and characterization of multiple sclerosis lesions in brain MR images. Magnetic Resonance Imaging **16** (1998) 311–318
8. Masulli, F., Schenone, A.: A fuzzy clustering based segmentation system as support to diagnosis in medical imaging. Artificial Intelligence in Medicine **16** (1999) 129–147
9. Bensaid, A.M., Hall, L.O., Clarke, L.P., Velthuizen, R.P.: MRI segmentation using supervised and unsupervised methods. In: Proceedings of the 13^{th} Annual International Conference of the IEEE Engineering in Medicine and Biology Society. Volume 30., Piscataway, NJ (1991) 111–126
10. Bezdek, J.C., Hall, L.O., Clarke, L.P.: Review of MR image segmentation techniques using pattern recognition. Medical Physics **20** (1993) 1033–1048
11. Bishop, C.: Neural Networks for Pattern Recognition. Clarendon Press, Oxford (1995)
12. Cohen, M.S., Dubois, R.M., Zeneith, M.M.: Rapid and effective correction of RF inhomogeneity for high field magnetic resonance imaging. Human Brain Mapping **10** (2000) 204–211
13. Friston, K.J., Ashburner, J., Frith, C.D., Poline, J.B.: Spatial registration and normalization of images. Human Brain Mapping **2** (1995) 165–189
14. Ashburner, J., Friston, K.J.: Voxel-based morphometry – the methods. NeuroImage **11** (2000) 805–821
15. Perrone, M.P., Cooper, L.N.: When networks disagree: ensemble methods for hybrid neural networks. Artificial Neural Networks for Speech and Vision (1993) 126–142
16. Jacobs, R.A., Jordan, M.I., Nowlan, S.J., Hinton, G.E.: Adaptive mixture of local experts. Neural Computation **3** (1991) 79–87
17. Hadjiprocopis, A., Smith, P.: Feed forward neural network entities. In J. Mira, R.M.D., Cabestany, J., eds.: Lecture Notes in Computer Science: Biological and Artificial Computation: From Neuroscience to Technology. Springer–Verlag (1997) 349–359

A Probabilistic Neural Networks System to Recognize 3D Face of People

Giancarlo Mauri and Italo Zoppis

Dipartimento di Informatica Sistemistica e Comunicazioni
Università degli Studi di Milano-Bicocca
Via Bicocca degli Arcimboldi 8, 20126 Milamo, Italy
{mauri,zoppis}@disco.unimib.it

Abstract. In this paper we describe a 3d face recognition system based on neural networks. The system consists of a modular architecture in which a set of probabilistic neural networks cooperate with the associated graphical models in recognising target people. The logic of this cooperation is quite simple: each network is able to discriminate between its "target" and all other samples of the training set. This is done by using only one characteristic piece of information among the available sets of L, U, V colours and Z coordinate. Every network provides its associated graph with estimates obtained during the training phase, while graphical models coordinate the answers of all the associated networks giving the posterior probability that the target corresponds to the person to be recognised. Then a decision-making criterium based on the maximum posterior probability is established to identify the recognised face.

1 Introduction

Computers still have relevant difficulty in reproducing some easy tasks carried out by humans. This is the case, for example, of people recognition or more generally, of recognition of patterns. Nevertheless, this task has become more and more meaningful in today's society, especially for those special contexts requiring security and control.

When it comes to face recognition, this need is particularly shown by a fair number of commercial products for professional, non professional tasks and even for home computer access control. Despite the fact that almost all available systems are still based on 2d data image there is an increasing number of researchers that are trying to explore both the third spatial dimension and the respective colour appearance of the face (the reader is refered to many exellent surveys, such as [1], [2], or [3]). In many of these researches, the use of multi-criteria based discrimination seems to provide a powerful tool to achieve various detection and recognition tasks (generally through "a priori" decomposition of the image into meaningful facial parts). In those researches, the compounded approaches have also improved sensitivity and reduced susceptibility to disturbing sources [4].

In this paper we use a multi-criteria based discrimination by analysing L, U, V colours [5] separately and by examining the information of an image that comes

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 158–164, 2003.

from Z coordinate. To do so, recognition is carried out following two main steps: first, a batch phase in which we train the system (section 1), second, the online recognition task (section 2). During the first phase, a set of specialised probabilistic neural networks [6] are able to discriminate the face of people, while during the second phase we use simple statistics from the training to create bayesian graphical models in the identification task.

2 Training

Once the faces are captured they can appear arbitrarily oriented in space. For this reason, we used only 3d images previously processed in order to have all the faces rotated in frontal view and translated from their initial position into the center of a fixed coordinate system.

In an attempt to explain this more clearly we can define O_L, O_U, O_V and O_Z as operators which extract from the set of faces Λ the respective colours and depths; for instance, $O_L : \Lambda \to \Lambda_L$, where Λ_L is the set of all subjects Λ described by their component L. Therefore, in order to reduce the size of an image (initially with a definition of 127 x 87), we computed the principal components on the set $\Lambda_L, \Lambda_U, \Lambda_V$ and Λ_Z. In the following discussion we assume these sets to be precisely the projections obtained through this data reduction step.

We also define the set $S_j^L = \{s_j \mid j \in \Lambda_L\}$ as images (projections from PCA) whose values correspond to the description of subject j by using only L components. Similarly, S_j^U, S_j^V, S_j^Z define the sets of sample (projections) faces for subject j whose values represent the U, V and Z components.

The goal of the training phase is to obtain for each specific person $j \in \Lambda$ a network able to discriminate between positive examples $S_j^X = \{s_j \mid j \in \Lambda_X\}$ - sets of targets described through component $X \in \{L, U, V, Z\}$ - and the rest of the images i.e. negative samples in the training data $\overline{S_j^X} = \Lambda_X - S_j^X$. Since we used 4 sources of information, we created 4 sets of PNN: NN^L, NN^U, NN^V and NN^Z respectively, being trained to classify faces according to their respective information. In other words we have 4 neural networks for each target j - i.e. $nn_j^L \in NN^L, nn_j^U \in NN^U, nn_j^V \in NN^V, nn_j^Z \in NN^Z$ - being trained to answer on the set $\{0,1\}$ depending on the identification of their target or not. We will call the set of their answer *evidence vector*.

Execution starts by simultaneously presenting the input vector (target $\mathbf{j}$) to all pattern networks. The transfer function for each radial basis neuron in each network is the unnormalised Gaussian, shown in expression 1

$$W(d) = e^{-d^2} \qquad \text{with } d = \parallel \mathbf{j} - \mathbf{w} \parallel b \; . \tag{1}$$

where $\mathbf{w}$ is the neuron's weight vector; d is the distance between the input vector and its weight vector and b is the *bias* [6].

The *bias* b allows the sensitivity of the radial basis neuron to be adjusted. Each *bias* is set up $\sqrt{-log(0.5)/spread}$. This provides radial basis functions that cross 0.5 when there are weighted input of $|spread|$. This determines the width

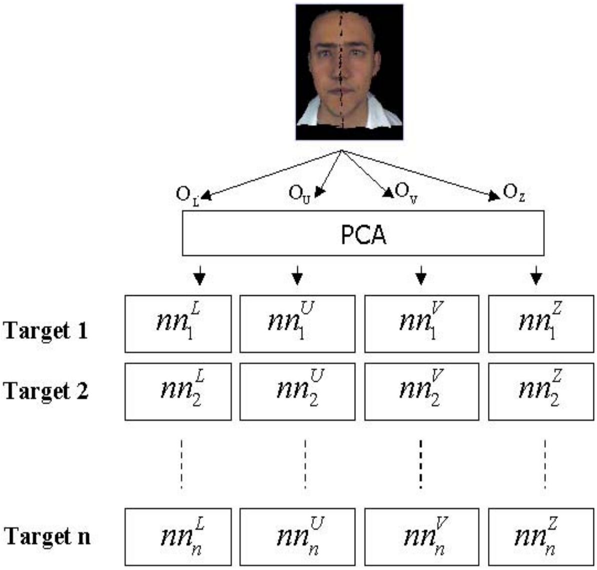


Fig. 1. Networks architecture.

of an input area to which each neuron responds. For instance, when *spread* is 4, each radial basis neuron will respond with 0.5 or more to any input vector within a vector distance of 4 from their vector weight.

The choice of *spread* can have a profound influence on the performance of the PNN. Values that are too small exert excessive influence on individual training. Often the values of *spread* that are too large cause so much blurring that the details of the density are lost and density estimates are badly distorted. For this reason training is performed through the help of a validation set. This is a well-known practice in the field of supervised-learning with neural networks (see for example [7]). The training phase is cyclically repeated until the system properly acts on the new set of samples (validation). Simple heuristic approaches are broadly used for this task: in our case, we save for the run time phase the network which gives the lowest error performance *ValErr* during its discrimination task . Outputs of the Algorithm 1 represent neural network weight matrix Net_j, and frequency of correct answers CPT_j (conditional probabilities table as defined in the framework of graphical models [8]), given that face j is presented to the system. The procedure *pnet* builds a new PNN based on the values of *spread*, training T and validation V.

3 Run Time

Run time environment is related to on-line acquisition and recognition steps. That is, the precise moment in which we capture and identify faces passing close to a camera. In order to recognise a person we established two main criteria that will separately guide our decision-making procedure.

Algorithm 1. Training procedure.

```
WHILE spread <= MaxSpread
      [net,ValErr,CPT] = pnet(T,V,Spread);
      IF (ValErr < MinValErr)
         Net_j = net;
         MinValErr_j = VaErr;
         CPT_j = CPT;
         spread = increment(spread);
      endIF
endWHILE
```

The first one is based on the optimisation of correct frequency answers in case the networks give a positive value as output. The second one is based on the computation of the posterior probability when the *evidence vector* of the neural networks is observed. In the first case we compute:

$$\max_{i \in A}\{p_i\} \ . \tag{2}$$

in which A is the set of networks whose answer is 1 (i.e. networks identify the face as target), and p_i is the frequency of correct answers of the i-th network collected during validation phase. Therefore, the person proposed by the system is the one associated with the network which provide (having performed the lowest frequency error) more reliability during the validation phase.

In the second case, we can represent the run time situation by simply producing a graphical model such as that shown in Figure 2. Such models are graphs in which nodes represent random variables and the lack of arcs represents conditional independence [8]. In our case, we suppose independence between networks and - without any prior information - discrete uniform distribution of *target* people passing close to the camera. With these assumptions in mind we can design a model based on the following variables:

- Random variable T_j (associated with the root):

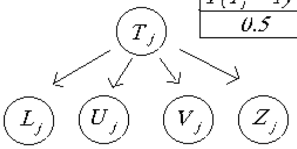
$$T_j = \begin{cases} 1 \text{ if “acquired face } j \text{ is target for respective} \\ \quad \text{graph”} \\ 0 \text{ otherwise} \end{cases}$$

- Random variables L_j, U_j, V_j and Z_j (associated with children nodes). For each scanned face, we have 4 indipendent neural networks being trained to discriminate that face according to the specific set of information; indeed we defined L_j as:

$$L_j = \begin{cases} 1 \text{ if “the observed network identify face } j \\ \quad \text{as target by using } L\text{-information only”} \\ 0 \text{ otherwise} \end{cases}$$

  In the same way we can define random variable U_j, V_j and Z_j by considering the associated information respectively.

$P(T_j = 1)$	$P(T_j = 0)$
0.5	0.5



T_j	$L_j = 1$	$L_j = 0$
1	$P(L_j = 1 / T_j = 1)$	$P(L_j = 0 / T_j = 1)$
0	$P(L_j = 1 / T_j = 0)$	$P(L_j = 0 / T_j = 0)$

Fig. 2. Bayesian graph in the Run Time situation when the face j is to be recognised. Conditional probability table (cpt) is shown for the L-values only.

After all networks have responded we have a vector of 4 *evidence* components for each specific trained face j. To draw a direct model, we must also specify the conditional probability distribution (cpt table in Figure 2) for each node.

Since during the training phase we collected the frequency of correct answers for a specific network, we can use these values to estimate the conditional probability that the network will correctly answer when the face j is captured by the camera. Therefore, we can compute for each graphical model j (we have as many graphs as the number of people in database) the posterior probability $p(T_j|L_j, U_j, V_j, Z_j)$. The decision-based procedure is then given by:

$$\max_{j \in \Lambda}\{p(T_j = 1|L_j = l, U_j = u, V_j = v, Z_j = z)\} \ . \tag{3}$$

where l, u, v and z are the realisation of the respective random variables. Hence, the person will be associated with the graph which gives the highest posterior probability value.

4 Concluding Remarks

The system proposed in this paper has been tested on a set of 19 3d face images. These images have been obtained at different times under natural indoor lighting conditions with cluttered background. During the acquisition of faces people were asked to maintain an inexpressive face. By taking into account these hypotheses it is clear that this experiment is far from providing conclusive outcomes. Anyway, the results obtained on this small sample seem to be encouraging. For each target person, Table 1 shows the answers of the system both with engine 1 and engine 2. If following the first trial the system does not properly provide the target, the system is asked again to output a maximum of 3 possibilities. Both the first and the second engine adequately answer during the first trial 17 times out of 19. If we allow the system to get two possibilities we obtain 18 correct answers out of 19 for engine 1 and 17 out of 19 for engine 2. Because of noisy images, especially regarding Z component, and the lack of a fairly good number of training samples, some networks can output unsafely.

Table 1. Outputs of system in decreasing probabilities order: for each acquired target first appears the system answer with the highest probability value.

target	engine 1	engine 2
1	3,1	3,8
2	2	2,8
3	15,5,8	15,8,5
4	4	4,8
5	5	5,8
6	6,2	6,8
7	9,7	9,7,8
8	8,11	8
9	9	9,8
10	10	10,8
11	11	11,8
12	12,1	12,8
13	13,17	13,8
14	14,1,2	14,8
15	15	15,8
16	16	16,8
17	17	17,8
18	18	18,8
19	19	19,8

A modular architecture is an attractive system especially when the context requires the recognition of greater numbers of people: in practical application, for example, this is the case of security systems. In those cases we do not need to train again the system but we simply add a specialised network which is able to discriminate its target. However, the problem continues to be represented by the difficulty in verifying to what extent some information Z, L, U and V can provide a major contribution to recognition. Our effort here is to show a strategy that harmonise different answers (PNN answers) by developing a simple modular system. In addition, the coordination of different answers through a bayesian graph is intended to achieve this task with more accuracy. It is this kind of usage which characterises the originality of our approach; in this context, probabilistic neural networks are used as a tool for an estimation of conditional probabilities required by Bayes' classification techniques. The bayesian model calculates, in turn, a posterior probability based on the estimation of the most reliable networks.

The framework described above has been implemented in Matlab within a complete prototype for authomatic face recognition system consisting of 4 parts: acquisition of an implicit representation of 3d data image (phase image) from a double CCD camera; reconstruction of the explicit 3d and colour face (XYZ surface and LUV data); extraction of significant features and finally identification of the person.

The authors wish to thank Comerson Srl, a leading company in the area of CCTV, for its grant of the 3d images in its proprietary format.

References

1. Valentin, D., Abdi, H., Otoole, A.J., Cottrell, G.W.: Connectionist Models of Face Processing - A Survey. Pattern Recognition, Vol. 27 (1994) 1209–1230
2. Chellappa, R., Wilson, C.L., Sirohely, S.: Human and Machine Recognition of Faces: A Survey. Proceedings of the IEEE, Vol. 83 (1995) 705–740
3. Zhao, W., Chellappa, R., Rosenfeld, A., Phillips, P.: Face recognition: A literature survey. UMD CfAR Technical Report CAR-TR-948 (2000)
4. Pehrsson, S.L., Shaffer, R.E., Hart, S.J., Williams, F.W., Gottuk, D.T., Strehlen, B.D., Hill, S.A.: Multi-criteria fire detection systems using a probabilistic neural network. Sensor and Actuators Part B: Chemical, Vol. 69 (2000) 325–335
5. Wyszecki, G., Stiles, W.S.: Color Science: concepts and methods, quantitative data and formulae. John Wiley and Sons, New York (1982)
6. Haykin, S.: Neural Networks: A comprehensive foundation. Prentice-Hall (1999)
7. Larsen, J., Hansen, L.K., Svarer, C., Ohlsson, M.: Design and Regularization of Neural Networks: The Optimal Use of A Validation Set. Proceedings of the IEEE Workshop on Neural Networks for Signal Processing (1996) 62–71
8. Jensen, F.V.: An Introduction to Bayesian Networks. Springer Verlag (1996)

Dereverberation of Acoustic Signals by Independent Component Analysis

Claudio Sistopaoli, Raffaele Parisi, and Aurelio Uncini

INFOCOM dept. - University of Rome “La Sapienza”
Via Eudossiana 18, 00184 Rome - Italy
parisi@infocom.uniroma1.it

Abstract. In this paper joint use of array processing techniques and Independent Component Analysis is proposed to separate acoustic signals in the presence of reverberation. The estimated time delay of arrival is employed as an index of performance, to show that the described approach is effective in performing the *dereverberation* of the received signals, as confirmed by several experimental tests in different environmental conditions.

1 Introduction

Independent Component Analysis (ICA) has raised the interest of many researchers in recent years, due to the high number of potential applications (e.g. signal and image enhancement, analysis of medical signals, source separation in telecommunications). Generally speaking, the goal of ICA is to recover original signals by observation of a mixture, having no knowledge of the mixing functions. In the simplest model, mixing is instantaneous (i.e. no convolutions are considered), thus neglecting any delays, reflections or noises.

An extensive literature on ICA is available. An efficient and popular algorithm was proposed in [1], and subsequently improved by Amari [2]. The proposed approach was able to separate up to ten mixed acoustic signals. Unfortunately, the instantaneous mixture model is not realistic when dealing with the problem of acoustic signal separation in reverberating environments. In this case, in fact, signals are convolved with the impulse response of the environment, which takes into account the effects of reflections and diffusion. The study of ICA in the presence of convolving mixtures thus requires a higher level of complexity.

A recent approach [3] proposed the joint use of array processing techniques and ICA to separate multiple acoustic sources in the presence of reverberation. In this work the algorithm presented in [3] is properly modified to separate signals at different reverberation levels. In particular, the time delay of arrival (TDOA) is introduced as a useful performance index in experiments, to show that the proposed approach has the effect of actually *dereverberating* the received signals.

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 165–170, 2003.

2 Separation Algorithm

2.1 Signal Model

Received signals include ambient noise, directional or not, and reflections from the walls of the room and other objects. Assuming M microphones and D sources, the following model in the frequency domain is commonly adopted:

$$\mathbf{x}(\omega, t) = \mathbf{A}(\omega) \cdot \mathbf{s}(\omega, t) + \mathbf{n}(\omega, t) \tag{1}$$

where $\mathbf{x}$ is the vector of the short-time Fourier transform (STFT) of the received signals, $\mathbf{s}$ is the STFT vector of the source signals, $\mathbf{A}$ is the mixing matrix and $\mathbf{n}$ is the STFT noise vector.

$\mathbf{A}$ is an $M \times D$ matrix, and its (m, n) element, $A_{m,n}(\omega)$, is the transfer function from the n-th source to the m-th microphone

$$A_{m,n}(\omega) = H_{m,n}(\omega) e^{-j\omega\tau_{m,n}} \tag{2}$$

In this equation, $\tau_{m,n}$ is the propagation delay from the n-th source to the m-th microphone. It can be noted that eq. (1) is formally similar to the case of instantaneous mixtures, if noise components are negligible. So, proper noise reduction techniques can be devised in order to make it possible the application of instantaneous separation algorithms to each frequency bin. A possible approach is the subspace method, quite popular in the array processing field, which is briefly reviewed in the following [3].

2.2 The Subspace Method

The subspace method is based on a proper decomposition of the *spatial correlation matrix*, defined as

$$\mathbf{R}(\omega) = E[\mathbf{x}(\omega, t)\mathbf{x}^H(\omega, t)] \tag{3}$$

Assuming uncorrelation between signal and noise and omitting ω for simplicity, from (1) it is possible to obtain

$$\mathbf{R} = \mathbf{A}\mathbf{Q}\mathbf{A}^H + \mathbf{K} \tag{4}$$

where $\mathbf{Q} = E[\mathbf{s}(t)\mathbf{s}^H(t)]$ and $\mathbf{K} = E[\mathbf{n}(t)\mathbf{n}^H(t)]$.

The hypothesis of uncorrelation between $\mathbf{s}$ and $\mathbf{n}$ is not strictly verified, since $\mathbf{n}(t)$ includes the reflections of $\mathbf{s}(t)$. However, if the STFT window is short enough and the delay time between the direct signal and the first reflection exceeds the window length, it can still be considered valid.

Computation of the generalized eigenvalue decomposition of $\mathbf{R}$ [4] leads to

$$\mathbf{R} = \mathbf{K}\mathbf{E}\boldsymbol{\Lambda}\mathbf{E}^{-1} \tag{5}$$

where $\mathbf{E} = [\mathbf{e}_1, \ldots, \mathbf{e}_M]$ is the eigenvector matrix, and $\boldsymbol{\Lambda} = diag(\lambda_1, \ldots, \lambda_M)$ is the diagonal matrix of the eigenvalues. Assuming that the signal power is high compared with the noise power, eigenvectors and eigenvalues have the following properties, that we recall from [3]:

1. The energy of the D directional components of the signal $\mathbf{s}(t)$ is concentrated in the D dominant eigenvalues.
2. The noise energy is uniformly distributed on all eigenvalues.
3. The eigenvectors $(\mathbf{e}_1, \dots, \mathbf{e}_D)$, corresponding to the dominant eigenvalues, constitute an orthonormal basis of $\Re(\mathbf{A})$, where $\Re(\mathbf{A})$ is the column space of $\mathbf{A}$ [4].
4. Eigenvectors $(\mathbf{e}_{D+1}, \dots, \mathbf{e}_M)$ constitute a basis of $\Re(\mathbf{A})^\perp$, where $\Re(\mathbf{A})^\perp$ is the orthogonal complement of $\Re(\mathbf{A})$.

Subspaces $\Re(\mathbf{A}) = \Re(\mathbf{E}_s)$ and $\Re(\mathbf{A})^\perp = \Re(\mathbf{E}_n)$ are called the *signal subspace* and the *noise subspace* respectively, being $\mathbf{E}_s = [\mathbf{e}_1, \dots, \mathbf{e}_D]$ and $\mathbf{E}_n = [\mathbf{e}_{D+1}, \dots, \mathbf{e}_M]$.

Properties 1 e 3 correspond to assume that directional components lie into the signal subspace, while noise is fairly distributed over all frequencies.

The *subspace filter* is finally defined as

$$\mathbf{W} = \Lambda_s^{-1/2} E_s^H \tag{6}$$

and processed signals are given by

$$\mathbf{y}(\omega, t) = \mathbf{W}(\omega)\mathbf{x}(\omega, t) \tag{7}$$

2.3 Scaling and Permutation

Since the separation algorithm is applied separately to every frequency bin, it is fundamental the determination of the correct amplitude and the correct permutation of the separated signals in order to correctly reconstruct the signals. In [3] the amplitude problem is solved by multiplying the output signals $u_\omega(t_s)$ by the pseudoinverse of the mixing matrix. If

$$\mathbf{u}(\omega, t) = \mathbf{B}(\omega)\mathbf{x}(\omega, t) \tag{8}$$

is the unmixing equation (where matrix $\mathbf{B}(\omega)$ takes into account the effects of both the subspace filtering and the ICA algorithm), the amplitude ambiguity can be solved by use of a proper scaling matrix $\tilde{\mathbf{B}}_m^\sharp(\omega)$ [3], where symbol $\sharp$ indicates pseudoinversion [4].

In order to solve the permutation problem, in [3] the *Inter-Frequency Coherency* (IFC) method is proposed. This technique is based on the consistency of the mixing matrix in adjacent frequencies. Assuming for the mixing matrix $A(\omega)$ the model (1) and assuming for simplicity $H_{m,n}(\omega) = 1$, the *steering vector* $\mathbf{a}_n$ is given by

$$\mathbf{a}_n(\omega) = [e^{-j\omega\tau 1n}, \dots, e^{-j\omega\tau Mn}]^T \tag{9}$$

At the adjacent frequency $\omega_0 = \omega - \Delta\omega$ it is

$$\mathbf{a}_n(\omega_0) = [e^{-j(\omega-\Delta\omega)\tau 1n}, \dots, e^{-j(\omega-\Delta\omega)\tau Mn}]^T \tag{10}$$

It can be seen that the location vector $\mathbf{a}_n(\omega)$ is equal to $\mathbf{a}_n(\omega_0)$ rotated by an angle θ_n. So, matrix $\mathbf{A}(\omega)$ can be written as

$$\mathbf{A}(\omega) = \mathbf{T}(\omega, \omega_0) \cdot \mathbf{A}(\omega_0) \tag{11}$$

where $\mathbf{T}$ is a *rotation matrix.* If the STFT frequency resolution $\Delta\omega$ is small enough, also angles θ_n are small and $\mathbf{T}(\omega, \omega_0) \simeq \mathbf{I}$ holds.

Indicating with $\bar{\mathbf{a}}_i(\omega)$ the column vector of the *estimated* mixing matrix $\bar{\mathbf{A}}(\omega)$, it is

$$\cos\theta_n = \frac{\bar{\mathbf{a}}_n^H(\omega) \cdot \bar{\mathbf{a}}_n(\omega_0)}{\|\bar{\mathbf{a}}_n(\omega)\| \cdot \|\bar{\mathbf{a}}_n^H(\omega_0)\|} \tag{12}$$

The following cost function $F(\mathbf{P})$ is introduced

$$F(\mathbf{P}) = \frac{1}{D} \sum_{n=1}^{D} \cos\theta_n \tag{13}$$

In order to avoid that an error in a frequency bin induces errors on the choice of the permutation for next frequencies, the reference frequency ω_0 is extended to an interval $\omega_0 = \omega - k \cdot \Delta\omega$ con $k = 1, \ldots, K$. The value of the cost function at ω_0 is denoted by $F(\mathbf{P}, k)$ and is properly exploited to determine the best possible permutation. More details can be found in [3].

3 Experimental Results

Two sources were considered in the experiments. The image method [8] was used to simulate the room response for different reverberation times [7]. The room size was $5.45m \times 4.15m \times 2.80m$ and 6 microphones were supposed to be placed on a wall, 20 apart from each other, at a height of $1.7m$. Sources were placed in front of the microphone array, at a distance of one meter.

Various experiments were performed, with different kinds of sources. The sampling rate was 16kHz.

Fig. 1 shows the overall structure of the proposed separation algorithm.

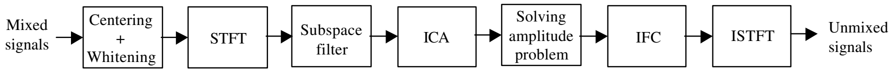


Fig. 1. Algorithm flowchart.

Input signals were first centered and whitened. STFT was then computed, using a Hamming window of $32ms$ and fifty percent overlapping. The FFT length was 512, as a compromise between performance and complexity.

The subspace filter was then applied, to remove noises and reflections and to reduce the number of signals to be processed. Finally ICA was performed, by use of the well-known Amari's algorithm [2]. Amari's algorithm was adopted in this work, since it does not require any matrix inversion and it yields a fast convergence rate. In particular, Amari's rule was properly adapted to take into account the complex nature of quantities of interest. In fact, Hermitian transposition was used instead of simple matrix transposition and the following activation function $f(\cdot)$ was adopted [5]

$$f(z) = \tanh(Re\{z\}) + j\tanh(Im\{z\}) \tag{14}$$

The filtering matrix was updated by use of a *learning coefficient* η, whose choice is critical for the success of the algorithm. In this application the value $\eta = 10^{-4}$ was chosen.

After solving the amplitude and permutation ambiguities, the inverse STFT was applied to yield the unmixed signals.

Several experiments were performed, at different reverberation levels. Specifically, gaussian noise, speech signals and music instruments were considered as sources, in various combinations. The reverberation time T_R was varied from 0 to 0.5 seconds.

As a performance index, the relative time delay of arrival (TDOA) between the separated signals on different pairs of microphones was adopted. It is well-known that in the presence of reverberation cross correlation methods often fail in determining the correct TDOA [9]. In this case generalized cross correlation (GCC) is usually employed [10]. In the present work, in order to analyze the performance of the proposed solution in terms of its *dereverberating effects*, the delay times estimated by the PHAT-GCC algorithm [10] applied on separated signals and on *single* reverberated signals were compared.

Figures 2 show the histograms of the estimated TDOA for a single microphone pair in the presence of two speech signals and $T_R = 0.3$. The vertical line indicates the true TDOA. Figure on the left shows the TDOA estimates obtained in the presence of the first signal alone, while figure on the right shows the TDOA histogram obtained from the same signal after separation. The improvement is clearly visible. Figure 3 shows the TDOA statistics in terms of number of anomalies, bias and standard deviation of the estimate, in the case of separation of one gaussian and one speech signals. The TDOAs are estimated for the gaussian signal, alone and after separation, at different reverberation times.

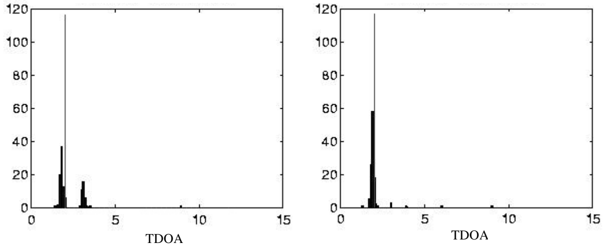


Fig. 2. TDOA histograms in the case of two speech signals and $T_R = 0.3$.

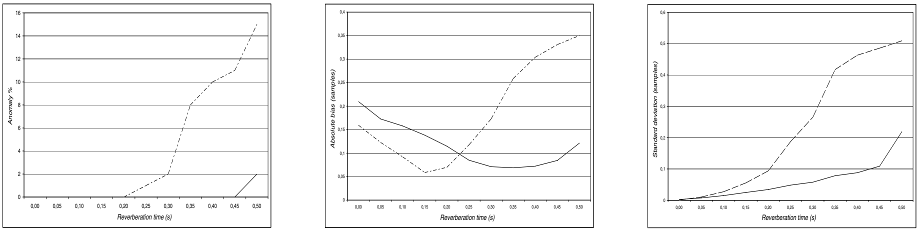


Fig. 3. TDOA statistics for gaussian and speech signals. Left to right: no. of anomalies, bias and standard deviation. Dashed line: gaussian signal alone, solid line: gaussian signal after separation.

More specifically, a TDOA estimate is an anomaly (or *outlier*) when it exceeds a prespecified threshold [9]. The number of anomalies is usually refferred to in TDOA estimation problems as a measure of an algorithm's robustness. Bias and deviation standard are evaluated on all non anomaly estimates. Also in this case the dereverberating effects of the proposed algorithm are evident.

References

1. A. J. Bell and T. J. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7(6):1129–1159, Nov. 1995.
2. S. Amari, A. Cichocki, and H.H.Yang. A new learning algorithm for blind source separation. In MIT Press, editor, *Advances in Neural Information Processing*, volume 8, pages 757–763, Cambridge, MA, 1996.
3. F. Asano, S. Ikeda, M. Ogawa, H. Asoh, and N. Kitawaki. A combined approach of array processing and independent component analysis for blind separation of acoustic signals. In *ICASSP2001*, Salt Lake City, May 2001.
4. G. H. Golub, C. F. Van Loan. *Matrix Computations*, 2nd edn. Johns Hopkins University Press, Baltimore, 1989.
5. N. Benvenuto and F. Piazza. On the complex backpropagation algorithm. *IEEE Transactions of Signal Processing*, vol. 40, April 1992, pages 967-969.
6. S. Ikeda and N. Murata. A method of ICA in time-frequency domain. *Proc. of Int. Workshop on Independent Component Analysis and Blind Signal Separation (ICA'99)*, 1999", pp. 365-371.
7. H. Kuttruff. *Room Acoustics*. Elsevier, London, 3rd edition, 1991.
8. J. B. Allen and A. Berkeley. Image method for efficiently simulating small-room acoustics. *J. Acoust. Soc. Am.*, vol. 65, no. 4, pp. 943–950, April 1979.
9. B. Champagne, S. Bedard, and A. Stephenne. Performance of time-delay estimation in the presence of room reverberation. *IEEE Trans. on Speech and Audio Processing*, vol. 4, no. 2, pp. 148–152, Mar. 1996.
10. C. H. Knapp and G. Clifford Carter. The generalized correlation method for estimation of time delay. *IEEE Trans. on Acoust., Speech and Signal Processing*, vol. 24, no. 4, Aug. 1976.

Supervised and Unsupervised Analysis Applied to Strombolian E.Q.

Cinzia Avossa[1], Flora Giudicepietro[2], Maria Marinaro[1], and Silvia Scarpetta[1]

[1] Dipartimento di Fisica “E.R.Caianiello” Univ. di Salerno
84081 Baronissi (SA) and INFM Sezione di Salerno, Italy
[2] INGV Osservatorio Vesuviano, Via Diocleziono, Napoli, Italy

Abstract. In this paper we analyze seismic signals recorded in September 1997 in Stromboli (Sicily) during explosive eruptions. First, we analyze the data via an unsupervised techniques using the Mixture of Gaussians algorithm (MoG) and the Principal Component Analysis (PCA) to discover the structure of the data. Experts distinguish two types of signals related to two different type of Strombolian explosive eruptions (Type 1 and Type 2). Using the MoG algorithm we can distinguish two classes that, with a good agreement, correspond to the two types of explosions given by experts. As a second step, we implement an supervised automatic system in order to discriminate between the two different types of explosive eruptions. The automatic system based on the MLP achieve a correct classification percentage of more then 98% on the test set (and 100% on the training).

1 Introduction

Stromboli may be considered one of the most active volcanos in the world, and its persistent but moderate activity, termed Strombolian, is only interrupted by occasional episodes of more vigorous activity accompanied by lava flows. In September 1997 the eruptive behavior was characterized by mild, intermittent explosive activity during which well-collimate jets of gases laden with molten lava fragments burst in short eruptions lasting 5-15s. The explosions were characterized by a typical rate of 20-30 events per hour (this was an occasional swarm activity, while explosions at Stomboli typically occur at a rate of 3-10 events per hour). In this period eruptive activity occurred mostly in two distinct vents located near the northern and southern perimeter of the crater. Experts are able to distinguish two different types of waveforms [2]: The type 1 events are associated with eruptions from the northern vent, characterized by canon-like blasts typically lasting a few seconds and producing well-collimated jets of incandescent gases laden with molten fragments. The type 2 events are representative of eruptions from the southern vent. These eruptions were much less impulsive than those from northern vent, lasted longer (up to 20s), and produced wider fans of ejects and significant amounts of ash.

The goal of our work is double, from one side, the aim is to visualize the data via a dimensionality reduction and to discover the structure of data via

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 173–178, 2003.

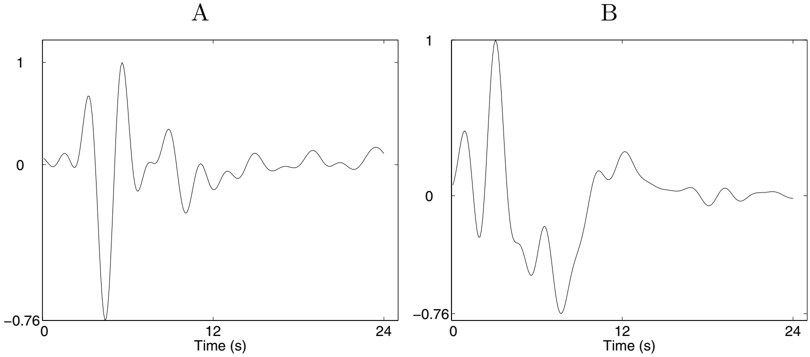


Fig. 1. Type 1 (A) and Type 2 (B) signals from our data set.

a probability distribution parametric model as the MoG, from the other side our aim is to implement a reliable automatic classifier able to discriminate the explosion quakes type 1 and type 2. A reliable automatic classifier may be useful to reduce the work of the experts, that usually discriminate the two kind of explosions by hand, case by case.

Our data set is composed of 353 events, of which 183 correspond to type 1 (northern vent) and 170 correspond to type 2 (southern vent).

In section 2 the signals are described, in sec. 3 an unsupervised approach is reported and in last section an automatic classified based on a multilayer perceptron (MLP) is designed and results are summarized.

2 Data

Our data were recorded, in September 1997, by a network of 21 three-component Guralp CMG-40T broadband (0.02-60s) seismometers deployed on the flanks of Stromboli. The network featured three rings of sensors surrounding the edifice, at crater level, mid elevations and near sea level, with stations ranging in distance between 0.3 and 2.2 km from the active crater. The stations remained in operation from September 18 through September 25 1997. The data set available to us is made of 353 records that had been filtered between 0.02 Hz and 0.6 Hz, and then have been re-sampled with a sampling frequency of 10 Hz. This filter enhances the very-long period (VLP) components present in the explosion signals and brings to light the repetitive action of two sources distinguished by their characteristic waveform. Figure 1 shows one signal of type 1 and one of type 2. Each signal of our data set is 24 seconds long (a vector with 240 components). There are 183 signals of type 1 and 170 of type 2.

3 Visualization and Unsupervised Analysis

Data visualization is an important means of extracting useful information from large quantities of raw data. The human eye and brain together make a formidable pattern detection tool, but to work the data must be represented in a low-dimension space, usually of two dimension. Principal Component Analysis (PCA) is a classical linear projection method for mapping data to a lower dimensional space. We apply the PCA to the full data set composed of 353 vectors in a 240-dimensional space. The data points are visualized by orthogonal projection on the principal components plane, spanned by the two leading eigenvectors (i.e those with the largest associate eigenvalues). The resulting graph is shown in figure 2. Note how the class denoted by crosses is scattered in the plane, while the other class is more tightly grouped. This linear map shows reasonably good separation between the two types, although they overlap in the central part of graph. So this information processing problem is solved by human brain in seemingly effortless fashion. To solve the problem with an automatic computation we recall that most pattern recognition tasks, as the classification, can be viewed in terms of probability density estimation. A powerful approach to density estimation is Gaussian mixture model (MoG)[1] We write our model for the density as a linear combination of component densities in the form

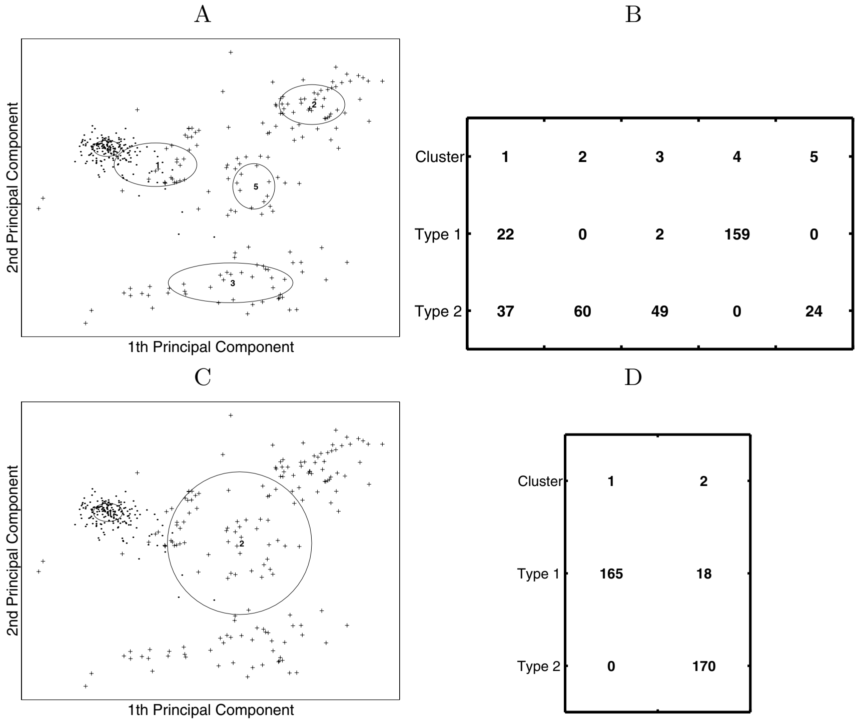
$$p(x) = (\sum_j p(x/j)p(j)) \tag{1}$$

in which the p(x/j) represent the individual components of the mixture. Using 5 components (j=1,2,3,4,5), with diagonals covariance matrices, results are shown in fig. 2A-B. Each signal is assigned to the cluster C_i such that $P(C_i \| \mathbf{x}) > P(C_j \| \mathbf{x})$.

Using only two components (j=1,2), with diagonals covariance matrices, to build a density model of the data $P(\mathbf{x})$, and computing posterior probabilities $P(C_j \| \mathbf{x})$ by Bayes' theorem, we get an unsupervised classification of the data into two clusters. Each signal will be assigned to the cluster C_i such that $P(C_i \| \mathbf{x}) > P(C_j \| \mathbf{x})$. The decision boundary is defined by the equation $P(C_i|\mathbf{x}) = P(C_j|\mathbf{x})$. The class corresponding to type 1 (2) will be the one that includes a greater number of type 1 (2) data with respect to the other and a lower number of type 2 (1) signals. The decision boundary is shown in fig. 3.A.

4 Supervised Analysis

An alternative was to develop a model that estimates the posterior probability directly via a supervised approach, this can be done using a neural network. We use a feedforward Multilayer Perceptron (MLP) with one hidden layer of units. The data set has been divided in training set (1/3 of data) and test set (2/3 of data) in order to get a robust estimation of the generalization performance of our classifier. In the preprocessing stage we use the PCA to extract the features of the 240-dimensional data. Then the MLP has been trained to classify the extracted features. Figure 3.B shows the eigenvalues, when PCA is applied to the training set, arranged in descending order. We can effectively reduce the dimensionality of the data, keeping about 95 % of information content, by retaining only the first 15 eigenvectors and discarding those with small eigenvalues. Thus



Cluster	1	2	3	4	5
Type 1	22	0	2	159	0
Type 2	37	60	49	0	24

Cluster	1	2
Type 1	165	18
Type 2	0	170

Fig. 2. A. Data projected in the two-dimensional principal space. MoG results with 5 gaussians are shown. (Dots shows Type 1 and crosses shows Type 2 data). B. Corresponding cluster composition. Cluster named 1 is mixed, while other clusters are composed of mainly one type of events. C. Data projected in the two-dimensional principal space. MoG results with 2 gaussians are shown. D. Corresponding cluster composition. Type 2 data fall all into a single cluster.

we project training and test data onto the space spanned by the first 15 eigenvectors. A 15-dimensional vector will be the input of the MLP classifier. The network architecture is the following: 15 input nodes, 3 hidden nodes and one output node. The net output y is given by

$$y \;=\; \sigma\;(\;\sum_i\; w_i\; tanh(\sum_j\; W_{ij}\; x_j)) \tag{2}$$

where

$$\sigma(a) \equiv \frac{1}{1 + \exp -a} \tag{3}$$

The logistic sigmoidal is the activation function of the output layer, while the hyperbolic-tangent, $tanh(a)$, is the non-linear activation function of the hidden, x_j is the input and W_{ij} and w_i are the parameters (or weights) optimized during the training procedure, minimizing the error function cross-entropy. We use a

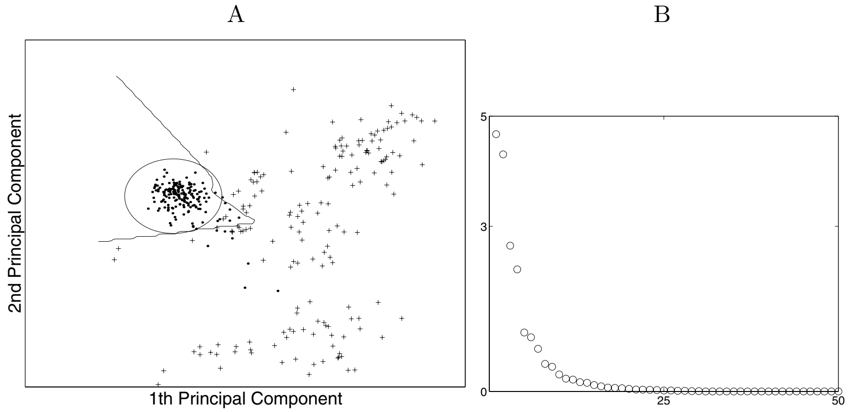


Fig. 3. A. The decision boundary using the Bayes rules in the unsupervised approach is shown (circle) together with the decision boundary of a MLP trained on data projected on the first two principal directions. B. Eigenvalues of the PCA decomposition of training set, arranged in descending order.

single output y, which represents the posterior probability $P(C_1|\mathbf{x})$, so the posterior probability of class C_2 is $P(C_2|\mathbf{x}) = 1 - y$. The target coding scheme is $t = 1$ if the input vector belongs to class C_1 (type 1) and $t = 0$ if it belongs to class C_2 (type 2). The target variable is binary and we use a Bernoulli random variable for the conditional distribution:

$$P(t|x) = y(x|\mathbf{W}, \mathbf{w})^t (1 - y(x|\mathbf{W}, \mathbf{w}))^{(1-t)} \tag{4}$$

Table 1. Confusion matrix of the MLP classifier, on the test set. The matrix confusion shows that three events type 1 are classified by MLP as type 2 and only one event type 2 is classified as a type 1.

	C_1	C_2
Type 1	116	3
Type 2	1	115

Taking the negative logarithm of the previous expression and summing over entire pattern set $\mathbf{x}_n$ we yield the cross-entropy error function:

$$E = -\sum_n \{t_n ln y_n + (1 - t_n) ln(1 - y_n)\} \tag{5}$$

The most widely known supervised algorithm to train multilayer neural network is the back-propagation gradient descent algorithm. Here we used a more efficient and fast back-propagation algorithm, the Quasi-Newton algorithm, that adjust the direction of descent by using second-order information, but which doesn't require the calculation of second derivatives [1]. After the training, we give to the MLP the new data, i.e. the test set projected along the principal directions extracted from the training set alone. We didn't use the test set in the feature extraction stage in order to assess in a unbiased way the generalization ability of the two-stage classification system by computing the performance of the system on the completely novel data (test set). To assess the system capabilities, five different data sets, each composed of a training and a test set were obtained through a permutation of all the available data and the network was separately trained and tested on each of them. The net performance is obtained averaging the percentage of correct classification obtained from each of the five test sets .This average percentage of correct classification is more then 98%. The confusion matrix for one choice of training/test set is shown in table 1. Note that we classified single station record. Each explosion quake is recorded by several stations at the same time. To classify the explosion event, we put together the classification results of all the corresponding single records, using majority rule. In this way the confidence of classification is enhanced.

References

1. C. Bishop, Neural Networks for pattern recognition, Oxford University Press 1995.
2. B. Choulet, P. Dawson, T. Ohminato, M. Martini, G. Saccorotti, F. Giudicepietro, G. De Luca, G. Milana, R. Scarpa. "Source mechanisms of explosions at Stromboli Volcano, Italy" Journal of Geophysical Research vol 108, n B1, 2019, 2003

Intracranial Pressure Signal Processing by Adaptive Fuzzy Network

Bruno Azzerboni[1], Mario Carpentieri[1], Maurizio Ipsale[1],
Fabio La Foresta[1], and Francesco Carlo Morabito[2]

[1] Dipartimento di Fisica della Materia e Tecnologie Fisiche Avanzate
Universitá degli Studi di Messina, salita Sperone, 31 C.P. 57, 98166 Messina, Italy
{azzerboni,carpentieri,ipsale,laforesta}@ingegneria.unime.it
[2] Dipartimento di Informatica Matematica Elettronica e Trasporti
Universitá *Mediterranea*, via Graziella Loc. Feo di Vito, 89100 Reggio Calabria, Italy
morabito@ing.unirc.it
http://neurolab.ing.unirc.it

Abstract. The aim of this work is the analysis of an intracranial pressure (ICP) signal, measured by means of an optical fiber catheter. We want to propose an alternative method to valuate the pressure inside the skull, without any knowledge of compliance curve, that can be valuated directly only by means of invasive and dangerous methods. First, we propose a classic Fourier processing in order to filter the ICP signal by its spectral components due at cardiac and respiratory activity. Then we perform the same analysis by wavelet transform, in order to implement a multiresolution analysis. The wavelet tool can perform also a very reliable data compression. We can demonstrate the advantages in using a neuro-fuzzy network on wavelet coefficients in order to obtain an optimal prediction of ICP signal. Various network structures are presented, in order to obtain several trade-off between computational time and prediction mean square error. Such analysis was performed by changing the fuzzy rule numbers, modifying the cluster size of the data. A real-time implementation was also proposed in order to allows the clinical applications.

1 Introduction

The skull can be modelled like a stiff box filled with the brain (80 % of full volume), blood (12 %) and cerebrospinal fluid (8 %). Brain must be perfused with regularity by blood flow. The Cerebral Perfusion Pressure (CPP) is the difference between the mean arterial pressure (MAP) and intracranial pressure (ICP): if blood pressure is inadequate, the organism starts a compensation mechanism to prevent ischemia or cerebral edema. Then the brain swells and ICP rises. At begin the organism lets go out Cerebrospinal Fluid (CSF) through spinal sac, with decrease of ICP. However, if pathology persists, further brain enlargement can induce to ernias. The compliance curve [1], shown in Fig. 1, explains how an intracranial volume increase produces an intracranial pressure rise. For little

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 179–186, 2003.

increases, a self-regulation system maintains ICP at low values that correspond to high compliance (points 1 and 2 in Fig. 1), whereby a very large intracranial volume increase brings ICP to dangerous high values (points 3 and 4 in Fig. 1). The compliance is a quantitative parameter done by the derivative of the intracranial volume with respect to th intracranial pressure.

$$compliance = \frac{dV}{dP} \tag{1}$$

In order to detect compliance value it is necessary to introduce a known volume of liquid in the skull. The compliance is calculated after measuring the corresponding intracranial pressure variation. Obviously, this is a very dangerous technique. For these reasons, the intracranial pressure measurement over time and its direct processing plays an important role in clinical neurological diagnosis and neurosurgery.

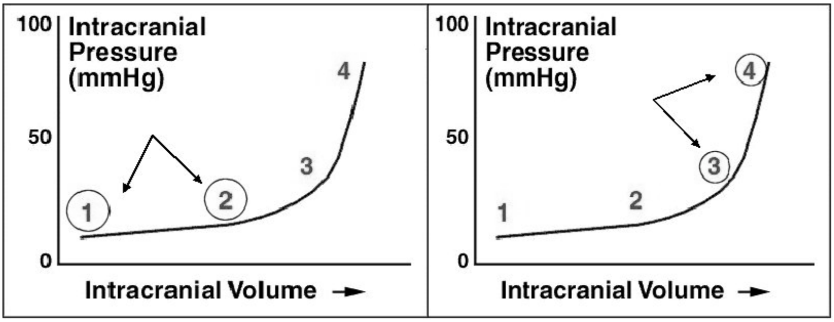


Fig. 1. Compliance curve.

2 Spectral and Time-Frequency Analysis Preprocessing

In this section we describe the preprocessing methods that we apply in order to perform ICP signal forecasting. In particular we introduce some theoretical informations needed to implement the *wavelet compressor* and the *wavelet expansor* that will be used in our system, as we will describe in next section.

2.1 Fourier Transform

First, we perform a Fourier analysis of ICP signal. We process a 2-hours recording of ICP, with a sampling frequency of 25 Hz, corresponding to 180000 samples of digital signal. By means of Fast Fourier Transform we obtain the ICP spectrum shown in Fig. 2. It is reasonable to identify respiratory activity in spectral components localized near 0.47 Hz (28.2 respiratory cycles/minute), whereby cardiac components are localized near 1.63 Hz (97 respiratory cycles/minute).

We propose two stop-band filters that eliminate these unwanted components. Fourier Analysis is the standard method to process ICP signal, and effectively it is able to detect unwanted spectral components. Really, classic processing is inadequate to analyze non stationary signals. The natural evolution of Fourier Analysis is the Short Time Fourier Transform (STFT), that uses same resolution for time and frequency. The disadvantage of this method is represented by Heisenberg uncertain principle, that limits the product of spectral and time resolutions. For this reason, we propose a multiresolution approach, like wavelet transform.

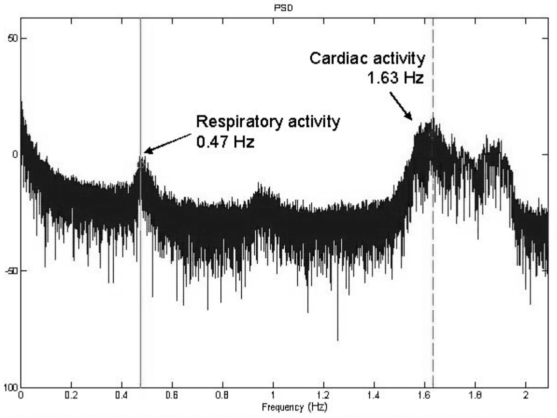


Fig. 2. Power Spectrum Density (dB) of ICP signal: zoom over lower frequencies.

2.2 Wavelet Transform

The advantage of wavelet transform on Fourier analysis consists in its ability to detect time instants when spectral components are localized [2]. Moreover, it allows a multiresolution analysis, characterizing the low frequencies in large time intervals, and the high frequencies in narrow time intervals. The variable *frequency* is substituted by *scale*, that is related to inverse frequency [3]. The basis functions of multiresolution analysis are named *mother wavelets*, that can

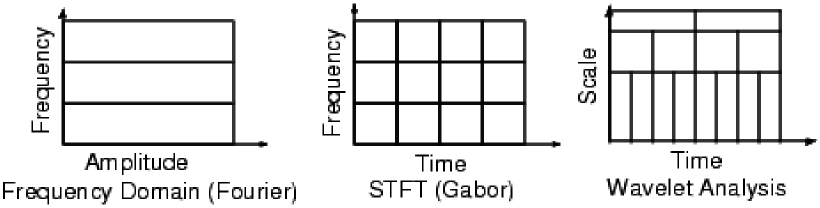


Fig. 3. Short Time Fourier Transform and Wavelet Transform.

inspect simultaneously times and scales. Given the input signal *f(t)*, its wavelet transform is defined as:

$$F(a,b) = \int_{-\infty}^{+\infty} f(t) \frac{1}{\sqrt{a}} \psi \left(\frac{t-b}{a} \right) dt \tag{2}$$

where a represents the scale parameter, b represents the translation parameter and ψ is the *mother wavelet.* If we choose a and b based only on powers of two then our analysis will be much more efficient and just as accurate. We obtain such an analysis from the *discrete wavelet transform* (DWT).

$$a = 2^j, \quad j \in N, \qquad b = k \cdot 2^j, \quad k \in Z$$

$$F(j,k) = \int_{-\infty}^{+\infty} f(t) \frac{1}{2^{j/2}} \psi \left(\frac{t-k\cdot 2^j}{2^j} \right) dt \tag{3}$$

When we apply DWT on the ICP signal, we obtain some coefficient vectors, each representing a time series containing only a predetermined frequency range. In these vectors, we can distinguish between approximation coefficients (that contain only low frequency spectral components) and detail coefficients (whose information is related on high frequency spectral components). By a parameter L (level of decomposition) we can decide which frequency range is contained on approximation coefficients. The approximation coefficients number, corresponding to low frequencies (high scales), is lower than the detail coefficients number (low scales). Practically, the discrete wavelet transform is performed by a filters bank, each having half bandpass than the previous one. Such analysis also allows data compression. Once the decomposition is performed, and the coefficients are calculated, we choose a threshold and discard all coefficients minor of that. Then we reconstruct the signal by the inverse DWT, taking only the remained coefficients. The result is a data compression that could be useful to lighten computation of algorithm without produce significant data deletion, since it contains nearly all useful information carried in original data signal.

3 System Implementation

In this section we present the structure of the implemented system[4,5]. After a data organization, we perform a compression by means of wavelet decomposition. Next, we select the coefficients of the best approximation of ICP signal and we input them in a neuro-fuzzy network that perform forecasting. Lastly, we utilize predicted coefficients in order to recostruct ICP signal. Fig.6 shows the block diagram of the entire procedure that will be described in deepened way in next subsections. In the real-time operation, the entire system receives in input 128 samples for every prediction step, and it can forecast other 128. By a continuous recording of ICP signal, the system can provide the ICP signal 128 samples ahead. The entire procedure has been implemented using Matlab©language as a support for simulations.

3.1 Wavelet Compressor

In the wavelet compressor block we perform the DWT, extracting approximation and detail coefficients. To obtain a good compression, we choose the *haar* function as mother wavelet, the older and the simplest wavelet. This function, corresponding to a modified *signum* function (i.e. -1 for $0 < t < 0.5$ and $+1$ for $0.5 < t < 1$), not only is simple, but also allows direct and inverse transform with the simplest structure of filters. Since the final aim of this work is the real-time application, the lightness of computational time and the compactness of algorithm justify the use of this function. We want to separate approximation from detail coefficients, making forecasting only on first of these. Haar wavelet can compress our data in the minimum number of approximation coefficients. These values are input in the neuro-fuzzy network that performs forecasting. The detail coefficients instead are input in the wavelet expansor, in order to make inverse discrete wavelet transform and to reconstruct predicted signal.

3.2 Neuro-fuzzy Network

The ICP signal can be viewed as a chaotic time series. In our analysis, we utilize known values until a time instant t, in order to forecast a value in a future time instant $t+s$. After choosing a forecasting step s and a dimension d, training data input for neuro-fuzzy network are organized as a d dimension vector composed as follow:

$$INPUT_{NFN}(t) = \left\{ x\Big[t-(d-1)s\Big] \; x\Big[t-(d-2)s\Big] \; ... \; x\Big[t-s\Big] \; x\Big[t\Big] \right\} \tag{4}$$

while training data output is:

$$OUTPUT_{NFN}(t) = \Big\{ x[t+s] \Big\} \tag{5}$$

The real time operation of entire algorithm is guaranteed by the data structure, that adds s coefficients for each step of prediction. First we implement a neural network for signal forecasting. Optimal results are obtained when network has four input variables, and one output variable, following data organization explained before. Moreover, we choose the prediction step s=2, i.e. we can forecast two samples of digital signal. The results of this approach consist in an acceptable prediction error, but a too much heavy computational weight. The neural network utilized, in fact, has a hidden layer with a very high number of neurons. This problem heavily influences the possibility of hardware implementation of our algorithm. Neural network is not able to generalize over various patients. Our choice has been concentrated on neuro-fuzzy system. These networks can be implemented easily in ST microcontrollers, which have an instruction set that well works with fuzzy systems. The membership functions are chosen as Gaussian curves, whereby the output is calculated as linear combination of fuzzy rules output. Training data (500 five-dimensional vectors, four dimensions for input

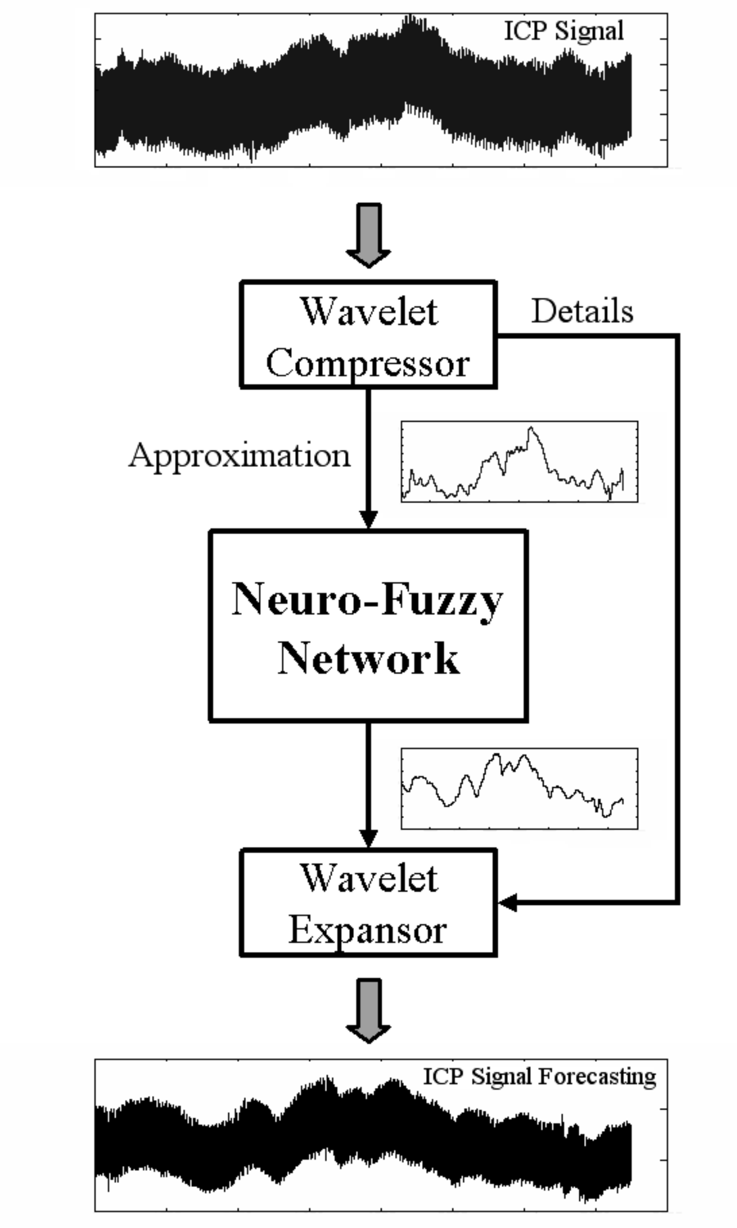


Fig. 4. Block diagram of the implemented system. Five seconds of the recorded signal (128 samples) are compressed by *wavelet compressor*. Only the approximation coefficients are input to *neuro-fuzzy network* that forecasts the approximation coefficients related to future five seconds. The task of the *wavelet expansor* performs wavelet reconstruction using the neuro-fuzzy output and not processed detail coefficients. The final output represents five seconds ICP signal forecasting.

and one for output) are supplied at the network, that allows the parameters of fuzzy rules and membership functions to vary. The number of rules is determined by the subtractive clustering algorithm, that characterizes the clusters situated in data with minimum number of rules necessary in order to distinguish fuzzy characteristics associated to every cluster. We vary radius of clusters in order to have a little number of rules with acceptable results. With a little radius we characterize a lot of little clusters (that corresponds to more rules); a big radius perform few clusters (that corresponds to few rules). The structure of neuro-fuzzy network is shown in Fig. 5. First hidden layer represents the membership functions (mf) associate with input (three mf for every input). The second layer is determined by three fuzzy rules that perform a linear combination of fuzzy inputs. Lastly, the output is a linear combination of fuzzy rules outputs. The linearity of algorithm, excluding fuzzyfication process, make entire procedure more light and compact in hardware realization than pure neural network. Moreover, the neuro-fuzzy network is able to generalize over all patients whose ICP is predicted.

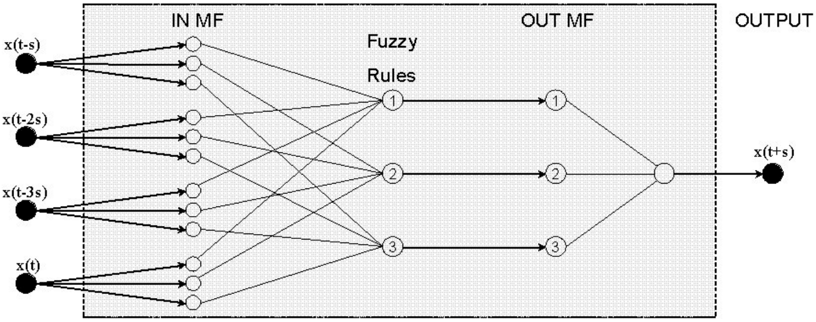


Fig. 5. Neuro-Fuzzy Network Structure.

3.3 Wavelet Expansor

Our fuzzy network can forecast two coefficients with a reasonably low error. The advantage in using wavelet compression is that two approximation coefficients (calculated in first block with Haar wavelet) correspond to 128 samples of ICP signal. The last block performs the wavelet reconstruction of ICP signal predicted, using approximation coefficients predicted by neuro-fuzzy network and detail coefficients extracted by wavelet compressor.

4 Conclusions and Experimental Results

Fig. 6 shows the predicted samples and the measured values (continuos line) of ICP signal. It is noteworthy to underline that a forecasting approach based only on neuro-fuzzy network would provide information limited to a few samples. The advantage of this method is that it turns to account both the wavelet ability to focus information in few parameters and the network performance to expand the forecasting to a larger set, improving the agreement between the experimental data and the predicted ones.

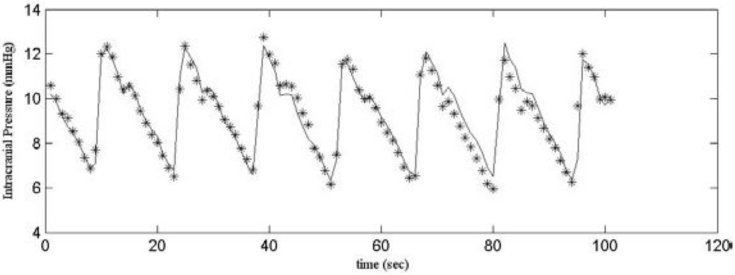


Fig. 6. Comparison between ICP signal measured (continue line) and predicted values.

References

1. G. Gambardella, D. d'Avella, R. Tomasello: Monitoring of brain tissue pressure with a fiberoptic device, Neurosurgery, vol. 31, pp. 918-922, (1992).
2. N. Hess Nielsen and M. V. Wickerhauser: Wavelets and Time-Frequency Analysis, Proceedings of the IEEE, vol. 84, No.4, pp. 523-529, Apr. 1996.
3. Wavelet Toolbox, Matlab, v.2.2, ch.1.
4. B. Azzerboni, G. Finocchio, M. Ipsale, F. La Foresta, F.C. Morabito: Intracranial Pressure Signals Forecasting with Wavelet Transform and Neuro-Fuzzy Network, Proceedings of The Second Joint EMBS/BMES Conference, Houston, TX, USA, pp. 1343-1344, (2002).
5. B. Azzerboni, G. Finocchio, M. Ipsale, F. La Foresta, F.C. Morabito: A Wavelet Approach to Intracranial Pressure Signal Forecasting, Proceedings of The Tenth Biennal IEEE Conference on Electromagnetic Field Computation, Perugia, Italy, p. 215, (2002).

Soft Computing Techniques for Classification of Bronze Age Axes

Lara Giordano[1], Claude Albore Livadie[2], Giovanni Paternoster[3],
Raffaele Rinzivillo[3], and Roberto Tagliaferri[1]

[1] Dipartimento di Matematica e Informatica, Università di Salerno
via S. Allende, 84081, Baronissi, Salerno, Italy
and I.N.F.M. unità di Salerno

[2] Istituto Universitario Suor Orsola Benincasa
Via Suor Orsola 10, 80135,Napoli

[3] Dipartimento di Scienze Fisiche, Università di Napoli "Federico II"
Via Cintia, 80126, Napoli, Italy

Abstract. In this paper, the application of Multiple Classifier Systems and Soft Computing techniques to the classification of Bronze Age axes found in Italian territory is shown. The methodology used from feature extraction to classification is detailed. The results are obtained by using a data set of 85 axes, with training accomplished by bootstrapping the data. The system has been tested on new axes to be classified and validated on an artificial data set generated following the covariance matrices of the original archaeological data.

1 Introduction

The study of the forms of the archaeological finds is one of the most accredited and immediate methods for the attribution of the dating and the affiliation of the manufactured article to a culture. This paper illustrates the application of Multiple Classifier Systems [7] [6] and Soft Computing techniques [5] [4] [10] for the classification of Bronze Age axes found in Italian territory. The features that differentiate the datings have been drawn out from the articles of the archaeologist G.L. Carancini [1].

The system is divided into two parts. The first part consists of extracting from the images of the axes the features necessary to the creation of a numerical data set. The second part analyzes the numerical data set applying the techniques of Soft Computing and Ensembles. The first step of the analysis is a selection of the remarkable features. The second step is the search of a measure of similarity suitable to represent the dispersions of the classes. The third step consists in the study and in the planning of a proper classifier for the data.

2 Image Measuring, Feature Extraction and Selection

To get the numerical data set on which to apply the techniques of Soft Computing for the classification, we started from the archaeological tables representing the

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 187–194, 2003.

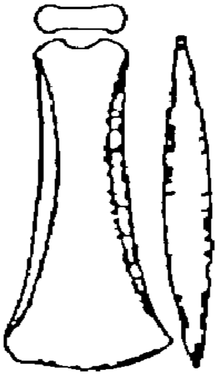


Fig. 1. Image of an axe of the date set with its three visual reports.

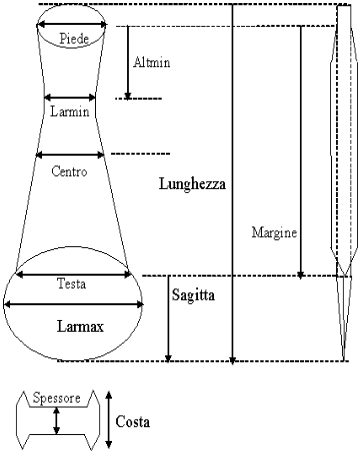


Fig. 2. Scheme of an axe with its parameters.

images of the axes, each one composed by three visual reports: frontal, side and thickness. From these images, with the use of algorithms opportunely conceived, we extracted the features of every single axe [2] [3]. We have so gotten a numerical data set composed by 85 objects that represent the axes, each of these with 10 features and a label that identifies the dating. The features derive from ratios among dimensional parameters of the axes. We used such ratio to make the measures absolute. The parameters of every single axe are represented in figure 2.

Feature Selection. The feature selection is a technique that allows to find the useful features that describe a dominion of application [4]. The problem of the feature selection can formally be defined as the selection of a minimal set composed by the M remarkable features taken from an initial set of the N original feature, where M ≤ N.

We used different techniques of feature selection, obtaining the best results with the Backward Selection and the Principal Component Analysis [4]. With both these techniques, we obtained a reduction from an initial number of 10 to a final number of 6, maintaining unchanged the capability of data classification.

3 Measure of Distance

One of the main difficulties of the archaeological data set is a strong overlap of the data. To avoid such problem, we tried to find a measure of distance suitable to represent the data. We made many experiments using different measures of distance and comparing the results with an opportune visualization. The measures of distance, that we used, are: the Euclidean, the Mahalanobis, the Manhattan and the Lukasiewicz' fuzzy similarity distances. Among these, the one that better identifies the class distributions is the Lukasiewicz fuzzy similarity distance. In the following subsection we give a brief definition of it [5].

Lukasiewicz Fuzzy Similarity Distance. Given two objects x and y, the distance of total similarity of Lukasiewicz among the two is given by the formula:

$$S\langle x, y\rangle = \frac{1}{N}\sum_{i=1}^{N} S_i\langle x, y\rangle, \tag{1}$$

where

$$S_i\langle x, y\rangle = \mu_X(x) \leftrightarrow \mu_X(y) \tag{2}$$

is the partial similarity among the two objects respect to the fuzzy set that characterizes the feature X, while ↔ is the operation of bi-residual defined by the logic of Lukasiewicz.

4 Classification of the Archaeological Data Set

k-Nearest Neighbor with Fuzzy Similarity. Once found the suitable measure of distance, we modified the k-Nearest Neighbor (k-NN) using such distance. The idea was to replace the simple Euclidean distance, with the distance of fuzzy similarity, so getting a new classifier, called k-Nearest Neighbor with Fuzzy Similarity (k-NNFS). The reason for such substitution is that the Euclidean distance do not identify in an appropriated way the distributions among the data. In the following we show one comparative table of the results obtained with the k-NNFS, the k-Means and the k-NN.

Table 1. Comparisons of the performances between the standard classifiers and the k-NNFS.

Classifier	Percentage of Correct Classification
k-NNFS	60%
k-NN	56,4%
k-Means	50,5%

Bootstrap and Neural Networks. The archaeological data set is composed by only 85 objects: the problem, then, is the impossibility to divide it into two different sets for training and testing a neural network. To solve such problem we replied the data set using the statistical technique of bootstrap [9]. With such technique it is possible to create a date set of appropriate dimensions. Once a sufficiently large data set (with around 1000 objects) was obtained, it was possible to use a Multi Layer Perceptron (MLP) to classify the data using the bootstrapped data set as training set and the original data set for testing.

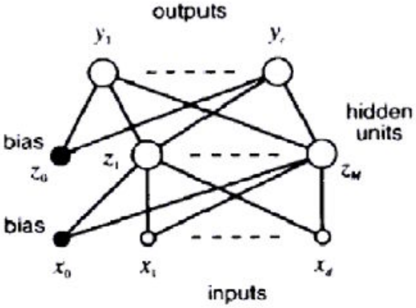


Fig. 3. Example of MLP with one hidden layer.

We used an MLP with 10 input neurons, 6 hidden neurons and 8 output neurons, SoftMax activation functions and QuasiNewton learning algorithm [4] [8].

The use of such technique carried an increase of the system performance of around the 10 percent, getting one percentage of correct classification equal to 71,7% on the whole data set.

Multiple Classifier Systems. In the field of pattern recognition, multiple classifier systems (MCS) based on the combination of outputs of a set of different classifiers have been proposed as a method for the development of high performance classification systems [7] [10]. A great number of methods for combining multiple classifiers have been proposed in the past ten years.

Roughly speaking, an MCS includes both an ensemble of different classification algorithms, and a decision function for combining classifier outputs. Therefore, the design of MCSs involves two main phases: the design of the classifier ensemble, and the design of the combination function [7] [6]. Although this formulation of the design problem should lead to think that effective design should address both phases, this is not the general rule: most design methods have only focused on one of these phases. Several different MCSs can be designed by coupling different techniques for creating classifier ensembles with different combination functions. However, the best MCS can only be determined by experimental performance evaluation.

For the classification of the archaeological data set some MCSs have been conceived and experimented and the best architecture is shown in figure 4. The first part of the classifier in figure 4 is composed by a set of five different classifiers in parallel: k-NNFS, kmeans, EM spherical, hier. centroid, hier. single. Each one works on the archaeological data set. Once the results of these classifiers were obtained, a combination function based on the concept of weight matrix is used. Every cell in the weight matrix represents the weight of the class i in the classifier j, normalized on the total number of objects belonging to the class i in the data set. From the output of the combination function we chose the objects classified in a wrong way. These objects were given in input to the first MLP classifiers. Subsequently, we add new MLP classifiers until the confusion matrix of the data was stabilized. Each MLP had as test set all the objects classified in wrong way from the previous MLP.

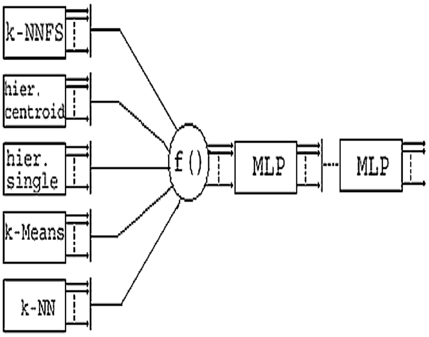


Fig. 4. Example of MLP with one hidden layer.

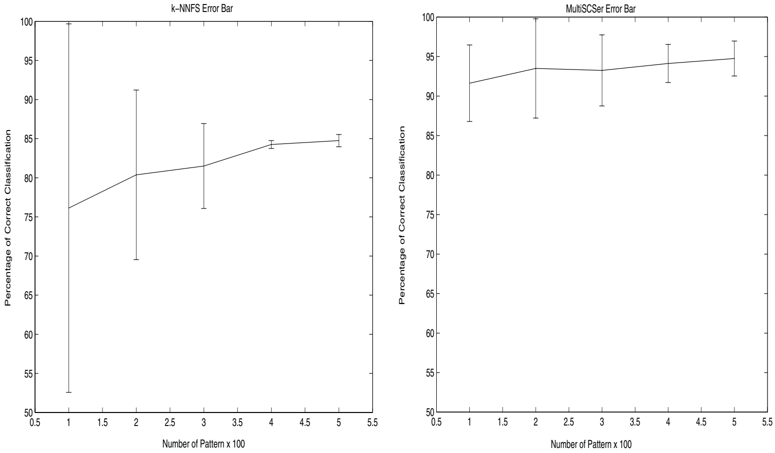
The results obtained with such classifier guaranteed an increase of the performances of about 10% on the archaeological data set in comparison with the MLP with bootstrap, getting a percentage of correct classification of 82,3%.

5 Validation of the Methods

In this paper, we have shown suitable techniques for the classification of the archaeological data set. To validate the conceived methods we thought to test

Table 2. Summary of the results.

Classifier	Percentage of Correct Classification
MultiSCSer	82,3%
MLP	71,7%
k-NNFS	60%



(a) Errror bar for k-NNFS

(b) Errror bar for MultiSCSer

Fig. 5. Two error bar.

the best classifier of the archaeological catalog on artificial data taken from known statistic distributions. The generation of the artificial data is based on the use of the Gaussian Mixture Models [4] [8].

The artificial data set has been created by the sampling of a Gaussian Mixture Model, with the same covariance matrices of the archaeological data. We used a Gaussian for each class of the original data set. Then, we created a data set of 1000 objects, from which we randomly extracted a set of 100 objects for the training and nine sets of 100 objects for the testing of the system. We then created all possible sets of 200, 300, 400, 500 objects combining the sets of 100 objects. From these, we chose in random way eight data sets for each dimension. We used the so created data sets as test set for the k-NNFS and MultiSCSer classifiers. From the obtained result, we calculated the mean and the variance. Figure 5 shows the plots with the error bar for the two classifier: figure 5(a) show the plot for the k-NNFS and figure 5(b) show the plot for the MultiSCSer.

6 Classification of Unlabelled Finds

To test our system with real data, we consider a data set composed by patterns extracted by further 22 axes of the Bronze Age, that were found in Campania's territory [11]. The axes came from two (early bronze age) unpublished caches, the former from S. Marcellino - Frignano site and the latter from the Patria Lake. All axes have some shared general features, as the edges raised, the foam cut more or less arched, about the same size and weight. Their bronze composition was measured and small differences were found to supply some indications on the age metalworking, founded on the same origin ores use and on the addition of the tin, in fixed percentage, and of the plomb, in some cases, perhaps to melt simplifying. This analysis did not permit to highlight possible differences in the caches and the archeologist C. Albore Livadie, that is dealt with such discoveries, has hypothesized that these new finds belonged to one period of cutover among two datings. With the use of the Multiple Classifier, described in a previous section, we obtained a full membership of these finds to the most recent of the two classes indicated by the archeologist, which is archaeologically reasonable.

7 Conclusion

In this paper, we introduced ensemble methods and Soft Computing techniques for the analysis of an archaeological data set. The first step was that to find a measure of distance suitable for the data; then we used such measure for the classification using the k-nn opportunely modified. With such classifier, we obtained a percentage of correct classification of 60%. Subsequently with the use of MLP and bootstrap, we increased the system performance of a 10%, reaching the percentage of correct classification of 71,7%. Finally, creating an appropriate Multiple Classifiers System, we obtained a further increase of 10%, for a total of correct classification of 82,3%.

The proposed system has been validated also on synthetic data generated with the same distributions of the archaeological catalog and can be easily generalized to other types of archaeological and non archaeological data sets.

Acknowledgement

The authors would like to thank the teacher A. di Nola for the kind collaboration and for the useful suggestions about fuzzy logic and fuzzy similarity.

References

1. Carancini,G.: L'età del bronzo in Italia: per una cronologia della produzione metallurgica. Ali&NO, Perugia (1999)
2. Jain, R. et al.: Machine Vision. Mc Graw Hill, Series in Computer Vision, (1995).
3. Gonzales,R. C., Woods, R. E.: Digital Image Processing. Addison-Wesley Publishing Company, Inc. (1992)
4. Bishop, C. M.: Neural networks for pattern recognition. Oxford University Press, Oxford UK (1995)
5. Turunen, E.: Mathematics behind fuzzy logic. Advances in Soft Computing, Physica-Verlag, Heidelberg (1999)

6. Roli, F., Giacinto, G., Vernazza, G.: Methods for Designing Multiple Classifier Systems. In J. Kittler and F. ROli(Eds.): MSC 2001 LNCS 2096 (2001) 78-87
7. Giacinto, G., Roli, F.: An approach to the automatic design of multiple classifier systems. Elsevier, Pattern Recognition Letters **22** (2001) 25-33
8. Nabney, I. T.: NetLab: algorithms for pattern recognition. Springer-Verlag, New York (2001)
9. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference and Prediction. Springer-Verlag, New York (2001)
10. Haykin, S.: Neural Networks. Prentice Hall, 2nd Edition (1998)
11. Livadie, C., Paternoster, G., Rinzivillo, R.: Nota sulle analisi mediante fluorescenza X in riflessione totale (TXRF) di asce provenienti da alcuni nuovi ripostigli del Bronzo antico della Campania. Boll. Acc. Gioenia Sci. Nat. 357 **33** (2000) 5-16.

Environmental Risk and Territorial Compatibility: A Soft Computing Approach

Silvio Giove[1] and Claudia Basta[2]

[1] Department of Applied Mathematics, University Ca' Foscari of Venice
Dorsoduro n. 3825/E - 30125 Venice, Italy
sgiove@unive.it

[2] Regional and Urban Planning Department - City of Venice viale Pertini
Mestre (Venice), Italy
claudia.basta@provincia.venezia.it

Abstract. The presence of risky establishments in urban areas is one of the most dangerous factor that the environmental planners have to consider to the end of a safe, compatible development of these part of the cities. The discrimination between acceptable and not acceptable risk is a central problem of this field. Particularly, in regard to the territorial risk analysis, the Italian law prescribes a methodology based on strict quantitative thresholds in order to separate acceptable risk from unacceptable one. This method might introduce some problems for those cases connected to the safety limit. The proximity depends on both the vulnerability indicator of the target, and on the distance from the limit of the damage-area. In both the cases, the fuzzy logic approach offers a good solution to the delicate question "how safe the target is?" by introducing a smoothing degree of compatibility rather than the use of rigid and pre-defined thresholds. By using fuzzy thresholds for each variable, we can compute a soft measure of compatibility between the target and the risk source. We underline the compensatory properties of this approach which improves the system currently in use.

1 Introduction

By following the general provision issued in the recent past by the UE Commission (*Seveso Bis* provision) in the field of the risk assessment and the regional and urban planning, the Italian legislator has imposed the use of a methodology based on fixed numbers of quantitative thresholds, with the purpose to discriminate the acceptable from the unacceptable risk. This approach introduces some difficulties in the evaluation of the cases *closed* to the safety limit. Despite the currently threshold's method, the fuzzy logic approach offers a good solution to the critical question *"how safe the target is?"*, introducing a *smoothing* degree of compatibility. In order to reach this end, we propose a soft computing solution for the evaluation of *territorial compatibility*, based on a fuzzy logic approach. Fuzzy logic was strongly applied to complex real world problems, refer to [10], however, only few attempts have been made for environmental risk analysis, refer

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 195–201, 2003.

to [5], [6],where the analysis is limited to the hazard characteristics of the plant. In this paper, we extend the fuzzy logic approach to the *territorial planning* risk analysis, by obtaining a soft compatibility index. The usefulness given by the use of a smooth value for such index relays on the compensatory property too. For instance, a very high level of vulnerability for the target can be compensated with a great distance from the risk plant, currently not included in the legal system, where a fixed threshold is imposed for each parameter. In this paper we propose a simple version of our method; in particular, the *domino* effect and not even the aggravating factors for relating parameters are considered. Moreover, the analysis should be performed for each type of damage which can arise in a risky situation. The approach can be very useful in conjunction with the risk analysis of the plant, to the end of developing a complete Decision Support System for the environmental policy, extending the ideas as referred in [9].

2 A Formal Approach to a Compatibility Measure in the Environmental Risk

In analyzing the environmental risk connected with major accidents, it is necessary to take into consideration two factors: the *risk source*, like an industrial plant, and one or more *vulnerable target*, like houses, hospital and other type of human installation inside the influence radius of the accident. The environmental compatibility takes into account the characteristics of the risk's source, like the probability of the event and its *magnitudo*, together with the *vulnerability* properties of each possible target. For the sake of simplicity, the following description considers the presence of one target only inside the maximum radius of damage. More than one target can be analyzed in the same way, by performing the analysis for one target time by time. In particular, the vulnerability is a function of one parameter whose minimum and maximum values are $[p_{min}, p_{max}]$ so that $p \in [p_{min}, p_{max}]$. Usually this parameter is related to the number of people present in the neighbor of the target. Moreover, the *magnitudo*, the measure of the damage impact, is a decreasing function of the distance between the target and the source (this function needs to be carefully identified, refer to [1]). Let be: P the probability of the catastrophic event, V the vulnerability of the target, R the distance between the target and the source. A target needs to be considered only if $R \leq R_{max}$, where R_{max} is the maximum radius of the damage, depending on the plant's characteristics, so that the damage intensity (magnitudo) inversely depends on R. The vulnerability V is a measure of the damage suffered from the target if the distance is very small (theoretically equal to zero); it is represented by the increasing function $V = l(p)$, $l : [p_{min}, p_{max}] \to [0, 1]$. Then, the theoretical *compatibility* function for a target can be defined as a function of $P, V, R : COM = f(P, V, R)$, with $f : [0, 1] \times [0, 1] \times [0, +\infty] \to [0, 1]$ satisfying some rationality conditions. Specifically, f is a decreasing function with respect to P, V and an increasing one with respect to R. Moreover, $\forall x, y \in [0, 1], \forall z > 0 : f(x, y, z) = 1$ if $R > R_{max}$, $f(0, y, z) = f(x, 0, z) = 1$, $f(1, 1, 0) = 0$.

3 The Evaluation of the Territorial and Environmental Compatibility in the Italian Approach

In the recent Italian law decree, dated 9 may 2001, refer to [2], the three variables *P, V,R*, compete together to the definition of the compatibility among the risk source and all the surrounding territorial and environmental elements. The compatibility depends on both the variables of the risk analysis, the probability and the *magnitudo* of the damage. The damage depends on the distance between the target and the source, since the distance is a *proxy* of the damage itself. In order to provide a common reference to the plant's management and to the employees in charge with the territorial planning, the legislator has pointed out some fixed thresholds that individualize: 4 classes of probability, 4 categories of damages (areas of *iso-damage*), and 6 classes of vulnerability. Then, the *probability* classification set[1] is composed of the set $U_P = \{Low, Medium, High\}$, whose elements are ordered from the best case to the worst one. The elements of the *damage* classification set are, from the best to the worst one: *reversible damage (RD), irreversible damage (ID), beginning lethality (BL), elevated lethality (EL)*, so that $U_D = \{RD, ID, BL, EL\}$. Finally, the 6 territorial categories of vulnerability, indexed from A up to F (*A: very vulnerable, F: no vulnerable*), constitute the *vulnerability* classification set $U_V = \{A, B, C, D, E; F\}$, from the worst case A to the best one, F. Furthermore, in this approach, each variable is represented by *ordered* classes. The ordered classes are formalized in a chart of territorial compatibility. Each couple of input variables, the class of probability (left column) and the area of iso-damage (higher line), correspond to the output of the maximum compatible territorial category. The *crisp* inference can be formalized by means of rules as the following:

$$(P = Medium) \cap (D = BL) \rightarrow (COM = \{C, D, E, F\}) \tag{1}$$

where $BL \in U_D$, and $Medium \in U_P$. If, for instance, $medium = [10^{-6}, 10^{-4}]$, the meaning of such a rule is: "If the probability of damage is *Medium* (i.e., in between 10^{-4} and 10^{-6}), and the distance between the risk source and the target is inside the zone of *Beginning Lethality*, than the compatibility among the source and the target is *at most* in the class *C* " (or, that is the same: "it *can* belong to the classes: *C, D, E, F* "). Note that this is equivalent to affirm that:

$$(P = Medium) \cap (D = BL) \cap (p \in \{C, D, E, F\}) \rightarrow COM \tag{2}$$

The rules data base is formed by a complete set of proposition like (2). Since it is sufficient that only one is true to guarantee the compatibility, we can also write:

$$COM = \vee_{k=1}^{n} (P \in A_k) \cap (D \in B_k) \cap (p \in C_k) \tag{3}$$

where $\vee$ is the maximum operator and $A_k \in U_P, B_k \in U_D, C_k \in U_p$. The formalization of (3) will be particularly useful in the fuzzy case, as described later.

[1] The names of the 4 elements of the set U_P were indicated by the 4 linguistic labels, in analogy with the fuzzy approach that we proposed.

Note that the vulnerability depends on the type of the building, and usually, even if more than one parameter could be considered, it is measured by a single parameter p. This parameter is a typical indicator defined for each public or residential buildings, like the number of the present persons, the index of construction, etc. The numerical thresholds for p produce intervals of different vulnerability. The evaluation procedure follows a precise methodology. At the beginning, the information about the plant is picked up, the areas of iso-damage are emphasized, and the cartographic base is updated. Next, the correspondence among the ministerial chart and every target inside the radius of damage is checked. A strict application of the evaluation procedure means that the iso-damage area, the probability of the event and the compatibility index are *crisp* values. A problem arises when a variable is *close* to the pre-defined threshold. A building in the vulnerability's class A, situated *just outside* of a not compatible area of iso-damage, necessarily needs a more precise evaluation compared to the one of the Italian law decree, which, even in these extreme cases, indicates *full* compatibility. Our end is the development of a territorial plane in the respect of an *acceptable degree* of compatibility, and, at the same time, we wish to avoid the use of a rigid *zoning* between safe areas and not-safe areas. It is evident that fuzzy logic can help us to resolve the problem characterized by such type of *uncertainty* in the definition of classes, see [6], [7], [10], [11].

4 A Fuzzy Approach to the Compatibility Measure

The categories of vulnerability are *ordered* classes, in the sense that the *i-th* one *precedes* the $(i+1)-th$ in relation with the characteristics of vulnerability; this means that A class is worst than B class B class is worst than C class and so on. Drawing the attention to the p parameter only, if it belongs to the $i-th$ class, say $Cl_i \in U_V$ the law decree supplies a clear criterion for allowing or refusing the compatibility. However, what happens if the value of the parameter should be close to the *border* located between two consecutive classes? The same question can be asked for the distance and the probability. The case might appear in the event of distance from the risk source being just a little bit greater than the fixed threshold between two iso-damage areas. Even if the law provision contains the answer to each possible case, it is not so reasonable that a *strict* mechanism can be imposed to the decisional process. In fact, some paradoxical situation might arise. For instance, it is useful to consider the "hypothetical" case in which the value of the p parameter is compared with the distance (most simply we do not take into consideration the probability). We can further suppose that, for the distance, the threshold between the BL and ID classes is 200, and between the ID and RD classes the threshold is 300, while for the parameter p, the threshold between C and D is 150. Consider the two following cases: a) $p = 151, D = 299$, b) $p = 149, D = 201$. Using the vulnerability classification set as imposed by the decree, let us suppose that the two following rules can be inferred from (1): "if the parameter p belongs to C, then the distance must belong to at *least* at the class BL", and: "if p belongs to class D, the distance must belong to *at least* the class ID". From such two rules, it follows immediately that case a) is no compatible, while case b) is classified as compatible. Such situation seems almost absurd. In fact, both in the case the parameter p has more or less the same value (from the case a) to the case b) it differs only for a little bit more than 1%, while the distance is much greater in the case a) than in the case b). Then, for what concerns the distance, the first case is quite more favourable than the second one. But the conclusion is that the case a) has to be rejected, while the case b) can be accepted! Really, fuzzy logic can help us to solve this undesired paradox. We need a tool that introduces a *smooth* degree when passing through the border of consecutive classes. Note that when in (3) we say, for instance $D \in B_k$, we mean that D belongs *at least* to the class B_k, being B_{k+1} a more favourable case than B_k. Then, we should write:

$$COM = \vee_{k=1}^{n}(P \in {A_k}^{leq}) \cap (D \in {B_k}^{leq}) \cap (p \in {C_k}^{leq}) \tag{4}$$

where $D \in B_k$ has to be intended as: "the distance belongs, in $fuzzy$ sense, *at least* to the class A_k", and similarly for the other variables. Thus, we have to formalize such type of $fuzzy$ proposition, that can be naturally be expressed as: "the distance is greater than ξ_k". To this we need to define a membership function (m.f. for brevity) of the proposition $D \geq \xi_k$, that can be represented by an increasing S-type fuzzy number, say $\nu_{\geq\xi_k}$, interpreting ξ_k as a fuzzy threshold. An S-type fuzzy number is represented by a monotonically (increasing or decreasing) function with the properties listed in [10]. Analogous definition can be given for decreasing S-type m.f., for instance $\eta_{\leq\tau_k}$, being τ_k an other *fuzzy threshold.* The conjunction operator $\cap$ can be implemented through a suitable *triangular norm*, see [4]. Moreover $D \geq \xi_k$, $P \leq \tau_k$ are computed by $\nu_{\geq\xi_k}(D)$ and $\eta_{\leq\tau_k}(P)$ respectively, and $p \in Cl_{k}^{\leq}$, computed by $\mu_k(p)$, is the membership degree of the proposition "the parameter p belongs to *at least* the class $Cl_k \in U_V$". Since this proposition is equivalent to "the parameter p is *equal or less* than the value λ_k", being λ_k a suitable threshold depending on the class Cl_k, the value of such proposition can be computed by a suitable S-type membership function $\mu_{\leq\lambda_k}(p)$. Note that, in the crisp sense, there is compatibility if (3) is satisfied. Then from (4) the soft compatibility can be finally computed as follows:

$$COM = \vee_{i=1}^{n}\{\mu_{\leq\lambda_k}(p) \cap \nu_{\geq\xi_k}(D) \cap \eta_{\leq\tau_k}(P)\} \tag{5}$$

5 Application to the Risk Evaluation Problem

We have applied the soft compatibility approach to a real case-study, a risky establishment containing gas - oil. The target is represented by the two platforms of the next railway station. Note that usually every target splits into three general categories: open and closed exposition, and intrinsically vulnerable targets, for which different type of the p parameters are used. In particular, open space and buildings are evaluated in the two first cases respectively, while the third one is used to evaluate hospitals, schools, and, more in general, all the other places where the inhabitants are mostly young or/and sick people. In the afford mentioned case, the value of the p parameter is $p = 6000$, and it refers to the average number of persons who daily travel by train. The categories of vulnerability for this kind of targets (railway stations) are represented by the B category, where p => 1000 daily travellers, and the C category, where p =< 1000; note that the target is classified as a B category, even if the value of p is 6 time over the threshold. The distance between the source of the the considered accident (a *pool - fire*) and the target is 40 meters; the correspondent iso -damage area is the one of "irreversible damage". Note that the next iso-damage area, the "beginning lethality" one, is not compatible and it has a radius of 30 meters. This case is a so called "case - limit", where the target has a very high value for what concerns the p parameter, and its distance from the not compatible iso-damage area is very small. However, the application of the Italian law decree furnished *compatibility* among the plant and the target. Next we applied the fuzzy approach with trapezoidal S-type membership functions $\mu_{\leq\lambda_k}(p)$, $\nu_{\geq\xi_k}(D)$ and $\eta_{\leq\tau_k}(P)$, each of them described by two parameters, refer to [10]. The conjunction operator was the triangular norm *minimum*, while the rules are expressed in the form (5). The obtained soft compatibility is $COM = 0.6$. This result indicates a low compatibility degree among the risk source and the target inside the radius. The low value for the soft compatibility should alert the urban planning that a risk situation can be present. It means that the actual crisp system is probably too much *optimistic*. Anywise, it is obvious that such result strongly depends on the parameters of the used membership functions. Due to the fact that this choice is critical, the parameters of the membership functions need to be carefully selected. To obtain this end, in the coming step of our activity we shall propose a suitable method for an optimal parameter design, using soft computing algorithms and methods, like *data mining* approach and *Group Decision Theory*.

6 Remarks and Conclusion

A fuzzy approach for the computation of a compatibility index for territorial risk analysis has been proposed. Some remarks about the usefulness of the described approach are in order, in particular with respect to the introduction of a compatibility index in $[0, 1]$:

a) in our proposal, some *compensatory* effects can be taken into account among the input variables (that is, a too low value for the distance can be compensated by a low value for the intrinsic parameter). Then a *minimal* acceptability of threshold can be defined for the compatibility

b) we can use the compatibility index as a *possible* compatibility index, from the optimal value, $COM = 1$, downward up to the worst one, $COM = 0$. In this case, different scenarios can be compared each others. For instance, an emergency plane can be scheduled in such a way as to give priority to the target with low compatibility.

We conclude remarking that, in the next future, we intend to develop some methods in order to optimize the parameters of the membership functions, thanks to the use of suitable *data mining* approaches and *Group Decision Theory*, together with the selection of proper conjunction operator.

Acknowledgements

The authors express their thanks to Prof. Stefano Stanghellini, Dept. of Urbanistic, IUAV, University of Venice, to Arch. Rosario Manzo, Di.Co.Ter., Italian Ministry of Infrastructures and Transports, and to Arch. Antonio Bortoli, Department of Regional and Urban Planning, City of Venice, for the helpful suggestions and the coordination of the overall research activity.

References

1. *Hazard evaluation procedures*, Battelle Columbus Division for The Centre for Chemical Process Safety of the American Institution of Chemical Engineers, 1985.
2. Manzo R. et al., *Pianificazione del territorio e rischio tecnologico- il D.M. 9 maggio 2001*, CELID, Italian Ministry of Infrastructures and Trasports, 2002.
3. Greco S., Matarazzo B., R. Slowinski Rought sets theory for multicriteria decision analysis, *European Journal of Operational Research*, 129, 2001, 1-41.
4. Klement E.P. Mesiar R., Pap E., *Triangular Norms*, Kluwer Academic Publishers, Dordrecht, 2000.
5. Mock R. Gheorghe A., Risk engineering: bridging risk analysis with stakeholders values, *Kluwer Academic Publishers, Dordrecht*, 1999.
6. Mock R., Krause J.P, Gheorghe A., Assessment of risk from technical systems: integrating fuzzy logic into the Zurich hazard analysis method, *International Journal of Environment and Pollution*, 5, 2/3, 1995.
7. Munda G., *Fuzzy information in multicriteria environmental evaluation models*, EUR 15602 EN, Ispra, Italy, 1994.
8. Nakayama H., Tanino T., Matsumoto K., Matsuo H., Inoue K., Eawaragi Y., Methodology for group decision making with an application to assessment of residential environment, *IEEE Trans. of Systems, Man and Cybernet.*, 9, 477-485, 1979.
9. Paruccini M., Decision support systems in the service of policy makers, *Final Report of the POP Sicily*, contract n. 10122-94-03 T1PC ISP I, Ispra, Italy, 1996.
10. Von Altrock C., *Fuzzy logic and neurofuzzy applications explained*, Prentice Hall, New York, 1995.
11. Zimmermann H. J., *Fuzzy sets, decision making and expert systems*, Kluwer Academic Publishers, Boston, 1993.

Neural Network Approach for Estimation and Prediction of Time to Disruption in Tokamak Reactors

Antonino Greco, Francesco Carlo Morabito, and Mario Versaci

DIMET, Universitá di Reggio Calabria, via Graziella – Loc. Feo di Vito
89100 Reggio Calabria, Italy
{greco,versaci,morabito}@ing.unirc.it

Abstract. This paper deals with the problem of predicting the onset of a disruption on the basis of some known precursors possibly announcing the event. The availability in real time of a large set of diagnostic signals allows us to collectively interpret the data in order to decide whether we are near a disruption or during a normal operation scenario. In this work, a database of disruptive discharges in Joint European Torus (JET) have been analyzed for the purpose. Neural Networks have been investigated as suitable tools to cope with the prediction problem. The experimental database has been exploited aiming to gain information about the mechanisms which drive to a disruption. The proposed processor will operate by implementing a classification of the shot type, and outputting a real number that indicates the time left before the disruption will take place.

1 Introduction to the Problem

The idea of generating nuclear fusion energy is largely based on the concept of magnetic confinement. To sustain the fusion reactions, the power liberated would have to be much greater than that lost via radiation and transport across the magnetic field. Thus, a key role in the Tokamak experiment is related to the energy confinement time. Some intrinsic physical limits related to the efficiency of the confinement and possible large instabilities could limit the operational regime of the tokamak, through a rapid falling to zero of the plasma current [1]. Consequently, the early prediction of the deterioration of magnetic confinement preceding the onset of a disruptive event, during the evolution of a plasma discharge in a Tokamak machine, represents an important step forward in order the experimental activity around nuclear fusion to achieve a practical industrial interest. From the physical viewpoint, the phenomenon of the disruption represents a transfer of energy of the plasma to the surrounding mechanical structures. During the sudden loss of confinement, the energy content of plasma collapse in an uncontrollable way, generating mechanical forces and heat loads which threaten the structural integrity of surrounding structures and vacuum vessel components. It is thus of primary importance to design an alarm system for detecting he onset of a disruption in tokamak plasma discharges. Neural Network models (NNs) have been proposed in the recent literature as forecasting

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 202–209, 2003.

systems, with the aim of predicting the occurrence of disruptions sufficiently far in advance for protecting procedures to be switched on [2, 3]. The design of such a system is constrained from the availability of experimental examples derived from the monitoring of disruptive shots. In this paper, the approach above mentioned, will be ameliorated to increase the advance of the prediction. For this purpose, an experimental database of discharge related to the JET device (Joint European Torus), has been exploited. The database represents a collection of measurements carried out by means of diagnostic sensors located along the contour of the vacuum vessel and of the corresponding time-to-disruption left before a disruptive event take place. The aim of the study is to devise a processing system that can be able to predict correctly the "time-to-disruption", based on the experience gained on the available "examples" through some sort of data analysis. The paper is organized as follows: section two describes the exploited experimental database; section three proposes a neural network approach for estimating time to disruption in tokamak reactors; section four reports a neural network approach for forecasting time to disruption and, finally, some conclusions.

2 The Experimental Database: An Overview

A disruption oriented database of a block of JET discharges has been set up by the JET Team. In this database, a set of measurements monitoring the plasma shots (disruptive and non disruptive shots) are stored. A large number of them were analysed with the purpose to find the technical causes, the precursors and the physical mechanisms of disruptions [4, 5]. The files under study derives from many years of experimental activity carried out at the Culham Center, Oxfordshire, London (United Kingdom). The database was built starting from the dynamics of a disruption. In particular, the study is concentrated in the zone of flat-top of the plasma current. During such phase, the plasma is monitored to obtain a constant plasma current (IPLA) and a stable confinement in terms of shape and position of the plasma. The selected variables are reported as following:

- Plasma current IPLA [A]
- Mode Lock [T/A]
- Total radiated power Prad [W]
- Plasma density ne $[1/m_3]$
- Total input power P_{inp} [W]
- Plasma internal inductance l_i
- Stored diamagnetic energy derivative W_{dia}
- Safety factor at 95% of minor radius q_{95}
- Poloidal β_p

The choice of such variables was done considering some observations performed on two different types of net; in fact, it was noted that a Multi-Layer-Perceptron network (MLP), by using these input variables to the time t, performed better than a recurrent network. This indicates that such variables have in some manner memory [6]. Unfortunately, some input variables are not available; in fact, some sensors do not work. For this reason, a study based on the plasma physics was carried out in order to find some signals that be able to replace the missing data. Knowing the phenomena concerning the incoming of a disruption, the interval of observation of the variables was circumscribed to the period ranging on [td-440ms; td-40ms], the last 40ms were omitted because, not relevant. The time of sampling is 20ms and, then, 20 samples for each channel. We have considered 1167 shots without disruption and 701 disruption, carried out between September 1997 and April 99. The outputs of the network are labelled by means of vtargetTS for training database, vtargetVS for validation database and vtargetTest for testing database. The outputs were carried out considering that, in correspondence of a shot without disruption, we need a series of 20 zeros that represent the risk of disruption. Concerning the incoming of disruption, the series of 20 values is carried out by means of a sigmoid that represents the risk of disruption (disruption t=time of disruption).

3 Neural Network Approach for Estimating of Time to Disruption

The philosophies of learning and the structures of the networks are complex, between the many kinds of NNs, after an attentive evaluation of the results obtained, we have decided to exploit Multi-Layer-Perceptron (MLP). Such choice is dictated from the type of variables that have used to train the network [7, 8]. The considered network associates to each input node a signal of diagnostic (9 input values to the time t); the output is an individual issue because the problem under study is a classification problem and that is codified by means of a real number ranging into [0, 1], that represents the risk of disruption. The input values are normalized into [0,1]. The target of the network for each disruptive shot is a sigmoid that represents the incoming of the risk of disruption, how represented in figure 1. If the shot is not disruptive, the target is considered zero. Therefore, the main problem remains the evaluation of the performance of the network. The quantification of such value is carried out by means of the number of false alarms (FA) and missed alarms (MA). In this work we use a MLP network with 9 inputs, 2 hidden layers of 6 and 5 neurones respectively; and 1 output (89 parameters totally). The exploited network is showed in figure 2.

The training step can be considered in two phases:

1. Exploitation of the training set;
2. Exploitation of the validating and testing sets to give a strong power of generalization.

For each phase, we have choice a set of parameters that optimize the performance of the network. The solution was chosen considering that the main target was a network with a strong power of generalization. A first method of comparison of the goodness of the obtained results is represented in the following. By dark blue lines, we have displayed the target and by green lines, the output of the

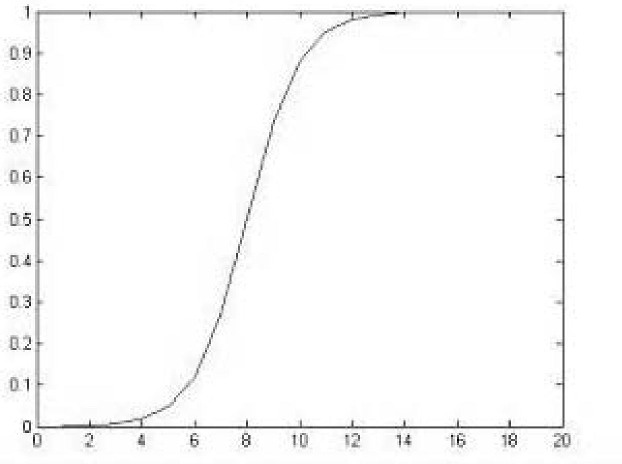


Fig. 1. The exploited sigmoid to compute the risk of disruption, the 20th sample correspond to t=td-40ms.

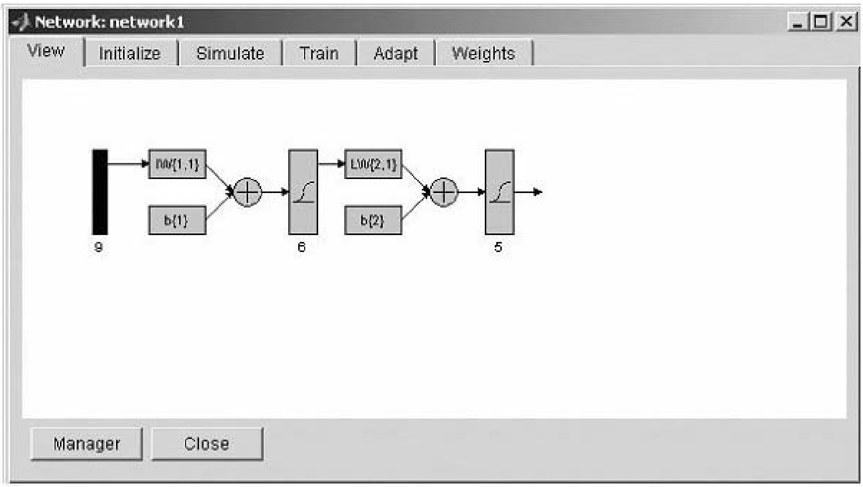


Fig. 2. Structure of the exploited network.

network (Figure 3). In particular, at the top of figure 3 we report the result concerning the training phase (the network has been trained by training data and the performance has been operated by means of training data); at the middle of figure 3, we report the result concerning the training and validating phases (the network has been trained by training and validating data and the performance has been operated by means of training data); and, finally, at the bottom of figure 3, we can see the result regarding the training validating and testing

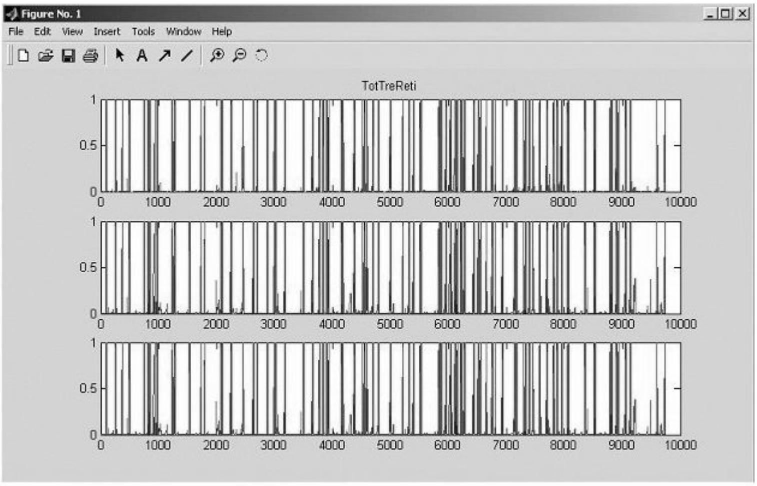


Fig. 3. From the top to the bottom: the performance of the network trained by training data, training-validating data, training+validating+testing data respectively when the input of simulation are training data.

phases (the network has been trained by training, validating and testing data and the performance has been operated by means of training data). By inspection of figure 3, we can see that the first simulation is better that the other ones. Therefore, even if the third simulation gives us the worse result, it has been trained by means of a larger database (generalized network); consequently, we focalise our attention on it. In particular, a zoom of the generalized network is reported in figure 4. Figure 5 reports a single disrupted shot. Table 1 displays the obtained errors for each database for each network respectively.

4 Neural Network Approach for Forecasting of Time to Disruption

The purpose of this section of the paper is, firstly, to subdivide the whole of the shots into two classes in which are stocked the disrupted and good shots respectively and, secondary, to classify a new shots in terms of good or disrupted shots. For this aim, we consider the sigmoid in terms of upper-bound of the disruption. In particular, we associate the value 1 to the disruption and 0 o the good shot. The main problem is to translate the information carried out from the network into the values 0 and 1. We consider the average value of the sigmoid in terms of the point of start for the decisional threshold. The values of the output of the network are averaged into the range [ttd-440ms, ttd-40ms]. If that value is grater than the decisional threshold, we can classify the shot as disrupted one, good shot otherwise. At this point, we have an array in which zeros and unities are present (vector of prediction); in other words, we have translated the risk of

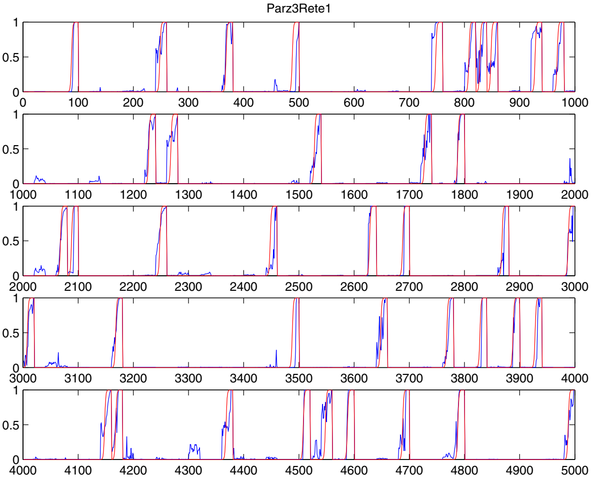


Fig. 4. Simulation at the bottom of Figure 3 (from 1 to 5000 samples). The red line is the target, the blue lines is the output of the network.

Table 1. Root Mean Square Errors (RMSE) for the exploited networks (estimation problem).

	Network 1	Network 2	Network 3
NTRAIN	0.129	0.181	0.181
NVALID			0.201
TEST			0.401

disruption (sigmoid) into sequence of zeros and unities. Now, we build a second array, labelled Real- array, in which are stocked zeros and unities carried out from the database. The final step of the procedure considers the construction of a new array in which each jth cell keeps the difference of the values of the correspondent cells in the predicted and real arrays. In the following, the possible values of the new array are reported.

$$Predicted - Real = \begin{cases} 0 \\ 1 \\ -1 \end{cases} \tag{1}$$

In particular, if the value is 0, then the network has carried out a good prediction; if the value is 1 the network has carried out a false alarm; if the value is -1 is a missing alarm. Table 2 displays the portions of missing and false alarms for each database under study.

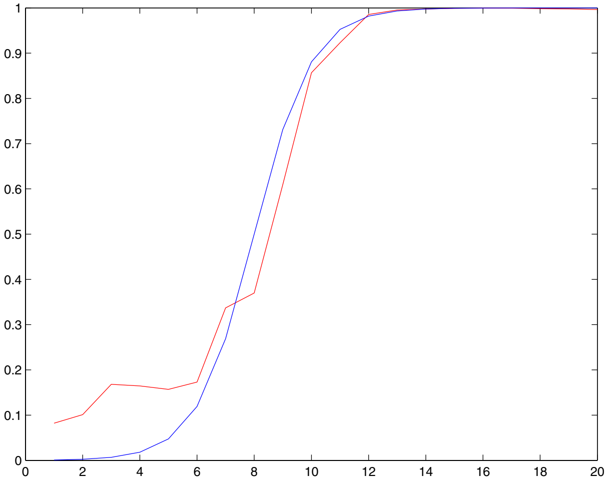


Fig. 5. The simulation of a single disrupted shot.

Table 2. Portions of Missing and False Alarms in each database (prediction problem).

Network Input	Training Set	Validation Set	Test Set
Missing Alarm	2/1760=0.0011	2/35=0.057	13/62=0.209
False Alarm	2/8000=0.0002	1/246=0.004	0/132

5 Conclusions

In this work, MLP Networks have been exploited for estimating and forecasting time-to-disruption in Tokamak Reactors, In particular, a database of disruptive discharges in JET machine have been analysed. Some diagnostics exploited for building the database are not available, then it need to replace them by means of a new set of diagnostics. That replacement is helding at the JET. Concerning the exploited database, the trained network can be considered as a good tool for the estimation and prediction of time-to-disruption. In particular, we can able to predict a disruption 40ms in advance, sufficient time to undertake control actions.

References

1. R. Dandy, Plasma Physics: “An Introductory Course”, Cambridge University Press, 1995, United Kingdom
2. A. Vannucci, et al., “Forecast of TEXT Plasma Disruptions Using Soft X Rays as Input Signal in a Neural Network”, Nuclear Fusion, 39, 1999, pp. 255.
3. D. Wroblewsky, “Neural Network valuation of Tokamak Current Profiles for Real Time Control”, Rev. Sci. Instrum, Vol. 68, 1997, p. 1281.

4. J.A. Wesson, et. al., “Disruptions in JET,” Nuclear Fusion, Vol 29, No.4, pp. 641-666, 1989. F.C. Schuller, “Disruptions in Tokamaks,” Plasma Physic Control Fusion, vol.37, pp. A135-A612, 1995.
5. E. Marongiu, “Neural Network Applications in Tokamaks,” Ph.D. Thesis University of Cagliari, January 30, 2002.
6. F. Milani, “Disruption Prediction at JET”, Ph.D. Dissertation, University of Aston in Birmingham, 1998.
7. G. Pautasso, et al., “Prediction and Mitigation of Disruptions in $ASDEX_{Upgrade}$”.

A Concurrent Neural Classifier for HTML Documents Retrieval

Giovanni Pilato[1], Salvatore Vitabile[1], Giorgio Vassallo[2], Vincenzo Conti[3], and Filippo Sorbello[1,3]

[1] Istituto di CAlcolo e Reti ad alte prestazioni
Italian National Research Council
Viale delle Scienze - 90128 Palermo - Italy
{g.pilato,s.vitabile}@icar.cnr.it
[2] Centro di Ricerche Elettroniche in Sicilia
Via della Regione Siciliana - Monreale (PA) - Italy
vassallo@cres.it
[3] Dipartimento di ingegneria INFOrmatica
University of Palermo
Viale delle Scienze - 90128 Palermo - Italy
sorbello@unipa.it,conti@csai.unipa.it

Abstract. A neural based multi-agent system for automatic HTML pages retrieval is presented. The system is based on the EαNet architecture, a neural network having good generalization capabilities and able to learn the activation function of its hidden units. The starting hypothesis is that the HTML pages are stored in networked repositories. The system goal is to retrieve documents satisfying a user query and belonging to a given class (i.e. documents containing the word "football" and talking about "Sports"). The system is composed by three interacting agents: the EαNet Neural Classifier Mobile Agent, the Query Agent, and the Locator Agent. The whole system was successfully implemented exploiting the Jade platform features and facilities. The preliminary experimental results show a good classification rate: in the best case a classification error of 9.98% is reached[1].

1 Introduction

With the recent increasing of digital libraries it has grown the necessity of designing automatic systems for information retrieval. Documents clustering and classification are two common and complex applications of text mining [11][12][13]. Many researchers proposed various approaches for dealing with this problem [6][7].

A lot of documents can be distributed over networked repositories; it is therefore important designing systems to help users to retrieve the requested information in short time and not overloading the network bandwidth.

[1] This research has been partially funded by Engineering Ingegneria Informatica SpA within the Teschet project.

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 210–217, 2003.

Multi-agent system platforms provide a smart approach to the design of intelligent data analysis tools for distributed environments [5][6]. In addition, mobile agents can be used for reaching the networked sources, performing computation locally and returning back the obtained results. In this paper a system, based on both neural networks and multi-agent paradigm, for automatic concurrent retrieval of HTML pages, is presented.

The system is composed by three agents: the Query Agent, the Locator Agent and the EαNet Classifier Mobile Agent. The proposed system has been implemented exploiting the features and facilities of the Jade platform[8], a multi-agent platoform FIPA specification compliant. At system start-up, an autonomous process for EαNet neural network training starts. The trained neural network is successively embedded into the EαNet Neural Classifier Mobile Agent. The user interacts, through the Query Agent, with the system in order to retrieve documents satisfying a query and belonging to a given class (i.e. documents containing the word "football" and talking about "Sports"). The EαNet Neural Classifier Mobile Agent receives the request and clones itself in the document repositories registered in the platform. Each clone interacts with the Locator Agent, present in each repository, in order to find the location of documents to be classified. After the classification task, each clones ends the results to the Query agent that shows them to the user.

The proposed system was trained using a random set of 1400 of pre-classified HTML pages. The downloaded pages belong to 7 classes of Google search engine: Arts, Business, Computers, Games, Health, Science, Sports. System performances were tested with 1400 new documents, uniformly distributed over the above classes. The documents have been distributed in three hosts and the preliminary experimental results show a classification error ranging from 9.98% to 29.16%.

2 Neural Architecture

The core of the proposed solution is based on EαNet architecture [1][3], a neural network architecture developed by the authors, whose performance has been measured using the classic method of the test set and the method of the quality factors[2].

To let the reading more fluent, both the methodology and the neural architecture are briefly summed-up in the following paragraphs.

2.1 Quality Factors

The authors have previously introduced three "quality factors" to give a measure, without using the test set, of the generalization capability of a feed-forward neural network[2]. In brief, the quality factors are referred to a typical feed forward architecture without shortcuts between its units[2]. The Quality factors are:

– The Learning Quality Factor Q_L quantifies the network learning capability over the training set samples.

- The Generalization Quality Factor Q_G computes the gradient of the network output function in the training points. It is an index inspired to the maximum entropy model and it quantifies *a priori* the network generalization capability.
- The Production Cost Quality Factor Q_P estimates the computational cost of the network during the production phase as the number of connections between the network units.

2.2 The EαNet Neural Network

The EαNet is a feed forward neural architecture capable to learn the activation function of its hidden units during the training phase giving a low value of both Q_G and Q_P quality factors compared to a traditional feed-forward network that uses sigmoidal activation functions for its hidden units [1][2][3].

This feature has been obtained through the combination of the CGD optimisation algorithm with the Powell restart conditions and the Hermite regression formula that uses the first R Hermite orthonormal functions to represent the activation function of each hidden unit. The starting number R of Hermite orthonormal functions per hidden unit is *a priori* chosen before starting the network learning process.

Each activation function changes until the minimum of Q_L has been obtained. The number of weights for each hidden unit changes dynamically using a pruning algorithm that exploits the information about the sensitivity of the Q_L factor with respect to each weight of each activation function belonging to a hidden unit.

The use of the Hermite approximation formula has been decided in order to obtain a smooth representation of the output function of any unit belonging to the output layer exploiting the properties of the Hermite polynomials. This leads to a very flexible neural architecture with good generalization capability [1][2][3].

3 The Multi-agent Platform

A multi-agent system (MAS) is an highly distributes system composed by autonomous, social and cooperative software agents which interact with a dynamic world. Each agent lives in its environment and is capable of interacting with the environment and with other agents.

The Foundation for Intelligent Physical Agents (FIPA) has formed for creating specifications for the implementation of multi-agent systems. The physical infrastructure in which agents can be deployed consists of: the hardware platforms, the operating system, the agents support software, the FIPA agent management components: DF, AMS, ACC, Internal Platform Message Transport.

According to the FIPA specifications, there must be at least one Directory Facilitator (DF) agent per platform; an agent can register its services with the DF. The DF allows *Yellow Pages* services and the agent can submit a query to the DF in order to find out services. Besides, the DF maintains accurate, complete, and up-to-date list of agents[9].

The Agent Management System is unique for the platform and it is responsible for managing the agents creation, deletion and migration. The Agent Communication Channel (ACC) routes messages between agents within the agent platform to agents resident on other agent platforms.

JADE (Java Agent DEvelopment Framework) is a software framework fully implemented in Java language which simplifies the implementation of multi-agent systems through a middle-ware that is FIPA-compliant and through a set of useful tools that support the debugging and deployment phase. It has a very flexible and efficient communication architecture and it allows the designing of agents capable to migrate or to clone themselves in other hosts[8].

4 The Proposed Concurrent Retrieval System

The proposed system (see fig. 1) is able to perform the document retrieval process on networked HTML pages repositories. The documents are processed locally and the user will receive only the adequate pages. This solution takes advantage of the EαNet generalization capabilities and of the use of the mobile agents paradigm, which allows the concurrent processing of the documents [6].

The proposed system is composed by three agents, each of one having its own function:

1. *The Query Agent*
2. *The Locator Agent*
3. *The EαNet Neural Classifier Mobile Agent*

Agent communication is based on messages exchange, according to the communication model supplied by the Jade platform.

4.1 The EαNet Training Phase

The training process is formed according to the following steps [4]:

- Documents downloading from Google Search Engine;
- HTML pages pre-processing and words frequency calculation;
- Dictionary of features construction;
- Training set building;
- EαNet Neural Classifier Training.

Google search engine downloaded documents are in English. The downloaded documents are then pre-processed as follows: it is extracted only the useful text, as HTML tags, JavaScript code, and so on. No word stemming has been used.

Documents are codified in multidimensional vectors as follows: for each word i and for each document belonging to the class k it is calculated the Term Frequency *Tf(i,k)*.

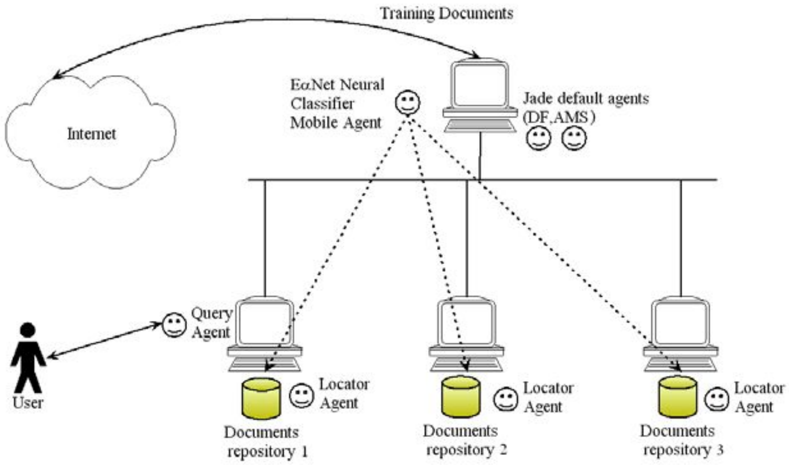


Fig. 1. The Concurrent Neural Classifier Architecture.

It is essential using a word weighting method to mark the words that are more suitable to build the vectors, therefore, for each class k and for each word i, the maximum value *MTf(i,k)* of the *Tf(i,k)* is computed. Then, for each class k and for each token i the percentage *Pf(i,k)* of documents belonging to the class k is found.

Let *Cf(i)* be the inverse of the number of classes in which the word k is present; the word k will have a score *S(i,k)*, computed as:

$$S\left(i,k\right) = \frac{MTf\left(i,k\right) \cdot Pf\left(i,k\right)}{Cf^{2}\left(i\right)} \tag{1}$$

Let N be the total number of document classes. The words representative of each class k will have the highest value of *S(i,k)*, therefore, it is chosen a number *Rp* of representative words belonging to the class k. These words constitute a set F_k that is a definite dictionary for the documents belonging to the class k. The number *Rp* is the same for all the N classes. The set F, defined as:

$$F = \bigcup_{i=1}^{N} F_i \tag{2}$$

forms the dictionary of features used for codifying the documents in vectors. The number of features can range between *Rp* and $N \cdot Rp$.

The pre-arranged downloaded web pages are then transformed in multidimensional vectors using the well-known *TF-IDF* formula and they will constitute the training set. The training set is used to train a set of M EαNet architectures with a pre-arranged number of hidden neurons: at the end of the selection process it is chosen the architecture which shows the lowest values of Q_L and Q_G quality factors, and the winner architecture is embedded into the Neural Classifier Mobile Agent.

4.2 The Query Agent

The Query Agent is running on each site accessible to users. Agent task is to get the user query associated with a desired class (i.e. documents containing the word “football” and talking about “Sports”). The agent submits a query to the DF in order to obtain the list of the agents providing the classification service, and sends them the *(query, class)* query. Then, it collects the answers and shows them to the user.

4.3 The Locator Agent

The Locator Agent is running on each repository where unorganised and unclassified HTML documents have been stored. The agent task is to communicate to the EαNet Neural Classifier Mobile Agent the location of the local documents directories. Each agent registers its own service with the DF agent at start-up and waits for the neural classifier requests.

4.4 The EαNet Neural Classifier Mobile Agent

At system start-up the EαNet Neural Classifier Mobile Agent registers itself with the DF service, then loads the dictionary F of the features and the trained EαNet architecture.

After receiving the *(query, class)* pair by the Query Agent, the EαNet Neural Classifier Mobile Agent clones itself on the library hosts. Each clone asks the location of the document directories to the Locator Agent, therefore it retrieves the documents satisfying the user query and transform them into a multidimensional vector using the following formula:

$$d_i = Tf\,(i) \cdot \log\left(\frac{C+1}{Df\,(i)+1}\right) \tag{3}$$

where C is the number of documents used for the training set, *Df(i)* is the document frequency of the word i in the training set and *Tf(i)* is the term frequency of the word i in the new document to be classified.

The transformed document is therefore given as input to the EαNet architecture in order to verify if it belongs to the desired class. After the retrieving process, each EαNet Neural Classifier Mobile Agent clone sends back to the Query Agent the list of the retrieved documents and their location.

5 Experimental Results

The whole system has been implemented and tested using the Jade platform exploiting its versatility and its tools for multi-agent system implementation as Agent Containers, Agent Management System for agents registration, Directory Facilitator for the yellow pages service and the Agent Communication Channel [8].

For the EαNet training phase were used a random set of C=1400 HTML documents belonging to the following Google classes: Arts, Business, Computers, Games, Health, Science, Sports. The documents are 200 for each class and they have not been pre-selected.

For the test phase, a new set 1400 documents, uniformly distributed on the above classes, have been spread on three hosts. The number of features per class and the number of hidden units for EαNet have been experimentally fixed, while the specific dictionary features and the number of Hermite orthonormal functions for each hidden unit have been fixed after the training phase. In the experimental trials we have chosen 150 features per class obtaining a database F composed by 978 features, i.e. 72 features are shared by the seven classes. For such number of features, the appropriate number of hidden units is 10.

A set of $M = 20$ neural architectures 978-10-7 has been launched, then the best one, i.e. the architecture with the lowest values of Q_L and Q_G has been selected and has been embedded into the EαNet Neural Classifier Mobile Agent.

The best trained EαNet architecture is characterized by the quality factors: Q_L=0, Q_G=370 $\cdot$ 10^3 and Q_P=9850.

In Table 1 the detailed classification error for the seven classes is reported. The spread ness of the classification error is due to the random choose of the training documents that can tarnish the set of illustrative example for the different classes.

Table 1. The classification error for each processed class.

Class	Arts	Business	Computers	Games	Health	Science	Sports
Classification Error	9.98%	17.87%	18.18%	16.13%	10.94%	29.16%	16.04%

6 Conclusions

In this paper a system, based on both neural networks and multi-agent paradigm, for automatic concurrent retrieval of HTML pages is presented. The proposed system has been implemented using the Jade platform and is composed by three agents: the Query Agent, the Locator Agent and the EαNet Classifier Mobile Agent. First experimental trials were conducted using a random set of 1400 HTML documents belonging to the Google classes search engine for system training and a new set 1400 documents, uniformly distributed on the above classes for system testing. The experimental results show that the proposed system is very efficient: the preliminary experimental results show that the best classification error is 9.98%.

References

1. Gaglio S., Pilato G., Sorbello F., Vassallo G.: Using the Hermite regression formula to design a neural architecture with automatic learning of the 'hidden' activation functions. Lecture Notes in Artificial Intelligence 1792, pages 226–237. Jan 2000 Springer-Verlag.

2. Pilato G., Sorbello F., Vassallo G.: An innovative way to measure the quality of a neural network without the use of the test set. International Journal of Artificial Computational Intelligence, Vol. 5 No 1, 2001, pp:31–36
3. Cirasa A., Pilato G., Sorbello F. Vassallo G.: An enhanced version of the αNet architecture: EαNet. Proc. of AI*IA Workshop of Robotics Parma, Italy, 1999.
4. Cirasa A., Pilato G., Sorbello F. and Vassallo G.: "EαNet: A Neural Solution for Web Pages Classification" - Proc. of 4th World MultiConference on Systemics, Cybernetics and Informatics - SCI'2000 - 23–26 July 2000, Orlando – Florida U.S.A.
5. Glitho R.H., Olougouna E., Pierre S.: "Mobile Agents and Their Use for Information Retrieval: a brief overview and an elaborate case study" - IEEE Network, 2002, 34–41
6. Brewington B., Gray R., Moizumi K., Kotz D., Cybenko G., Rus D.: Mobile agents in distributed information retrieval Intelligent Information Agents, 1999, Springer-Verlag, 1–32
7. Kosala R., Blockeel H.: Web Mining research: a survey ACM SIGKDD Explorations, 2000, Vol 2, No 1, 1-15
8. http://jade.cselt.it
9. http://www.fipa.org
10. Yiu-Kai Ng, June Tang, Michael Goodrich: A Binary-Categorization Approach for Classifyng Multiple-Record Web Documents Using Application Ontologies and a Probabilistic Model, Database Systems for Advanced Applications, 2001. Proceedings. Seventh International Conference on , 2001
11. Jian Liang, David Doermann, Matthew Ma, Jinhong K. Guo: Page Classification Through Logical Labelling, Pattern Recognition, 2002. Proceedings. 16th International Conference on, Volume: 3 , 2002
12. Nuanwan Soonthornphisaj, Boonserm Kijsirikul: The Effects of Different Feature Sets on the Web Page Categorization Problem Using the Iterative Cross-Training Algorithm, Proceeding of the International Conference on Enterprise and Information System (ICEIS), Setubal, Portugal,July 2001
13. Sankar K. Pal, Varun Talwar, Pabitra Mitra: Web Mining in Soft Computing Framework: Relevance, State of the Art and Future Direction, IEEE Transaction on Neural Networks, Vol. 13, No.9, September 2002

Bayesian Modelling for Packet Channels

Pierluigi Salvo Rossi[1], Gianmarco Romano[2],
Francesco Palmieri[2], and Giulio Iannello[1]

[1] Dipartimento di Informatica e Sistemistica, Università di Napoli "Federico II"
Via Claudio 21
80125 Napoli, Italy
{salvoros,iannello}@unina.it

[2] Dipartimento di Ingegneria dell'Informazione, Seconda Università di Napoli
Real Casa dell'Annunziata Via Roma 29
81031 Aversa (CE), Italy
{francesco.palmieri,gianmarco.romano}@unina2.it

Abstract. Performance of real-time applications on network communication channels are strongly related to losses and temporal delays. Several studies showed that these network features may be correlated and present a certain degree of memory such as bursty losses and delays. The memory and the statistical dependence between losses and temporal delays suggest that the channel may be well modelled by a Dynamic Bayesian Network with an appropriate hidden variable that captures the current state of the network. In this paper we propose a Bayesian model that, trained with a version of the EM-algorithm, seems to be effective in modelling typical channel behaviors.

1 Introduction

Gilbert and Elliott works [1][2] on modelling burst-error channels for bit transmission showed how a simple 2-state Hidden Markov Model (HMM) was effective in characterizing real communication channels. As in the case of bit-transmission channels, end-to-end packet channels show burst-loss behavior.

Jiang and Schulzrinne [9] investigated lossy behavior of packet channels finding that a Markov model is not able to describe appropriately the inter-loss behavior of channels. They also found that delays manifest temporal dependency, i.e. they should not be assumed as a memoryless phenomenon.

Salamatian and Vaton [10] found that a HMM trained with experimental data seems to capture channel loss behavior. They found that a HMM with 2 to 4 hidden states fits well experimental data.

These works suggest us that a Bayesian model should be effective in capturing the dynamic behavior of losses and delays on end-to-end packet channels. Our objective is to build a comprehensive model that jointly describes losses and delays.

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 218–225, 2003.

2 The Model

Fig.1 shows our reference model with a periodic source traffic with inter-departure period T and fixed packet size. The network randomly cancels and delays packets according to current congestion.

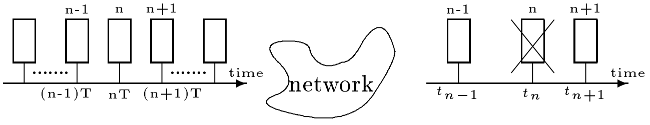


Fig. 1. End-to-end packet channel.

Let us number transmitted packets, $n = 1, 2, \ldots$, and let us denote with t_n and τ_n the arrival time and the accumulated delay of the *n-th* packet respectively, i.e.

$$\tau_n = t_n - nT. \tag{1}$$

We want to model the system in a way which carries information on current congestion that may determine variable loss rates and average delays. Loss phenomenon shows a bursty behavior, i.e. it cannot be thought as a memoryless stochastic process. Moreover, several works [6][8][9] showed that temporal delays seem not to be well modelled as independent identically distributed (iid) random variables, i.e. also delays, as losses, present a certain degree of memory. In real communication networks, losses and delays are strongly correlated; it has been observed [9] that in proximity of a loss, larger delays tend to occur.

Memory presence in phenomena we want to model suggests us to introduce a hidden state variable that stochastically influence losses and delays. State variable is hidden because our knowledge about it comes from observation of loss and delays, and there is no way to access directly to it.

In the following x_n, l_n and τ_n respectively will denote "state", "loss" and "delay" at time nT,

$$\begin{aligned} x_n &\in \{s_1, s_2, \ldots, s_N\} \\ l_n &\in \{v_1, v_2\} \end{aligned} \tag{2}$$

where s_i is the *i-th* state of the network and v_1 (resp. v_2) means the absence (resp. presence) of a loss.

It should be noted that being in the presence of a loss, the delay has no real value, it can be considered infinite. For easy use we consider,

$$\begin{aligned} \tau_n/\{l_n = v_1\} &\in [0, +\infty) \\ \tau_n/\{l_n = v_2\} &= -1 \end{aligned} \tag{3}$$

Our reference Bayesian model is shown in Fig.2 where the arrows represent statistical dependence among variables. More specifically, the set of parameters characterizing the model is $\Lambda = \{\mathbf{A}, \mathbf{p}, f_1(\tau), f_2(\tau), \ldots, f_N(\tau)\}$, where

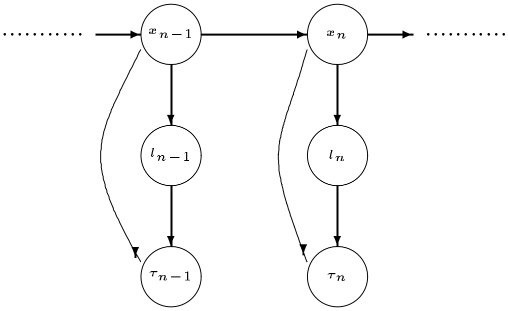


Fig. 2. The Bayesian model for packet channel.

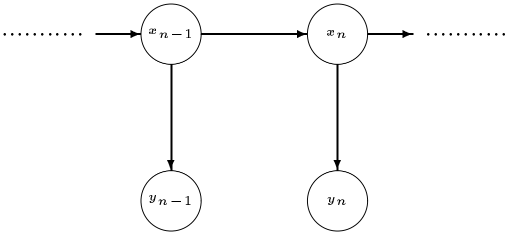


Fig. 3. Hidden Markov Model.

- **A** is the state transition matrix, i.e.

$$a_{ij} = Pr(x_{n+1} = s_j / x_n = s_i) \quad {}_{\substack{j\in\{1,2,\ldots,N\} \\ i\in\{1,2,\ldots,N\}}} \tag{4}$$

- **p** is the loss probability vector, i.e.

$$\begin{cases} p_i = Pr(l_n = v_1 / x_n = s_i) \\ 1 - p_i = Pr(l_n = v_2 / x_n = s_i) \end{cases} \quad {}_{i\in\{1,2,\ldots,N\}} \tag{5}$$

- $f_i(\tau)$ is the delay conditional pdf, i.e.

$$Pr(\tau_n > t / x_n = s_i, l_n = v_1) = \int_t^{+\infty} f_i(\tau) d\tau \tag{6}$$

The model can be reduced to a HMM, as seen on Fig.3, with a hidden variable x_n and an observable variable y_n that represents jointly loss and delay as

$$y_n = \begin{cases} \tau_n \ if \ l_n = v_1 \\ -1 \ if \ l_n = v_2 \end{cases} \tag{7}$$

Summarizing:

- x_n is a discrete random variable whose dynamic behavior follows Eq.(4)

- y_n is a hybrid random variable characterized, given $\{x_n = s_i\}$, by the following conditional pdf,

$$b_i(t) = p_i f_i(t) + (1 - p_i)\delta(t + 1) \tag{8}$$

It should be noted, as shown in Fig.4 that y_n is a hybrid variable obtained as a mixture of two components (one continuous, one discrete), when introduced the "trick" of associating $\tau_n = -1$ to a loss in order to have non-overlapping distributions for continuous and discrete components. The continuous component describes network delays behavior in the absence of losses, whereas the discrete component describes losses behavior.

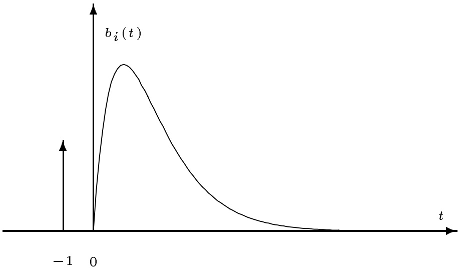


Fig. 4. An example of the conditional pdf $b_i(t)$ for the hybrid variable y_n.

If π is the stationary state probability distribution, i.e.

$$\pi_i = \lim_{n \to \infty} \{Pr(x_n = s_i)\} \tag{9}$$

the loss probability and the average delay of the model are:

$$Pr(loss) = \sum_{i=1}^{N} \pi_i (1 - p_i) \tag{10}$$

$$\overline{delay} = \sum_{i=1}^{N} \pi_i \int_0^{+\infty} t f_i(t) dt \tag{11}$$

3 Learning Parameters of the Model

The Expectation-Maximization algorithm [7] is an optimization procedure searching for a new set of parameters for a stochastic model according to improvements of the likelihood of a given sequence of observable variables. For structures like HMM of Fig.3 this optimization technique reduces to the Forward-Backward algorithm [3][4][5] studied for discrete and continuous observable variables with a broad class of allowed conditional pdf.

More specifically, given a sequence of observable variables $\mathbf{y} = (y_1, y_2, \ldots, y_K)^T$, and a set of parameters $\lambda = \{\mathbf{A}, \mathbf{p}, \mu\}$, where

$$\mu_i = E[\tau_n / \{x_n = s_i\}] = \int t f_i(t) dt \tag{12}$$

the update $\hat{\lambda} = \{\hat{\mathbf{A}}, \hat{\mathbf{p}}, \hat{\mu}\}$ of λ follows the recursions

$$\hat{a}_{ij} = \frac{\sum_{k=1}^{K-1} \alpha_k(i) a_{ij} b_j(y_{k+1}) \beta_{k+1}(j)}{\sum_{k=1}^{K-1} \alpha_k(i) \beta_k(i)} \tag{13}$$

$$\hat{p}_i = \frac{\sum_{k=1}^{K} \rho_k(i) \beta_k(i)}{\sum_{k=1}^{K-1} \alpha_k(i) \beta_k(i)} \tag{14}$$

$$\hat{\mu}_i = \frac{\sum_{k=1}^{K} \rho_k(i) \beta_k(i) y_k}{\sum_{k=1}^{K-1} \rho_k(i) \beta_k(i)} \tag{15}$$

where

$$\alpha_k(j) = \sum_{i=1}^{N} \alpha_{k-1}(i) a_{ij} b_j(y_k) \tag{16}$$

$$\beta_k(i) = \sum_{j=1}^{N} a_{ij} b_j(y_{k+1}) \beta_{k+1}(j) \tag{17}$$

are the forward and backward partial likelihood, and where

$$\rho_k(j) = \sum_{i=1}^{N} \alpha_{k-1}(i) a_{ij} p_j \left. \frac{\partial b_j(t)}{\partial p_j} \right|_{t=y_k} \tag{18}$$

The iteratively procedure will reach a local maximum point of the likelihood function,

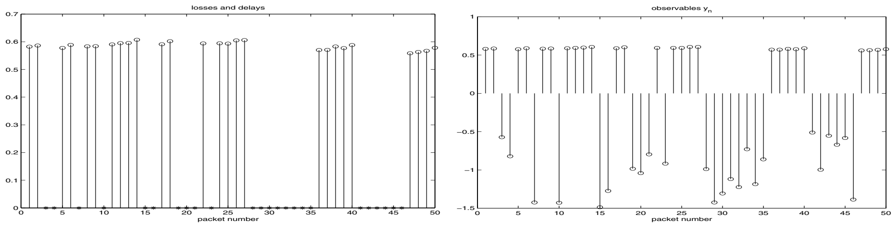
$$L(\mathbf{y}; \lambda) = Pr(\mathbf{y}/\lambda) = \sum_{i=1}^{N} \alpha_K(i) \tag{19}$$

which typically depends on the starting point λ. When necessary, repeated starts with different initial conditions provide the global solution.

The problem of the Dirac-impulse in the conditional pdf (8) was avoided considering a modified function

$$\tilde{b}_i(t) = p_i f_i(t) + (1 - p_i) g(t) \tag{20}$$

where $g(t)$ is any pdf such that $g(t) = 0 \ , \quad \forall \ t \geq 0$, in order to have non-overlapping supports between $f_i(t)$ and $g(t)$. Obviously, while the set $\{f_i(t)\}_{i=1}^{N}$ will be adjusted by the iterative procedure, $g(t)$ will remain unchanged, as only its area is relevant. This means that losses, in the algorithm have to be randomized according to $g(t)$.



(a)Losses and delays sequence. (b) Corresponding observable sequence.

Fig. 5. Portion of a measured trace on real network.

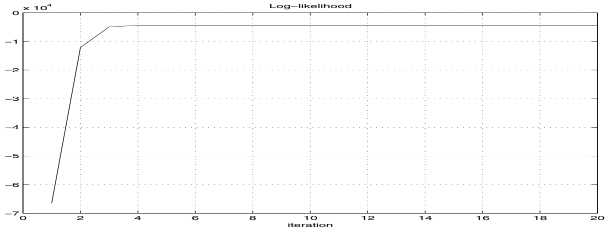


Fig. 6. Log-likelihood trend.

4 Experimental Results

Measures of losses and delays have been performed between the *Dipartimento di Informatica e Sistemistica, Universitá di Napoli "Federico II"*, and the *Dipartimento di Ingegneria dell'Informazione, Seconda Universitá di Napoli*, using the software Internet Traffic Generator (ITG) [12].

ITG, a new version of Mtools [11], can generate both traffic at transport layer and "layer 4-7". It implements both TCP and UDP traffic generation according to several statistical distributions both for inter-departure times and packet sizes. ITG allows simulations of complex traffic sources furnishing information about transmitted and received packets.

The characteristics of generated traffic are: inter-departure period $T = 5 \cdot 10^{-3}$ *sec.* and packet size of 1000 *bytes*, $(bit - rate = 1.6\ Mbps)$.

A typical trace obtained is shown in Fig.5(a), while in Fig.5(b) the corresponding sequence used for learning procedure is shown.

The learning procedure was applied on observable sequences of 500 to 1000 samples, finding in reasonable time (10*sec.*) acceptable estimation of network behavior. Fig.6 shows a typical trend of the log-likelihood obtained in the learning procedure.

Our choice of conditional pdf's for delays was Gamma distributions, as suggested by several works [6][8],

$$f_i(t) = \frac{t^{\gamma_i - 1} e^{-t}}{\Gamma(\gamma_i)} u(t), \tag{21}$$

while losses was randomized according to a uniform distribution, i.e.

Table 1. Example of parameters learning with a 2-states model.

	$Pr(loss)$	$\overline{delay}$
measured	0.494	$543.69\,ms$
starting model	0.117	$131.06\,ms$
trained model ($10 iterations$)	0.495	$492.58\,ms$
trained model ($20 iterations$)	0.494	$565.81\,ms$

Table 2. Example of parameters learning with a 3-states model.

	$Pr(loss)$	$\overline{delay}$
measured	0.494	$543.69\,ms$
starting model	0.365	$143.12\,ms$
trained model ($10 iterations$)	0.495	$492.58\,ms$
trained model ($20 iterations$)	0.494	$536.92\,ms$

$$g(t) = \begin{cases} 1 & t \in [-3/2, -1/2] \\ 0 & t \in R - [-3/2, -1/2] \end{cases} \tag{22}$$

Tabs.1 and 2 show loss probability and average delay of a measured trace used as training sequence, and of a model with 2 and 3 states before and after learning procedure, according to (10),(11).

Actually we are investigating on generalization capability. Preliminary tests show that the model is able to follow the channel behavior until its characteristics can be considered almost stationary.

5 Conclusion and Future Work

In this paper we have proposed a Bayesian Network whose objective is to model end-to-end packet channel behavior, jointly capturing losses and delays characteristics. The proposed model generalizes the HMM description of real channels introducing a memory stochastic modelling of delays. Preliminary results are encouraging and future works will be focused on model improvements and coding strategies.

Acknowledgement

This work has been carried out partially under the financial support of the Ministero dell'Istruzione, dell'Università e della Ricerca (MIUR) in the framework of the FIRB Project "Middleware for advanced services over large-scale, wired-wireless distributed systems (WEB-MINDS)".

References

1. Gilbert, E.N.: Capacity of a burst-noise channel. In: Bell System Technical Journal, Vol. 39 (Sept. 1960), 1253–1265

2. Elliott, E.O.: Estimates of error-rate for codes on burst-noise channels. In: Bell System Technical Journal, Vol. 42 (Sept. 1963), 1977–1997
3. Liporace, L.A.: Maximum Likelihood Estimation for Multivariate Observations of Markov Sources. In: IEEE Transactions on Information Theory, Vol. IT-28(5) (Sept. 1982), 729–734
4. Juang, B.H., Levinson, S.E., Sondhi, M.M.: Maximum Likelihood Estimation for Multivariate Mixture Observations of Markov Chains. In: IEEE Transactions on Information Theory, Vol. IT-32(2) (Mar. 1986), 307–309
5. Rabiner, L.R.: A tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. In: Proceedings of the IEEE, Vol. 77(2) (Feb. 1989), 257–285
6. Bolot, J.C.: Characterizing End-to-End Packet Delay and Loss in the Internet. In: Journal of High-Speed Networks, Vol. 2(3) (Dec. 1993), 305–323
7. Bilmes, J.A.: A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models. Technical Report, ICSI-TR-97-021, University of Berkeley,CA, 1998.
8. Paxson, V.: End-toEnd Internet Packet Dynamics. In: IEEE Transactions on Networking, Vol. 7(3) (June 1999), 277–292
9. Jiang, W., Schulzrinne, H.: Modeling of Packet Loss and Delay and Their Effect on Real-Time Mulrimedia Service Quality. In: 10th International Workshop on Network and Operating System Support for Digital Audio and Video, June 2000.
10. Salamatian, K., Vaton, S.: Hidden Markov Modeling for network communication channels. In: ACM Sigmetrics/Performance, Vol. 29 (2001), 92–101
11. Avallone, S., D'Arienzo, M., Esposito, M., Pescapè, A., Romano, S.P., Ventre, G.: Mtools. In: Networking Column on IEEE Network, Vol. 16(5) (Sept./Oct. 2002), pag.3
12. Avallone, S., Pescapè, A., Ventre, G.: Analysis and Experimentation of Internet Traffic Generator. Submitted to: ACM Workshop on Models, Methods and Tools for Reproducible Network Research

Neural Networks for Photometric Redshifts Evaluation

Roberto Tagliaferri[1,2], Giuseppe Longo[3], Stefano Andreon[4], Salvatore Capozziello[5], Ciro Donalek[2,3,6], and Gerardo Giordano[3]

[1] DMI - University of Salerno, 84081, Baronissi (SA), Italy
robtag@unisa.it
[2] INFM, Unità di Salerno, 84081 Baronissi, Italy
[3] Department of Physical Sciences, University Federico II of Naples, I-80126, Italy
[4] INAF-Osservatorio Astronomico di Brera, Milano
[5] Dipartimento di Fisica, Università di Salerno, Baronissi, Italy
[6] Dipartimento di Matematica Applicata, University Federico II Naples
I-80126, Italy

Abstract. We present a neural network based approach to the determination of photometric redshift, which is a very important parameter to find the depth of astronomical objects in the sky. The method was tested on the Sloan Digital Sky Survey Early Data Release reaching an accuracy comparable and, in some cases, better than Spectral Energy Distribution template fitting techniques. We used Multi-Layer Perceptrons operating in a Bayesian framework to compute the parameter estimation, and a Self Organizing Map to estimate the accuracy of the results, evaluating the contamination between the classes of objects with a good prediction rate and with a poor one. In the best experiment, the implemented network reached an accuracy of 0.020 (robust error) in the range $0 < z_{phot} < 0.3$, and of 0.022 in the range $0 < z_{phot} < 0.5$.

1 Introduction

Redshifts number among the most crucial cosmological parameters. They, in fact, are a conventional term to denote the recession velocity of galaxies and trough the Hubble law which establishes a linear relationship between distance and recession velocity, redshifts become the most effective way to evaluate galaxy distances. The accurate knowledge of the redshifts for large samples of galaxies is therefore a pre-condition for most extragalactic and cosmological studies. Unfortunately, the measurement of accurate redshifts requires low/medium resolution spectroscopy with large telescopes, a technique which is very demanding in terms of (expensive) telescope time. An alternative (even though less accurate) approach is the evaluation of the so called "photometric redshifts", id est the derivation of redshift estimates starting from photometric data obtained in several broad or intermediate photometric bands. This technique exploites the fact that in wide field astronomical images tens of thousands of objects are recorded at the same time and only a few exposure are required to provide the needed input data. Many different approaches have been proposed to the evaluation of

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 226–234, 2003.

photometric redshifts (see for instance [1–4]). An approach, which is in the same line of the one discussed here, can be applied only to what we shall call 'mixed surveys', *id est* datasets where accurate and multiband photometric data for a large number of objects are supplemented by spectroscopic redshifts for a small but statistically significant subsample of the same objects. In this case, the spectroscopic data can be used to constrain the fit of a polynomial function mapping the photometric data [5–7].

Interpolative methods offer the great advantage that they are trained on the real Universe and do not require strong assumptions on the physics of the formation and evolution of stellar populations. Neural Networks (hereafter NNs) are known to be excellent tools for interpolating data and for extracting patterns and trends (cf. the standard textbook by Bishop [8]) and in this paper, we shall discuss the application of a set of neural tools to the determination of photometric redshifts in large "mixed surveys". The Multi Layer Perceptron (MLP) in the framework of the Bayesian learning was used to interpolate the photometric redshfit with a very good predictive result on objects until a given depth, while Self Organising Maps (SOM) were used to identify the confidence of the objects to belong to good prediction classes and to evaluate the degree of contamination of the final redshift catalogues.

2 Neural Networks

NNs, over the years, have proven to be a very powerful tool capable to extract reliable information and patterns from large amounts of data even in the absence of models describing the data [8] and are finding a wide range of applications also in the astronomical community: catalogue extraction [9], star/galaxy classification [10, 9], galaxy morphology [11, 12], classification of stellar spectra [13–15], data quality control and data mining [16].

The AstroMining software [17] is a package written in the Matlab environment to perform a large number of data mining and knowledge discovery tasks, both supervised and unsupervised, in large multiparametric astronomical datasets. The package relies also on the Matlab "Neural Network", the "SOM" [18] and the "Netlab" [19] toolboxes.

Using AstroMining, via interactive interfaces, it is possible to perform a large number of operations: i) manipulation of the input data sets; ii) selection of relevant parameters; iii) selection of the type of neural architecture; iv) selection of the training validation and test sets; v) etc. The package is completed by a large set of visualization and statistical tools which allow to estimate the reliability of the results and the performances of the network. The user friendly interface and the generality of the package allow both a wide range of applications and the easy execution of experiments (more details on other aspects of the AstroMining tool which are not relevant to the present work may be found in [16].

2.1 The Multi-layer Perceptron – MLP

A NN is usually structured into an input layer of neurons, one or more hidden layers and one output layer. Neurons belonging to adjacent layers are usually fully connected and the various types and architectures are identified both by the different topologies adopted for the connections and by the choice of the activation function. Such networks are generally called Multi Layer Perceptron (MLP; [8]) when the activation functions are sigmoidal or linear. Due to its interpolation capabilities, the MLP is one of the most widely used neural architectures. We implemented an MLP with one hidden layer and n input neurons, where n is the number of parameters selected by the user as input in each experiment.

It is possible to train NN's also in the Bayesian framework, which allows to find the more efficient among a population of NN's differing in the hyperparameters controlling the learning of the network [8], in the number of hidden nodes, etc.

The Bayesian method allows the values of the regularization coefficients to be selected using only the training set, without the need for a validation set.

The implementation of a Bayesian framework requires several steps: initialization of weights and hyperparameters; training the network via a non linear optimization algorithm in order to minimize the total error function. Every few cycles of the algorithm, the hyperparameters are re-estimated and eventually the cycles are reiterated.

2.2 The Self-organizing Maps

The SOM algorithm [20] combines a competitive learning principle with a topological structuring of nodes such that adjacent nodes tend to have similar weight vectors. The training is unsupervised and it is entirely data-driven and the neurons of the map compete with each other [18]. These networks are Self Organizing in that, after training, nodes tend to attain weight vectors that capture the characteristics of the input vector space. SOM allows an approximation of the probability density function of the training data, the derivation of prototype vectors best describing the data, and a highly visualized and user friendly approach to the investigation of the data. This property turns SOM into an ideal tool for KDD and expecially for its exploratory phase: data mining [18].

During the training phase, one sample vector $\mathbf{x}$ from the input data set is randomly chosen and a similarity measure is calculated between it and all the weight vectors of the map. The Best-Matching Unit (BMU), denoted as c, is the unit with weight vector having the greatest similarity with the input sample $\mathbf{x}$. The similarity is usually defined by means of a distance measure, typically an Euclidean distance. After finding the BMU, the weight vectors of the SOM are updated. The training is usually performed into two phases. In the first phase, relatively large initial α value and neighborhood radius are used. In the second phase both the α value and the neighborhood are small from the beginning. This procedure corresponds to first tuning the SOM approximately to the same space as the input data and then fine-tuning the map. The SOM toolbox [18] includes the tools for the visualization and analysis of SOM. Another advantage of SOM is that it is relatively easy to label individual data, *id est* to identify which neuron is activated by a given input vector. The utility of these properties of the SOM will become clear in the next paragraphs.

3 Application to the SDSS-EDR Data

A preliminary data release (Early Data Release or EDR) of the SDSS was made available to the public in 2001 [21]. This data sets provide photometric, astrometric and morphological data for an estimated 16 millions of objects in two fields: an Equatorial 2^{circ} wide strip of constant declination centered around δ=0 and a rectangular patch overlapping with the SIRTF First Look Survey.

The EDR provides also spectroscopic redshifts for a little more than 50.000 galaxies distributed over a large redshift range and is therefore representative of the type of data which will be produced by the next generation of large scale surveys. In order to build the training, validation and test sets, we first extracted from the SDSS-EDR a set of parameters (u, g, r, i, z, both total and petrosian magnitudes, petrosian radii, 50% and 90% petrosian flux levels, surface brightness and extinction coefficients, [21] for all galaxies in the spectroscopic sample.

In this data set, redshifts are distributed in a very dishomogeneous way over the range $0 - 7.0$ (93% of the objects have $z < 0.7$).

It needs to be stressed that the highly dishomogeneous distribution of the objects in the redshift space implies that the density of the training points dramatically decreases for increasing redshifts, and that: i) unless special care is paid to the construction of the training set, all networks will tend to perform much better in the range where the density of the training points is higher; ii) the application to the photometric data set will be strongly contaminated by the spurious determinations.

3.1 The Photometric Redshift Evaluation

The experiments were performed using the NNs in the Matlab and Netlab Toolboxes, with and without the Bayesian framework. All NNs had only one hidden layer and the experiments were performed varying the number of the input parameters and of the hidden units. Extensive experiments lead us to conclude that the Bayesian framework provides better generalization capabilities with a lower risk of overfitting, and that an optimal compromise between speed and accuracy is achieved with a maximum of 22 hidden neurons and 10 Bayesian cycles.

In Table 1, we summarize some of the results obtained from the experiments and, in Figure 1, we compare the spectroscopic redshifts versus the photometric redshifts derived for the test set objects in the best experiment.

3.2 Contamination of the Catalogues

In practical applications, one of the most important problems to solve is the evaluation of the contamination of the final photometric redshift catalogues or, in other words, the evaluation of the number of objects which are erroneously attributed a z_{phot} significantly (accordingly to some arbitrarily defined threshold) different from the unknown z_{spec}. This problem is usually approached by means of extensive simulations. The problem of contamination is even more relevant

Table 1. Column 1: higher accepted spectroscopic redshift for objects in the training set; column 2: input parameters used in the experiment; column 3: number of neurons in the hidden layer; column 4: robust errors evaluated on the test set; column 5: number of objects used in each of the training, validation and test set.

Range	**parameters**	**h.n.**	**err.**	**obj.s**
$z < 0.3$	r, u-g, g-r, r-i, i-z	18	0.029	12000
$z < 0.5$	r, u-g, g-r, r-i, i-z	18	0.031	12430
$z < 0.7$	r, u-g, g-r, r-i, i-z	18	0.033	12687
$z < 0.3$	r, u-g, g-r, r-i, i-z, radius	18	0.025	12022
$z < 0.5$	r, u-g, g-r, r-i, i-z, radius	18	0.026	12581
$z < 0.7$	r, u-g, g-r, r-i, i-z, radius	18	0.031	12689
$z < 0.3$	r, u-g, g-r, r-i, i-z, radius, p. fluxes, s. brightness	22	0.020	12015
$z < 0.5$	r, u-g, g-r, r-i, i-z, radius, p. fluxes, s. brightness	22	0.022	12536
$z < 0.7$	r, u-g, g-r, r-i, i-z, radius, p. fluxes, s. brightness	22	0.025	12680

in the case of NNs based methods, since NNs are necessarily trained only in a limited range of redshifts and, when applied to the real data, they will produce misleading results for most (if not all) objects which "in the real word" have redshifts falling outside the training range. This behaviour of the NNs is once more due to the fact that while being good interpolation tools, they have very little, if any, extrapolation capabilities.

Moreover, in the SDSS-EDR spectroscopic sample, over a total of 54,008 objects having $z > 0$, only 88%, 91% and 93% have redshift z lower than, respectively than 0.3, 0.5 and 0.7. To train the network on objects falling in the above ranges implies, respectively, a minimum fraction of 12%, 9% and 7% of objects in the photometric data set having wrong estimates of the photometric redshift.

An accurate estimate of the contamination may be obtained using unsupervised SOM clustering techniques over the training set.

In Figure 2 we show the position of the BMU as a function of the redshift bin. Each exagon represents a neuron and the figures inside it give the number of input vectors (in a given range) which have that neuron as BMU. It is clearly visible that low redshift objects ($z < 0.5$) tend to activate neurons in the lower right part of the map, intermediate redshift ones ($0.5 < z < 0.7$) neurons in the lower left part and, finally, objects with redshift higher than 0.7 activate only the neurons in the upper left corner. The labeling of the neurons (shown in the upper left map) was done using the training and validation data sets in order to avoid overfitting, while the confidence regions were evaluated on the test set. In Figure xx we split the objects into two groups: in the first one we included all objects with $z < 0.5$ and in the second one all those with $z >= .5$. Each cell is labeled as class 1 or class 2 accordingly to the relative distribution of input vectors belonging to a given group which activate that cell. Therefore, test set may be used to map the neurons in the equivalent of confidence regions and

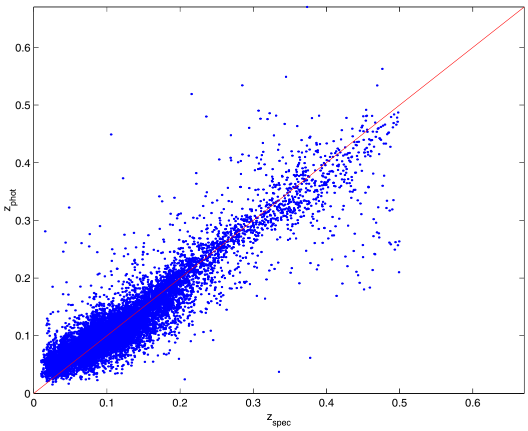


Fig. 1. Photometric versus spectroscopic redshifts obtained with a Bayesian MLP with 2 optimization cycles, 50 learning epochs of quasi-Newton algorithm and 5 inner epochs for hyperparameter optimization. Hyperparameters were initialized at α=0.001 and β=50.

to evaluate the degree of contamination to be expected in any given redshift bin. Conversely, when the network is applied to real data, the same confidence regions may be used to evaluate whether a photometric redshift correspondent to a given input vector may be trusted upon or not.

The above derived topology of the network is also crucial since it allows to derive the amount of contamination. In order to understand how this may be achieved, let us take the above mentioned NN, and consider the case of objects which are attributed a redshifts $z_{phot} < 0.5$. This prediction has a high degree of reliability only if the input vector activates a node in the central or right portions of the map. Vector producing a redshift $z_{phot} < 0.5$ but activating a node falling in the upper left corner of the map are likely to be misclassified. In our experiment, out of 9270 objects with $z_{phot} < 0.5$, only 39 (*id est*, 0.4% of the sample) have discordant spectroscopic redshift. A confusion matrix helps in better quantifying the quality of the results. In Table 2, we give the confusion (or, in this case, 'contamination') matrix obtained dividing the data into three classes accordingly to their spectroscopic redshifts, namely class I: $0 < z < 0.3$, class II: $0.3 < z < 0.5$, class III: $z > 0.5$. The elements on the diagonal are the correct classification rates, while the other elements give the fraction of objects belonging to a given class which have been erroneously classified into another class. Furthermore, in the redshift range $(0, 0.3)$, 95.4% of the objects are correctly identified and only 4.6% is attributed a wrong redshift estimate. In total, 94.2% are correctly classified. By taking into account only the redshift range $0 < z < 0.5$, this percentage becomes 97.3%. From the confusion matrix, we can therefore derive a completeness of 97.8% and a contamination of about 0.5%.

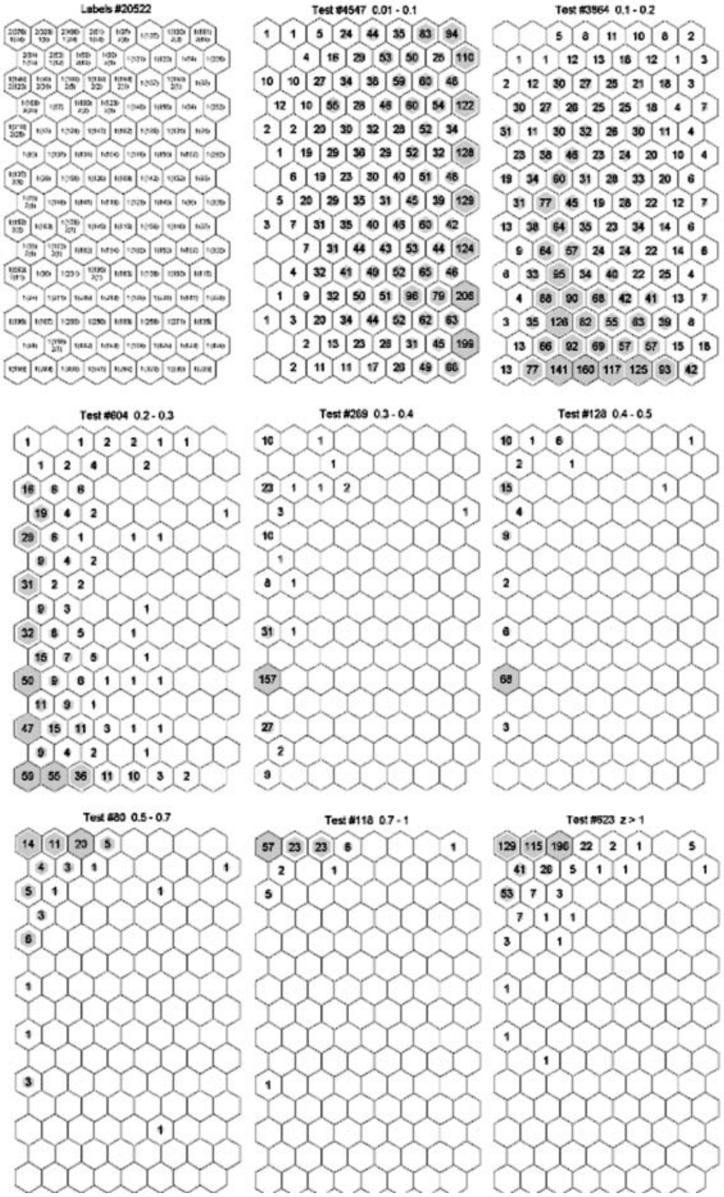


Fig. 2. Maps of the neuron activated by the input data set. Exagons represent the NN nodes. In the map in the upper left corner, for a given node, the figures n(m) can be read as follows: n is the class (n=1 meaning $z < 0.5$ and n=2 meaning $z > 0.5$) and m is the number of input vector of the correspondent class which have activated that node. This map was produced using the training and validation data sets. The other maps, produced each in a different redshift bin, indicate how many input vector from the test data set activated a given node.

Table 2. Confusion matrix for the three classes described in the text.

	objects	Class I	Class II	Class III
Class I	9017	95.4%	2.96%	1.6%
Class II	419	6.4%	76.6%	16.9%
Class III	823	3.8%	2.1%	94.2%

4 Summary and Conclusions

The application of NNs to mixed data, *id est* spectroscopic and photometric surveys, allows to derive photometric redshifts over a wide range of redshifts with an accuracy equal if not better to that of more traditional techniques.

The method makes use of two different neural tools: i) an MLP in Bayesian framework used to estimate the photometric redshifts; ii) an unsupervised SOM used to derive the completeness and the contamination of the final catalogues. On the SDSS-EDR, the best result (robust error = 0.020) was obtained by a MLP with 1 hidden layer of 22 neurons, after 10 Bayesian cycles.

The method fully exploits the wealth of data provided by the new digital surveys since it allows to take into account not only the fluxes, but also the morphological and photometric parameters.

The proposed method will be particularly effective in mixed surveys, *id est*, in surveys were a large amount of multiband photometric data is complemented by a small subset of objects having also spectroscopic redshifts.

References

1. Koo D.C., 1999, astro-ph/9907273
2. Fernandez-Soto A., Lanzetta K.A., Chen H.W., Pascarelle S.M., Yakate N., 2001, ApJSS, 135, 41
3. Massarotti M., Iovino A., Buzzoni A, 2001a, AA, 368, 74
4. Massarotti M., Iovino A., Buzzoni A., Valls-Gabaud D., 2001b, AA, 380, 425
5. Connolly A.J., Csabai I., Szalay A.S., Koo D.C., Kron R.G., Munn J.A., 1995, AJ, 110, 2655
6. Wang Y., Bachall N., Turner E.L., 1998, AJ, 116, 2081
7. Brunner R.J., Szalay A.S., Connolly A.J., 2000, ApJ, 541, 527
8. Bishop C.M., 1995, Neural Networks for Pattern Recognition, Oxford University Press
9. Andreon S., Gargiulo G., Longo G., Tagliaferri R., Capuano N., 2000, MNRAS, 319, 700
10. Bertin E., Arnout S., 1996, AAS, 117, 393
11. Storrie-Lombardi M.C., Lahav O., Sodré L. jr, Storrie-Lombardi L.J., 1992, MNRAS, 259, 8
12. Lahav O., Naim A., Sodré L. jr., Storrie-Lombardi M.C., 1996, MNRAS, 283, 207
13. Bailer-Jones C.A.L., Irwin M., von Hippel T., 1998, MNRAS, 298, 361
14. Allende Prieto C., Rebolo R., Lopez R.J.G., Serra-Ricart M., Beers T.C., Rossi S., Bonifacio P., Molaro P., 2000, AJ, 120, 1516
15. Weaver W.B., 2000, ApJ, 541, 298

16. Tagliaferri R., Longo G., Milano L., Acernese F., Barone F., Ciaramella A., De Rosa R., Donalek C., Eleuteri A., Raiconi G., Sessa S., Staiano A., Volpicelli A., 2003, Neural Networks. Special Issue on Applications of Neural Networks to Astrophysics and Geosciences, R. Tagliaferri, G. Longo, D'Argenio B. eds.
17. Longo G., Tagliaferri R., Sessa S., Ortiz P, Capaccioli M., Ciaramella A., Donalek C., Raiconi G., Staiano A., Volpicelli A., in Astronomical Data Analysis, J.L. Stark and F. Murtagh eds., SPIE n. 4447, p.61
18. Vesanto J., 1997, Ph.D. Thesis, Helsinky University of Technology
19. Nabney I.T., Bishop C.M., 1998, Netlab: Neural Network Matlab Toolbox, Aston University
20. Kohonen T., 1995, Self-Organizing Maps, Springer:Berlin-Heidelberg
21. Stoughton C., Lupton R.H., Bernardi M., Blanton M. R., et al., 2001, AJ, 123, 485

Prediction of Signal Peptide in Proteins with Neural Networks

Piero Fariselli, Giacomo Finocchiaro, and Rita Casadio

Department of Biology/CIRB University of Bologna
via Irnerio 42, 40126 Bologna, Italy
{Piero.Fariselli,Rita.Casadio}@unibo.it
http://www.biocomp.unibo.it

Abstract. In this paper we present a new Neural-Network-based predictor trained and tested on a set of well annotated proteins to tackle the problem of predicting the signal peptide in protein sequences. The method trained on a set of experimentally derived signal peptides from Eukaryotes and Prokaryotes, identifies the presence of the sorting signal and predicts their cleavage sites. The accuracy in cross-validation is comparable with previously presented programs reaching the 97%, 96% and 95% for Gram negative, Gram positive and Eukaryotes, respectively.

1 Introduction

The detection of signal peptides is of great relevance in the field of automatic genomics annotation. Several methods have been developed to predict the presence of the signal peptides and their cleavage sites. Starting from the von Heijne's pioneering paper [1] to the most advanced machine learning approaches, which include Support Vector Machines (SVM) [2], Hidden Markov Models (HMM) [3] and hybrid methods (HMM and Neural Networks) [4]. We implemented two types of neural networks, one for the detection of the cleavage sites (*CleavageNet*) and the other for the N-terminus discrimination (*SignalNet*). Although our architectures are similar to those previously described by other authors [4], our training was carried out using a new data set of experimentally determined signal peptides [5]. Moreover we implement a different way of combining the two networks, reaching the same level of accuracy achieved before [4], without using hidden Markov models. We train three different predictors: for gram negative, for gram positive and for Eukaryotes. In all cases we adopted a four-fold cross validation procedure (taking care of eliminating sequence with detectable homology among training and testing sets). The results for the best performing predictors are reported below, together with the application of the prediction to *Escherichia coli* annotated proteome.

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 237–244, 2003.

2 System and Methods

2.1 Measure of Accuracy

Two measures are used here to score the methods. The accuracy which ranges from 1 (perfect prediction) to 0 (all wrong)

$$Q_2 = \frac{(number \quad of \quad correct \quad predictions)}{whole \quad set} . \tag{1}$$

and the correlation coefficient

$$C = \frac{(tp * tn - fp * fn)}{\sqrt{(tp + fp)(tp + fn)(tn + fp)(tn + fn)}} . \tag{2}$$

where tp, tn, fp, and fn are respectively the true positive, true negative false positive and false negative. C ranges from 1 (perfect prediction) to -1 (inverse prediction) and is 0 for a random assignment.

2.2 The Sequence Data Base

Recently, a reliable data set comprising experimentally annotated proteins which contain (and do not contain) signal peptides has become available [5]. The authors exploited the information present in the SWISSPROT data base to derive this cleaned set [5]. In Table 1 we report the number of sequences for each organism type present in that data base. We also highlight the subsets of the proteins containing the signal peptides (Positive Set) and the complementary set (Negative Set).

Table 1. Number of sequences in the data base

Set Type	Eukaryotes	Gram negatives	Gram positive	All
Positive Set	1158	301	132	1591
Negatve Set	1142	297	129	1568

2.3 Neural Network System

In this paper we use standard feed-forward neural networks with the back-propagation learning algorithm [6]. A thorough search into the parameter space defined six different neural network architectures (which are reported below). All the evaluations were carried out using a four fold cross-validation procedure, taking care of eliminating detectable sequence identities among the corresponding learning and testing sets. This was done using $N \times N$ BLAST searches [7] and applying to the obtained sets the transitive closure algorithm to identify sequence clusters.

We implemented two different types of neural networks, one for the prediction of the signal peptide *SignalNet*, which associates to each residue the probability of belonging to the signal peptide or not, and *CleavageNet* which predicts the position of the cleavage site. Both types have one output neuron, but they differ in the number of hidden and input units. In particular, we allowed asymmetric sliding window, considering that there is more information in the left part than on the right one. The input layers account for the sliding windows and the residue encoding. Each residue is coded by a 21 binary input vector. The first 20 positions of the vector represent the residue types, while the 21-th element codes for the empty positions (it occurs when the sliding window is located at the N-terminus).

Since SignalNet and CleavageNet have one output neuron we need a threshold to classify a given residue in the signal peptide class or not. Then, if $O_S(i)$ is the network output for the position i we have

$$O_S(i) \geq \theta \quad \Rightarrow i \in signal \quad peptide \; . \tag{3}$$

In this paper we set the decision threshold $\theta = 0.5$. Analogously for CleavageNet output O_C we define

$$O_C(i) \geq 0.5 \quad \Rightarrow i \quad is\ a\ cleavage\ site \; . \tag{4}$$

3 Results and Discussion

Signal peptides identify the destination of protein sequences, by promoting the translocation through cell membranes (inner membrane for gram negative). The signal peptides have a central hydrophobic core, a N-terminus part of polar or positive charged residues, and the C-terminus part (close to the cleavage site) polar or negative charged. Although Prokaryotes (gram positive and gram negative) and Eukaryotes signal peptides share this common features, their length distribution and hydrophobicity picks are particular of each class of organisms. We computed with the Kyte-Doolittle [8] hydrophobicity scale the average value (h) for each signal peptide position i as

$$h_{av}(i) = 1/(2W+1) \sum_{j=-W}^{W} h(i+j) \; . \tag{5}$$

where we set $W = 9$. In this way we can identify the location of the maximum of h_{av}. In Table 2 we report the minimal, the maximal, the average length and the average position of the hydrophobic pick (h_{av}) for each organism.

This simple analysis (confirmed also with a more thorough study, data not shown) indicates that a particular care should be used in the prediction of this sorting signal. In particular, we always train and evaluate the performance of our method using the subsets defined by the different types of organisms.

Table 2. Signal peptide lengths in different organisms

Organism	Average Length	Minimal Length	Maximal Length	Hydrophobicity Maximal Position
Eukaryotes	22	7	56	13
Gram Positive	25	11	58	15
Gram Negative	33	21	63	19

3.1 SignalNet and CleavageNet Performance

After a thorough search in the neural network parameter space, we ended up with six best performing different predictors (two networks for each organism type). All neural networks have a hidden layer and one output neuron, and differ for the number of neurons in the input and hidden layers. In order to test the predictor, we examined only the first 65 residue of each protein chain, since if present, the signal peptide falls inside this N-terminal region.

In Table 3 we report the accuracy obtained with the best three network systems for each organism type. As it can be seen SignalNet performs very well when the window is symmetric (left=right). The level of accuracy measured per residue is also very high, reaching 95% for Eukaryotes and 92% and 91% for gram positive and gram negative bacteria, respectively. This values together with those computed for the correlation coefficient (C) show the very high level of precision achieved by our networks.

Table 3. SignalNet accuracy

Organism	Window Length	C	Q2
Eukaryotes	11-1-11	0.75	0.92
”	13-1-13	0.82	0.94
”	15-1-15	0.83	0.95
Gram positive	11-1-11	0.76	0.90
”	13-1-13	0.78	0.91
”	15-1-15	0.79	0.92
Gram negative	9-1-9	0.72	0.88
”	11-1-11	0.78	0.92
”	13-1-13	0.77	0.91

When the task of the exact cleavage site is tackled, the accuracy tends to be apparently higher, due to the fact that the *non-cleavage* positions are far more abundant than the single positions present into the proteins containing the signal peptide. However, this is reflected into a lower probability of exactly locating the cleavage position, which consequently can be visible as a drop of the correlation coefficient. This claim is confirmed from the results reported in Table 4, where we show the accuracy obtained with the best three CleavageNet systems for each organism type. Nevertheless, the values achieved by the correlation coefficient are noteworthy. It is also interesting to note that the best performing architectures in the case of CleavageNet are asymmetric, indicating that in this case the information present into the left part of the window dominates.

Table 4. CleavageNet accuracy

Organism	Window Length	C	Q2
Eukaryotes	15-1-2	0.61	0.97
”	17-1-2	0.58	0.96
”	19-1-2	0.59	0.96
Gram positive	19-1-2	0.55	0.96
”	20-1-3	0.56	0.96
”	21-1-2	0.56	0.95
Gram negative	9-1-2	0.61	0.95
”	11-1-2	0.62	0.96
”	13-1-2	0.61	0.95

3.2 Combining SignalNet with CleavageNet

When a large scale analysis is required, as in the case of genome (proteome) annotation, more than the precise location of the cleavage site (which is of course still relevant) the prediction of the presence or absence of this sorting signal would be extremely useful, both for classification of the protein location and for a further detailed protein sequence processing. In this respect we define a filtering procedure which utilizes the notion that in order to assign a signal peptide we must have strong predictions for a great number of adjacent residues in the first part of the protein sequence. We then introduce the function

$$A_S(i) = \frac{1}{2D+1} \sum_{j=-D}^{D} O_S(i+j) \; . \tag{6}$$

which computes the average value of the network outputs in the symmetric neighborhood $[-D+i, i+D]$, and the function

$$C_S(i) = \frac{1}{i} \sum_{j=1}^{i} O_S(j) \; . \tag{7}$$

that computes the cumulative sum of the i^{th} residues starting from the first position. In order to predict a signal peptide we then require that the position of the maximal score

$$P_S = \max_{i \in [1,65]} \{ A_S(i) \geq S \quad AND \quad C_S(i) \geq L \} \; . \tag{8}$$

must be in a given interval $[P_{min}, P_{max}]$. The values of D, P_{min}, P_{max}, S and L are determined using the training sets. Adopting this new criterion we obtain a significant improvement over the original SignalNet performance, as it is shown in Table 5 where we report the results obtained for the different sets. This is particularly evident considering the correlation coefficient values (compare with Table 3).

Table 5. Signal peptide detection accuracy

Measure	Gram positive	Gram negative	Eukaryotes	All
Q_2	0.97	0.97	0.95	0.95
C	0.93	0.93	0.90	0.91

3.3 Predicting the Cleavage Sites

Even though the accuracy obtained for the CleavageNet is remarkable, it is not easy to sort out the exact position of the cleavage site. For this reason, we used an idea similar to that previously used [4] which takes advantage of both type of neural networks.

We then introduced a function

$$Comb(i) = \sqrt{dS(i)^2 \cdot O_C(i)}\ . \tag{9}$$

where $O_C(i)$ is the usual CleavageNet output for the residue position i and $dS(i)$ is a kind of numerical derivative of the SignalNet. Considering the SignalNet output (O_S)

$$dS(i) = \frac{\sum_{k=1}^{w+a} O_S(i-k) - \sum_{k=0}^{w} O_S(i+k)}{w+a}\ . \tag{10}$$

where w (usually set around 10) and a (around 10) are used to take into account the N-terminal part with respect to the upstream positions.

An example is reported in Fig. 1. It is evident that more than one picks are generated by CleavageNet, and the one centered in $i = 31$ (the expected observation) is not the highest. However, taking into account the values of SignalNet, as describe by 9, *Comb* assigns the correct location.

3.4 Comparison with the Best Performing Method

We compare our system with the best performing one, SignaP [4]. We introduce here two tests, one is based on our dataset [5], and the second is a wide scale test (against the annotated *Escherichia coli* proteome).

In Table 6 we report the results on both tests. The method here introduced (*Our*) is more balanced with respect to SignalP. Even though, the latter performs better on the sets of proteins containing the signal peptides (Positive Set), it

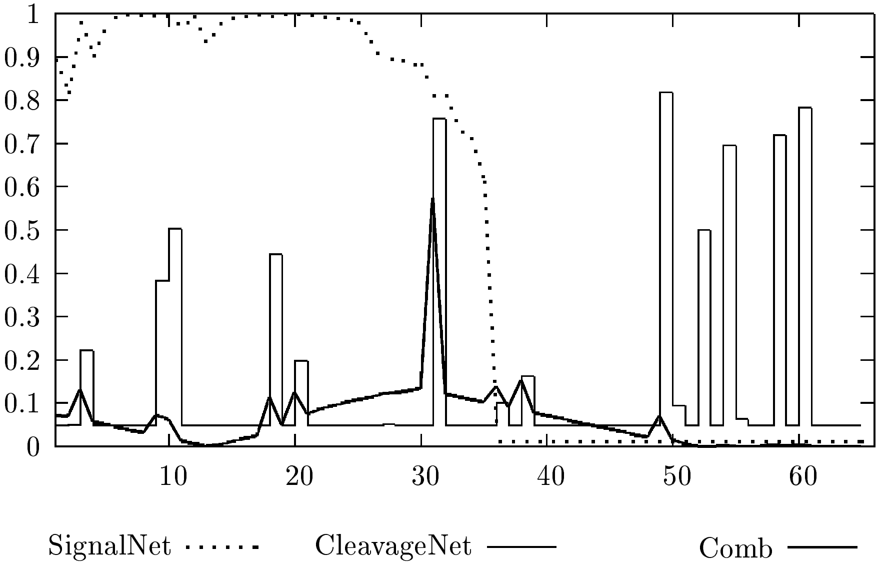


Fig. 1. An example of the prediction of AMO_ECOLI using the SignalNet, CleavageNet and the Comb methods. Y-axis: method-outputs. X-axis: residue position along the chain

Table 6. Q_2 comparison between our method (SPEP) with SignalP [4]

Method	Eukaryotes (Positive Set)	Eukaryotes (Negative Set)	Prokaryotes (Positive Set)	Prokaryotes (Negative Set)	*E. coli* proteome
SignalP	0.99	0.85	0.99	0.93	0.95
Our	0.97	0.94	0.97	0.96	0.96

tends to overpredict, as is pointed out by the Negative set scores. In the case of *E. coli* proteome, where the accuracy is computed summing up both positive and negative examples, our method is comparable with SignalP, indicating that the method presented here is among the best performing described so far.

Acknowledgements

This work was partially supported by a grant of the Ministero della Università e della Ricerca Scientifica e Tecnologica (MURST) for the project Hydrolases from Thermophiles: Structure, Function and Homologous and Heterologous Expression and for the project Development and implementation of algorithms for predicting protein structure , a project on Molecular Genetics and one on Functional Genomics, both of the Italian Centro Nazionale delle Ricerche (CNR), and a PNR 2001-2003 (FIRB art.8) project on Postgenomics, delivered to RC.

References

1. von Heijne, G.: A new method for predicting signal sequence cleavage sites. Nucleic Acids Research **14** (1986) 4683-4690.
2. Vert, J.P.: Support vector machine prediction of signal peptide cleavage site using a new class of kernels for strings. Pac Symp Biocomput. (2002) 649-660.
3. Nielsen, H., and Krogh, A.: Prediction of signal peptides and signal anchors by a hidden Markov model. In: Proceedings of the Sixth International Conference on Intelligent Systems for Molecular Biology, Menlo Park, CA, 1998 AAAI Press (1998) 122-130.
4. Nielsen, H., Brunak, S., von Heijne, G.: Machine learning approaches for the prediction of signal peptides and other protein sorting signals. Protein Eng. **12** (1999) 3-9.
5. Menne, K.M.L., Hermjakob, H., and Apweiler,R.: A comparison of signal sequence prediction methods using a test set of signal peptides. Bioinformatics 16 (2000) 741-742.
6. Baldi,P. Brunak,S.: Bioinformatics: the Machine Learning Approach.
7. Altschul,S.F., Madden,T.L., Schaffer,A.A., Zhang,J., Zhang,Z., Miller,W., Lipman,D.J.: Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. Nucleic Acid Res. **25** (1997) 3389-3402.
8. Kyte, J., Doolittle, R.F.: A simple method for displaying the hydropathic character of a protein. J. Mol. Biol. **157** (1982) 105-132 MIT Press (2001)

Proteomic Profiling of Inherited Breast Cancer: Identification of Molecular Targets for Early Detection, Prognosis and Treatment, and Related Bioinformatics Tools

Giovanni Cuda, Mario Cannataro, Barbara Quaresima, Francesco Baudi,
Rita Casadonte, Maria Concetta Faniello, Pierosandro Tagliaferri,
Pierangelo Veltri, Francesco Costanzo, and Salvatore Venuta

University "Magna Græcia" of Catanzaro, Via T. Campanella 115
88100 Catanzaro, Italy
{cuda,cannataro,quaresima,baudi,casadonte,
faniello,tagliaferri,veltri,fsc,venuta}@unicz.it
http://www.unicz.it

Abstract. Proteomic-based approaches are quickly becoming a powerful and widely used technique to identify specific "molecular signatures" in several pathologic conditions. In particular, cancer, which is one of the most challenging and socially important diseases, is currently under intensive investigation in order to overcome limitations still affecting conventional diagnostic strategies. In particular, one of the major goals in this field is the identification of reliable markers for early diagnosis, as well as for prognosis and treatment. Among cancer, breast carcinoma is the most important malignant disease for western women. A hereditary form has been identified which is related to inherited cancer-predisposing germ-line mutations. Germ-line mutations of BRCA1 gene have been identified in 15-20% of women with a family history of breast cancer and 60-80% with family history of both breast and ovarian cancer. Pathological as well as molecular profiling studies support the concept that inherited breast tumors are different forms of disease, suggesting the intriguing possibility of tailored chemopreventive and therapeutic approaches in this setting. Bioinformatics, and in particular pattern recognition learning algorithms, offer the enabling analysis tools, so the paper also discusses a software environment to conduct such data-intensive computations over Computational Grids.

1 Introduction

Neoplastic diseases represent a major issue, especially in western countries, where the progressive improvement of social, economic and health conditions has markedly lengthnened life expectancy. Breast cancer is among the most prevalent forms of solid tumors. In Italy, the incidence of this particular neoplasia is estimated in 95 new cases each 100.000 women.

Even though the sporadic forms are largely predominant, 5 to 10% of breast cancer is genetically inherited [1]. Among this subset of tumors, those in which a BRCA1-gene predisposing "founder" mutation has been identified show particular interest. The surprisingly high prevalence of such mutations may be the result of historical events

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 245–257, 2003.

leading to the accumulation of an ancient “founder” effect initially occurring in very homogeneous populations. Founder effects are well documented among Icelanders or Askhenazi Jews [2, 3].

Our group has reported the identification of a recurrent mutation in the BRCA1 gene in a patient population with breast cancer from the Calabria region [4]. Such mutation (5083del19), which is highly prevalent in patients undergoing genetic counseling in our institution (72% of the overall BRCA1 gene defects), is found in about 10% of the overall breast tumors occurring in Calabria and is reported 19 times in the Breast Cancer Information Core database (research.nhgri.nih.gov/bic/). The central/western Europe ethnicity described in 5 of these cases, might reflect consistent calabrian migrations to those areas that took place during the late 1800’s and early 1900’s.

Allelotype analysis performed in our study population by using 5 highly polymorphic microsatellite markers has demonstrated a shared haplotype, strengthening the hypothesis of a “founder” effect. Interestingly, the same mutation was detected in four patients, living in the North of Italy and refereed to us by collaborative research groups, as well as in two individuals from United States and Canada. Haplotype analysis confirmed the common origin of the BRCA1 gene defect. Neoplastic malignancies, and breast cancer in particular, are characterized by a high degree of phenotypic complexity.

It is critical, therefore, to focus research strategies not only on the detection of the mutational event(s), but, more importantly, on the identification of new molecular markers, potentially useful in diagnosis and treatment. An mRNA transcript or a protein, whose expressions are significantly modified in tissue specimens or serum from a patient with a genetically-inherited cancer compared to a subject with a sporadic form or to a control individual, should be considered as biological markers.

Last year, Lance Liotta’s group at the National Institutes of Health, USA, using SELDI-TOF mass spectrometry, has identified in the serum of patients with ovarian cancer a cluster pattern that completely segregated cancer from noncancer [5]. These results, characterized by a high degree of sensitivity and specificity, represent an extraordinary step forward in the early detection and diagnosis of ovarian cancer and justify a prospective population-based assessment of proteomic pattern technology as a screening tool for all stages of ovarian cancer in high-risk and general populations. Similar studies performed on different types of neoplastic diseases have confirmed the importance of identification of “molecular profiles or signatures” (either at RNA or protein level) as a powerful tool for innovative diagnostic and therapeutic approaches [6, 7].

We are presently involved in a project whose rationale is to take advantage from the homogeneity of the genetic background of calabrian population to study the proteomic profiling of tissue specimens and serum from breast cancer patients harboring the 5083del19 BRCA1 gene mutation and to compare it with patterns obtained from affected individuals non carriers of this BRCA1 mutation, as well as with protein profiles derived from serum analysis of control subjects. The hypothesis that will be tested is that the founder mutation produces, either at tissue or serum level, the development of a pattern of protein expression which is highly specific for this particular mutation and therefore can be used as a valuable model to be translated to the highly genetically heterogeneous setting of hereditary breast tumors. So, an important task of the project will be the advanced analysis of proteomic data produced by mass spectrometry, as well as finding correlations between these data and other data regarding patients. Recently, pattern recognition learning algorithms (such as data mining, able to learn, adapt and gain experience over time) combined with mass spectrometry profiles of serum, have been successfully used for early detection of cancer. Such new analytical model is named *proteomic pattern diagnostics* [8].

The rest of the paper is organized as follows. Section 2 describes the proposed proteomic techniques for profiling inherited breast cancer. Section 3, after introducing Computational Grids and their use in Bioinformatics, presents the preliminary design and architecture of PROTEUS, a Problem Solving Environment for proteomic analysis on Computational Grids. Finally, Section 4 concludes the paper.

2 Proteomic Profiling of Inherited Breast Cancer

In this Section we discuss an ongoing project recently started at our University. The aim of the project is the prospective collection of biological samples (tumor, normal breast specimens and serum) from affected 5083del19 BRCA1-gene mutation carriers, from affected individuals non harboring this particular mutation, as well as from healthy control subjects.

The general strategy for sample collection will be based on the organizative framework that is presently acting within the Calabrian referral center for genetic counselling at the Magna Græcia University (www.oncologia.unicz.it), specifically devoted to identification and management of the hereditary malignancies in Calabria.

Moreover, additional samples from the 5083del19 BRCA1 gene mutation carriers will be made available through the italian network established on the basis of ongoing AIRC and MIUR grants. A web network (www.hereditarycancer.it), based in our institution, is currently operating and will be of major help during the project.

Proteomics is a fastly developing area of biochemical investigation [9]. The basic aim of proteomic analysis is the identification of specific protein patterns from cells, tissues and biological fluids related to physiological or pathological conditions. It provides a different view as compared to gene expression profiling, which does not evaluate post-transcriptional, post-translational modifications as well as protein compartimentalization and half-life changes (for instance ubiquitination and proteosome-driven degradation). All these characteristics make the protein profile much more complex but more informative compared to gene expression profiling.

Several approaches have been used to perform proteomic analysis; among them, the most common are 2D-gel electrophoresis and mass spectrometry (MS)-based methods (MALDI-TOF, SELDI-TOF). The main difference between the traditional protein sequencing approach and the more innovative proteomic technologies is the tremendous amount of information made available by the possibility of multiple protein identification by computer-assisted comparative analysis with large databases commonly available on the Web (NCBI or SWISS-PROT/TrEMBL).

A recent development in protein separation and identification of protein profiles has been provided by the isotope-coded affinity tag-labeling procedure (ICAT), followed by liquid chromatography coupled to tandem mass spectrometry (LC/MS/MS). It involves the site-specific labeling of proteins with isotopically normal or heavy ICAT reagents, proteolysis of the combined, labeled protein mixture, followed by the isolation and mass spectrometric analysis of the labeled peptides [10]. The high-throughput proteomic technology requires sophisticated management and analysis of the output data. Specifically devoted computer programs allow protein identification and/or identification of protein patterns related to a particular pathophysiological condition. These data can be produced either by spontaneoulsy generated profiles (unsupervised), which depict a molecular pathology framework, or even generated on the basis of a selected outcome (supervised approach).

In our project, samples obtained from BRCA1 founder mutation carriers will be compared with analogous samples from BRCA1 founder mutation-negative breast cancer individuals and from healthy control subjects. Moreover, the identified protein patterns will also be compared with protein patterns obtained from the BRCA1-defective HCC1937 experimental cell line, where we have been able to demonstrate a specific drug sensitivity profile which is modified by the reconstituted BRCA1 expression following transfection of the wild type gene (manuscript submitted for publication). The research program will be subdivided in the following tasks:

- Task 1. Selection, sampling and storage of biological materials.
- Task 2. Proteomic analysis of biological materials and single protein or multiple protein pattern identification.
- Task 3. Proteomic analysis of BRCA1-deficient and reconstituted HCC1937 cell line.
- Task 4. Data integration and identification of potential diagnostic/therapeutic fallout.

Task 1. Selection, Sampling and Storage of Biological Materials. During this phase, we plan to collect biological samples from breast cancer patients with or without the founder mutation in BRCA1 gene. On the basis of the epidemiological data of breast cancer incidence in Calabria (640 expected new cases/year), the founder mutation-related tumor population should be close to 60-70 cases/year. Our goal is to reach an accrual of at least 25-30% of the yearly estimated cases. The flow of biological material and clinical data will follow the current network leaded by the Calabrian referral center for genetic counselling at the Magna Græcia University which has produced so far a bank of 15 samples. Mutation carriers non selected by genetic counselling, will be offered to be informed about mutation status and eventually enrolled in our genetic counselling program. All counselling procedures are approved by the Ethical Committee of our institution.

Task 2. Proteomic Analysis of Biological Materials and Single Protein or Multiple Protein Pattern Identification.

Study design:

- Tissue: Identification of tumor-specific protein patterns (derived by comparative analysis of neoplastic vs normal specimens) from BRCA1 gene founder mutation carriers vs. non BRCA1-related cancer patients (see Figure 1).

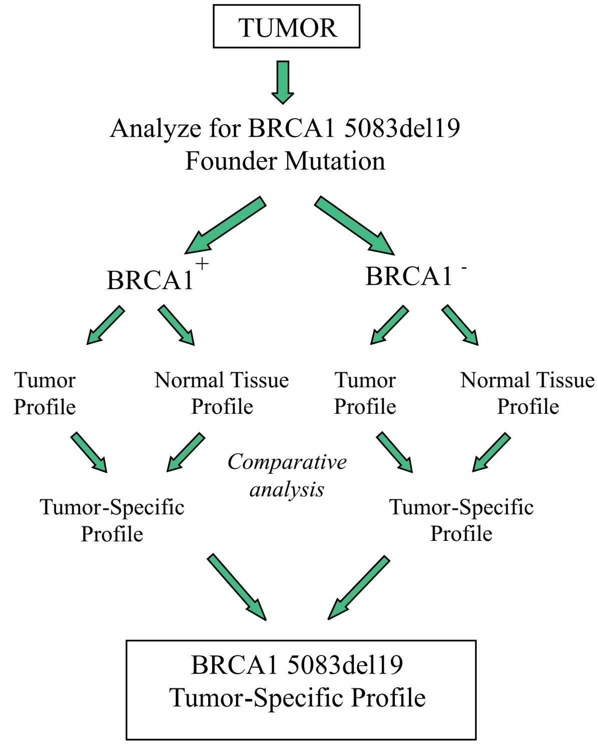


Fig. 1. Identification of tumor-specific protein patterns in tissue.

- Serum: Identification of tumor-specific protein patterns from BRCA1 gene founder mutation carriers vs non BRCA1-related cancer patients vs control individuals (see Figure 2).

Proteomics on tissue specimens:

a. Tissue collection at the University Hospital and at the city Hospitals throughout the regions involved in the network leaded by the Calabrian referral center for genetic counselling at the Magna Græcia University.
b. Histological diagnosis, confirmed by two independent pathologists before use in this study.
c. Tissue processing:
   1. Preparation of uncoated glass slides with 8 μm sections of normal and neoplastic patient-matched breast tissue.
   2. Isolation of breast tissue subtypes by Laser Capture Microdissection.
   3. Sample prefractionation (subproteomics) in order to focus on differential cellular compartments (membranes and nucleus in particular).
   4. Protein separation by two-dimensional SDS polyacrylamide gel electrophoresis (SDS-PAGE).

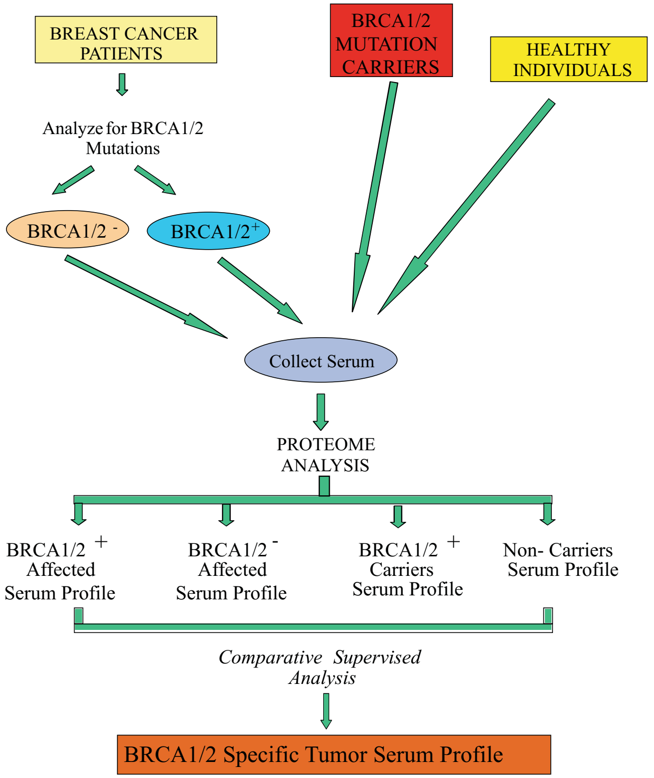

BREAST CANCER PATIENTS
BRCA1/2 MUTATION CARRIERS
HEALTHY INDIVIDUALS
Analyze for BRCA1/2 Mutations
BRCA1/2 -
BRCA1/2+
Collect Serum
PROTEOME ANALYSIS
BRCA1/2 + Affected Serum Profile
BRCA1/2 - Affected Serum Profile
BRCA1/2 + Carriers Serum Profile
Non- Carriers Serum Profile
Comparative Supervised Analysis
BRCA1/2 Specific Tumor Serum Profile

Fig. 2. Identification of tumor-specific protein patterns in serum.

5. Over or under-expressed protein estimation by isotope-coded affinity tag-labeling procedure (ICAT).
6. Analysis of peptide mixtures by reverse phase liquid chromatography coupled to tandem mass spectrometry (LC/MS/MS).
7. Identification of the peptide(s) by computer-assisted comparative analysis with large databases commonly available on the Web (NCBI or SWISS-PROT/TrEMBL).

Proteomics on serum:

a. Case and Control Serum specimen collection.
b. Albumin and Globulin depletion.
c. Over or under-expressed protein estimation by isotope-coded affinity tag-labeling procedure (ICAT) with/without preventive prefractionation via 1D-gel electrophoresis.
d. Analysis of peptide mixtures by reverse phase liquid chromatography coupled to tandem mass spectrometry (LC/MS/MS).

e. Automatic Protein Identification and Quantitation of the peptide(s) by leveraging access to large databases commonly available on the Web (NCBI or SWISS-PROT/TrEMBL).

Re-run by tandem mass spectrometry for assessing the over and the under-expressed proteins with low Codon Bias.

Task 3. Proteomic Analysis of BRCA1-Deficient and Reconstituted HCC1937 Cell Line. During this phase, proteomic analysis will be performed on cultured human breast cancer cell line HCC1937 which completely lacks BRCA1 function and on the HCC1937/ $^{WT}BRCA1$ derivative, where BRCA1 function has been reconstituted by transfection of a full length wild type cDNA coding for the BRCA1 gene. The aim of this work is the identification of proteins differentially expressed in the same cell line in the presence or absence of BRCA1 cell function, taking in account their potential role in determining the differences in the chemosensitivity profile displaced by the two cell lines.

Task 4. Data Integration and Identification of Potential Diagnostic/Therapeutic Fallout. The final task will be the integration of data derived from tumor analysis and the data coming from tumor cell lines in order to identify differentially expressed proteins and their potential relevance to the phenotypic features displayed by the tumors in the different conditions. Data from serum profiling will be evaluated again comparing breast cancer 5083del19 BRCA1-gene mutation carriers vs. breast cancer non mutation carriers as well as vs. healthy control subjects in the attempt to identify a serum signature of tumor arising in a well defined hereditary setting. These finding will be eventually used as a tool to develop innovative strategies for early diagnosis and treatment of breast cancer. Moreover, we plan to implement clustering techniques based on data mining algorithms. Mass spectrometry raw data is first collected from cancer and healthy patients, then after having organized them in a suitable format for data mining algorithm, we plan to find representative clusters for, respectively, healthy and non healthy patients. The final task will be the classification of incoming test data to find if a new patient belongs to a cancer cluster or to a healthy one.

3 PROTEUS: A Problem Solving Environment for Proteomic on the Grid

According to D. Walker, [11] “A Problem Solving Environment (PSE in short) is a complete, integrated computing environment for composing, compiling, and running applications in a specific area. A PSE may also incorporate many features of an expert system and can assists users in formulating problems, running the problem on an appropriate platform, and viewing and analyzing results. In addition a PSE may have access to virtual libraries, knowledge repositories, sophisticated execution control systems, and visualization environments.”

The main motivations to build a PSE is to provide software tools and assistance to scientists in a user-friendly environment, allowing rapid prototyping of ideas and higher productivity, leaving the user free to concentrate on application and not on software programming [12]. Basic components of PSEs are:

- A set of components or modules (software and data sources).
- A uniform access system (module interface).
- A (application) composing interface (visual, APIs, ...).
- An execution manager.

To deal with data produced in the first phases of proteomic analysis in breast cancer, we plan to apply our experience on parallel and distributed data mining on Grids [13] to design and implement PROTEUS, a PSE for the design and execution of Bioinformatics (in particular Proteomics) applications on the Grid.

To design PROTEUS we will apply some ideas developed in the KNOWLEDGE GRID [13], a PSE to design and execute distributed data mining applications on the Grid. In the following, after introducing Computational Grids and reviewing their use in Bioinformatics, we discuss the main requirements and architecture of PROTEUS.

3.1 Computational Grids

Computational Grids are geographically distributed environments for high performance computation that manage a large number of systems offering them as a unified system accessible to users via a single interface [14]. Grid middleware offers services for managing large numbers of diverse computational resources administered by independent organizations, and provides application developers with a simplified view of the resulting computational environment. Grids provide the computational infrastructure for powerful new tools for scientific investigation and virtual organization operations, including desktop supercomputing, smart instruments, collaborative environments, and distributed supercomputing. From the Bioinformatics point of view, Computational Grids can offer both the basic services to access and use distributed computational resources, and to form virtual organizations working on specific tasks [15].

During the development and use of Grids, people realized that access to distributed data is typically as important as access to distributed computational resources. Thus in the rest of the Section Data Grids and Knowledge Grids are briefly described.

Data Grids. Distributed scientific and engineering applications typically require access to large amounts of data; moreover, Grid applications also require widely distributed access to data from many places by many people (for example, as in virtual collaborative environments). In many cases, application requirements call for the ability to read large datasets (e.g. protein databases) and to create new datasets (e.g. mass spectrometry proteomic data). They can require the ability to change (updating) existing datasets; consequently, the so-called Data Grids have been developed. A Data Grid is a distributed infrastructure that allows to store large local datasets, has locally-stored replicas of datasets from remote locations, and accesses remote datasets that are not replicated locally. In order to support these features, Data Grid middleware provides both a means of managing different types of datasets and a number of basic mechanisms that generalize the requirements of most data intensive applications. The main goal is to offer a generic infrastructure in the form of core data transfer services and generic data management libraries. Three main important research projects aiming at the development of Data Grids are the Globus Data Grid [16], the European Data Grid [17], and the Particle Physics Data Grid.

Data Grids could be usefully applied for:

- the efficient sharing of biological databases,
- the effective building and updating of large-scale virtual biological databases, obtained aggregating and integrating local databases,
- the realization of large scale experiments involving both datasets and databases owned by different organizations.

Knowledge Grids. Knowledge Grids offer high-level tools and techniques for the distributed mining and extraction of knowledge from data repositories available on the Grid. The KNOWLEDGE GRID [13] is a PSE to design and execute Distributed Data Mining applications on the Grid. The KNOWLEDGE GRID architecture uses basic Grid services, such as authentication, data sharing, execution management, to build specific knowledge extraction services. The current implementation is based on the Globus Toolkit, the *de facto* standard for Grid middleware [18]. The KNOWLEDGE GRID services allow a user to design distributed data mining applications as composition of pre-existing data mining software installed on the nodes of a Grid, by using a visual programming environment that allows to search, locate and access remote software and data sources.

3.2 Bioinformatics Grids

In this Section we report some projects developed on Grids whose aim is to deal with both bioinformatics data and tools: we refer to these projects as BioGrids. The availability of the Grid as a promising enabling infrastructure supporting the collaboration of people and resources and the consciousness in the Bioinformatics community that scientific processes need, other than computing support, also a lot of semantic modelling, quality of resources being used and full support of cooperation between scientists, are the main motivations for the following BioGrid projects.

Many of these projects concentrate on the sharing of computational resources and the efficient, large-scale data movement and replication, in different Bioinformatics fields, such as high throughput sequence analysis, simulations and access to remote instrumentations. Indeed, the myGrid project attempts to introduce semantic modelling of Bioinformatics processes. Our project PROTEUS goes along this direction, as described in the rest of the Section. In the following, some BioGrids are briefly described.

myGrid. myGrid is a large UK eScience pilot project (http://mygrid.man.ac.uk) targeted at developing open source high-level Grid middleware to support data-intensive bioinformatics on the Grid, in particular personalized “in silico” experiments in biology. In myGrid the emphasis is on data integration, workflow, personalization and provenance. In particular, database integration is obtained both by dynamic distributed query processing, or creating virtual databases through federations of local databases. Introducing workflow and knowledge management in bioinformatics means to formalize the bioinformaticians knowledge in such a way it is usable in a machine based form, and to control data flows between tools to automate and make more reliable the bioinformatics processes. In fact, usually Bioinformatics experts know which tools to use, their input and output data formats, how to invoke them, how to move data between tools and in which order to use them. To this end, bioinformatics services and resources are modelled and classified through ontologies [19]. The main goal of myGrid is to offer to application providers an environment for the development of distributed bioinformatics application for the community of biologists. Some early prototypes of myGrid services for the functional analysis of clusters of proteins have been developed.

EUROGRID Bio-GRID. The EUROGRID (Application Testbed for European GRID Computing) project (http://www.eurogrid.org/), funded by the European Community, aims to demonstrate the use of Grids in selected scientific and industrial communities. In particular, the Bio-GRID work package (http://biogrid.icm.edu.pl/) is developing an access portal for biomolecular modelling resources [20]. The project will develop various interfaces to biomolecular applications and databases that will allow chemists and biologists to submit work to high performance computing facilities, hiding Grid programming details.

North Carolina BioGrid. The North Carolina Bioinformatics Grid (NC BioGrid) is an initiative of the North Carolina Biotechnology Center to promote genomics, proteomics, and bioinformatics research in North Carolina (http://www.ncbiogrid.org/). The NC BioGrid, that will connect the members of the established North Carolina Genomics and Bioinformatics Consortium, will accumulate and make available to researcher and educators the enormous library of genomics and proteomics data being created throughout the world, combined with non-proprietary data from Consortium members. The NC BioGrid will concentrate on the sharing of software for mining, analyzing, and modelling data, in genomics and proteomics.

Asia Pacific BioGRID. The Asia Pacific BioGRID (APBioGrid) is an initiative of the Asia Pacific Bioinformatics Network (http://www.apbionet.org/apbiogrid/). The APBioGrid is attempting to build a customized, self-installing version of the Globus Toolkit comprising well tested installation scripts for a defined set of OS platforms, and automated configuration scripts with BioGRID specifics, avoiding to deal with Globus details. Various bioinformatics and biological applications suitable for Grid will be or are currently deployed, among these EMBOSS, EMBASSY, PHYLIB, etc.

Canadian BioGrid. The Canadian BioGrid (Canadian Bioinformatics Resource, CBR) (http://cbr-rbc.nrc-cnrc.gc.ca/) is dedicated to providing the Canadian research community with convenient, effective access to widely used bioinformatics tools and databases. Among these, it offers access to the following software: BLAST, Clustal, Decypher, DNA Fold, EMBOSS, GeneMatcher, MAGPIE, Mascot, Primer3, ReadSeq, RNA Fold, SeqsBlast, SRS, WebPhylib, Cogeme, ExPASy, and to more than 80 databases.

Biomedical Informatics Research Network. The Biomedical Informatics Research Network (BIRN) (http://www.nbirn.net/) is an NCRR (National Center for Research Resources) initiative that will create a testbed to enable the biomedical access and analysis of biomedical data located at diverse sites throughout the country. The BIRN testbed will provide the hardware and software necessary to build a scalable network of biomedical databases and high performance resources. Issues of user authentication, data integrity, security, and data ownership will also be addressed. In addition to scalability, the produced software will be reusable and extensible, so the testbed, that will initially be applied to neuroimaging research, will be deployed to different research centers working on different research topics.

3.3 Requirements and Architecture of PROTEUS

To help scientists in Proteomic/Bioinformatics research, PROTEUS will embed Bioinformatics knowledge about:

- bioinformatics/proteomics processes (involved resources, workflow, user interaction, etc.),
- biological databases (description of Protein Data Banks, etc.),
- bioinformatics tools and software (EMBOSS, BLAST, etc.).

Proteus will be based on the use of Open Source Bioinformatics software, like EMBOSS, and the integration of public-available biological databases. Private databases (i.e. databases accessible with registration via Web) will also be considered.

EMBOSS (European Molecular Biology Open Software Suite) is a package of high-quality Open Source software for sequence analysis (http://www.emboss.org/). The EMBOSS suite provides a comprehensive set programs (approximately 100), covering: Sequence alignment, Rapid database searching with sequence patterns, Protein motif identification, including domain analysis, Nucleotide sequence pattern analysis, Codon usage analysis for small genomes, Rapid identification of sequence patterns in large scale sequence sets, Presentation tools for publication. EMBOSS provides a set of core software libraries (AJAX and NUCLEUS) and supports all common Unix platforms including Linux, Digital Unix, Irix, Tru64Unix and Solaris.

The main drawback in using EMBOSS or similar packages is that users need to know both the biological domain and the semantic and details of available software components. Moreover, the access to such components is often available by command line only. To overcome these problems, PROTEUS will leverage existing software tools simplifying their use by modelling applications through ontology and adding metadata to available software. Thus PROTEUS will offer pre-packaged bioinformatics applications in different fields (e.g. proteomics) using the computational power of Grids.

The access to resources will be simplified by using a two layer metadata schema: at the top layer an ontology will be used to model the rationale of bioinformatics applications, whereas at the bottom layer specific metadata about available (i.e. installed) bioinformatics software and data sources will be defined. The top layer ontology will guide the user in the choice of the available software tools or complete applications on the basis of her/his requirements (ontology-based application design) [21], whereas the low layer metadata will be used to really access software tools and databases, providing information like installed version, format of input and output data, parameters, constraints on execution, etc. The ontology will be updated whenever new software tools or data sources are added to the system, or new applications are developed (i.e. designed through composition of software components). This will enable the realization of a Knowledge Base of applications/results, that is enriched whenever new applications are developed or new results are obtained. Thus, new users will be able to gain knowledge about pre-existing experiments.

Currently we are designing the PROTEUS architecture extending the KNOWLEDGE GRID approach. Main components of Proteus are:

- bioinformatics applications ontology and components metadata (i.e. software, databases and data sources),
- visual ontology-based application design (concept-based search of basic components, as well as entire applications)
- workflow-based Grid execution manager and scheduler.

Proteomics will be the first investigated domain and we plan to embed more open source software, in useful fields such as data mining, neural networks and genetic algorithms. Moreover, we will describe data sources produced by commercially available equipments such as Mass Spectrograph, to allow the advanced analysis of such data.

4 Conclusions and Future Work

The impact of proteome analysis in cancer research, and specifically in the field of inherited tumors, is rapidly increasing thanks to the development and improvement of the instrumentations and bioinformatic tools currently available. High performance-computers, pattern recognition learning algorithms, and mass spectrometry platforms are able to provide an enormous amount of data and informations on a high throughput basis. The challenge now is to take advantage from all these data and use them to identify proteomic profiles that might be useful as molecular markers for early detection, prognosis and tailored-treatment of cancer. Together with genomics, proteomics research and the bioinformatic structure supporting both approaches are going to make a tremendous change in the understanding of the basis of tumorigenesis and, hopefully, in the expectancy and quality of life of many individuals suffering from cancer.

References

1. Nathanson KL, Wooster R, Weber BL, Nathanson KN. Breast cancer genetics: what we know and what we need. Nat Med 7: 552-556, 2001.
2. Johannesdottir G, Gudmundsson J, Bergthorsson JT, Arason A, Agnarsson BA, Eiriksdottir G, Johannsson. High prevalence of the 999del5 mutation in icelandic breast and ovarian cancer patients. Cancer Res 56: 3663-5, 1996.
3. Warner E, Foulkes W, Goodwin P, Meschino W, Blondal J, Paterson C, Ozcelik H, Goss P, Allingham-Hawkins D, Hamel N, Di Prospero L, Contiga V, Serruya C, Klein M, Moslehi R, Honeyford J, Liede A, Glendon G, Brunet JS, Narod S. Prevalence and penetrance of BRCA1 and BRCA2 gene mutations in unselected Ashkenazi Jewish women with breast cancer. J Natl Cancer Inst 91:1241-7

4. Baudi F, Quaresima B, Grandinetti C, Cuda G, Faniello C, Tassone P, Barbieri V, Bisegna R, Ricevuto E, Conforti S, Viel A, Marchetti P, Ficorella C, Radice P, Costanzo F, Venuta S. Evidence of a founder mutation of BRCA1 in a highly homogeneous population from southern Italy with breast/ovarian cancer. Hum Mutat. 2001 Aug;18(2):163-4
5. Petricoin EF, Ardekani AM, Hitt BA, Levine PJ, Fusaro VA, Steinberg SM, Mills GB, Simone C, Fishman DA, Kohn EC, Liotta LA. Use of proteomic patterns in serum to identify ovarian cancer. Lancet. 2002 Feb 16;359(9306):572-7
6. Hedenfalk I, Duggan D, Chen Y, Radmacher M, Bittner M, Simon R, Meltzer P, Gusterson B, Esteller M, Kallioniemi OP, Wilfond B, Borg A, Trent J. Gene-expression profiles in hereditary breast cancer. N Engl J Med. 2001 Feb 22;344(8):539-48
7. Wulfkuhle JD, Sgroi DC, Krutzsch H, McLean K, McGarvey K, Knowlton M, Chen S, Shu H, Sahin A, Kurek R, Wallwiener D, Merino MJ, Petricoin EF 3rd, Zhao Y, Steeg PS. Proteomics of human breast ductal carcinoma in situ. Cancer Res. 2002 Nov 15;62(22):6740-9.
8. Wulfkuhle JD, Liotta LA, Petricoin EF. Proteomic applications for the early detection of cancer. Nature Review Vol 3, 267 - 275 (April 2003).
9. Pandey A, Mann M. Proteomics to study genes and genomes. Nature 405, 837 - 846 (15 Jun 2000).
10. Smolka MB, Zhou H, Purkayastha S, Aebersold R. Optimization of the isotope-coded affinity tag-labeling procedure for quantitative proteome analysis. Anal Biochem 2001 Oct 1;297(1):25-31)
11. D. Walker, O. F. Rana, M. Li, M. S. Shields, and Y. Huang, The Software Architecture of a Distributed Problem-Solving Environment. Concurrency: Practice and Experience, Vol. 12, No. 15, pages 1455-1480, December 2000.
12. Gallopoulos, S.; Houstis, E. N.; Rice, J. Computer as Thinker/Doer: Problem-Solving Environments for Computational Science. In *IEEE Computational Science and Engineering*.
13. Cannataro, M.; Talia, D.; The Knowledge Grid. *CACM*, 46(1):89–93.
14. Foster, I.; Kesselman, C. Globus: a Metacomputing Infrastructure Toolkit. *Intl. J. Supercomputer Applications*, 11:115–128.
15. Foster, I.; Kesselman, C.; Tuecke, S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. *Intl. J. Supercomputer Applications*, 15(3).
16. Chervenak, A.; Foster, I.; Kesselman, C.; Salisbury, C.; Tuecke,S. The Data Grid: Towards an Architecture for the Distributed Management and Analysis of Large Scientific Datasets. *J. of Network and Computer Applications*, 23:187–200.
17. W. Hoschek, J. Jaen-Martinez, A. Samar, H. Stockinger, K. Stockinger. Data Management in an International Data Grid Project. Proc. IEEE/ACM Int. Workshop on Grid Computing (Grid 2000). LNCS vol. 1971, Springer Verlag, pp. 77-90, December 2000.
18. The Globus Project. http://www.globus.org/.
19. Chris Wroe, Robert Stevens, Carole Goble, Angus Roberts, Mark Greenwood. A suite of DAML+OIL Ontologies to Describe Bioinformatics Web Services and Data. International Journal of Cooperative Information Systems special issue on Bioinformatics, March 2003. ISSN: 0218-8430 (2003)
20. J. Pytlinski, L. Skorwider, P. Bala, M. Nazaruk and K. Wawruch BioGRID - uniform platform for biomolecular applications Euro-Par 2002. Parallel Processing (Lecture Notes in Computer Science 2400) Eds. B. Monien, R. Feldman Springer-Verlag 2002 pp. 881-884
21. Cannataro, M.; Comito, C. A Data Mining Ontology for Grid Programming. *Proceedings 1st Int. Workshop on Semantics in Peer-to-Peer and Grid Computing (SemPGrid2003)*, 113-134, www.isi.edu/ stefan/SemPGRID.

Browsing Large Pedigrees to Study of the Isolated Populations in the "Parco Nazionale del Cilento e Vallo di Diano"

Giuliano Antoniol[1], Michele Ceccarelli[1], Fabio Rollo[1],
Wanda Longo[2], Teresa Nutile[2], Marina Ciullo[2],
Enza Colonna[2], Antonietta Calabria[2], Maria Astore[2], Anna Lembo[2],
Paola Toriello[2], and M. Grazia Persico[2]

[1] RCOST - Research Centre on Software Technology
University of Sannio, Department of Engineering
Palazzo ex Poste, Via Traiano, I-82100 Benevento, Italy
[2] Institute of Genetic and Biophysics, "A. Buzzati Traverso", CNR, Napoli

Abstract. The paper reports a flow analysis framework for data exploration, knowledge discovery and visualization of large-scale Pedigrees represented as directed graphs. Indeed, when large Pedigrees are involved in biological studies researchers need to interact with multiple tools such as databases (storing genetic as well as phenotype information), graph browsers, graph visualization tools, etc.
We have already collected the last three centuries genealogical data of the population in the villages of Campora and Gioi, situated on the hills and mountains of the National Park of Cilento and Vallo of Diano, an area in the Southern Italy. At the present the villages have a population of 600 and 1200 residents respectively. The size of the today population as well as the collected genealogy requires sophisticated Software methods to support the storage, the handling, the analysis and the visualization of the data. In particular, visualization may become an impossible task when large Pedigrees need to be represented and browsed: very often the result is a screen cluttered by to much information.
The amount of collected information requires reliable and powerful software tools. The paper describes the key elements we are organizing into a Software system allowing to analyze, manage and visualize the large Pedigree. In particular, we report the structure of a database, the main framework inspired by flow analysis to extract properties from the pedigrees and a preliminary visualization tool based on GraphWiz the ATT graph visualization environment which allows to use applet to display graphs into WEB browser.

Keywords: Large Pedigrees, databases, flow analysis

1 Introduction

Complex diseases like cardiovascular diseases, tumors, neurodegenerative disorders and diabetes are the more frequent cause of morbidity and mortality in the

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 258–268, 2003.

industrialized countries. One approach useful to identify loci and genes responsible for complex diseases takes advantage of the populations of small villages that have been geographically isolated for centuries. We looked at the villages situated on the hills and mountains of the National Park of Cilento and Vallo of Diano, an area in the Southern Italy. We chose twelve of these villages on the basis of the degree of endogamy (marriages between people born in the same village) in the last two centuries (higher than 80%).

We have already collected the last three centuries genealogical data of the population in the villages of Campora and Gioi, which at the present have a population of 600 and 1200 residents respectively, in order to build their genealogy. The size of the today population as well as the collected genealogy requires sophisticated software methods to support the storage, the handling, the analysis and the visualization of the data. In particular, visualization may become an impossible task when large pedigrees need to be represented and browsed: very often the result is a screen cluttered by to much information.

For example, data collected on the village of Campora include: the genealogy obtained from the demographical data; the genetic structure of the population; the characterization of the "state of health" of the Campora's inhabitants obtained by clinical analysis performed from blood samples, medical interview and examination. At the time of writing the pedigree contains more than 7000 individuals, the analysis for about 500 people (each analysis characterizes 45 features). All in all the software has to be able to handle a forest of graphs with a number of nodes ranging from dozen to thousands. Each node represents an individual and thus the individual status (e.g., dead or alive), measured features (e.g., blood cholesterol level) and marker values has to be graphically associated and accessed. To store the data we have defined the structure of a relational database that can be accessed by network; an XML representation decouples the database structure from the representation layers. To support pedigree and more generally information browsing we are developing a portable program inspired by the paradigm of the fisheye, a paradigm often adopted in software engineering to browse large graphs. Fisheye has been implemented as a variable number of visualized generations around the current highlighted individual. The researcher is left in charge to define the number of forward and backward generation thus allowing to finely control the number of displayed individuals and the depth of the visualized sub-pedigree.

This paper presents the overall database structure, the XML interface, the preliminary visualization software we are developing to support pedigree browsing and an easily customizable framework to analyze and query large pedigree. Within this framework the extraction of properties of the graph and of individual nodes is modeled as a flow analysis problem, each node implementing a input-output mapping depending of a composition rule and four associated ets. The flow information is propagated through the graph until convergence. We show how to query the structure of the graph by customizing the composition rules and the associated sets. The algorithm has a quadratic worst-case behavior, however we have observed linear increasing of execution times in nearly all real world experimental conditions. This means that the number of iterations for the convergence of the graph relaxation procedure weakly depend on the number of nodes.

The paper is organized as follows. First we will briefly summarize the structure of the database then a framework inspired to flow analysis allowing to extract information and sub-pedigrees from a larger one. Finally, we will present some preliminary pedigree visualization snapshots.

2 The Database

The collected information were stored in a relational database, its scheme is depicted in Fig. 1, Database tables models entities of the application domain, they can be divided into two broad categories: tables describing the essential information related to the population and tables describing the measurement process or more generally information of the genotype and phenotype collection process. The following tables describing the population were identified:

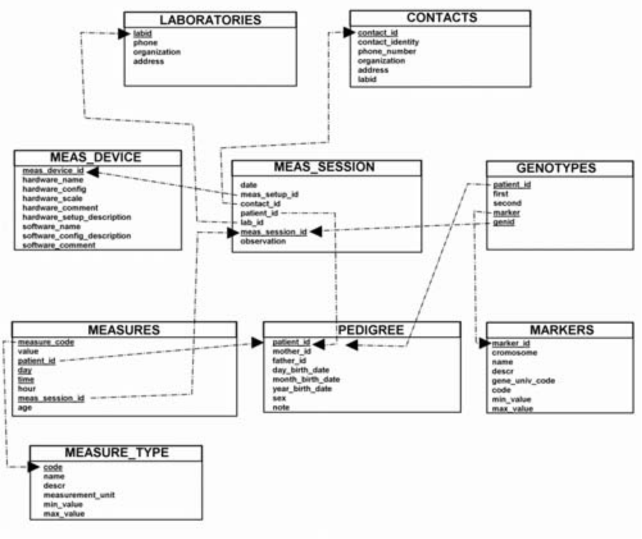


Fig. 1. An excerpt of UML database schema.

- *pedigree*: to store the population pedigree as well as birth-date, sex and notes and other patient related information;
- *measure*: containing the values of analyzes (e.g., body mass index, lymphocyte concentration, etc);
- *genotypes*: storing alleles values for the genotyped markers;

The above tables are complemented by two tables encoding the marker and laboratory analysis information respectively. Other tables store the data collection process details allowing to represent essential feature such as the calibration of the analysis devices or the day when the blood sample was collected. More precisely the following tables were populated:

- *contacts*: to store information on the nurse collecting the blood samples, on the laboratory of analysis, etc;
- *meas_device*: specifying the brand, model and setup of various devices;
- *meas_sessions*: to store temporal information and other relevant facts on each measurement session (i.e., blood sample collection, patient interview, etc);
- *laboratories*: to identify the different laboratories, hospitals or structures involved in the process of handling and analyzing collected samples.

The data base schema was represented via XML Schema to promote software and tools interoperability. An excerpt of the XML Schema is shown in Fig. 2, where the central table storing the pedigree information is summarized. Details were omitted from Fig. 2 due to space problem.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<xsd:element name="Pedrigree" >
 <xsd:element name="sex">
  <xsd:simpleType>
    <xsd:restriction base="xsd:string">
      <xsd:pattern value="M|F"/>
    </xsd:restriction>
  </xsd:simpleType>
 </xsd:element>
 <xsd:element name="year">
  <xsd:simpleType>
   <xsd:restriction base="xs:integer">
    <xsd:pattern value="[0-2][0-9][0-9][0-9]"/>
   </xsd:restriction>
  </xsd:simpleType>
 </xsd:element>
  ...
 <xsd:complexType>
  <xsd:sequence>
   <xsd:element name="Persona">
    <xsd:complexType>
      <xsd:element name="PatientID" type="xsd:integer"/>
      <xsd:element name="MotherID" type="xsd:integer"/>
      <xsd:element name="FatherID" type="xsd:integer"/>
      <xsd:element name="Day_Birth_Date" type="xsd:day"/>
      <xsd:element name="Month_Birth_Date" type="xsd:month"/>
      <xsd:element name="Year_Birth_Date" type="xsd:year"/>
      <xsd:element name="Sex" type="xsd:sex"/>
      <xsd:element name="Note" type="xsd:string"/>
    </xsd:complexType>
   </xsd:element>
  </xsd:sequence>
 </xsd:complexType>
</xsd:element>
</xsd:schema>
```

Fig. 2. XML Schema excerpt

Foreign keys and triggers were defined between tables to ensure data integrity. To speed up complex queries computation indexes on a subset of the tables attributes were created.

It is worth mentioning that the information we store in the database does not cover all the project details. For example, the standard operating procedure (e.g., the details on the PCR-SSCP analysis or the biochemistry laboratory tests) are stored in specific documents and thus they were not modeled by the database.

Finally, when selecting the database engine, the characteristics we privileged were reliability, transaction support, support for trigger, foreign keys and nested queries. PostgreSQL was selected as the project database management system. PostgreSQL is the worlds most advanced Open Source database software and it includes all the required characteristics.

3 The Flow Analysis Framework

Flow analysis techniques have historically been important for gathering information about certain properties of graphs and programs represented as directed graphs [4,15,13,6,2,10,11,16].

Given a population, the relation between individuals, ancestors and descendants can be thought of and modeled via a directed graph and thus flow analysis may be applied to collect relevant facts on the populations or to extract subgraphs with a given property (e.g., all the ancestors of an individuals).

Normally, the information is gathered for each point in the Directed Graph (DG). Examples of traditional flow analysis include finding dominators (i.e., those individuals on all path from an ancestor to a given individual). Other analysis such as live variables are more tied to software engineering and do not have a correspondent interpretation in terms when analyzing the DG representing a population (live variables are those variables that might be used after each point in the program).

Flow analysis techniques are extremely appealing in that there are several problems related to extract sub-population or pedigree visualization that can be easily and concisely described in term of flow equations.

A general theory and detailed application of flow analysis is beyond the scope of the present paper, the interested reader may refer to the *Dragon Book* [1], on the newest book [3] or in more specific conference and journal publication [12, 14,18,17,8,9]. Basic notion on flow analysis are summarized in the following in the context of a forward flow analysis (i.e., flow information is propagated from the ancestors to the leaves of the graph).

Let $G = (N, E)$ be the DG representing the pedigree P of the population, where

- The node set N is the set of all individuals in the pedigree P.
- The edge set E is the set of pairs of individuals $(n_i, n_j) \in N \times N$ if n_i is a parent of n_j.
- Assume that there is a special node: a *start* - v_1 corresponding to a common unknown founder of the population.

Led D be the *universal* set for the flow information, each node $n \in N$ has the following associated sets:

- $IN_n \subset D$: representing the incoming flow information;
- $OUT_n \subset D$: representing the outgoing flow information;
- $GEN_n \subset D$: representing the node *generated* flow information;
- $KILL_n \subset D$: representing the information *killed* by the node.

Furthermore, each node may be associated with information specific to the modeled problem, such as the gender information, whether or not the individual is still alive, etc.

Flow analysis is a general framework that can be specialized to solve particular problem settings by customizing the following features:

- The kind of flow information that is propagated.
- The rules describing how each node transforms its input flow information into output flow information.
- The direction for the flow propagation.
- The rules describing how flow information is merged at the joint points.
- The initial values.

Flow information is forward propagated in the control flow graph according to equations:

$$IN_n = \bigoplus_{p \in pred(n)} OUT_p$$

$$OUT_n = GEN_n \bigcup (IN_n - KILL_n)$$

where *pred*(n) is the set of adjacent nodes to n in the graph, and $\bigoplus$ is the *confluence* or *merge* operator, describing how to merge flow information at the joint points, and depending on the analysis being performed. The graph is traversed and flow information propagated until convergence according to the forward flow propagation algorithm shown in Fig. 3;

Since there are N nodes in the graph, and each iteration, in the worst case, modifies a single set with a single piece of information, in at most N iterations convergence will be achieved. Thus the algorithm has a worst case complexity $O(N^2)$. However, most of the problems and flow analysis specializations exhibit a linear or almost linear complexity.

3.1 Specializing Flow Analysis

The main advantage of the framework relies in the fact that particular queries over the pedigree can be casted as a flow analysis problem. The user simply needs to write a $\bigoplus$ merge operator and the code to initialize (or generate) the flow analysis sets (i.e., IN, OUT, GEN and $KILL$). Very often those sets are initialized with the empty set, with the universal set for flow information (e.g., the set N of the graph nodes) or with the label identifying the node. The merge

```
for each $n \in N$ do initialize $IN_n$ and $OUT_n$
$change \leftarrow$ true
while $change$ do begin
        $change \leftarrow$ false
        for each $n \in N$ do begin
                $IN_n \leftarrow \bigoplus_{p \in pred(n)} OUT_p$
                $OLDOUT_n \leftarrow OUT_n$
                $OUT_n \leftarrow GEN_n \bigcup (IN_n - KILL_n)$
                if $OUT_n \neq OLDOUT_n$ then
                        $change \leftarrow$ true
        end
end
```

Fig. 3. Flow propagation algorithm.

operator is almost always implemented as a set operation: union, intersection, set difference, min or max.

Suppose we are interested of identifying, for each patient in the pedigree, the set of ancestors. Let v_1 be the common founder, ancestors may be easily computed by setting:

- Universal flow information $D = N$.
- Transformation rules:
  $GEN_n = \{n\}$ $KILL_n = \{\}$
- Flow information is propagated *forward.*
- Confluence operator $\bigoplus$ is the union.
- Initial values:
  $\forall n \in N, n \neq v_1, IN_n = OUT_n = \{\}$
  $IN_{v_1} = \{\}, OUT_{v_1} = \{v_1\}$

After convergence, the IN sets contain the ancestors while the OUT sets contain the ancestors plus the current node. Notice that to compute descendants we only need to reverse the graph. To extract the sub-pedigree, with a specified depth (ie., number of subsequent generation) say d, of a given node, say m, the following equations may be applied:

- Universal flow information $D = N$.
- Transformation rules:
  $GEN_n = \{\}$ $KILL_n = \{\}$
- Flow information is propagated *forward.*
- Confluence operator $\bigoplus$ is $[min_{p \in Pred(n)} IN_n] - 1$ if the value is greater or equal to zero, zero otherwise.
- Initial values:
  $\forall n \in N, n \neq m, IN_n = OUT_n = \{\}$
  $GEN_m = \{d\}$

Descendants are those nodes with a positive value in the input set. By reversing the graph, predecessors at a given depth may be computed. Fish eye visualization may be implemented based on these analysis thus allowing to focus on a specified sub-pedigree. Finally, suppose we need to extract the sub-graphs encompassing individuals with a specified value of a property P; let $< n,m >$ be an edge connecting the parent n to the child m:

- Universal flow information $D = E$.
- Transformation rules:
  $GEN_n = \{< p, n >\}$ if predecessor p of n has property P; $KILL_n = \{\}$ if n has property P, D otherwise.
- Flow information is propagated *forward.*
- Confluence operator $\bigoplus$ is union;
- Initial values:
  $\forall n \in N$, $n \neq v_1$, $IN_n = OUT_n = \{\}$

3.2 Implementation Issues

We have implemented the flow analysis in Perl; the development was based on the `Graph` package available at CPA, the Perl comprehensive archive. File input was implemented by sub classing the `Graph::Reader` to match out format. Much in the same way, the output was obtained via sub classing the `Graph::Writer` component. Perl was chosen since it is available on all hardware and software architecture, furthermore, Perl components are easily wrapped to obtain distribute system via WEB services.

In addition, the key element to implement a flow analysis is an efficient set implementation, which is not a trivial task. Compromises may be necessary between time and space complexities; for example, a bit vector implementation, hash tables or sorted arrays are reasonable choices ensuring on average efficient insertion, element search, intersection, and union operations. Details on time complexity and alternative implementation can be found in [1, 5]. Meanwhile, space complexity problems may arise from a naive set implementation. Suppose we are modeling a population with 10000 individuals; a parsimonious, but naive, bit vector implementation may require up to 4 Kbytes per node, with an overall of about 40 Mbytes. A non efficient hash table or array implementation may easily blow up 400 Mbytes. To overcame the problem at least two solutions are available. The first is based on the idea of sharing a pool of sets between graph nodes. By tying sets a space reduction is obtained at the price of decreasing time performance. However, since operation involving the universal set for flow information are trivial, another approach is to specialize the empty and universal flow information set implementations. For example, the union with an empty set or the intersection with the universal flow information set will not alter any sets.

4 Visualization

The visualization tool is aimed at supporting the task of browsing through the collected information. According to Tores and Barillot [19], it is difficult to find a program that draws pedigree diagrams perfectly. However, since a pedigree is just a directed graph (drawn according to a particular convention) it is foreseeable to exploit directed graph layout programs to visualize pedigrees.

GraphWiz, the ATT environment to create, visualize and browse graphs implements an efficient algorithm to compute the layout of directed graphs [7] via the GraphWiz `dot` component.

Pedigree and individual related information are extracted via query over the data base and filtered via Perl scripts based on the described flow information framework. Further processing is required to format a pedigree in a way that GraphWiz can be used to draw it. We implemented the transformation proposed by Limsoon Wong (see the author WEB site http://sdmc.lit.org.sg:8080/ Elimsoon). The idea is simple and effective: by adding extra nodes to the directed graph a suitable layout is easily obtained. Special attention needs to be devoted to the cases of multiple marriages of the same individual or to marriages between consanguineous which may causes edge crossing, cluttering the resulting diagram. Visualizing the family tree originated by a given founder, causes a further problem. Individuals that do not descend from that founder but married to one of his/her descendants do not belong to the sub-pedigree. However, they must be added to complete the diagram. This is of course a common situation and we implemented a suitable heuristic to allow drawing a nice pedigree (see Fig 4) in presence of such a common phenomenon. The strategy to augment a pedigree with the required extra nodes and edges to exploit GraphWiz was implement in a Perl script `mkpedigree`. `mkpedigree` allows to reuse the GraphWiz environment including the `grappa` Java system without modifying `dot`.

At the time of writing, a preliminary release with limited capability was available. DG are saved as Dotty files (the tool can be downloaded from the ATT WEB site http://www.research.att.com/sw/tools/graphviz/). Graphs are transformed via the *mkpedigree* Perl script and the layout is determined by `dot`. Visualization is accomplished via Java applet (developed with the `grappa` jar) that allows to access pedigree information by means of any WEB browser.

The Java applet allows to interact, zooming in out or interacting with the database.

5 Conclusion

We reported a flow analysis framework for data exploration, knowledge discovery and visualization of large-scale pedigrees represented as directed graphs. The reference database contains more than 7000 individuals, the analysis for about 500 people (each analysis characterizes 45 features). All in all the software has to be able to handle a forest of graphs with a number of nodes ranging from dozen to thousands. Each node representing an individual and thus the individual status, measured features and marker values has to be graphically associated and accessed. We have presented the overall database structure, the XML interface, the preliminary visualization software we are developing to support pedigree browsing and an easily customizable framework to analyze and query large pedigree.

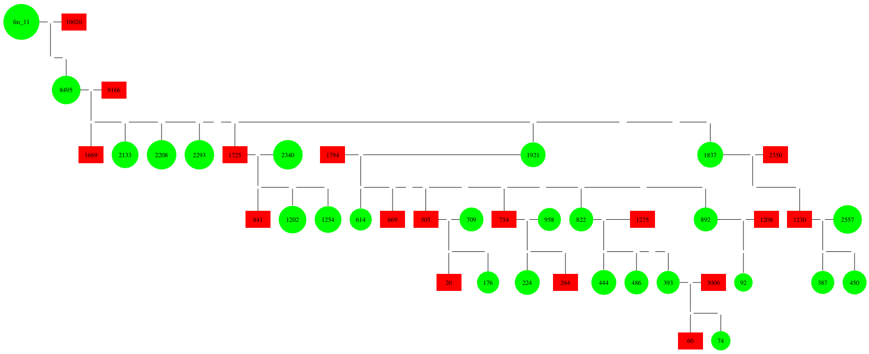


Fig. 4. Example of sub-pedigree visualization in Dotty.

Within this framework the extraction of properties of the graph and of individual nodes is modeled as a flow analysis problem, each node implementing a input-output mapping depending of a composition rule and four standard initialization sets. The flow information is propagated through the graph until convergence. We showed how to query the structure of the graph by customizing the composition rules and the initializer. The algorithms is quadratic, nevertheless we have observed linear increasing of execution times in nearly all real world experimental conditions. This means that the number of iterations for the convergence of the graph relaxation procedure weakly depend on the number of nodes.

References

1. A. V. Aho, R. Sethi, and J. D. Ullman. *Compilers. Principles Techniques and Tools.* Addison-Wesley Publishing Company, Reading, MA, 1985.
2. A. V. Aho, R. Sethi, and J. D. Ullman. *Compilers: principles, techniques, tools.* Addison-Wesley, 1986.
3. A. W. Appel. *Modern Compiler Implementation in Java.* Cambrige University Press, 1998.
4. D. C. Atkinson and W. G. Griswold. Implementation techniques for efficient data-flow analysis of large programs. *to appear in ICSM 2001.*
5. T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introductions to Algorithms.* MIT Press, 1990.
6. E. Duesterwald, R. Gupta, and M. L. Soffa. A practical framework for demand-driven interprocedural data flow analysis. *ACM Transactions on Programming Languages and Systems*, 19(6):992–1030, 1997.
7. E. Ganser et al. A technique for drawing directed graphs. *IEEE Transactions on Software Engineering*, 19(3):214–230, 1993.
8. K. B. Gallagher and J. R. Lyle. Using program slicing for software maintenance. *IEEE Transactions on Software Engineering*, 17(8):751–761, 1991.
9. S. Horwitz, T. Reps, and D. Binkley. Interprocedural slicing using dependence graphs. *ACM Transactions on Programming Languages and Systems*, 12(1):26–60, 1990.

10. J. B. Kam and Jeffrey D. Ullman. Global data flow analysis and iterative algorithms. *Journal of the ACM*, 23(1):158–171, January 1976.
11. G. A. Kildall. A unified approach to global program optimization. In *Conference Record of the ACM Symposium on Principles of Programming Languages*, pages 194–206. ACM SIGACT and SIGPLAN, ACM Press, 1973.
12. R. Komondoor and S. Horwitz. Using slicing to identify duplication in source code. *Proc. of the 8th International Symposium on Static Analysis*, 2001.
13. T. Reps. Program analysis via graph reachability. *Information and Software Technology*, 40(11-12):701–726, 1998.
14. S. Sinha, M. J. Harrold, and G. Rothermel. System-dependence-graph-based slicing of programs with arbitrary interprocedural control flow. *Proceedings of the 21st International Conference on Software Engineering*, pages 432–441, 1999.
15. S. Sinha, M.J. Harrold, and G. Rothermel. Computation of interprocedural control dependencies. *ACM Transactions on Software Engineering and Methodology*, 10(2):209–254, 2001.
16. K. Stirewalt and L. Dillon. A component-based approach to building formal analysis tools. *Proc. of IEEE International Conference on Software Engineering*, 2001.
17. F. Tip. A survey of program slicing techniques. *Journal of Programming Languages*, 3(3):121–189, 1995.
18. P. Tonella, G. Antoniol, and E. Merlo. Flow insensitive c++ pointers and polymorphism analysis and its applications to slicing. *Proceedings of International Conference on Software Engineering (ICSE)*, pages 90–99, 1997.
19. F. Tores and E. Barillot. Optimizing pedigree drawing using interval graph theory. *Currents in Computational Molecular Biology*, pages 194–195, April 2000.

Monitoring of Car Driving Awareness from Biosignals*

Bruno Apolloni, Simone Bassis, Andrea Brega, Sabrina Gaito,
Dario Malchiodi, Norberto Valcamonica, and Anna Maria Zanaboni

Dipartimento di Scienze dell'Informazione, Università degli Studi di Milano
Via Comelico 39/41, 20135 Milano, Italy
{Apolloni,Bassis,Brega,Gaito,Malchiodi,Valcamonica,Zanaboni}@dsi.unimi.it

Abstract. We infer symbolic rules for deciding the awareness state of a driver on the basis of physiological signals traced on his body through non invasive techniques. We use a standard device for collecting signals and a three-level procedure for: 1) extracting features from them, 2) computing Boolean independent components of the features acting as propositional variables, and 3) inferring Boolean normal forms on these variables deciding the driver awareness. In spite of their symbolic form, these formulas are not easily interpretable, rather they represent a sort of Boolean wavelets for describing the driver emotional state. A set of experiments are shown on a benchmark expressly drawn from a car driver simulator by the Psychology Department of Queen University of Belfast.

1 Introduction

New generation man-machine interfaces are required to be transparent, in the spirit of disappearing/pervasive computing trend. Thus interactions on the part of man will be ruled not by logic commands expressly minded by him, rather by physiological signals automatically produced by his body. We challenged this commitment in the special case of noticing to the machine an unawareness or awareness state of a human called to do time to time an attention requesting task. Namely, we monitor the attention state of a subject playing a driving simulator in two different conditions: normal condition and accident avoidance. We work on a quadruplet of tracks of physiological data like in Figure 1(a), where the first track reports the electrochardiographic signals (ECG), the second the respiration (RSP), the third the galvanic skin response (GSR) an the fourth the skin temperature (SKT). The output of our procedure is a log like the thick line in Figure 1(b) denoting the occurrence of excited attention episodes in correspondence to the peaks. The curves of the physical parameters of the simulated dynamics put in definite connection the highest peaks with the occurrence of accident avoidance requests along the driving story. Due to the complexity and noisiness of the collected signals on one side and the difficulty of the inference target on

* Work partially funded by the E.C. contract No. IST-2000-26091, "ORESTEIA: mOdular hybRid artEfactS wiTh adaptivE functIonAlity".

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 269–277, 2003.

the other, the above curve required the intervention of a long procedure and sophisticated techniques. Actually, in order to maintain some understandability of the awareness detection rules, or at least its formal manageability within wider contexts, we required these rules to be symbolic, being expressed, in particular, as either conjunctive or disjunctive normal forms. On the other side we manage features and their massive interactions in an essentially subsymbolic way, relying on some cost functions to both extracting their independent components (with a further requirements of being Boolean) and assessing the above normal forms, mainly intended as basis functions of a sort of Boolean wavelet representation. We resume the main points of this procedure in Section 2. A discussion of early numerical results and forewords are delivered in Section 3.

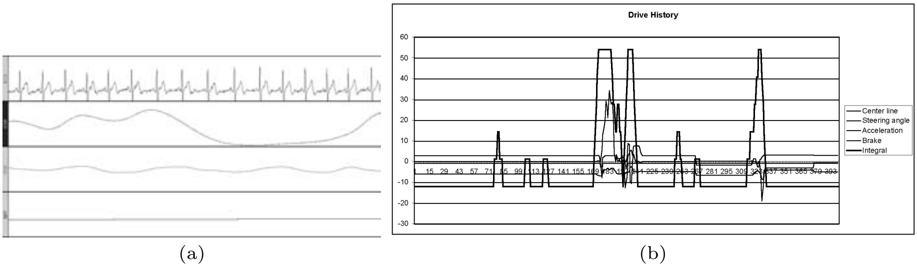


Fig. 1. Input and output of the procedure: (a) a specimen of the four signals used; (b) the awareness monitoring track. (see Figure 7 for cars identification.)

2 A Procedure from Sensory Data to Rules

We may figure the whole procedure hosted by a hybrid multilayer perceptron as in Figure 2 where first layers compute mainly subsymbolic functions like conventional neural networks, while the latter are arrays of Boolean gates computing formulas of increasing complexity. From an informational perspective the signal flows left to right evolving through representations increasingly compressed where the information is distributed in both the states of the processing elements and their connections. The way of training this machinery is reported elsewhere [1]; here we will rather discuss the local targets pursued by the learning algorithm and the features of the trained machinery.

2.1 Data

A trial is constituted by a road trip that must be followed by the driver sitting at a car driving simulator with different boundary conditions. This simulator, which is in Queen University of Belfast [2], is equipped with a Biopac device [3] having some non invasive sensors to record the signals mentioned in the introduction at the sampling rate of 200 Hz. We sharply distinguish between unsoliciting conditions, by default, and soliciting conditions represented by the sudden road

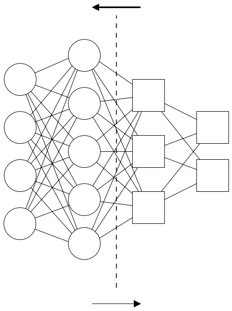


Fig. 2. Balancing the data explanation in an extended neural network. Circles and squares denote respectively neural and symbolic units. Arrow: data flow direction

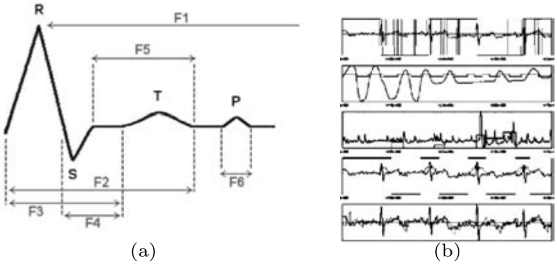


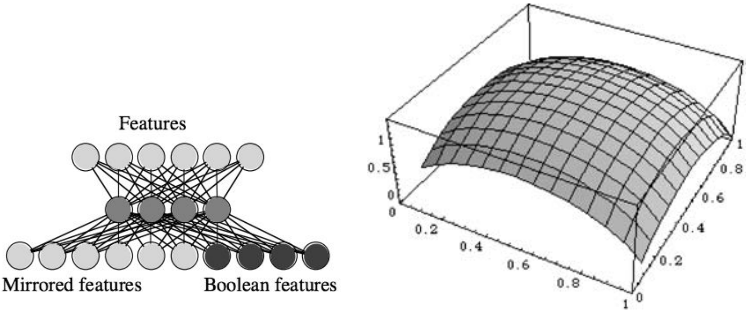
Fig. 3. A sketch of the extracted features: (a) ECG parameters; (b) coded signals

crossing by a pedestrian and periodically fast direction changes induced by cones in front of the car.

The main cinematic parameters of the car, such as speed, acceleration, and wheel angle, are recorded during the trial.

2.2 Feature Extraction

Since SKT is a constant due to simulation shortness, we extract ten features just from first three signals, that we avoided to filter as this operation generally destroys relevant informations. We also didn't manage for removing motion artifacts since the features we will consider are insensitive to them. Eight features are conventional, consisting in distances between relevant points of the signals according to medical knowledge [4] (as sketched in Figure 3) , while the other two are power spectrum and neural prediction shift. The power spectrum captures global properties of the ECG pulse and is computed within every pulse through usual Fast Fourier Transform processing. Finally a “neural” distance is computed between the actual ECG trajectory and the one 10 steps ahead forecasted by a specially featured recurrent neural network. This network is fed by (last line in Figure 3(b)) the current values of the three physiological signals and is embedded with both conventional sigmoidally activated neurons and symbolic



(a) (b)

Fig. 4. The Boolean ICA extractor.
a. the hourglass neural network; b. a two arguments edge pulling function

ones expressly computing second and fifth order dynamics as baselines of the RSP and ECG signals respectively.

2.3 Boolean Independent Components Extraction

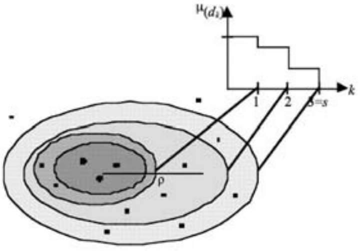
We give a neural network the task of producing symbols from the set of features with the key commitment of getting a *syntactic mirroring* of these data. We look for a vector of Boolean variables whose assignments reflect (possibly through a distorting mirror) the relevant features of the original (possibly continuous) data. This means that an almost “one to one” correspondence must exist between the two parts, where Boolean assignments may coincide when they code data patterns with the same values of the features of interest to us. In greater detail, we split the syntactic mirroring in two parts: a true mirroring of the data pattern and a projection of a compressed representation of the pattern (obtained as a side effect of the first part) into the space of Boolean assignments. Mirroring is pursued through a RAAM (Recursive Auto Associative Memory) architecture [5]: a three-layer network with the same number of units in both input and output layers and a smaller number of units in the hidden layer. Therefore the hidden layer constitutes a bottleneck which collects in the state of its nodes a compressed representation of the input. Our output layer is augmented and partitioned into two parts (see Figure 4(a)). Part B outputs a copy of the input and is trained via a usual back-propagation algorithm according to a quadratic error function. Part A computes the Boolean assignments that will be used by the subsequent symbolic layers. To this aim we require the network minimazing the following error function:

$$E_p = \log\left(\prod_{j=1}^{n} z_{p_j}^{-z_{p_j}}(1 - z_{p_j})^{-(1-z_{p_j})}\right) \tag{1}$$

where z_{p_j} denotes the j^{th} among the n Boolean outputs when pattern p is presented as input.

```
Begin
    DNF=$\emptyset$;
    for each positive example $\mathbf{u}$
        $\mathsf{m} = \bigcap_{i|u_i=1} x_i$
        DNF = DNF $\cup\{\mathsf{m}\}$
    end for
    return DNF;
End
```

(a)



(b)

Fig. 5. (a) Algorithm for building an inner border; (b) The fuzzy border of a monomial. Dark region → m; lessening gray regions → progressive enlargements after removal of d_1, d_2 and d_3 from set(m); $\mu_{\mathsf{m_d}}(d_k)$ as in (3); ρ as in 4

This function, which we call the *edge pulling function*, has the shape of an entropy measure and was selected because it is minimized for output values close to the limits of the activation space (see Figure 4(b)). Via the Jensen inequality [6] on $g(x) = x \ln x$ it is easy to prove that the normalized sum of E_p over the patterns is majorized by the empirical entropy $\widehat{H}$ of the joint distribution of the propositional variables Z_j, when they are supposed to be identically Bernoulli distributed and independent. Namely, denoting r the training set size:

$$\begin{aligned}\breve{H} &\equiv \frac{1}{r}\sum_p E_p \leq \sum_j \left[-\sum_p \frac{z_{p_j}}{r} \ln\Big(\sum_p \frac{z_{p_j}}{r}\Big) - \Big(1 - \sum_p \frac{z_{p_j}}{r}\Big) \ln\Big(1 - \sum_p \frac{z_{p_j}}{r}\Big)\right] \\ &= \widehat{H} \end{aligned} \tag{2}$$

Thus, the left part of this inequality has a minimum for each z_{p_j} Boolean. The right part has a minimum too when the assignments are pushed towards independent Z_j's still preserving the original data entropy. Now, the sole entropy we are interested on lies on a consistent partition of positive and negative coded patterns that we expressly check during the $\breve{H}$ descent. Moreover, for sparse $\mathbf{z}$'s $g(x)$ behaves almost linearly. Thus the cost function is close to promoting the extraction of Boolean independent components from the biodata.

2.4 Learning Boolean Wavelets

Given a set Σ of examples in the Boolean hypercube of the Boolean assignments $\mathbf{u}$ to the vector $\mathbf{x}$ of propositional variables, we simply compute a *minimal* DNF (union of monotone monomials m) D including all *positive points* (the examples coming from an attention requiring episode) through the algorithm in Figure 5(a), where minimality denotes that no other DNF with same property exists included in D.

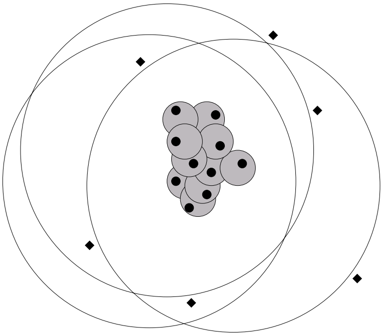


Fig. 6. Inner and outer borders, in the sample space, of a concept at two abstraction levels. Inner borders are delimited by the union of formulas bounded by positive examples, outer borders by the intersection of formulas bounded by negative examples. Bullets: positive examples; diamonds: negative examples

We managed the simplification of the formulas as an optimization task of their membership functions. In synthesis, we want to balance a desirable shortening of a formula with the undesirable loss of its description power [7] (in terms of negative points included and positive ones excluded). Starting from a crisp monomial m at a given abstraction level, we fuzzy enlarge its contour by annexing boundary regions of the formula. Namely, we denote by the sequence $\mathbf{d} = (d_1, \ldots, d_s)$ the ordered (in any way) set, set(m), of literals in m, by $\mathbf{d}^k$ its prefix of length k and by $\mathsf{m}_{\mathbf{d}^k}$ the monomial obtained by removing these literals from m (with $\mathsf{m}_{\mathbf{d}^0} = \mathsf{m}$). By definition, the support of (i.e. the set of assignments $\mathbf{u}$ satisfying) any $\mathsf{m}_{\mathbf{d}^k}$ includes the m's. Denoting with $\sigma(\mathbf{d}^k)$ the cardinality of the Σ subset belonging to $\mathsf{m}_{\mathbf{d}^k} \backslash \mathsf{m}$, we rely on a membership function $\mu_{\mathsf{m}_\mathbf{d}}(d_k)$ of a literal d_k in respect to $\mathsf{m}_\mathbf{d}$ as follows:

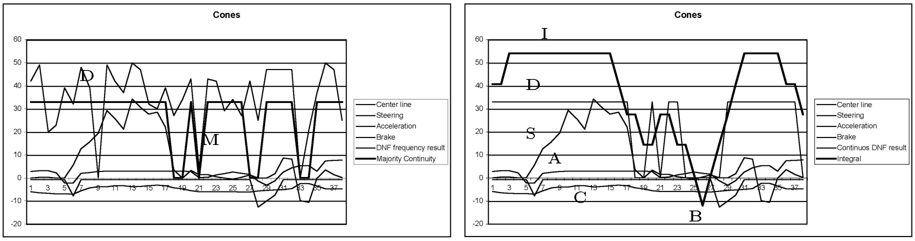
$$\mu_{\mathsf{m}_\mathbf{d}}(d_k) = 1 - \frac{\sigma(\mathbf{d}^k)}{\sigma(\mathbf{d})} \tag{3}$$

The monotonicity of the membership function, referred to assignments as points in the unitary hypercube, induces a dummy metrics where points annexed by one literal in $\mathbf{d}$ are more far from the crisp monomial than the ones annexed by previous literals (see Figure 5(b)). In this framework we can define the radius of the fuzzy contour as the mean value of the distances of points belonging to each enlargement slice from the crisp monomial m

$$\rho_i = \sum_{k=1}^{s} \mu_{\mathsf{m}_\mathbf{d}}(d_k) \tag{4}$$

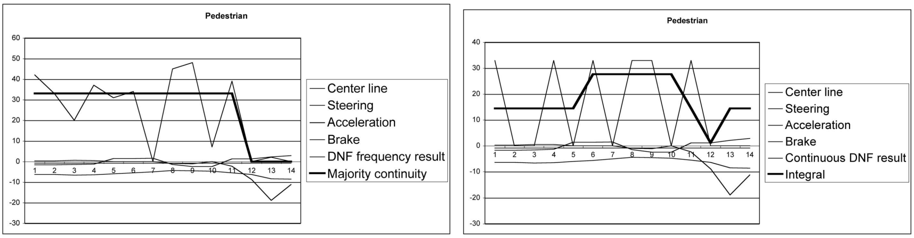
With the above notation, we want to minimize the following cost function with respect to a DNF f consisting of ℓ monomials:

$$O(f, \boldsymbol{\lambda}) = \lambda_1 \sum_{i=1}^{\ell} L_i + \lambda_2 \sum_{i=1}^{\ell} \rho_i + \lambda_3 \nu_0 \tag{5}$$



(a) (b)

Fig. 7. Classification in case of cones



(a) (b)

Fig. 8. Classification in case of pedestrian

where L_i is the number of literals in the i-th monomial, ν_0 is the percentage of positive examples left out from the support of f, and the free parameter $\boldsymbol{\lambda}$ guarantees a proper balance between the cost components.

A complementary procedure may be pursued in respect to the negative points (not attention requiring situations) to achieve a *maximal* CNF (intersection of monotone clauses). Minimal and maximal forms are consistent with the whole training set. They represent an inner and outer borders between inside the function describing the attention excited state may be found. If we preserve the consistency during the simplification procedure, we may view at these borders as a pair of symbolic wavelet representations of the goal function. In fact, as any function of $L^2(\mathbb{R})$ can be represented as a linear combination of wavelets [8], any function in the hypercube may be written as a sum of monomials (and similar for clauses). Monomials may be viewed as symbolic wavelets since they are a basis for DNF representation, localized in space as wavelets are in $\mathbb{R}$. The property of localization in the frequency domain, typical of wavelets, reverses in the small number (actually one) of points binding the simplification of a monomial or a clause [9]. The above fuzzy relaxation procedure looks as the companion of the classical thresholding methods adopted in wavelet analysis to disregard useless details.

3 Numerical Results and Forewords

Figures 7 and 8 summarize a numerical experiment on the biodata log of one subject called to i) skipping cones on the road (the former) and avoiding a pedestrian (the latter). Part (a) of the figures synthesizes the results of 50 learning experiments performed with a cross validation technique. Starting from a sequence of 870 examples, one for each driver heart pulse, 57 of which refer to the two events (thus being labeled positive), we took 50 different random partitions of it into a training and a test set, by halving the positive points and annexing in the training set only 20% of the negative points in order to balance their effects. On each training set the procedure of Section 2 computes a DNF and a CNF discriminating positive from negative points. While the truth table (Table 1) denotes acceptable but not brilliant classification results, obtained with relatively short formulas in any case, the figures do greater justice to the method. We trace the performance by two curves. Majority line reports on each example the label (0 or 1) computed by the majority of the 50 rules. We smooth this curve with some contiguity rules, excluding for instance changes of label of one example surrounded by more than 4 examples with the complementary label. Frequency line reports exactly the number of the rules outputting label 1. Taking into account that each point figures in the training set of around half of the rules, the two curves agree in recognizing in Figure 7 attention excited points at beginning of the hard steering angle variation and of the slalom around cones, while the other points are relatively relaxing the driver attention. Similarly the driver spend the most of his attention right before starting the manoeuvre for avoiding running over the pedestrian.

Part (b) of the figures reports the classification labels of a single rule. The curve is smoothed by averaging these values within a window of 5 points and rescaling the values for giving them a better readability. We see that even a single rule is able to correctly classify the examples.

Table 1. Performance of 50 trials cross-validation test. FP: false positives, FN: false negatives

	DNF			CNF		
	length	FP	FN	length	FN	FP
AVG	44.42	22.91	27.62	71.06	22.41	28.22
STD DEV	7.35	5.36	8.79	20.37	7.49	5.44

These results are definitely preliminary. They base on poor features and a limited bench of data. A wider experimental campaign is starting focusing on really attention demanding episodes coupled with a variety of ancillary conditions in order to both stress the method and realize the extensibility of the found formulas to *similar* episodes and/or subjects. A key point will be the realization of the effect of distracting conditions such as hearing musics or answering a mobile.

References

1. Apolloni, B., Malchiodi, D., Orovas, C., Palmas, G.: From synapses to rules. Cognitive Systems Research **3** (2002) 167–201

2. Balomenos, T., Cowie, R.: Oresteia deliverable nd1.5: Data description, sampling methods, experiments design (2003) `http://www.image.ntua.gr/oresteia`.
3. Biopac website (2003) `http://www.biopac.com`.
4. Plonsey, R., Fleming, D.G.: Bioelectric phenomena. McGraw-Hill Series in Bioengineering. McGraw-Hill (1969)
5. Pollack, J.: Recursive distributed representations. Artificial Intelligence **1-2** (1990) 77–105
6. Wilks, S.S.: Mathematical Statistics. Wiley Publications in Statistics. John Wiley, New York (1962)
7. Vapnik, V.: The Nature of Statistical Learning Theory. Springer, New York (1995)
8. Mallat, S.: A wavelet tour for signal processing. Academic Press (1999)
9. Apolloni, B., Baraghini, F., Palmas, G.: Pac meditation on boolean formulas. In Koenig, S., Holte, R.C., eds.: Abstraction, Reformulation and Approximation. Springer, Berlin (2002) 274–281

Markov Analysis of Protein Sequence Similarities

Chakra Chennubhotla[1] and Alberto Paccanaro[2]

[1] Department of Computer Science, University of Toronto, Canada
chakra@cs.toronto.edu
[2] Bioinformatics Unit, Queen Mary University of London, UK
a.paccanaro@qmul.ac.uk

Abstract. In our previous work, we explored the use of graph-theoretic spectral methods for clustering protein sequences [7]. The nodes of the graph represent a set of proteins to be clustered into families and/or super-families. Edges between nodes are undirected and weighted by the similarities between proteins. We constructed a novel similarity function based on BLAST scores. The similarity values are in turn used to construct a Markov matrix representing transition probabilities between every pair of connected proteins. By analyzing the perturbations to the stationary distribution of the Markov matrix (as in [6,4]), we partition the graph into clusters. In this paper, we compare our method with TribeMCL, which modifies random walks, by reinforcing strong edges and pruning weak ones, such that clusters emerge naturally from the graph [3]. We compare these two methods with respect to their ease of use and the quality of the resulting clusters.

1 Markov Analysis and Spectral Clustering

A major challenge in bio-informatics is the grouping together of protein sequences into functionally similar families. Biological function of a newly determined sequence is often inferred by similarity to a sequence of known function [2]. Grouping together sequences into families on the basis of how similar they are to one another may give insight into the general features associated with the biological role of that family. Clustering may also facilitate the discovery of relationships between protein sequences that are not otherwise apparent due to the transitive nature of sequence homology [8]. Moreover, clustering of the protein sequence data would clearly be of benefit in the selection of targets for structural genomics and in determining their use as comparative modelling templates [9].

In our previous work, we explored the use of graph-theoretic spectral methods for clustering protein sequences [7]. Each node in the graph corresponds to a protein sequence and the edges correspond to the strength (or similarity) with which two protein sequences belong to a group. We consider partitioning one such weighted undirected graph G into a set of discrete clusters. Ideally, the nodes in each cluster should be connected with highly-similar edges, while different clusters are either disconnected or are connected only by a few edges with low

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 278–286, 2003.

similarity. The problem is to identify these tightly coupled clusters, and cut the inter-cluster edges.

Following the framework in [4], we consider an undirected graph $G = (V, E)$ with vertices v_i, for $i = 1, \ldots, n$, and edges $e_{i,j}$ with non-negative weights $s_{i,j}$. Here the weight $s_{i,j}$ represents the similarity of vertices v_i and v_j. The edge weights are assumed to be symmetric, that is, $s_{i,j} = s_{j,i}$. In matrix notation the similarities are represented by a symmetric $n \times n$ matrix S with elements $s_{i,j}$. The degree of a node j is defined as: $d_j = \sum_{i=1}^{n} s_{i,j} = \sum_{j=1}^{n} s_{i,j}$. We represent D for a diagonal matrix of degrees: $D = \mathrm{diag}(d_1, \ldots, d_n)$. A Markov chain is defined using these affinities by setting a transition probability matrix $M = SD^{-1}$, where the columns of M each sum to 1. The transition probability matrix M defines the random walk of a particle on the graph G.

In this paper, we compare two different ways of propagating the Markov chain on the graph, leading to two different clustering algorithms: (1) our previous work ([7]) which involves analyzing perturbations to the stationary distribution of the Markov transition matrix, captured via the eigenvectors of the Markov matrix (as in [6,4]); and (2) TribeMCL which modifies the random walks to promote the emergence of natural clusters in the graph [3].

K-Means Spectral Clustering — Spectral methods use the leading eigenvectors of the Markov transition matrix derived from the similarity information. They allow one to study *global* properties of a dataset by making only *local* (pairwise) similarity measurements between data points. Here we notice that the matrix M is in general *not* symmetric. So, for a stable eigendecomposition of M, it is convenient to consider the normalized similarity matrix L given by, $L = D^{-1/2} M D^{1/2}$. Note L is symmetric and hence $M = D^{1/2} L D^{-1/2} = D^{1/2} U \, \Lambda \, U^T D^{-1/2}$, where $U = [\boldsymbol{u}_1, \boldsymbol{u}_2, \cdots, \boldsymbol{u}_n]$ are the eigenvectors and Λ is a diagonal matrix of eigenvalues $[\lambda_1, \lambda_2, \cdots, \lambda_n]$ of L, sorted in decreasing order. While the eigenvectors are orthonormal, the eigenvalues are real and have an absolute value bounded by 1, $|\lambda_k| \leq 1$. Thus an eigenvector $\boldsymbol{u}$ of L corresponds to an eigenvector $D^{1/2}\boldsymbol{u}$ of M with the same eigenvalue λ.

Consider propagating the Markov chain for β iterations. The transition matrix after β iterations can be represented as: $M^{\beta} = D^{1/2} U \Lambda^{\beta} U^T D^{-1/2}$. Therefore the probability distribution of a particle after β iterations is $\boldsymbol{p}^{\beta} = M^{\beta} \boldsymbol{p}^{0}$, where $\boldsymbol{p}^{0}$ is the initial distribution. As $\beta \to \infty$, the Markov chain approaches the stationary distribution $\boldsymbol{\pi}$, $M^{\infty} = \boldsymbol{\pi} \mathbf{1}^T$. The stationary distribution is given by $\boldsymbol{\pi} = \mathrm{diag}(D) / \sum_{i=1}^{n} d_i$ and $\boldsymbol{\pi}$ is unique as the graph is undirected and connected. Observe that $\boldsymbol{\pi}$ is an eigenvector of M as it is easy to show that $M\boldsymbol{\pi} = \boldsymbol{\pi}$ and the corresponding eigenvalue is 1.

Assuming the graph G is connected with edges having non-zero weights, it is convenient to interpret the Markovian relaxation process as perturbations to the stationary distribution, $\boldsymbol{p}^{\beta} = \boldsymbol{\pi} + \sum_{j=2}^{n} \lambda_j^{\beta} D^{1/2} \boldsymbol{u}_j \boldsymbol{u}_j^T D^{-1/2} \boldsymbol{p}^{0}$, where $\lambda_1 = 1$ is associated with the stationary distribution $\boldsymbol{\pi}$. The second eigencomponent $\boldsymbol{u}_2$ perturbs the stationary distribution the most as $\lambda_2 \geq \lambda_k$ for $k > 2$. Because $\boldsymbol{u}_2 \boldsymbol{u}_2^T$ is a matrix with elements $\{u_{2,i} \times u_{2,j}\}$, the probability of transitions between the points that share the same sign as $u_{2,i}$ is increased, while that of transitions across points with different signs is decreased [4]. Thus, a simple recipe for partitioning the graph into two clusters is to assign points based on the sign of the elements of $\boldsymbol{u}_2$: $u_{2,i} > 0 \Rightarrow$ assign point i to cluster 1, otherwise cluster 0. In [4] it was shown how this property translates to a condition of *piecewise constancy* on the form of the leading eigenvectors, i.e. elements of the eigenvectors have approximately the same value within each cluster. Specifically, it was shown that for K weakly coupled clusters the leading K eigenvectors of the stochastic matrix M will be roughly piecewise constant. The K-Means spectral clustering procedure is a particular manner of employing the standard K-Means algorithm on the elements of the leading K eigenvectors to extract K clusters simultaneously. For details see [6].

There are two issues left to be discussed: (1) How to define a suitable metric for pairwise similarity? (2) How to select K, the total number of clusters in a dataset?

Our starting point for defining a similarity measure between sequence i and sequence j, was constituted by their BLAST E-values, $b_{i,j}$ [1]. We then converted such values into probabilities using a negative-sigmoidal function: $p_{i,j} = \left[1 + \exp[\log_{10}(b_{i,j}) - \log_{10}(1 \cdot 10^{-1})]^4\right]^{-1}$. While the choice for the parameters in the sigmoid is subjective, we propose it as an attempt to capture the effects of relatively weak BLAST hits. The BLAST E-values, once converted into probabilities, are collected into a matrix P. In general P will not be symmetrical. We obtain a symmetrical matrix, S, by assigning to each $s_{i,j}$ and $s_{j,i}$ the higher of the two values $p_{i,j}$ and $p_{j,i}$. This amounts to a conservative interpretation of BLAST E-values. We used S as the similarity matrix in the K-Means spectral clustering algorithm that we applied to protein sequences [7].

In order to choose the total number of clusters K we analyze the eigenspectrum of the normalized similarity matrix L. We begin by noticing that if a graph is connected such that there is a path between every two nodes, then the leading eigenvalue will be 1 and all the other eigenvalues are bounded away from 1: $\lambda_k < 1, \forall\, k > 1$. This implies that the graph is one large connected component which can be divided possibly into several smaller clusters. In order to estimate the number of clusters from the eigenvalue spectrum, we first compute the *eigengaps*, which are the ratios of successive eigenvalues. We then apply a threshold on the eigengaps to select the number of clusters: $K = \min\{i : \lambda_i / \lambda_{i+1} > \epsilon\}$, where ϵ is predefined. We found this method adequate for the datasets that we considered.

TribeMCL — A cluster in a graph is characterized by the presence of many strong edges between its members and perhaps few weak connections with other clusters. If a graph contains such clusters, then it is very *unlikely* that random walks based on Markov transition probabilities will jump between two clusters too frequently. Hence, if the random walks can somehow be biased, say by pruning weak edges and reinforcing strong edges simultaneously, clusters may emerge naturally from the graph.

This is the idea behind the Markov Cluster (MCL) algorithm, which induces bias in random walks by alternating between two operators called *expansion* and *inflation* [10]. Expansion is similar to the propagation we discussed in previous section where the Markov transition matrix M is raised to a power. Inflation corresponds to taking powers of M entrywise, followed by a normalization step, such that the matrix elements in each column sum up to 1 again. Expansion makes the differences between nodes less distinguishable, while inflation has the effect of boosting probabilities on strong *intra-cluster* walks and demoting weak *inter-cluster* walks. The inflation process is dictated by a parameter r. Increasing r has the effect of increasing the tightness of clusters. Iterative application of expansion and inflation operators approaches an equilibrium state and the resulting graph is then examined for cluster information [10].

MCL constitutes the core component of TribeMCL, an algorithm used for clustering protein sequences [3]. In TribeMCL, the similarity measure between two proteins is built on the BLAST E-value. In particular, a similarity matrix S is put together by taking the average of the pairwise $-\log_{10}$(E-value) values, thus resulting in a symmetric matrix. The similarity matrix is then converted into a Markov transition matrix for the application of expansion and inflation operators.

2 Results

We compared the performance of the K-Means spectral clustering algorithm with TribeMCL on different datasets of proteins extracted from SCOP [5]. This is an expertly curated manual database in which proteins have been classified on the basis of their 3-dimensional structure as well as other relevant information. The SCOP classification is hierarchical and within super-families lie families. A pair of proteins within a family generally display discernible sequence similarity, while this may not be the case for two proteins each in a different family but both belonging to the same super-family. Proteins in the same super-family are believed to have the same function. We use the SCOP assignments in our clustering experiments to see how well protein sequences belonging to the same super-family are grouped together.

Performance Measures. If we consider the grouping provided by SCOP super-families as the "ground truth", it is possible to use the so called external quality measures to evaluate a clustering. We used the Precision, Recall and F_1-measure. For a certain protein set, let us call K the categorization into super-families provided by SCOP, and let us denote by λ the clustering returned by a certain clustering algorithm for that set. Let n_l^h denote the number of objects that are in cluster l according to λ as well as in class h given by K. The precision is then defined as the fraction of correctly retrieved proteins out of all the proteins in the cluster n_l: $P(C_l, K_h) = n_l^h/n_l$ The recall is defined as the fraction of correctly retrieved proteins out of all the proteins in the class n^h: $R(C_l, K_h) = n_l^h/n^h$ The F_1- measure, that combines precision and recall with equal weights. For the entire clustering the total F_1-measure is defined as: $F_1(\lambda, K) = 1/n \sum_{h=1}^{g} n^h \max_l \frac{2n_l^h}{n_l+n^h}$.

First we shall present results obtained on 3 datasets, which were built by hand-choosing super-families in such a way that the datasets had a simple structure, thus enabling the performance of the algorithm to be appreciated visually.

Dataset 1 consists of 108 protein sequences belonging to 3 different super-families namely Globin-like, Cytochromes and Fibronectin type III. We picked 36 sequences at random from each super-family, such that no two pairs of sequences have more than 40% sequence identity.

Dataset 2 consists of 511 protein sequences belonging to 7 super-families namely Globin-like (88), Cupredoxins (78), Viral coat and capsid proteins (106), Trypsin-like serine proteases (73), FAD/NAD (P)-binding domain (64), MHC antigen-recognition domain (51), Scorpion toxin-like (51). The maximum pairwise identity in this set was 95% but within each super-family there were up to 5 families, making this a rather challenging problem.

Dataset 3 consists of 507 sequences belonging to 6 super-families, namely Globin-like (88), EF-hand (83), Cupredoxins (78), (Trans)glycosidases (83), Thioredoxin-like (81), Membrane all-alpha (94). The last 2 super-families contained 12 and 13 families respectively. The maximum pairwise identity in this set was also 95%.

When running TribeMCL on these datasets, we found that the setting of the inflation parameter was crucial for obtaining good results. First, we tried adjusting the inflation parameter to get the roughly the same number of clusters as there are super-families in the dataset. This amounts to a “low” setting of the inflation parameter. However we observed that the resulting clusters typically have members from various different super-families all grouped together — when the inflation parameter is set low, TribeMCL tends to cluster all the proteins into few large clusters. If it was set higher, results get better and the procedure tends to identify more correct subdivisions. Below, we report results with the best setting for the inflation parameter that we could find, with regards to the quality of the overall clusters, over several independent runs.

Results on Dataset 1. The eigengap plot (not shown for lack of space) for Dataset 1 indicates that there are 3 relevant clusters in the data. If K is set to 3, the K-means spectral algorithm returns a perfect separation of the proteins into the 3 constituent families. While this dataset may seem easy to work with because the sequences in each group belong to a single family, remember that they had less than 40% sequence identity. For this reason, a simple procedure like the single-linkage analysis fails to reveal the right clusters. On this dataset, despite the wide range of choices we made for the inflation parameter, we found TribeMCL unable to return three distinct clusters. Furthermore, the largest cluster typically had members from at least two of the three families. Consequently, while for the solution obtained using the K-means spectral algorithm $F_1 = 1$, the best value for F_1 for a TribeMCL solution that we were able to obtain was 0.60.

Results on Datasets 2 and 3. Datasets 2 and 3 are much more difficult, involving a hierarchy of families and super-families. The K-means algorithm algorithm finds 9 clusters in Dataset 2 ($F_1 = 0.74$) and 6 clusters in Dataset 3

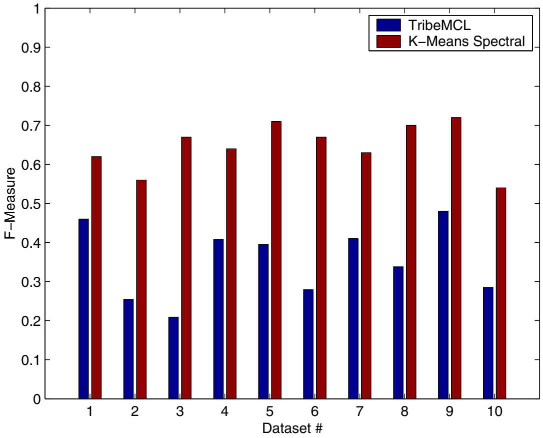
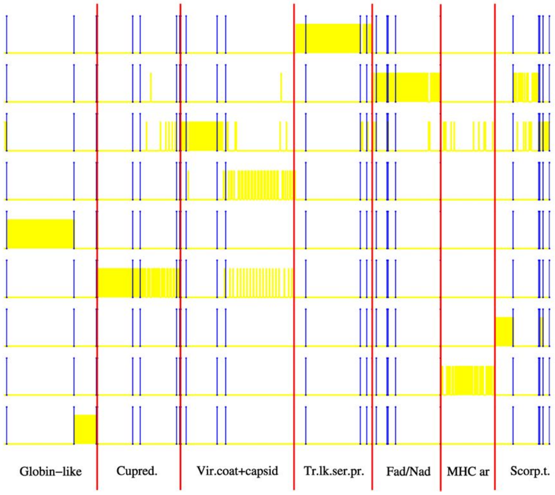


Fig. 1. F_1-measures for 10 randomly drawn subsets from SCOP. In each experiment the inflation parameter for TribeMCL was set to 1.6.

($F_1 = 0.81$). The clusters of the proteins obtained by K-Means do not correspond to the exact super-families in the set. For brevity, we analyze the results just on Dataset 2 (Fig. 2 Top). Some super-families are clearly separated. For example, the Globin-like super-family (column 1) splits into 2 clusters, namely Globins and Phycocyanin-like. The majority of sequences in the Trypsin-like serine protease super-family (column 4) are clustered but these are almost all the eukaryotic proteases (51) and some proteases from other smaller families are incorrectly assigned. Similarly the majority of the FAD/NAD (P)-binding domain sequences (column 5) are clustered together. A large group of animal virus proteins (the last grouping in the Viral coat super-family (column 3) cluster together but these appear to split into 2 sub-families. Figure 2 Bottom, shows the results obtained for the same dataset using TribeMCL, with the inflation parameter set to the best value that we could find, which is 1.55. With this setting TribeMCL finds 28 distinct clusters. Comparing with Fig. 2 Top, we notice that some of the larger clusters obtained by TribeMCL group disparate families. Also the TribeMCL procedure appears to create many spurious clusters (i.e., oversegmentation) containing only few proteins, and sometimes even a single one. Overall, the clusters returned by the TribeMCL procedure appear to lack coherency. The F_1 value for this solution was 0.54.

Finally, we compared the performance of the two algorithms on a group of 10 datasets which we generated from astral-95 by adding random super-families to a dataset until it contained at least 500 proteins. To ensure a fair selection of super-families, the super-families were chosen by selecting a random protein from astral-95 and then including all members of the corresponding super-family in the dataset. Fig. 1 presents the F_1-measure for the results obtained using K-means spectral and TribeMCL on these datasets. We can see that the K-means algorithm gives consistently an improved performance over TribeMCL.



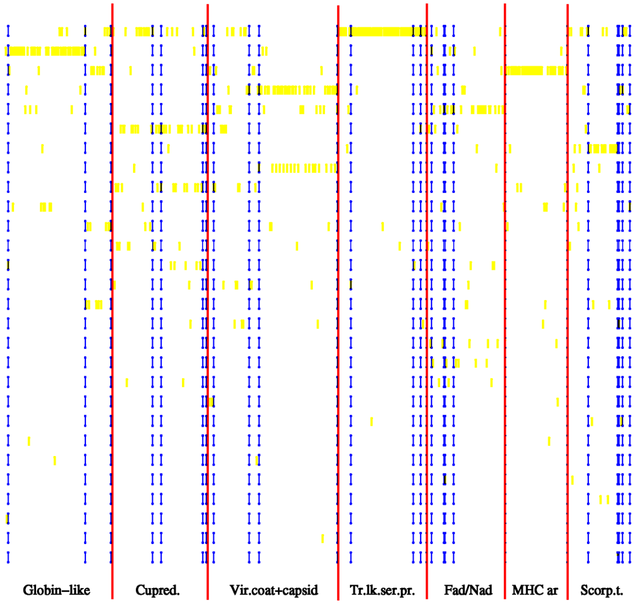


Fig. 2. Clustering results for Datasets 2 obtained using K-Means (Top) and TribeMCL (Bottom). The protein sequences in the datasets are arranged horizontally with each row corresponding to a different cluster. Short (yellow) bars represent the assignment of a protein sequence to a cluster in the following way: each protein has a bar in only one of the rows (clusters); the presence of the bar means that the protein is assigned to that cluster. Boundaries between super-families are shown by thick long vertical (red) lines; boundaries between families within each super-family are shown by tall (blue) lines.

3 Discussion

In this paper we have looked at two different methods for clustering protein sequences: K-Means spectral clustering [7] and TribeMCL [3]. The starting point for both algorithms is a set of pairwise BLAST E-values between sequences. From these values, each method builds a Markov transition matrix, M. The algorithms differ in the way in which they propagate the Markov chain on the graph: while K-Means spectral clustering analyzes perturbations to the stationary distribution of M, TribeMCL modifies the random walks to promote the emergence of natural clusters in the graph.

The K-Means spectral clustering algorithm is very simple to implement. The eigenvectors of the Markov transition matrix are obtained by singular value decomposition (SVD), which is a very stable process. The similarity matrices tend to be sparse and there are well known procedures in numerical analysis to speed up the computation of SVD. A Matlab implementation of the algorithm, running on a 1.8GHz Pentium 4 machine, takes about few seconds to cluster the biggest dataset presented here. The K-means procedure gave very stable results for several different runs. TribeMCL is also quite simple, involving only basic linear algebra operations.

We have compared the results obtained by the two methods on several difficult datasets. Both algorithms require the user to input a parameter. For the K-Means spectral clustering algorithm we found that the eigengap computation provides a good indication for a reasonable setting of the number of clusters in the dataset. In comparison, our experience with TribeMCL shows that there is a considerable trial and error involved in the selection of the inflation parameter.

Regarding the quality of the clusters, our algorithm seem to identify many of the relevant family/super-family groupings as defined by SCOP. However, the main clusters identified by TribeMCL appear to group disparate families, and TribeMCL also tends to create many spurious clusters containing only few proteins (see figure 2) .

These results are quite preliminary. The K-Means spectral clustering method has been tried on few problems, and more extensive testing is needed. Moreover, while it is known that TribeMCL can cluster thousands of sequences [3], we need to see how our spectral method scales to datasets that are much larger in size.

Acknowledgment

Thanks to Mansoor Saqi, James Casbon, Paul Grosvenor and Allan Jepson for discussions. AP acknowledges the support from MRC/DTI (grant G0100109).

References

1. Altschul S.F., Gish W., Miller W., Myers E.W. and Lipman D.J. Basic Local Alignment Search Tool *J Mol Bio* 1990 215:403-410
2. Brenner S.E., Koehl P. and Levitt M. The ASTRAL compendium for protein structure and sequence analysis. *Nucleic Acids Res* 2000 Jan 1;28(1):254-6

3. Enright A.J., Van Dongen S. and Ouzounis C.A. An efficient algorithm for large-scale detection of protein families. *Nucleid Acids research* 2002 30(7):1575-1584
4. Meila M. and J. Shi A random walks view of spectral segmentation. *Proc. International Workshop on AI and Statistics*, 2001.
5. Murzin A.G., Brenner S.E., Hubbard T. and Chothia C. SCOP: a structural classification of proteins database for the investigation of sequences and structures. *J Mol Biol* 1995 Apr 7;247(4):536-40
6. Ng A., M. Jordan and Y. Weiss On Spectral Clustering: analysis and an algorithm *NIPS*, 2001.
7. Paccanaro A., C. Chennubhotla, J. Casbon and M. Saqi Spectral Clustering of Protein Sequences *IJCNN*, 2003.
8. Sasson O., Vaaknin A., Fleischer H., Portugaly E., Bilu Y., Linial N. and Linial M. ProtoNet: hierarchical classification of the protein space. *Nucleic Acids Res* 2003 Jan 1;31(1):348-52
9. Sali A. and T.L. Blundell Comparative protein modelling by satisfaction of spatial restraints. *J. Mol Bio* 234, 779-815, 1993
10. van Dongen, S. Graph Clustering by flow simulation. Ph. D. Thesis, University of Utrecht, The Netherlands.

Divergence Projections for Variable Selection in Multi–layer Perceptron Networks

Antonio Eleuteri[1], Roberto Tagliaferri[2], and Leopoldo Milano[3]

[1] Dipartimento di Matematica ed Applicazioni "R. Caccioppoli"
Università di Napoli "Federico II", Napoli
and INFN sez. Napoli, via Cintia, I-80126 Napoli, Italia
eleuteri@na.infn.it

[2] DMI, Università di Salerno via S. Allende, I-84081 Baronissi (Sa), Italia
and INFM unità di Salerno
robtag@unisa.it

[3] Dipartimento di Scienze Fisiche, Università di Napoli "Federico II"
and INFN sez. Napoli, via Cintia, I-80126 Napoli, Italia

Abstract. In this paper an information geometric–based variable selection method for MLP networks is shown. It is based on divergence projections of the Riemannian manifold defined by a MLP network on submanifolds defined by MLP networks with reduced input dimension. We show how we can take advantage of the layered structure of the MLP to simplify the projection operation, which cannot be accurately done by using only the Fisher information metric. Furthermore, we show that our selection algorithm is more robust and gives better results than other well known selection algorithms like Optimal Brain Surgeon. Some examples are shown to validate the proposed approach.

1 Introduction

Multi–layer perceptron networks are well known for their capability to learn complex relationships between sets of variables. Most often, however, the phenomenon we are trying to model is very complex, and there is not a priori knowledge which can be used to select the input variables which are relevant to modeling the input–output relationship we seek. So, the usual approach is to use all the available input variables. This is true in particular in biostatistics applications, in which researchers are not willing to exclude possibly relevant variables.

The inclusion of many variables however has many drawbacks: it is more difficult to interpret the resulting model, irrelevant variables act as noise worsening the generalization capability of the model, data gathering can be much more costly, small datasets become less useful because of the curse of dimensionality.

Variable selection methods are a main subject of research in statistics, with application to commonly used modeling techniques [4]. However, it turns out that much of the developed theories and algorithms are not usable in the case of neural networks due to their complex structures. Therefore, in the last years many *ad*

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 287–295, 2003.

hoc methods have been developed which are taylored to neural networks. Since variable selection requires the definition of a selection criterion and a search procedure, several methods can be defined by combining instances of the two. For a review of some variable selection algorithms we refer to [6],[8].

All of the currently existing methodologies, however, although successful to various degrees, are more or less heuristic in nature and lack a solid theoretical foundation. Furthermore, many methods require retraining of the network, and have therefore a very high computational cost. In this paper we propose a selection criterion which has its roots in information geometry, and in this framework we develop an algorithm which makes full use of the natural differential geometric structure possessed by MLP's [1], [2].

2 Variable Selection

Let $\mathbf{x}, \mathbf{y}$ be random variables representing the input and output of a statistical model, respectively. The goal is to reduce the dimension of the input vector to a smaller number while preserving as much as possible the prediction capability of the model. We suppose to have a dataset $C = \{(\mathbf{y}_i, \mathbf{x}_i)\}_{i=1...N}$ which we use to train the model.

Removal of a set of inputs $\{x_k\}, k \in K$ is equivalent to the following conditions on the weights connecting the inputs with the hidden units:

$$w_{ik} = 0, \forall k \in K, \forall i \in \{1 \dots H\} \tag{1}$$

where H is the number of hidden units.

If we consider the parameters in a model as the coordinates of some space, we get an immediate geometric intuition about the nature of variable selection. Setting some parameters to zero can be seen as a projection of the original space on a lower dimensional space. To make the projection, however, we must take into account the nature of the space and its geometric structure, and some kind of distance measure must be defined.

The complete description of the available data is in terms of a joint distribution $p(\mathbf{x}, \mathbf{y})$. We can factorize this distribution as:

$$p(\mathbf{x}, \mathbf{y}) = p(\mathbf{y}|\mathbf{x}, \mathbf{w})r(\mathbf{x}) \tag{2}$$

where $\mathbf{w}$ is the vector of parameters of the stochastic input–output relation realized by the model. Given a vector with a smaller number of parameters, we get a different description of the joint distribution:

$$q(\mathbf{x}, \mathbf{y}) = p(\mathbf{y}|\mathbf{x}, \mathbf{u})r(\mathbf{x}) \tag{3}$$

We would like $\mathbf{u}$ to be the projection of $\mathbf{w}$, while preserving as much as possible the distance from $p(\mathbf{x}, \mathbf{y})$ to $q(\mathbf{x}, \mathbf{y})$. We need then to define a distance between models.

A distance appropriate for probability measures is the Kullback divergence [2]:

$$D(p\|q) \equiv \int \log \frac{p(z)}{q(z)}\ p(z)\ \mathrm{d}z\ . \tag{4}$$

As we shall see later, the Kullback divergence is the natural choice given the geometric structure of the manifold of the MLP parameters.

Evaluation of the divergence from $p(\mathbf{x}, y)$ to $q(\mathbf{x}, y)$ gives:

$$\begin{aligned} D(p(\mathbf{x},\mathbf{y})\|q(\mathbf{x},\mathbf{y})) &= D(r(\mathbf{x})\|r(\mathbf{x})) + \mathrm{E}_x[D(p(y|\mathbf{x},\mathbf{w})\|p(y|\mathbf{x},\mathbf{u}))] = \\ &= \mathrm{E}_x[D(p(y|\mathbf{x},\mathbf{w})\|p(y|\mathbf{x},\mathbf{u}))]\ . \end{aligned} \tag{5}$$

The projection of $\mathbf{w}$ can then be found by solving the nonlinear program:

$$\arg\min_{\mathbf{u}} \mathrm{E}_x\left[D(p(\mathbf{y}|\mathbf{x},\mathbf{w})\|p(\mathbf{y}|\mathbf{x},\mathbf{u}))\right] \tag{6}$$

In a Bayesian framework we can estimate the divergence by evaluating the posterior expectation of the divergence with respect to the posterior distribution of network weights [13]. However, in general this is not analytically possible in the case of a complex model like the MLP, so we must make a Monte Carlo approximation using a sample of parameters from the posterior obtained for example with a MCMC method [11]. For each parameter vector we shall then solve a minimization problem to find the corresponding projected parameter vector.

It should be noted that for a complex model like the MLP, the solution of the nonlinear program may be hard to find (and there is no a priori guarantee that it is unique!). Such a "brute force" approach, therefore, is not likely to work.

3 Taylor Series Expansion of the Divergence

A possible solution is to find an approximation to the divergence which could make the projection task easier. We expand the divergence in Taylor series (as a function of $\mathbf{u}$) around $\mathbf{w}$ up to second order:

$$D = \frac{1}{2}(\mathbf{u}-\mathbf{w})^T G(\mathbf{w})(\mathbf{u}-\mathbf{w}) + o(\|\mathbf{u}-\mathbf{w}\|^2)\ , \tag{7}$$

where the elements of the matrix $G(\mathbf{w})$ can be proven to be:

$$g_{ij}(\mathbf{w}) = \mathrm{E}_x\left[\int p(\mathbf{y}|\mathbf{x},\mathbf{w})\frac{\partial}{\partial w_i}\log p(\mathbf{y}|\mathbf{x},\mathbf{w})\frac{\partial}{\partial w_j}\log p(\mathbf{y}|\mathbf{x},\mathbf{w})\mathrm{d}\mathbf{y}\right] \tag{8}$$

which are the elements of the Fisher or information metric, the Riemannian metric of the manifold of the network parameters.

Although the use of the information metric may seem as a powerful and easy tool, simulations show that this method gives quite poor results when the manifold on which we project has a low dimension (i.e., we "set to zero" many parameters). The second order approximation then is not accurate enough for our purposes. It can be easily proven that the Optimal Brain Surgeon algorithm (OBS) [5] and its variants [12], [7] can be considered as special cases of eq.(8), since the information metric is equivalent to the Hessian of the error criterion. OBS is derived under the additional hypothesis that the network has been trained to a minimum of the error function.

4 Exploiting the Layered Structure: The Layered Projection Algorithm

We shall show how the divergence can be efficiently minimized by exploiting the layered structure of the MLP; this will lead us to the formulation of the Layered Projection algorithm.

Let us consider a stochastic extension of the MLP in which each hidden unit activation has a noise model whose density is $p(z_i|\mathbf{x}, \mathbf{v}_i^{(1)})$, where $\mathbf{v}_i^{(1)}$ is the vector of first layer weights feeding the unit. Since the activations are independent, we can write the joint activation for the hidden layer:

$$p(\mathbf{z}|\mathbf{x}, \mathbf{v}^{(1)}) = \prod_i^H p(z_i|\mathbf{x}, \mathbf{v}_i^{(1)}) \ . \tag{9}$$

Given a noise model on the outputs, $p(\mathbf{y}|\mathbf{z}, \mathbf{v}^{(2)})$, where $\mathbf{v}^{(2)}$ is the vector of second layer weights, we can then define the joint density of output and hidden (independent) activations:

$$\begin{aligned} p(\mathbf{y}, \{z_i\}|\mathbf{x}, \mathbf{v}^{(1)}, \mathbf{v}^{(2)}) &= p(\mathbf{y}|\{z_i\}, \mathbf{v}^{(2)}) \prod_i^H p(z_i|\mathbf{x}, \mathbf{v}_i^{(1)}) = \\ &= p(\mathbf{y}|\mathbf{z}, \mathbf{v}^{(2)}) p(\mathbf{z}|\mathbf{x}, \mathbf{v}^{(1)}) \ . \end{aligned} \tag{10}$$

This factorization is possible due to the flow of information in a MLP, in fact the outputs depends on the first layer weights through the hidden activation, which does not depend on the second layer weights. We can then use eq.(10) in the definition of the divergence, which for a finite dataset it can then be shown to take the form:

$$\begin{aligned} D = &\frac{1}{N} \sum_k^N \int p(\mathbf{z}|\mathbf{x}_k, \mathbf{w}^{(1)}) \log \frac{p(\mathbf{z}|\mathbf{x}_k, \mathbf{w}^{(1)})}{p(\mathbf{z}|\mathbf{x}_k, \mathbf{u}^{(1)})} \, \mathrm{d}\mathbf{z} + \\ &+ \frac{1}{N} \sum_k^N \int p(\mathbf{y}|\mathbf{z}_k, \mathbf{w}^{(2)}) \log \frac{p(\mathbf{y}|\mathbf{z}_k, \mathbf{w}^{(2)})}{p(\mathbf{y}|\mathbf{z}_k, \mathbf{u}^{(2)})} \, \mathrm{d}\mathbf{y} = D_1 + D_2 \ . \end{aligned} \tag{11}$$

It should be noted that, in the case the variables we integrate over are not continuous, the integrals become sums (this happens if the models are described in terms of probability distributions instead of densities).

The structure of eq.(11) reflects the layered structure of the MLP, and the chain of dependencies between parameters and activations suggests that the two terms can be minimized in sequence: minimize the first term using information from inputs, first layer weights and hidden activations of the starting network; then minimize the second term using information from first layer weights and hidden activations of the projected network, second layer weights and output activation of the starting network.

It can be shown that the above scheme can be iterated for an arbitrary number of layers. We have thus obtained the following result:

Proposition 1 *Divergence projection for variable selection in MLP networks only requires solving a sequence of single–layer minimization problems.*

We call the resulting algorithm *Layered Projection*.

4.1 Hidden Noise Model

If, without loss of generality, we assume that the hidden activation functions are logistic sigmoids, then a simple noise model is given by a Bernoulli distribution for each hidden activation. In this case, since each hidden unit activation can be seen as a probability, the resulting activation model is:

$$p(z_i|\mathbf{x}, \mathbf{v}_i^{(1)}) = f(\mathbf{v}_i^{(1)} \cdot \mathbf{x})^{z_i} \, (1 - f(\mathbf{v}_i^{(1)} \cdot \mathbf{x}))^{1-z_i} \ . \tag{12}$$

This expression is formally equivalent to a logit model which relates the hidden activations to the $\mathbf{x}$ variables. Note that the hidden activation of the deterministic MLP is just the expectation of the activation of the stochastic MLP. This choice of noise model is not restrictive, since we just create a stochastic model of a deterministic network.

The activation for each hidden unit can be put in exponential family form:

$$p(r_i|\theta(\mathbf{x}, \mathbf{v}_i^{(1)})) = \exp\left(\theta(\mathbf{x}, \mathbf{v}_i^{(1)}) r_i - \psi(\theta(\mathbf{x}, \mathbf{v}_i^{(1)}))\right) \tag{13}$$

where $\theta = \log(f/(1-f)), r = \delta(z-1), \psi = \log(1 + \exp(\theta))$. For a set of N independent observations, the activation is still an exponential family of larger dimension:

$$\begin{aligned} p(\{z_{i,k}\}_k|\{\mathbf{x}_k\}, \mathbf{v}_i^{(1)}) &= \prod_{k=1}^{N} p(r_{i,k}|\theta(\mathbf{x}_k, \mathbf{v}_i^{(1)})) = \\ &= \exp\left(\sum_k \theta(\mathbf{x}_k, \mathbf{v}_i^{(1)}) r_{i,k} - \sum_k \psi(\theta(\mathbf{x}_k, \mathbf{v}_i^{(1)}))\right) = \\ &= \exp\left(\boldsymbol{\theta}_{i,N}^* \cdot \mathbf{r}_{i,N}^* - \psi_N\right) \end{aligned} \tag{14}$$

where

$$\begin{aligned}
\mathbf{r}^*_{i,N} &= (r_{i,1}, \dots, r_{i,N}), \\
\boldsymbol{\theta}^*_{i,N} &= (\theta(\mathbf{x}_1, \mathbf{v}_i^{(1)}), \dots, \theta(\mathbf{x}_N, \mathbf{v}_i^{(1)})), \\
\psi_N &= \sum_k \psi(\theta(\mathbf{x}_k, \mathbf{v}_i^{(1)})) \ .
\end{aligned} \tag{15}$$

If we denote by $S_{i,k} = \{p(r_{i,k}|\theta(\mathbf{x}_k, \mathbf{v}_i^{(1)}))\}$ the manifold of distributions corresponding to the k-th pattern, then the joint distribution of eq.14 is an element of the product manifold $S^*_{i,N} = S_{i,1} \times S_{i,2} \times \cdots \times S_{i,N}$.

Since $\boldsymbol{\theta}^*_{i,N}$ is a function of $\mathbf{v}_i^{(1)}$, it cannot take arbitrary values, and is therefore restricted to a region of $S^*_{i,N}$. This region is called a *curved exponential family* in $S^*_{i,N}$, with inner coordinate system $\mathbf{v}_i^{(1)}$.

The dualistic structure of the manifold induces a divergence which is equal to the Kullback divergence. This fact gives a justification for the use of the Kullback divergence as a distance between models [2].

The projection can then be seen as an m-projection from a manifold M_w with inner coordinates $\mathbf{w}_i^{(1)}$ to a submanifold M_u with inner coordinates $\mathbf{u}_i^{(1)}$. Furthermore, since this submanifold is e-flat, the m-projection is unique [2].

The projection corresponds to a Maximum Likelihood (ML) estimation of the coordinates of the projected submanifold, which can be carried out by solving the likelihood equations $\nabla_{\mathbf{u}_i^{(1)}} D_1 = 0$.

For a finite data set, it can be shown that the ML equations are given by:

$$\sum_j \frac{\exp(\mathbf{u}_i^{(1)} \cdot \mathbf{x}_j)}{1 + \exp(\mathbf{u}_i^{(1)} \cdot \mathbf{x}_j)} \mathbf{x}_j = \sum_j \frac{\exp(\mathbf{w}_i^{(1)} \cdot \mathbf{x}_j)}{1 + \exp(\mathbf{w}_i^{(1)} \cdot \mathbf{x}_j)} \mathbf{x}_j \ . \tag{16}$$

Note that eq.(16) can be seen as the ML equations for a Generalized Linear Model (GLM) with logistic canonical link [10], to which the Iteratively Reweighted Least Squares (IRLS) algorithm [10] can be applied to solve the problem in a very fast and efficient way.

For the second layer we solve $\nabla_{\mathbf{u}_i^{(2)}} D_2 = 0$, using the information obtained from the solution of the first problem. We again get ML equations for a GLM, with canonical link dependent on the kind of task.

We have thus shown that the problem of divergence projection can be reduced to a sequence of minimization problems, which can be easily and fastly solved.

It should be noted that some algorithms have been been devised which make layered fittings, and which are similar to Layered Projection although derived on heuristic grounds and with many approximations. For a description we refer to [6] and the quotations therein.

A similar approach, in a more classical statistical context, has been proposed in [3] and applied to GLMs. Our work can be seen as an information geometric generalization of their approach.

4.2 Properties of Divergence Projections

In the present context, it can be shown that the divergence enjoys two fundamental properties: transitivity and additivity. These two properties are fundamental, because they ensure that the order in which we select variables is not relevant, and that embedded models are always at a greater "distance" from the starting model than embedding models. These properties ensure the monotonicity of the divergence, which in turn implies that fast, exact search algorithms (e.g. branch and bound, [4]) can be applied without having to explore the full features space.

5 Experiments

In this section we shall make some experiments to validate the proposed approach. In each experiment the networks have been trained in the Bayesian framework using a Gaussian approximation of the posterior distribution of the parameters [9]. After training, the parameter vector was projected by using both the Layered Projection (LP) and the OBS algorithm. Furthermore, a two–tailed, two–sample Kolmogorov–Smirnov (K–S) test at the significance level 0.05 has been performed to assess the null hypothesis that the outputs of the starting network and the projected one have the same distribution.

5.1 Experiment 1: Noisy Redundant Regression

In this problem, we want to build an approximation to the function:

$$y = \sin(2\pi x_1) + n \tag{17}$$

where $n \sim \mathcal{N}(0, 0.05^2)$, $x_1 \sim \mathcal{N}(0, 1)$. However, we use three more regression variables as input: $x_2 \sim \mathcal{N}(x_1, 0.02^2)$, $x_3 \sim \mathcal{N}(0.5, 0.2^2)$, $x_4 \sim \mathcal{N}(0.1, 0.2^2)$. Therefore, only x_1 is strictly relevant to the task, x_2 is correlated with x_1 while x_3 and x_4 are completely useless. We expect that the variable selection algorithm is able to make a projection of the trained network which reproduces the correct function thus identifying the least relevant inputs. The training and test data sets are composed by 100 and 200 patterns, respectively.

The starting network has 4 inputs, 3 hidden units and one output. In tab.(1) the results of the projections are shown.

Note that the null hypothesis cannot be rejected for the projected networks in the case of the LP algorithm, while it is strongly rejected in the case of the OBS algorithm.

It can be seen that the LP algorithm is quite robust, being able to remove the irrelevant inputs without adverse effects on the prediction.

The information metric method does not give a good approximation. Instead, the removal of the irrelevant inputs completely changes in the worse the mapping performed by the network, as also indicated by the p–values. The failure of this method can be explained by the fact that the algorithm modifies all the weights in the network, and the removal of many weights makes the second order approximation a poor one.

Table 1. Results of the projection operations for the regression problem. We report the name of the algorithm, the dimension of the projected manifold and the labels of removed inputs, the mean absolute error of the prediction, the error variance and the result of the K–S test.

Algorithm	Dim (Removed input)	Mean abs error	Error σ^2	Reject?
None	19 (none)	0.03	0.001	-
LP	16 (4)	0.0252	3.06e-4	No
OBS	16 (4)	0.91	0.049	Yes
LP	13 (4,3)	0.0144	9.46e-4	No
OBS	13 (4,3)	1.083	0.054	Yes
LP	10 (4,3,2)	0.014	1.06e-4	No
OBS	10 (4,3,2)	1.029	0.08	Yes

5.2 Experiment 2: Binary Redundant Classification

In this problem, we want to classify data produced by a three component Gaussian mixture. One of the components is assigned to class one, two of the components are assigned to class two. The prior probabilities for the components are 0.5, 0.25, 0.25. Each of the components has its full covariance matrix. To make the task more complex, two more variable are considered as inputs to the network, also if they do not contribute to the classification. The first variable is $\mathcal{N}(3, 3^2)$, the second one is $\mathcal{U}(-3, 3)$. The training and test data sets consist in 200 patterns each. The starting network has 4 inputs, 6 hidden units and one output. In tab.(2) the results of the projections are shown.

Table 2. Results of the projection operations for the classification problem. We report the name of the algorithm, the dimension of the projected manifold and the labels of removed inputs, the mean classification error, the cross class errors and the result of the K–S test.

Algorithm	Dim (Removed input)	Mean class error	c_1/c_2	c_2/c_1	Reject?
Bayes	-	.17	0.218	0.11	-
None	37 (none)	.22	0.2	0.244	-
LP	31 (4)	.19	0.236	0.13	No
OBS	31 (4)	.42	0.218	0.67	Yes
LP	25 (4,3)	.175	0.236	0.1	Yes
OBS	25 (4,3)	.405	0.236	0.61	Yes

For the LP algorithm, the null hypothesis cannot be rejected for the projected network in the case of one removed input, while for two removed inputs, the null hypothesis is rejected. In the case of the OBS algorithm it is always strongly rejected.

The p–values indicate statistically equivalent predictions for the first LP projected network, while for the second one they are not statistically equivalent, and in fact the projected network performs much better, with performance near the optimal Bayes classifier. Also in this problem, the OBS projected networks give very different (and worse) predictions.

6 Conclusions

The Layered Projection algorithm has many remarkable features. From a theoretical viewpoint it sheds some light into the variable selection problem for MLPs. From a practical viewpoint it exhibits a low computational cost, it does not require retraining of the reduced network, it uses the same data used to train the original network, it lends well to the application of fast exact search procedures and it is applicable to all kind of modeling tasks. Last but not least, its implementation is easy, since existing software can be used with little or no changes.

References

1. S. Amari. Information geometry of EM and em algorithms for neural networks. Neural Networks 8 (1995) 1379–1408
2. S. Amari. Methods of Information Geometry. Translations of Mathematical Monographs 191, Oxford University Press, Oxford (2000)
3. J. A. Dupuis, C. P. Robert. Bayesian Variable Selection in Qualitative Models by Kullback–Leibler projections. J.Statistical Planning and Inference 111 (2003) 77–94
4. K. Fukunaga. Introduction to Statistical Pattern Recognition, second edition. Academic Press (1990)
5. B. Hassibi, D. G. Stork. Second order derivatives for network pruning: Optimal Brain Surgeon. In S. J. Hanson, J. D. Cowan adn C. L. Giles (eds), Advances in Neural Information Processing Systems 5: Proceedings of the 1992 Conference. Morgan Kaufmann, San Mateo (1993) 164–171
6. P. van de Laar, T. Heskes, S. Gielen. Partial retraining: A new approach to input relevance determination. International Journal of Neural Systems A 9 (1999) 75–85
7. Y. LeCun, J. S. Denker, S. A. Solla. Optimal Brain Damage. In D. S. Touretzky (ed), Advances in Neural Information Processing Systems 2: Proceedings of the 1989 Conference. Morgan Kaufmann, San Mateo (1990) 598–605.
8. Ph. Leray, P. Gallinari. Feature Selection with Neural Networks. Behaviormetrika (special issue on Analysis of Knowledge Representation in Neural Network Models) 26(1) (1999) 145–166
9. D. J. C. MacKay. A practical Bayesian framework for backpropagation networks. Neural Computation 4(3) (1992) 448–472
10. P. McCullagh, J. Nelder. Generalized Linear Models. Chapman and Hall, London (1989)
11. R. M. Neal. Bayesian Learning for Neural Networks. Springer, New York (1996)
12. A. Stahlberger, M. Riedmiller. Fast network pruning and feature extraction using the unit–OBS algorithm. In M. Mozer, M. Jordan, T. Petsche (eds), Advances in Neural Information Processing Systems 9. MIT Press, Cambridge (1997) 655–661
13. C. P. Robert, G. Casella. Markov Chain Monte Carlo Methods. Springer, New York (1999)

Large Scale Prediction of Protein Interactions by a SVM-Based Method

Francesco Marangoni[1], Matteo Barberis[1], and Marco Botta[2]

[1] Fondazione per le Biotecnologie, Torino, Italy
[2] Dipartimento di Informatica, Università di Torino, Torino, Italy
botta@di.unito.it

Abstract. A number of techniques have been developed in order to address issues such as genome, trascriptome and proteome analysis. However, a time and cost effective technique for interactome analysis is still lacking. Lots of methods for the predicion of protein - protein interacions have been developed: some of them are based on high quality alignment of sequences, others are based on the tridimensional features of proteins, but they all bear strong limitations that make impossible their large scale application. Recently, an SVM-based machine learning approach has been used to address this topic. Although the method was able to correctly classify 80% of the test samples, it was not applied to the prediction of yet unknown interactions. In this work, we address this topic and show that an optimized, SVM-based machine learning approach trained with combinations of shuffled sequences as examples of lack of interaction is unable to make large scale predictions of interaction.

1 Introduction

The recent sequencing of human and other species genomes has given a strong impulse to the large scale study of genes and their products. A number of techniques have been developed in order to address issues such as genome, trascriptome and proteome analysis.

Unfortunately, and in contrast with other cell molecules, a bench technique that allows a large scale, cost and time effective analysis of interactome is still lacking. To overcome this limitation, several bioinformatic approaches have been developed to make predictions, which subsequently have to be validated by classical experiments of wet biology.

The search for pairs of proteins that interact is really a challenging issue. Given that a protein has from 2 to 10 interacting partners within a cell [5], it is clear that a bunch of interesting proteins has to be distinguished from a proteome of noisy ones.

Some bioinformatic approaches rely on the alignment of sequences coming from different organisms. The principal methods are philogenetic profiling, gene order conservation, "Rosetta stone", co-evolution analysis, in silico two hybrid [5] [8]. Even if these methods are considered preferable [1], they bear strong limitations. First of all, they need high quality sequences that are not always available. Moreover, they are based on the comparison of homolog sequences from different organisms. Inferring that an organism has (or has not) a homolog for a given sequence means that the organism has

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 296–301, 2003.

to be fully sequenced at a good qualitative standard. As a consequence, even if those methods provide good results, they can be hardly ever applied.

Other methods, namely the "docking" methods, try to infer an interaction by analysing the tridimensional complementarity of two proteins. The strongest limitation of such methods is that the tridimensional structure is known for a little minority of the proteins. This problem could be partially overcome by the so-called "threading" procedures, which combine homology modelling, structure refining and tridimensional complementarity analysis. Despite of this, the method cannot be applied to the sequences that lack structurally resolved homologs.

Recently, an SVM-based prediction method has been proposed that, looking exclusively at the primary structure and related physico-chemical properties would be functional and could be applied without limitations. Such method learns from examples of interacting and non - interacting proteins, represented as physico-chemical characteristic patterns, and tries to rule out the regularity (which is often unknown) that permits to discriminate between the two classes [2].

SVM is a supervised, kernel machine learning method that is based upon the concept of structural risk minimization [1]. It has recently been used for bioinformatic purposes [2] [9] because of some interesting features. First of all, it can efficiently manage huge amounts of data; then, the phenomenon of overfitting is strongly reduced; moreover it exists a publicly available implementation in C language which is computationally very efficient [4]. An SVM-based method has been used by Bock and Gough [2] to classify pairs of interacting and non-interacting proteins, with a precision of 80%. It must be underlined here that, while interacting pairs of proteins are easily found in specialized databases, it does not exist a database of non-interacting proteins; then, negative examples were generated by shuffling the sequence of interacting proteins. The strongest limitation of this approach is that the authors did not apply the method to the real problem of predicting new interactions.

The present work has three purposes: to determine the crucial parameters for the construction of good SVM input vectors, to ameliorate the classification performance of 80% found by Bock and Gough, and to make predictions of novel interacting pairs of proteins through our optimized SVM-based method. The paper is organized as follows: in the next section a review of the methods used is given, followed by a summary of the obtained results, and finally a general discussion about the results is provided.

2 Matherials and Methods

Hardware and software. The computer used all over this work is an AMD DURON 750 MHz, with 576 Mb RAM. The software implementing the SVM algorithm was written in C language and kindly made accessible to the scientific community by Dr. T. Joachims [4]. All the other scripts needed for this work were developed in PERL 5.0 by the authors.

Databases and flatfiles. The Database of Interacting proteins (DIP) was used as source of information about interactions (24th January 2002 and 28th June 2002 releases) and sequences of interacting proteins (2nd June 2002 release). AAindex database provided the information about the aminoacid related physico-chemical properties (31st

July 2002 release). Saccharomyces Genome Database (SGD) was the source of the Saccharomyces cerevisiae whole proteome file.

Sets of physico-chemical properties. Two indexes for each of the five major index classes were chosen on the basis of their completeness and the lack of correlation with the other chosen indexes. We chose DAYM780101 and HUTJ700101 for composition, SWER830101 and JANJ780101 for hydrophobicity, CHAM820101 and DAYM780201 for miscellaneous physico-chemical properties, GEIM800105 and NAGK730102 for beta-propension, GEIM800101 and NAGK730101 for alpha-and turn propension. Two sets of aminoacidic features were built choosing one index per class.

Generation of vectors representing interactions. The vectors representing the supposed interaction between two proteins were generated as described in [2]. Briefly, an aminoacid index was used to substitute the aminoacid residues of the primary sequence. This vector was standardized to a fixed length, and vectors originated from different indexes joined. Pairs of interacting proteins were represented by the joining of the two respective vectors. Negative examples were generated in the same way starting from shuffled sequences of interacting proteins.

Learning sets and Test sets. All learning sets were composed by a variable number of examples of interaction and the same number of examples of non-interaction. Where not otherwise specified, the test set was composed by the 2300 interactions that are uniquely found in the more recent release of DIP database, and 2300 examples of lack of interaction.

Output data analysis. Pairs of proteins were considered as interacting when SVM returned a value greater than zero; otherwise they were considered as non-interacting.

3 Summary of Results

In order to optimize the SVM-based classifier, we trained SVM with 2000 examples of interaction and 2000 examples of lack of interaction varying the standardization length of the vectors and the sets of physico-chemical properties. Length of 100, 200 and 400, and three sets of physico-chemical properties (Set 1 and Set 2 contained 5 indexes, and Set 3 contained all the ten indexes) were chosen. As shown in Fig. 1, we found that the classification performance grows when the standardization length grows as well.

Moreover, one of the 5-indexes set (Set 1) got better performances than the others in classifying examples of either interaction or non interaction. Surprisingly, Set 3, comprising all 10 indexes, failed to improve the classification performance: in fact, it gave an overall performance strictly similar to that obtained by the worst 5-indexes set (Set 2).

Then, we studied the influence of the number of training instances on the classification performance. As shown in Fig. 2, the curves that link the number of learning examples to the classification performance were logarithmic-like for both instances of interaction and lack of interaction. Some events characterized the curves when >500 learning examples of interaction were used: both curves showed a very similar performance, the standard deviation of each point became extremely low, and a plateau was reached. We chose as optimal the number of 2200 learning examples of interaction, be-

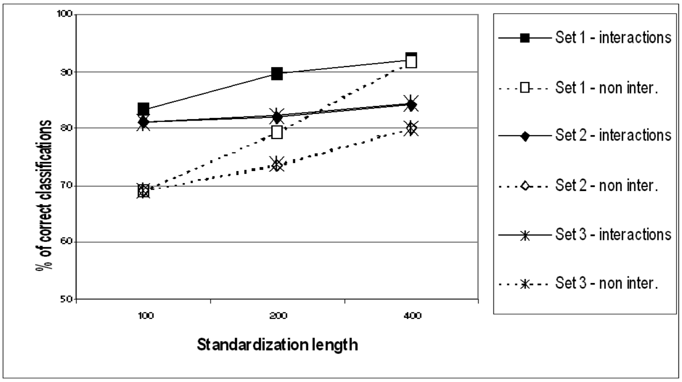


Fig. 1. Effect of the standardization length and of the three sets of physico-chemical aminoacidic features on the SVM-method classification performance. For every set, the classification performance was studied for both interacting and non-interacting pairs of proteins

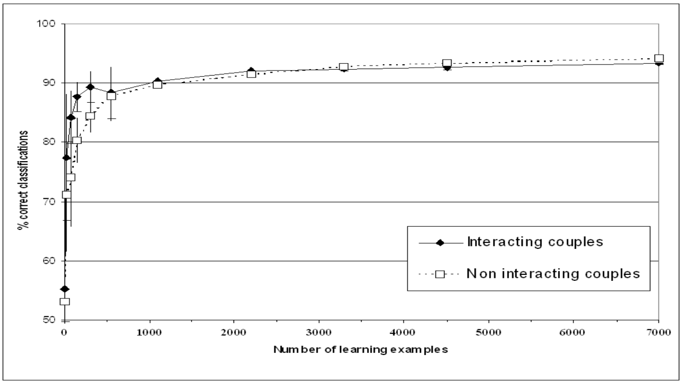


Fig. 2. Link between the number of learning examples used and the classification performance obtained. Every dot represents the average ± standard deviation of 5 independent experiments

cause it corresponded to the plateau point that was computed using the smallest number of learning examples, thus using the lowest amount of computational resources.

The optimizing parameters were then applied to build a Saccharomyces cerevisiae restricted classifier. This classifier had a classification performance of 92% for both interacting and non-interacting proteins.

4 Discussion

Three parameters were individuated as crucial for the construction of interaction-representing vectors of good quality: the standardization length of the vectors, the set of physico-chemical properties, and the number of learning examples.

Another crucial point was the right choice of examples of lack of interaction. Since a database of non-interacting protein does not exist, we needed to generate artificial examples of non-interacting proteins. Bock and Gough proposed that shuffled sequences from former interacting proteins could act as a non-interacting pair of proteins [2]. Even if it appears a very artificious method, we kept it for the present work.

As far as the standardization length is concerned, we found that the best performances were achieved with standardization length of 400, while lower lengths led to lower classification performances. The analysis of the classification errors occurring using a smaller standardization length clarified that errors were made mostly when classifying pairs of protein composed by at least one much bigger than the average (data not shown). This could lead to the hypothesis that "long" proteins constrained in "small" space lose the features needed for a correct classification.

Two sets of 5 physico-chemical properties were tested, and one of them gave performances reaching 92% correct classification rate. We then tried to join those sets, but unexpectedly we found that it had a classification performance tightly resembling that of the worst 5-properties set. An interference effect due to the redundancy of information was supposed, accordingly to Zavaljevski et al. [9], that reported an interference phenomenon on another application of SVM.

We noticed that the law that connects the number of learning examples to the classification performance is logarithmic-like and so it reaches a plateau. The best number of learning examples was determined to be the lowest computational effort needed to reach the best performance. We determined that the best amount of examples was 2200 for both interacting and non-interacting pairs.

We then moved to another topic, and tested whether our SVM-based classifier could be used as a predictor of protein interactions in a yeast model. We considered three Saccharomyces cerevisiae nuclear proteins involved in the yeast cell cycle: Sic1, Cdc28 and Clb5 [3] [6] [7], and we looked for their partners in the whole Saccharomyces cerevisiae proteome.

First of all, we needed to adapt the SVM-based classifier to an SVM-based predictor. This issue can be addressed by the Bayes' formula, in which the number of expected interactors and the total number of proteins are crucial. In our case, we had nearly 10000 proteins to test, among them we expected 10 interactors for each of our model proteins, according to Marcotte et al. [5]. It is immediately clear that the method in our hands would have given only 1.125% of overall correct predictions, while 8% of the whole proteome would have been proposed as interacting. Moreover, to obtain a prediction percentage >50%, we will need to build a classifier with a performance of 99.9% and a false discovery rate of 0.1%.

Despite this, we applied our SVM-based predictor to the whole Saccharomyces cerevisiae proteome, in order to check if we could obtain the theoretical percentage of 8% of the whole proteome as being potentially interacting for each of our model proteins. Surprisingly, the SVM-based classifier recognized nearly all the proteome as being potentially interacting with the studied proteins. In particular, Sic1 was associated with 98.6%, Clb5 with 96.0% and Cdc28 with 99.4% of the Saccharomyces cerevisiae proteome.

One of the main reasons for this could be that our SVM-based predictor recognized real proteins from shuffled sequences, and did not distinguish interacting from non-interacting pairs of proteins. An evidence of this is that our predictor classified as non-interacting almost all the shuffled sequences we submitted (data not shown). Thus, in contrast to what Bock and Gough proposed, the assumption that shuffled sequences can well surrogate examples of lack of interaction appears here to be wrong.

Future research will be focused on further optimization of vector construction and on the generation of more reliable examples of lack of interaction. This could lead to the minimization of the false discovery rate, and finally to a reliable SVM-based method for the large scale prediction of protein interactions.

Acknowledgements

The authors wish to thank Dr. Elena Spoldi, Dr. Claudia Mondino and Dr. Ferdinando Di Cunto for the outstanding organization and scientific structure of the Master course in Bioinformatics, and Dr. Daniela Morale for the insightful discussions and help.

References

1. P. Baldi and S. Brunak, *Bioinformatics: A machine learning approach. 2nd edition*, MIT-Press (2002).
2. J.R. Bock and D.A. Gough, Predicting protein-protein interactions from primary structure *Bioinformatics*, 17 (2001) 455-460.
3. L. Dirick, T. Bšhm and K. Nasmyth, Roles and regulation of Cln-Cdc28 kinases at the start of the cell cycle of *Saccharomyces cerevisiae*, *EMBO J.*, 14 (1995) 4803-4813.
4. T. Joachims, B. Sholkopf, C. Burges, and A. Smola, Making large-scale SVM learning practical, *Advances in kernel methods - support vector learning* MIT-Press (1999).
5. E.M. Marcotte, M. Pellegrini, H.L. Ng, D.W. Rice, T.O. Yeates, and D. Eisenberg, Detecting protein function and protein-protein interactions from genome sequences, *Science*, 285 (1999) 751-753.
6. M.D. Mendenhall, W. Al-jumaily, and T.T. Nugroho, The Cdc28 inhibitor p40Sic1 *Prog. Cell Cycle Res.*, 1 (1995) 173-185.
7. E. Schwob and K. Nasmyth, CLB5 and CLB6, a new pair of of B cyclins involved in DNA replication in *Saccharomyces cerevisiae Genes Dev.*, 7 (1994) 1160-1175.
8. A. Valencia and F. Pazos, Computational methods for the prediction of protein interactions, *Curr. Opin. Struct. Biol.*, 12 (2001) 368-373.
9. N. Zavaljevski, F.J. Stevens, and J. Reifman, Support vector machines with selective kernel scaling for protein classification and identification of key amino acid positions, *Bioinformatics*, 8 (2002) 689-696.

Gene Selection Using Random Voronoi Ensembles

Francesco Masulli[1,2] and Stefano Rovetta[1,3]

[1] INFM-Istituto Nazionale per la Fisica della Materia
Via Dodecaneso 33, I-16146 Genova, Italy
[2] DI-Dipartimento di Informatica, Università di Pisa
Via F. Buonarroti 2, 56127 Pisa,Italy
masulli@di.unipi.it
[3] DISI-Dipartimento di Informatica e Scienze dell'Informazione
Università di Genova, Via Dodecaneso 35, 16146 Genova, Italy
rovetta@disi.unige.it

Abstract. In this paper we propose a flexible method for analyzing the relevance of input variables in high dimensional problems with respect to a given dichotomic classification problem. Both linear and non-linear cases are considered. In the linear case, the application of derivative-based saliency yields a commonly adopted ranking criterion. In the non-linear case, the method is extended by introducing a resampling technique and by clustering the obtained results for stability of the estimate. The method was preliminarly validated on the data published by T.R. Golub et al. on a study, at the molecular level, of two kinds of leukemia: Acute Myeloid Leukemia and Acute Lymphoblastic Leukemia (Science 5439-286, 531-537, 1999). Our technique indicates that, among the top 20 genes found by the final cluster analysis, 8 of the 50 genes listed in the original work feature a stronger discriminating power.

1 Introduction

In pattern recognition the problem of input variable selection has been traditionally focused on technological issues, e.g., performance enhancement, lowering computational requirements, and reduction of data acquisition costs. However, in the last few years, it has found many applications in basic science as a model selection and discovery technique, as shown by a rich literature on this subject, witnessing the interest of the topic especially in the field of bioinformatics. A clear example arises from DNA microarray technology that provides high volumes of data for each single experiment, yielding measurements for hundreds of genes simultaneously.

The problem statement is as follows. We are given a two-class labeled training sample $\left\{\mathbf{x} \in \Re^d\right\}$ of n observations. On the basis of the analysis of the decision surfaces, we want to assign an importance ranking to each individual input variable x_i with the aim of pointing out which input variables contribute most to the classification performance. This ranking can be used for the actual selection step.

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 302–307, 2003.

2 Linear Case

We assume that the normalization parameters for the data are known with sufficient statistical confidence. This is not always true, although in the case of microarray data accurate normalization is part of the standard preparation of data [3].

Let $r = g(\mathbf{x}) \in \Re$ be the discriminant or decision function, the discrimination criterion being $y = \mathrm{sign}(r)$. We assume a classifier $r = g()$ capable of good generalization performance. We adopted Support Vector Machines [5], which provide optimal solutions with a minimum of parameter tuning.

To analyze what input variables have the largest influence over the output function, we evaluate the derivatives of r with respect to each variable, to point out which one is responsible, for a given perturbation, of the largest contribution to sign inversion (which denotes switching from one class to another). This is the so-called *derivative-based saliency*. It is a way to assess the sensitivity of the output to variations in individual inputs, and has been used in many contexts.

Since we are interested in zero crossings, the analysis should be done in a neighborhood of the locus $\{\mathbf{x}|g(\mathbf{x}) = 0\}$, and of course requires $g()$ to be locally differentiable. The latter assumption is reasonable (obviously, on a local basis) since smoothing is required by the discrete sampling of data. However, the more complex the decision surface $\{\mathbf{x}|g(\mathbf{x}) = 0\}$, the smaller the regions in which this assumption holds around any given point.

Standard input selection criteria [14] justify the application of the above technique to linear classifiers, although some small-sample issues, such as the previous consideration on normalization, are often overlooked. This technique is described for instance in [4] and [16]. In the linear case, $r = g(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x}$ and $\nabla r = \mathbf{w}$. The single feature r discriminates between the two classes ($r > 0$ and $r < 0$). This feature is given by a linear combination of inputs, with weights $\mathbf{w}$. Thus, by sorting the inputs according to their weights, the "importance" ranking is directly obtained. In the analysis, we examine relative importances, $\mathbf{t} = \mathbf{w}/\max_i \{w_i\}$ (w_i components of $\mathbf{w}$). The approach can be justified from many perspectives: statistical, geometrical, or in terms of classification margin.

3 Non-linear Case

In the general, non-linear case, it is not possible to define a single ranking which holds in any region of the input space. A global approach employing statistical saliency evaluation based on data [13] requires large datasets which are not generally affordable, especially with DNA microarrays. Our approach involves partitioning the decision function $g()$, and performing local saliency estimates in sub-regions where $g()$ can be approximated with a linear decision function. To this end we apply a Voronoi tessellation [1], defined by drawing a set of points (termed Voronoi sites). Each Voronoi site defines a localized region in the data space, that is the locus of points in the data space for which that site is the nearest of all sites.

We can identify *empty regions* (with no data points); *homogeneous regions* (with points from one class only); *general regions* (with points from both classes).

Table 1. Random Voronoi Ensemble method for feature selection

```
(1) Establish a random Voronoi partitioning of the data space;
(2) Discard homogeneous and empty Voronoi cells;
(3) Compute a linear classifier on each remaining Voronoi cell;
(4) Store the obtained saliency vector along with the cell site;
(5) Repeat steps 1-4 until a sufficient number of saliency vectors
    are obtained;
(6) Perform joint clustering of the saliency vectors and cell
    centers;
(7) Retrieve cluster centers and use them as estimated local
    saliency rankings.
```

In the simplest approach, local linearization is made on the basis of an arbitrary partitioning (local subsampling) of the data space; to perform random partitioning, the Voronoi sites are drawn randomly. Homogeneous and empty regions are then discarded. Within each general region, a local linear classifier is built. Thus a single random Voronoi tessellation defines a set of classifiers, each performing a local analysis.

This basic method has several drawbacks: lower confidence of classifiers (trained on sub-samples); artifacts from Voronoi borders superposed to the separating surface; lack of a criterion for the number of regions; need to combine saliency rankings obtained in different regions.

To address all these issues, we propose a method we term "Random Voronoi Ensemble" since it is based on random Voronoi partitions as described above; these partitions are replicated by resampling, so the method actually uses an ensemble of random Voronoi partitions. Ensemble methods are described for instance in [6]. The method is outlined in Tab. 1.

Since a purely random partition is likely to generate many empty regions, the Voronoi sites are initialized by a rough vector quantization step, to ensure that sites are placed within the support of the data set. Subsequent random partitions are obtained by perturbation of the initial set of points. Within each Voronoi region, a linear classification is performed using Support Vector Machines (SVM) with a linear kernel.

To build a classifier ensemble, a resampling step is applied by replicating the basic procedure. The subsequent clustering step acts as the integrator, or arbiter: its role is to integrate the individual outcomes and to output a global response. It results in a set of "prototypical" saliency patterns, corresponding to different local classification criteria. These patterns are "prototypical" in the same sense as the centroids of k-means partitions [7] are representative of the respective clusters.

Resampling helps in smoothly covering the whole data set and, by averaging, contributes to the stability of the outcomes. Unfortunately, it is difficult to obtain theoretical guidelines on how many replications are required. Theoretical results on stability of Voronoi neighbors are available only for low dimensions [17], and typically cannot be generalized to higher dimensions.

To integrate the outcomes of the ensemble, we use the Graded Possibilistic Clustering technique to ensure an appropriate level of outlier insensitivity ([12]). This technique

Table 2. Relevant inputs for the Leukemia data

Gene description	Gene accession number	Correlated class	Sign of saliency
GPX1 Glutathione peroxidase 1	Y00787	AML	$-$
PRG1 Proteoglycan 1, secretory granule	X17042	AML	$-$
CST3 Cystatin C (amyloid angiopathy and cerebral hemorrhage)	M27891	AML	$-$
Major histocompatibility complex enhancer-binding protein mad3	M69043	AML	$-$
Interleukin 8 (IL8) gene	M28130	AML	$-$
Azurocidin gene	M96326	AML	$-$
MB-1 gene	U05259	ALL	$+$
ADA Adenosine deaminase	M13792	ALL	$+$

is a generalization of the Possibilistic approach to fuzzy c-Means clustering of Keller and Krishnapuram [10][11] in which cluster membership can be constrained to sum to 1 (as in the standard fuzzy clustering approaches [2]), unconstrained (as in the Possibilistic approach), or partially constrained. Partial constraints allow the implementation of several desirable properties, among which there is a user-selectable degree of outlier insensitivity. The number of cluster centers is assessed by applying a Deterministic Annealing schedule [15] to the parameter β, which directly influences the width of clusters and is a measure of the "resolution" of the method.

4 Experimental Results

The method was preliminarily validated on the data published in [8], a study, at the molecular level, of two kinds of leukemia, Acute Myeloid Leukemia (AML) and Acute Lymphoblastic Leukemia (ALL). Data were obtained by an Affymetrics high-density oligonucleotide microarray, revealing the expression level of 6817 human genes plus controls. Observations refer to 38 bone marrow samples, used as a training set, and 34 samples from different tissues (the test set).

In this experiment, we used only the training data to discriminate ALL from AML. Classes are in the proportion of 27 ALL and 11 AML observations. Parameters: 4 Voronoi sites; β from 0.1 down to 0.01 in 10 steps, exponential decay law; uniform perturbation of maximum amplitude 0.5, independent on each input coordinate; 100 perturbations resulting in 400 random partitions of which 61% useful (general).

Results are summarized in Tab. 2, comparing the most important genes with those obtained by the original authors. Genes that were indicated both in [8] and by our technique are listed with the sign of their saliency value. Our technique indicates that, among the top 20 genes found by the final cluster analysis, 8 of the 50 genes listed in the original work feature a stronger discriminating power. We restrict the analysis to few genes, since a good cluster validation step is not included in the method yet. However, the results may indicate that not all of the genes found by Golub et al. contribute to the actual discrimination to the same extent.

5 Discussion and Conclusions

We have described a flexible method for analyzing the relevance of input variables in high dimensional problems. The method, which is in an early phase of development, has nevertheless shown the ability to tackle dichotomic problems even in the presence on non-linear separating surfaces. Its behavior has also been validated by comparing the results obtained on a real microarray data set with those published by the original authors.

We can underline some issues can we plan to addressed in the future work. For example, the number of Voronoi sites is an important parameter, since it is related to the scale of the tessellation (size of cells). Large cells will tend to contain segments of the separating surface which are difficult to linearize, while small cells will lead to excessively small data subset cardinality, and therefore to low generalization ability. The selection of the number of sites can be based on estimates of the problem complexity such as those proposed in [9], which are based on geometrical characterization of the data rather than the more usual statistical or information-theoretical consideration. However these must be combined with estimates of generalization to account for the trade-off outlined above.

Moreover, we have based our analysis on decision surfaces. This implies that the most natural setting of the problem is given by dichotomic (two-class) cases. Any polychotomic problem can be stated as a set of dichotomic problems, and this is what is usually done when using Support Vector Machines for classification. However a possible development of the method could imply the analysis of multi-class decision criteria, such as soft-max.

We point out that the proposed method for feature selection is especially well suited to parallel implementation at many levels, since the various steps can be pipelined, the subsamples can be processed in parallel, and the Voronoi resampling and clustering phases themselves can be implemented in parallel. All these steps involve very reduced communication. For instance, parallel resampling can be implemented by completely independent random partitions, and communication of subsamples for parallel analysis can be obtained by passing the index of selected patterns. Therefore a Beowulf-type workstation cluster may be proficiently used with limited adaptation effort.

The technique to generate the random perturbations themselves can be also optimized, to reduce the number of empty/homogeneous regions, since the data sets are expected to be extremely sparse in the data space. Perturbations can therefore be limited to a subspace, for instance by constraining them to the directions spanned by the versors of the data patterns (e.g., referring to the leukemia data, this is a basis which spans a 38-dimensional subspace of the 6817-dimensional data space).

Acknowledgements

Work funded by the Italian National Institute for the Physics of Matter (INFM) and by the Italian Ministry of Education, University and Research (“Prin2002”).

References

1. F. Aurenhammer, Voronoi diagrams-a survey of a fundamental geometric data structure, *ACM Computing Surveys*, 3 (23) (September 1991), 345-405.
2. J.C. Bezdek, *Pattern recognition with fuzzy objective function algorithms*, Plenum, New York (1981).
3. M. Bilban, L.K. Buehler, S. Head, G. Desoye, V. Quaranta, Normalizing DNA microarray data, *Curr Issues Mol Biol*, 4 (2) (2002) 57-64.
4. J. Brank, M. Grobelnik, N. Milic-Frayling, D. Mladenic, Feature selection using linear support vector machines, Tech. Rep. MSR-TR-2002-63, Microsoft Research (June 2002).
5. N. Cristianini, J. Shawe-Taylor, *An Introduction to Support Vector Machines*, Cambridge Univ. Press (2000).
6. T.G. Dietterich, Machine-learning research: Four current directions *The AI Magazine* 4 (18) (Winter 1998) 97-136.
7. R.O. Duda, P.E. Hart, *Pattern Classification and Scene Analysis*, John Wiley and Sons, New York (USA) 1973.
8. T.R. Golub *et al.*, Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring, Science 5439 (286) (1999) 531-537.
9. Tin Kam Ho and Mitra Basu, "Complexity measures of supervised classification problems", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 3, pp. 289–300, March 2002.
10. R. Krishnapuram, J.M. Keller, A possibilistic approach to clustering, *IEEE Trans. on Fuzzy Systems*, 2 (1) (May 1993) 98-110.
11. R. Krishnapuram, J.M. Keller, The possibilistic c-Means algorithm: insights and recommendations, *IEEE Trans. on Fuzzy Systems*, 3 (4) (August 1996) 385-393.
12. F. Masulli, S. Rovetta, Soft transition from probabilistic to possibilistic fuzzy clustering, DISI Technical Report DISI-TR-03-02, Department of Computer and Information Sciences, University of Genoa, Italy (April 2002).
    URL: http://www.disi.unige.it/person/RovettaS/research/techrep/DISI-TR-02-03.ps.gz.
13. C. Moneta, G. Parodi, S. Rovetta, R. Zunino, Automated diagnosis and disease characterization using neural network analysis, in *Proc. of the 1992 IEEE Int. Conf. on Systems, Man and Cybernetics*, Chicago USA (October 1992) 123-128.
14. B.D. Ripley, *Pattern recognition and neural networks*, Cambridge Univ. Press (1996).
15. K. Rose, Deterministic annealing for clustering, compression, classification, regression, and related optimization problems, *Proceedings of IEEE*, 11 (86) (November 1998) 2210-2239.
16. V. Sindhwani, P. Bhattacharya, S. Rakshit, Information theoretic feature crediting in multiclass support vector machines, in *1st SIAM Int. Conf. on Data Mining, Chicago, USA*. (April 2001) SIAM, Philadelphia.
17. F. Weller, Stability of Voronoi neighborhood under perturbations of the sites, in *Proc. of Ninth Canadian Conf. on Computational Geometry*, Kingston, Ontario, Canada (August 1997).

QSAR/QSPR Studies by Kernel Machines, Recursive Neural Networks and Their Integration

Alessio Micheli[1], Filippo Portera[2], and Alessandro Sperduti[2]

[1] University of Pisa, Department of Computer Science
[2] University of Padova, Department of Pure and Applied Mathematics

Abstract. We present preliminary results on a comparison between Recurrent Neural Networks (RecNN) and an SVM using a string kernel on QSPR/QSAR problems. In addition to this comparison, we report on a first attempt to combine RecNN with SVM.

1 Introduction

Recursive Neural Networks (RecNN) (see, for example, [8]) have been proposed as an extension of standard neural networks to the adaptive processing of structured data, such as chemical compounds. Especially in Chemistry, RecNN have been successfully applied to QSPR (quantitative structure-property relationships) and QSAR (quantitative structure-activity relationships) studies (see, for example, [1]). Recently, kernels for structured data have been proposed and some interesting results on their application is emerging. The aim of this paper is to start a comparison of the above approaches on specific QSPR/QSAR applications. In particular, a Recursive Cascade Correlation network is considered as representative for the RecNN, while for structured kernels we considered a string kernel [9]. For both approaches we report preliminary results obtained for QSPR on Alkanes (predicting the boiling point) and QSAR of a class of Benzodiazepines (predicting the non-specific activity (affinity) towards the Benzodiazepine/GABA$_A$ receptor). In addition to this comparison, we report on a first trivial attempt to combine RecNN with SVM, where SVM is used to perform a post-processing of the internal representations of RecNN.

2 QSPR/QSAR Tasks

Here we describe the QSPR/QSAR regression tasks we have considered.

QSPR Task: Alkanes
The problem consists in the prediction of the boiling point for a group of acyclic hydrocarbons (alkanes). The prediction task is well characterized for this class of compounds, since the boiling points of hydrocarbons depend upon molecular size and molecular shape, and vary regularly within a series of compounds, which

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 308–315, 2003.

means that there is a clear correlation between molecular shape and boiling point. Moreover, the relatively simple structure of these compounds[1] is amenable to very compact representations such as topological indexes and/or vectorial codes, which are capable of retaining the relevant information for prediction. For these reasons, standard multilayer feed-forward networks using “ad hoc” representations yield very good performances.

In order to perform a comparison with the considered methods, we decided to use as reference point the work described in [2] which uses standard multilayer feed-forward networks. For this problem, RecCC has been proved to be competitive with respect to *ad-hoc* techniques (see [1]). In fact, the obtained results compares favorably versus the approach proposed by Cherqaoui et al. [2], which represented the *state-of-the-art* results. The data set used in [2] comprised all the 150 alkanes with 10 carbon atoms: 135 compounds were used for training and 15 compounds for test[2].

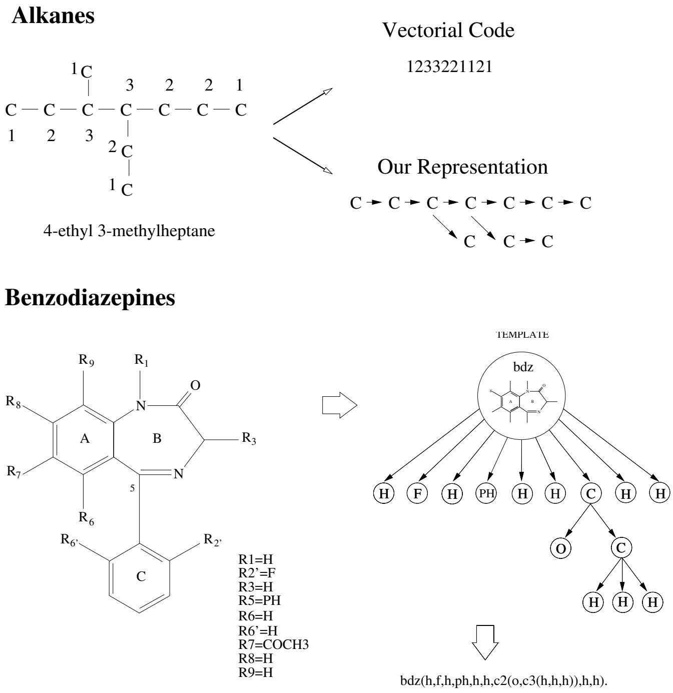


Fig. 1. Example of representation for an alkane and a benzodiazepine.

[1] No explicit representation of the atoms and bound type is required.

[2] It must be noted that Cherqaoui et al. use a vectorial code representation of alkanes based on the n-tuple code for the encoding of trees (see Fig. 1). So they represent each alkane as a 10 numerical components vector with the last components filled by zeros when the number of atoms of the compound was less than 10. The single component encodes the number of bounds of the corresponding carbon node.

QSAR Task: Benzodiazepines
The ability of predicting the biological activity of chemical compounds belonging to classes of therapeutical interest constitutes the major aspect of the drug design. Benzodiazepines, for example, has been extensively studied since the 70s, as this class of compounds plays the major role in the field of minor tranquilizer, and several QSAR studies have been carried out aiming at the prediction of the non-specific activity (affinity) towards the Benzodiazepine/GABA$_A$ receptor. The affinity can be expressed as the inverse of the logarithm of the drug concentration C (Mol./liter) able to give a fixed biological response[3].

A group of Benzodiazepines (Bz) (classical 1,4-benzodiazepin-2-ones) previously analyzed by Hansch et al. [3] through the traditional QSAR equations, was analyzed. The total number of molecules was 77, of which 5 are used as test set. The analyzed molecules present a common structural aspect given by the Benzodiazepine ring (see Fig. 1) and they differ each other because of a large variety of substituents at the positions showed in Fig. 1.

3 Neural Networks for Structures and the Tasks

Recursive neural networks (RecNN) [8] are neural networks able to perform mappings from a set of directed ordered acyclic graphs (DOAGs) (with labeled nodes) $\mathcal{I}^{\#}$ to the set of real vectors. Specifically, the class of functions which can be realized by RecNN can be characterized as the class of functional graph transductions $\mathcal{T} : \mathcal{I}^{\#} \to \mathbb{R}^k$, which can be represented in the following form $\mathcal{T} = g \circ \hat{\tau}$, where $\hat{\tau} : \mathcal{I}^{\#} \to \mathbb{R}^m$ is the *encoding* function and $g : \mathbb{R}^m \to \mathbb{R}^k$ is the *output* function. Specifically, given a DOAG $\boldsymbol{Y}$, $\hat{\tau}$ is defined recursively as

$$\hat{\tau}(\boldsymbol{Y}) = \begin{cases} \boldsymbol{0} \text{ (the null vector in } \mathbb{R}^m) & \text{if } \boldsymbol{Y} = \xi \\ \tau(s, \boldsymbol{Y}_s, \hat{\tau}(\boldsymbol{Y}^{(1)}), \ldots, \hat{\tau}(\boldsymbol{Y}^{(o)})) & \text{otherwise} \end{cases} \tag{1}$$

where a (*stationary*) τ can be defined as $\tau : \mathbb{R}^n \times \underbrace{\mathbb{R}^m \times \cdots \times \mathbb{R}^m}_{\text{o times}} \to \mathbb{R}^m$, $\mathbb{R}^n$ is the label space, the remaining domains represent the encoded subgraphs spaces up to the maximum out-degree of the input domain $\mathcal{I}^{\#}$, o is the maximum out-degree of DOAGs in $\mathcal{I}^{\#}$, $s = source(\boldsymbol{Y})$, $\boldsymbol{Y}_s$ is the label attached to the source of $\boldsymbol{Y}$, and $\boldsymbol{Y}^{(1)}, \ldots, \boldsymbol{Y}^{(o)}$ are the subgraphs pointed by s. A possible neural realization for τ is $\tau(\boldsymbol{l}, \boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(o)}) = \boldsymbol{F}(\boldsymbol{W}\boldsymbol{l} + \sum_{j=1}^{o} \widehat{\boldsymbol{W}}_j \boldsymbol{x}^{(j)} + \boldsymbol{\theta})$, where $\boldsymbol{F}_i(\boldsymbol{v}) = f(v_i)$ (sigmoidal function), $\boldsymbol{l} \in \mathbb{R}^n$ is a label, $\boldsymbol{\theta} \in \mathbb{R}^m$ is the bias vector, $\boldsymbol{W} \in \mathbb{R}^{m \times n}$ is the weight matrix associated with the label space, $\boldsymbol{x}^{(j)} \in \mathbb{R}^m$ are the vectorial codes obtained by the application of the encoding function $\hat{\tau}$ to the subgraphs $\boldsymbol{Y}^{(j)}$, and $\widehat{\boldsymbol{W}}_j \in \mathbb{R}^{m \times m}$ is the weight matrix associated with the jth subgraph space.

Concerning the output function g, it can be defined as a map $g : \mathbb{R}^m \to \mathbb{R}^k$. Specifically, in this paper, we use linear output neurons: $g(\boldsymbol{x}) = \boldsymbol{m}^T \boldsymbol{x} + \beta$, where $\boldsymbol{m} \in \mathbb{R}^m$ and β is the output threshold.

[3] In order to characterize the fixed response, the drug concentration able to give half of the maximum response (IC$_{50}$) is commonly used.

The architecture we have adopted for the experiments reported in this paper is Recursive Cascade Correlation (RecCC), which is described in detail in [8].

Molecular Structure Representation

The main requirement for the use of the RecNN consists in finding a representation of molecular structures in terms of DOAGs. The candidate representation should retain the detailed information about the structure of the compound, atom types, bond multiplicity, chemical functionalities, and finally a good similarity with the representations usually adopted in chemistry. The main representational problems are: to represent cycles, to give a direction to edges, to define a total order over the edges. An appropriate description of the molecular structures analyzed in this work is based on a labeled tree representation. In fact, concerning the first problem, since cycles are absent in the examined alkanes and they mainly constitute the common shared template of the benzodiazepines compounds, it is reasonable to represent a cycle (or a set of connected cycles, as in the benzodiazepines case) as a single node where the attached label carries information about its chemical nature. The second problem was solved by the definition of a set of rules based on the I.U.P.A.C. nomenclature system[4]. Finally, the total order over the edges follows a set of rules mainly based on the size of the sub-compounds. Examples of representations for alkanes and benzodiazepines are shown in Fig. 1.

4 A String Kernel for Trees

We start defining some notation to provide a kernel function for strings that will be applied to a string representation of chemical compounds (see Fig. 1). Let $\mathcal{A}$ a finite set of *characters* called *alphabet*. A *string* is a element $x \in \mathcal{A}^k$ for $k = 0, 1, 2, \ldots$. Let $|x|$ the length of x and let $v, x \in \mathcal{A}^k$, we say that $v \sqsubseteq x$ if v is a substring of x. Let $\mathrm{num}_y(x)$ the number of occurrences of y in x. Let $\bar{\phi}(x)$ a function that maps $x \in \mathcal{A}^k$ in a feature space $\mathbb{N}^d$ with $d = \sum_{i=1}^{k} |\mathcal{A}|^i$, where each dimension represents a particular string of $\mathcal{A}^k$. Defining $\phi(x)_i = \mathrm{num}_{s_i}(x)$, we can consider a dot product between vectors on this feature space [9]:

$$k(x, y) = \sum_{s_i \in \mathcal{A}^*} \phi(x)_i \phi(y)_i |s_i| \qquad (2)$$

Note that, by definition, this is a kernel function since it represents a dot product in $\mathbb{N}^d \times \mathbb{N}^d$. Therefore, the kernel function $k(x, y)$ depends on the co-occurrences of substrings s_i both in x and in y. A match is then weighted with the length of the common substring s_i. The function (2) can be computed in time $O(|x| + |y|)$ building the matching statistics with the Suffix Tree algorithm [9].

Having a kernel function that is able to compare variable length strings is helpful when input patterns are trees. In fact, given a tree T let $T' \sqsubseteq T$ if T'

[4] The root of a tree representing a benzodiazepine is determined by the common template, while the root for alkanes is determined by the I.U.P.A.C. nomenclature system.

is a subtree of T. A possible kernel for trees is based on matching subtrees and has been proposed in [7]: $k(T_1, T_2) = \sum_{t_1 \sqsubseteq T_1, t_2 \sqsubseteq T_2} w_{t_1} \delta_{t_1,t_2}$, where $\delta_{t_1,t_2} = 1$ if $t_1 = t_2$ and 0 otherwise. A possible alternative is a string representation of trees and the use of the string kernel in eq. (2). Let T be a tree with l nodes and suppose that $\forall n \in T$, label(n) is a label associated to node n. Then for each node n in T assume that there exists a lexicographic order between the children nodes of n based on label(n). For each leaf node n, define tag(n) = label(n), for each internal node i let tag(n) = [label(n)tag$(n1)\ldots$tag(n_c)] following the order defined on the children nodes of n. Thus, it can be shown [9] that the tag of the root node of T is a unique identifier for T and it can be computed in $O(l \log_2 l)$ time complexity.

5 Experiments

We performed experiments on 2 splits of the Alkanes dataset, called *a2* and *a9*, and the Benzodiazepines dataset called *Bz*. For these datasets, in Table (1) we have reported the results obtained by two instances of RecCC, denoted by the symbols a and b.

In the same table we have reported preliminary results of the string kernel as defined in eq. (2), calibrated on the test set to understand the potentiality of the approach that relies on a string representation of a molecule. In particular, the symbolic tree representation of the molecule used to generate the input for RecCC (by substituting symbols with binary vectors), is transformed into a string by the technique described in Section 4. These experiments were performed by integrating the string kernel in SVMLight 5.0 ([5]), that supports SVM regression.

We then explored the possibility to combine RecCC with a SVM. The first trivial attempt was to implement the output function $g()$ of RecCC by a SVM. Thus, we used the activity of the hidden units of an *already trained* RecCC (either a or b) as the input pattern for a SVM for regression with a RBF kernel. For each of the three problem instances we used both RecCC a and b. For this experiment, besides to SVMLight 5.0, we also used BSVM 2.04[5] [4]. We applied a 3-fold cross validation in the training set to fix the hyper-parameters C and γ and ϵ (the regression tube width) of the learning task. These results are denoted in Table 1 with *fair*. In addition, we report the results obtained by calibrating the learning algorithm on the test set.

As a final experiment of combination, we considered the same approach described above while using a dataset where each molecule is represented by the concatenation of the hidden activity of RecCCa and RecCCb. The results are shown in Table 2.

Table 3 reports a comparison of the results obtained with all the algorithms described in this paper. We did not report the results obtained by the concate-

[5] BSVM relies on a slight modification of the SVM primal quadratic model [6] that corresponds to a dual quadratic problem without general linear constraints. This explains the different results obtained with SVMLight and BSVM.

Table 1. Results on the *a* and *b* splits of datasets *a2*, *a9*, and *Bz*.

Problem	*max abs test err*	*mean abs test err*	*C*	γ	*epsilon-tube* [max abs train err]
a2(a) RecCC	0.0774	0.02714	max abs train err ≤ 0.080		
a2(a) SVMLight fair	0.0605	0.02426	1E4	0.001	0.015 [0.139]
a2(a) BSVM fair	0.0663	0.02532	1E4	0.001	0.015 [0.135]
a2(a) SVMLight	0.0587	0.01987	2E4	0.001	0.019 [0.143]
a2(a) BSVM	0.0590	0.02293	2E4	0.001	0.019 [0.110]
a2(b) RecCC	0.0647	0.02638	max abs train err ≤ 0.080		
a2(b) SVMLight fair	0.0583	0.02147	1E4	0.001	0.015 [0.076]
a2(b) BSVM fair	0.0615	0.02205	1E4	0.001	0.015 [0.074]
a2(b) SVMLight	0.0672	0.01919	2E4	0.001	0.0252 [0.074]
a2(b) BSVM	0.0689	0.01909	2E4	0.001	0.0252 [0.074]
a2 String Kernel	0.2879	0.06794	1	0.001	1.0E-8 [0.107]
a9(a) RecCC	0.3256	0.05112	max abs train err ≤ 0.080		
a9(a) SVMLight fair	0.3199	0.04847	1E4	0.001	0.01 [0.083]
a9(a) BSVM fair	0.3255	0.04822	1E4	0.001	0.01 [0.083]
a9(a) SVMLight	0.322	0.04809	1E4	0.001	0.009 [0.086]
a9(a) BSVM	0.3269	0.04844	1E4	0.001	0.009 [0.085]
a9(b) RecCC	0.2224	0.04819	max abs train err ≤ 0.080		
a9(b) SVMLight fair	0.2111	0.04744	1E5	0.001	0.048 [0.065]
a9(b) BSVM fair	0.2164	0.04804	1E5	0.001	0.048 [0.075]
a9(b) SVMLight	0.1846	0.04137	1E5	0.001	0.004 [0.099]
a9(b) BSVM	0.1766	0.04282	1E5	0.001	0.004 [0.123]
a9 String Kernel	0.1672	0.05070	1	0.001	1.0E-6 [0.476]
Bz(a) RecCC	0.06460	0.02542	max abs train err ≤ 0.040		
Bz(a) SVMLight fair	0.09019	0.03466	560	0.001	1E-9 [0.058]
Bz(a) BSVM fair	0.09048	0.03452	560	0.001	1E-9 [0.057]
Bz(a) SVMLight	0.05067	0.02320	9500	0.001	2E-3 [0.061]
Bz(a) BSVM	0.04612	0.02160	9500	0.001	2E-3 [0.062]
Bz(b) RecCC	0.08014	0.02307	max abs train err ≤ 0.040		
Bz(b) SVMLight fair	0.09075	0.03083	790	0.001	2E-5 [0.054]
Bz(b) BSVM fair	0.08956	0.03102	790	0.001	2E-5 [0.056]
Bz(b) SVMLight	0.07343	0.01986	5E4	0.001	2E-10 [0.028]
Bz(b) BSVM	0.07213	0.02220	5E4	0.001	2E-10 [0.033]
Bz String Kernel	0.08628	0.03770	0.1	0.001	1.5E-5 [0.014]

nation of the hidden activity of RecCC*a* and RecCC*b* because it involves two indipendent RecCC models and so it would not be fair to compare these results with the ones obtained by a single model.

Preliminary results obtained by using a tree kernel based on the spectrum of a tree (i.e., number of shared subtrees), seems to give improved results with respect to the adoption of the string kernel. The tree kernel seems to be especially effective for dataset *a9*.

Table 2. Results obtained with feature concatenation of splits *a* and *b* on datasets *a2*, *a9*, and *Bz*.

Problem	*max abs*	*mean abs*	*C*	γ	*epsilon-tube*
	test err	*test err*			[max abs train err]
a2 RecCC Best	0.0647	0.02638	max abs train err $\leq$ 0.080		
a2 SVMLight fair	0.0719	0.02748	18	0.001	0.015 [0.077]
a2 BSVM fair	0.0683	0.02844	18	0.001	0.015 [0.078]
a2 SVMLight	0.0324	0.01974	39	0.001	0.0031 [0.104]
a2 BSVM	0.0316	0.01842	39	0.001	0.0031 [0.104]
a9 RecCC Best	0.2224	0.04819	max abs train err $\leq$ 0.080		
a9 SVMLight fair	0.1633	0.05102	1E5	0.001	0.008 [0.009]
a9 BSVM fair	0.1593	0.05102	1E5	0.001	0.008 [0.009]
a9 SVMLight	0.1853	0.04618	96	0.001	0.0385 [0.045]
a9 BSVM	0.1848	0.04625	96	0.001	0.0385 [0.039]
Bz RecCC	0.08014	0.02307	max abs train err $\leq$ 0.040		
Bz SVMLight fair	0.07335	0.03814	140	0.001	0.008 [0.070]
Bz BSVM fair	0.07178	0.03802	140	0.001	0.008 [0.071]
Bz SVMLight	0.02386	0.01228	4.7E4	0.001	1E-6 [0.014]
Bz BSVM	0.01851	0.00835	4.7E4	0.001	1E-6 [0.020]

Table 3. Summary of the results obtained with several algorithms on datasets *a2*, *a9*, and *Bz*.

Problem	*max abs*	*mean abs*	*C*	γ	*epsilon-tube*
	test err	*test err*			[max abs train err]
a2 RecCC Best	0.0647	0.02638	max abs train err $\leq$ 0.080		
a2 RCC+SVM Best	0.0689	0.01909	2E4	0.001	0.0252 [0.074]
a2 String Kernel	0.2879	0.06794	1	0.001	1.0E-8 [0.107]
a9 RecCC Best	0.2224	0.04819	max abs train err $\leq$ 0.080		
a9 RCC+SVM Best	0.1846	0.04137	1E5	0.001	0.004 [0.099]
a9 String Kernel	0.1672	0.05070	1	0.001	1.0E-6 [0.476]
Bz RecCC	0.08014	0.02307	max abs train err $\leq$ 0.040		
Bz RCC+SVM Best	0.07343	0.01986	5E4	0.001	2E-10 [0.028]
Bz String Kernel	0.08628	0.03770	0.1	0.001	1.5E-5 [0.014]

6 Conclusions

The aim of this paper was to start a comparison of Recursive Neural Networks (RecCC) versus kernel for structures (string kernel). The results for the string kernel seems to be worst with respect to the ones obtained by the Recursive Cascade Correlation network, however, these results, especially for the string kernel, are still very preliminary and need to be assessed. Moreover, a tree kernel will be considered in the future. More promising seems to be the preliminary results obtained by the simple attempts to combine RecNN with SVM, where SVM is used to perform a post-processing of the internal representations of the neural network for structure.

References

1. A.M. Bianucci, A. Micheli, A. Sperduti, and A. Starita. Application of cascade correlation networks for structures to chemistry. *Journal of Applied Intelligence (Kluwer Academic Publishers)*, 12:117–146, 2000.
2. D. Cherqaoui and D. Villemin. Use of neural network to determine the boiling point of alkanes. *J. Chem. Soc. Faraday Trans.*, 90(1):97–102, 1994.
3. Dimitra Hadjipavlou-Litina and Corwin Hansch. Quantitative structure-activity relationships of the benzodiazepines. a review and reevaluation. *Chemical Reviews*, 94(6):1483–1505, 1994.
4. C. Hsu and C. Lin. A comparison of methods for multi-class support vector machines, 2001.
5. T. Joachims. Text categorization with support vector machines: learning with many relevant features. In *Proceedings of ECML-98, 10th European Conference on Machine Learning*, pages 137–142, 1998.
6. O. L. Mangasarian and D. R. Musicant. Successive overrelaxation for support vector machines. *IEEE-NN*, 10(5):1032, September 1999.
7. M.Collins and N.Duffy. Convolution kernels for natural language, 2001.
8. A. Sperduti and A. Starita. Supervised neural networks for the classification of structures. *IEEE Transactions on Neural Networks*, 8(3):714–735, 1997.
9. S.V.N. Vishwanathan and Alexander J. Smola. Fast kernels for string and tree matching. In *NIPS 2002 Proceeedings*, 2003.

An Application of Low Bias Bagged SVMs to the Classification of Heterogeneous Malignant Tissues

Giorgio Valentini[1,2]

[1] DSI, Dipartimento di Scienze dell'Informazione
Università degli Studi di Milano, 20135 Milano
[2] INFM, Istituto Nazionale per la Fisica della Materia, 16146 Genova, Italy
valenti@disi.unige.it

Abstract. DNA microarray data are characterized by high-dimensional and low-sized samples, as only few tens of DNA microarray experiments, involving each one thousands of genes, are usually available for data processing. Considering also the large biological variability of gene expression and the noise introduced by the bio-technological machinery, we need robust and variance-reducing data analysis methods. To this purpose, we propose an application of a new ensemble method based on the bias–variance decomposition of the error, using Support Vector Machines (SVMs) as base learners. This approach, that we named *Lo*w bias *bag*ging (*Lobag*), tries to reduce both the bias and the variance components of the error, selecting the base learners with the lowest bias, and combining them through bootstrap aggregating techniques. We applied Lobag to the classification of normal and heterogeneous malignant tissues, using DNA microarray gene expression data. Preliminary results on this challenging two-class classification problem show that Lobag, in association with simple feature selection methods, outperforms both single and bagged ensembles of SVMs.

1 Introduction

DNA hybridization microarrays [6] supply information about gene expression through measurements of mRNA levels of a large amount of genes in a cell. Typically thousands of genes are used and analyzed for each microarray experiment.

Several supervised methods have been applied to the analysis of cDNA microarrays and high density oligonucleotide chips [2,7,10]. In particular, ensembles of learning machines have been applied to gene expression data analysis, as they can reduce the variance due to the low cardinality of the available training sets, and the bias due to specific characteristics of the learning algorithm. Indeed, in recent works, combinations of binary classifiers (one-versus-all and all-pairs) and Error Correcting Output Coding (ECOC) ensembles of MLP, as well as ensemble methods based on resampling techniques, such as bagging and boosting, have been applied to the analysis of DNA microarray data [5,12,16].

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 316–321, 2003.

In this paper we propose to apply a new ensemble method, that we named *Lobag*, that is *Lo*w bias *bag*ging, to the classification of normal and heterogeneous malignant tissues, using DNA microarray data. In order to reduce the very high dimensionality of the data, a simple feature selection method is applied [8], and the classification results are compared with single and bagged ensembles of SVMs.

2 Lobag: Low Bias Bagged SVMs

The basic idea behind Lobag consists in selecting low-bias SVMs through an analysis of bias–variance decomposition of the error [4], and then in combining them through bootstrap aggregating techniques in order to reduce both the bias and the variance components of the error. The selection of low bias SVMs, is performed by means of a relatively inexpensive estimate of bias and variance, according to Breiman's out-of-bag techniques [1].

The high level algorithmic scheme of Lobag is the following:

1. Estimate bias-variance decomposition of the error for different SVM models
2. Select the SVM model with the lowest bias
3. Perform bagging using as base learner the SVM with the estimated lowest bias.

The pseudocode of Lobag, together with the description of the procedures applied to estimate the bias and variance components of the error are described in detail in [14]. An implementation of Lobag has been developed using and extending the *NEURObjects* C++ library [15].

3 Gene Selection

We used a simple filter method, that is a gene selection method applied before and independently of the induction algorithm, originally proposed in [8]. The mean gene expression value across all the positive (μ_+) and negative (μ_-) examples are computed separately for each gene, together with their corresponding standard deviations (σ_+ and σ_-).

Then the following statistic (a sort of signal-to-noise ratio) c_i is computed:

$$c_i = \frac{\mu_+ - \mu_-}{\sigma_+ + \sigma_-} \tag{1}$$

The larger is the distance between the mean values with respect to the sum of the spread of the corresponding values, more related is the gene to the discrimination of the positive and negative classes. Then the genes are ranked according to their c_i value, and the first and last m genes are selected.

With the *GCM* data set we applied a permutation test to automatically select a set of marker genes:

1. Calculate for each gene the signal-to-noise ratio (eq. 1)
2. Perform a gene-specific random permutation test:

(a) Generate n random permutations of the class labels computing each time the signal-to-noise ratio for each gene.
(b) Select a p significance level (e.g. $0 < p < 0.1$)
(c) If the randomized signal-to-noise ratio is larger than the actual one in less than $p \cdot n$ random permutations, select that gene as significant for discrimination at p significance level.

This simple method has a $\mathcal{O}(nd)$ time complexity, where n is the number of examples and d the number of features (genes). Moreover the permutation test is distribution independent: no assumptions about the functional form of the gene distribution are supposed.

4 Data Set and Experimental Set-Up

We used the *GCM* data set obtained from the Withehead Institute, Massachusetts Institute of Technology Center for Genome Research [11]. It is constituted of 300 human normal and tumor tissue specimens spanning 14 different malignant classes. In particular it contains 280 samples, 190 tumoral pertaining to 14 different classes, plus other 20 poorly differentiated tumor samples and 90 normal samples. We grouped together the 14 different tumor classes and the poorly differentiated tumor samples to reduce the multi-class classification problem to a dichotomy in order to separate normal from malignant tissues.

The 300 samples sequentially hybridized to oligonucleotide microarrays contain a total of 16063 probe sets (genes or ESTs) and we performed a stratified random splitting of these data in a training and test set of equal size. We preprocessed raw data using thresholding, filtering and normalization methods as suggested in [11]. Performances of Lobag ensembles of SVMs were compared with standard bagging and with single SVMs, using subsets of genes selected through the simple feature-filtering method described in Sect. 3.

5 Results and Discussion

Using the above filter selection method, we selected 592 genes correlated with tumoral examples ($p = 0.01$) (set A) and about 3000 genes correlated with normal examples ($p = 0.01$) (set B). Then we used the genes of set A and the 592 genes with highest signal-to-noise ratio values of set B to assemble a selected set composed by 1184 genes.

The results of the classifications with single SVMs, with and without gene selection are summarized in Tab. 1. A statistical significant increment in accuracy at 0.05 confidence level (McNemar test [3]) is registered using SVMs trained with the selected subset of genes. Note that polynomial kernels without feature selection fail to classify normal from malignant tissues.

In light of these results, with bagged and lobag ensembles of SVMs we considered only expression data with the selected subset of genes. Tab. 2 summarizes the results of bagged SVMs on the *GCM* data set. Even if not always there is a statistical significant difference (according to Mc Nemar test) between single

Table 1. *GCM* data set: results with single SVMs

Kernel type and parameters	**Err.all genes**	**Err.sel. genes**	**Relative err.red.**
Dot-product, C=20	0.2600	0.2279	12.31 %
Polynomial, deg=6 C=5	0.7000	0.2275	—-
Polynomial, deg=2 C=10	0.6900	0.2282	—-
Gaussian, σ=2 C=50	0.3000	0.2185	27.33 %

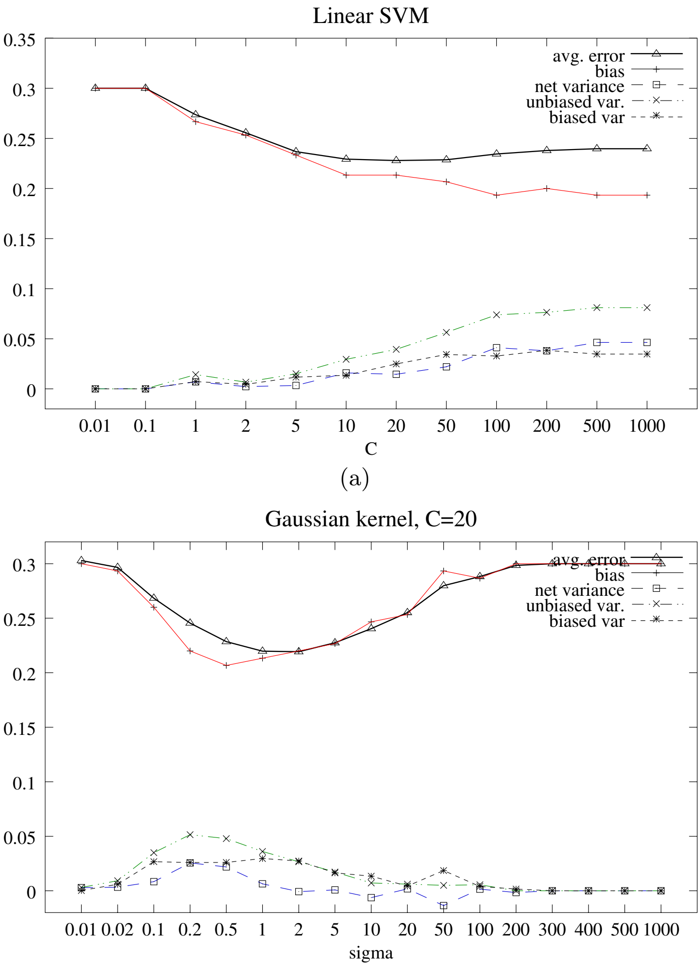
Table 2. *GCM* data set: compared results of single and bagged SVMs

Kernel type and parameters	**Error SVMs**	**Error bagged**	**Relative err.red.**
Dot-product, C=10	0.2293	0.2200	4.06 %
Dot-product, C=20	0.2279	0.2133	6.41 %
Polynomial, degree=6 C=5	0.2275	0.2000	12.09 %
Polynomial, degree=2 C=10	0.2282	0.2133	6.53 %
Gaussian, sigma=2 C=50	0.2185	0.2067	5.40 %
Gaussian, sigma=10 C=200	0.2233	0.2067	7.44 %

and bagged SVMs, in all cases bagged ensembles of SVMs outperform single SVMs. The degree of enhancement depends heavily on the possibility to reduce the variance component of the error, as bagging is mainly a variance-reduction ensemble method.

Indeed, performing a bias–variance analysis of the error of single SVMs on the *GCM* data set [13], we note that bias largely overrides the variance components of the error, and in this case we cannot expect a very large reduction of the error with bagging (Fig. 1). Nonetheless we can see that both with linear SVMs (Fig. 1 a), and gaussian (Fig. 1 b) SVMs, the minimum of the estimated error and the estimated bias are achieved for different learning parameters, showing that in this case Lobag could improve the performance, even if we cannot expect a large reduction of the overall error, as the bias largely dominates the variance component of the error (Fig. 1). Indeed with Lobag the error is lowered, both with respect to single and bagged SVMs (Tab. 3). As expected, both bagged and lobag ensembles of SVMs outperform single SVMs, but with lobag the reduction of the error is significant at 0.05 confidence level, according to Mc Nemar's test, for all the applied kernels, while for bagging it is significant only for the polynomial kernel. Moreover Lobag always outperforms bagging, even if the error reduction is significant only if linear or polynomial kernels are used. Summarizing Lobag achieves significant enhancements with respect to single SVMs in analyzing DNA microarray data, and also lowers the error with respect to classical bagging.

Even if these results seem quite encouraging, they must be considered only as preliminary, and we need more experiments, using different data sets and using more reliable cross-validated evaluations of the error, in order to evaluate more carefully the applicability of the lobag method to DNA microarray data analysis. Moreover we need also to assess the quality of the classifiers using for instance ROC curves or appropriate quality measures as shown, for instance, in [9].



(a)

(b)

Fig. 1. *GCM* data set: bias-variance decomposition of the error in bias, net-variance, unbiased and biased variance, while varying the regularization parameter with linear SVMs (a), and the kernel parameter σ with gaussian SVMs (b)

Table 3. *GCM* data set: compared results of single, bagged and Lobag SVMs on gene expression data. An asterisk in the last three columns points out that a statistical significant difference is registered ($p = 0.05$) according to the Mc Nemar test

Kernel type	**Error SVMs**	**Error bagged**	**Error Lobag**	**Err.red. SVM− > bag**		**Err.red. SVM− > Lobag**		**Err.red. bag− > Lobag**	
Dot-product	0.2279	0.2133	0.1933	6.41 %		15.18 %	*	9.38 %	*
Polynomial	0.2275	0.2000	0.1867	12.09 %	*	17.93 %	*	6.65 %	*
Gaussian	0.2185	0.2067	0.1933	5.40 %		11.53 %	*	6.48 %	

Acknowledgments

This work has been partially funded by DSI, Università di Milano.

References

1. L. Breiman. Random Forests. *Machine Learning*, 45(1):5–32, 2001.
2. M. Brown et al. Knowledge-base analysis of microarray gene expression data by using support vector machines. *PNAS*, 97(1):262–267, 2000.
3. T.G. Dietterich. Approximate statistical test for comparing supervised classification learning algorithms. *Neural Computation*, (7):1895–1924, 1998.
4. P. Domingos. A unified bias–variance decomposition. Technical report, Department of Computer Science and Engineering, University of Washington, Seattle, WA, 2000.
5. S. Dudoit, J. Fridlyand, and T. Speed. Comparison of discrimination methods for the classification of tumors using gene expression data. *JASA*, 97(457):77–87, 2002.
6. M. Eisen and P. Brown. DNA arrays for analysis of gene expression. *Methods Enzymol.*, 303:179–205, 1999.
7. T.S. Furey, N. Cristianini, N. Duffy, D. Bednarski, M. Schummer, and D. Haussler. Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics*, 16(10):906–914, 2000.
8. T.R. Golub et al. Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. *Science*, 286:531–537, 1999.
9. I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene Selection for Cancer Classification using Support Vector Machines. *Machine Learning*, 46(1/3):389–422, 2002.
10. J. Khan et al. Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nature Medicine*, 7(6):673–679, 2001.
11. S. Ramaswamy et al. Multiclass cancer diagnosis using tumor gene expression signatures. *PNAS*, 98(26):15149–15154, 2001.
12. G. Valentini. Gene expression data analysis of human lymphoma using support vector machines and output coding ensembles. *Artificial Intelligence in Medicine*, 26(3):283–306, 2002.
13. G. Valentini and T.G. Dietterich. Bias–variance analysis and ensembles of SVM. In *Multiple Classifier Systems. Third International Workshop, MCS2002, Cagliari, Italy*, volume 2364 of *Lecture Notes in Computer Science*, pages 222–231. Springer-Verlag, 2002.
14. G. Valentini and T.G. Dietterich. Low Bias Bagged Support Vector Machines. In *Proc. ICML 2003, The Twentieth International Conference on Machine Learning*, Washington D.C., USA, 2003.
15. G. Valentini and F. Masulli. NEURObjects: an object-oriented library for neural network development. *Neurocomputing*, 48(1–4):623–646, 2002.
16. G. Valentini, M. Muselli, and F. Ruffino. Bagged Ensembles of SVMs for Gene Expression Data Analysis. In *IJCNN2003, The IEEE-INNS-ENNS International Joint Conference on Neural Networks*, Portland, USA, 2003.

Narrative Thought in Childhood and Adolescence Hierarchical Models of Story Composition and Interpretation

Anne McKeough[1], Barb Wigmore-MacLeod[2], and Randy Genereux[3]

[1] Division of Applied Psychology
University of Calgary
2500 University Drive NW Calgary, Alberta Canada T2N 1N4
[2] Holland College
[3] Mt. Royal College

Abstract. Stories have been a fundamental feature of human existence for millennia. Since well before recorded history we have been spinning tales with words, images, music, and dance in order to entertain, enchant, enculturate, and entrance. The crafting of compelling narratives has been enshrined as one of our highest forms of artistic achievement. Furthermore, formal training in reading, writing, and interpreting stories has become a core component of our educational curriculum. There can be no doubt that narrated is deeply embedded in the individual and in the cultural mind. In this paper, I will first briefly discuss the nature of narrative thought, presenting its theoretical base and a smattering of pivotal studies that have provided empirical evidence for how narrative develops. I will then describe, in more detail, my own developmental work, which is based on Case's neo-Piagetian theory of cognitive development [1][2]. I will discuss the impact of individual working-memory capacity as well as cultural factors. I will end the paper with a brief discussion of how this work relates to and supports findings from neurological study.

1 The Nature of Narrative Thought

1.1 Theoretical Positions

Over the past thirty years, there has been a renewed interest in the study of narrative, perhaps, in large part, because of the level of information it provides researchers and practitioners regarding social, discursive, and cultural *forms of life* [3][4][5][6]. The advantages that are believed to arise from the assemblage and narrative analysis of direct accounts of experience are based on the assumption that individuals think, perceive, imagine, and make moral choices in accordance with narrative knowledge and narrative thought. From an early age, narrative becomes the way in which individuals organize and make sense of their daily experiences [7][3][4][5]. McCabe [8][9][10] stated that "narrative is a linguistic crossroads of culture, cognition, and emotion and serves the dual functions of sense making and self-presentation". Her view of narrative emphasizes the cultural "embeddedness" of any particular narrative, while simultaneously

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 325–337, 2003.

acknowledging the subjectivity voiced by the narrator. This view attributes increasing coherence in children's narratives as resulting not only from their desire to become more proficient storytellers, but also from the need to organize a coherent life story. Through narrative, then, children come to know and understand themselves in relation to the world around them. Narratives also hold important roles in our lives because they are used to communicate, to motivate, to teach, and to entertain. In Western cultures, a narrative is generally thought to be comprised of several essential components, including: 1) a beginning, or an initiating event, 2) a simple reaction (by the protagonist), 3) a goal (formulated by the protagonist), 4) an attempt (by the protagonist) to achieve that goal, 5) an outcome, and 6) an ending, that includes the protagonist's reactions and the long-term consequences [7]. More recently, Yussen and Ozcan [11] have defined narrative thought as involving "any cognitive action (activity) - be it listening, speaking, reading, writing, imagining, or recollecting - in which the individual contemplates one or more people engaged in some activity or activities in specific settings for a purpose". Bruner [3] proposed that narrative is one of the two main modes of thought, the other being the paradigmatic mode. He suggested that these are the two distinct ways by which experience is ordered and reality is constructed. The goal of the paradigmatic or logico-mathematical mode of thought is to explain and to predict natural phenomena, in objective terms, based on the discovery of cause-effect laws. Throughout history, the paradigmatic mode has been used as the main method of studying scientific phenomena. Recently, however, there has been increased recognition of the limitations of the empirical or scientific method for application to the human sciences. This is due, in large part, to the inability of this mode to make much sense of such constructs as human desire, goals, and social interaction [3][12][13][14]. In contrast to the paradigmatic mode, the narrative mode of thought allows for the explanation and understanding of specific human behaviors and events by ascertaining their inherent meaning and interpreting their significance [5]. This is made possible because of the dual organizational component of narrative thought which operates by temporally and causally ordering events that occur in the physical world, or what Bruner [4][5] referred to as the landscape of action, while simultaneously accounting for mental activity, or the landscape of consciousness. This allows individuals to understand both the tangible aspects of an experience, as well as their own and others' intentional states [4][5][6]. Bruner [3] stated that masters of the paradigmatic mode refrain from saying more than they mean in efforts to avoid using unfounded assumptions in their search for an objective truth. In contrast, masters of the narrative mode, such as accomplished poets and novelists, are most effective when they mean more than they are able to say. A good story, therefore, offers each of its listeners something unique and can generate a multitude of meanings, allowing each person to attach his or her own personal significance to it. Through narrative, individuals can experience authentic thoughts, emotions, and desires that become incorporated into consciousness even though they may arise from the vicarious experience of adventures and interactions.

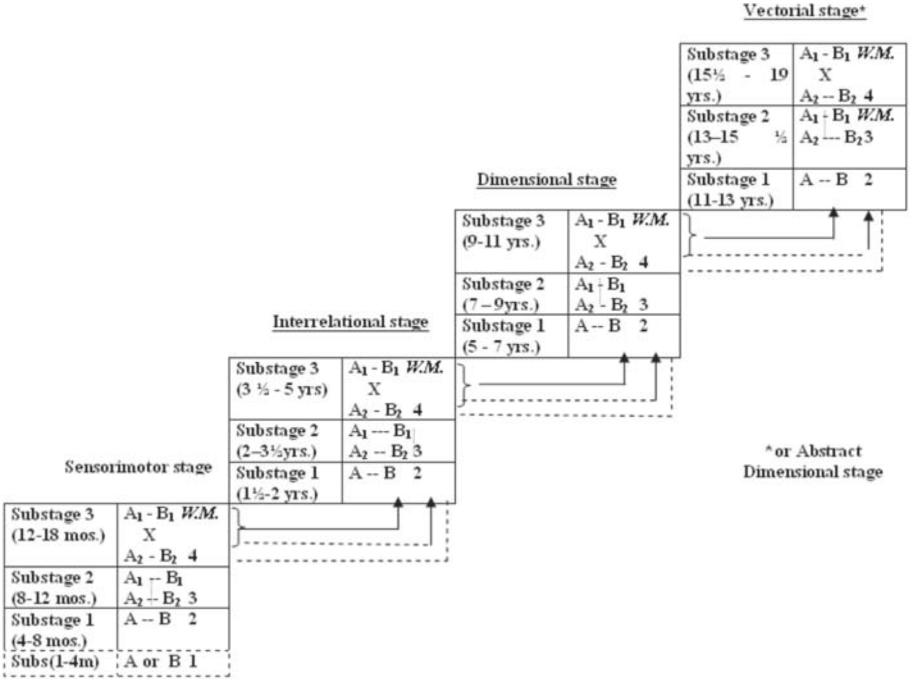


Fig. 1. Case's proposed stages and substages of cognitive development

1.2 Empirical Support for Developmental Trajectories

The pervasiveness of stories in our lives, as well as Bruner's [3] proposed elevated status of narrative as one of the two primary modes of thought, elucidates the need for understanding how narrative knowledge develops from childhood through adulthood. Applebee [15] recognized the developmental transformation of narrative complexity, reporting increases in the structural components of narrative with age and verbal capability. Similarly research investigations by Stein [17] and Peterson and McCabe [16] provided evidence of age-related changes in narrative structure. Mandler's [7] definition of a basic story episode involves six elements - a beginning, a simple reaction, a goal, an attempt, an outcome, and an ending. Several developmental researchers have concluded by about 8 years of age, children have mastered this [18][19][20][21]. At Age 10, children are capable of composing narratives that are more complex than the basic story episode, demonstrating multiple attempts by the protagonist to achieve the goal at hand [18][19][21]. While there has been little research investigating increases in the structural complexity of narrative after Age 10, there is one line of research that sheds light upon further increases in structural complexity of narrative thought. McKeough's (1992) model of narrative development, which stems from Case's [1][2] neo-Piagetian theory of cognitive development, will, therefore, be presented in some detail.

2 A Neo-structural Model of Narrative Development

Recently, the basic elements used to comprise a story, and the ways in which these elements are coordinated or integrated, have been studied as sources of the underlying levels of cognition engaged in by storytellers [19][22]. Following Case [1][2], McKeough's model of narrative development takes into account both maturationally-based increases in information-processing capacity, as well as culturally-based learning experiences. Case suggested that brain maturations and development allows for an increase in processing capacity that is recursively reorganized through childhood and adolescence. This, in turn, enables the development of cognitive structures of increasing complexity within any cognitive domain. Case further elaborated that as children are exposed to and engage in higher forms of cognition within their culture (e.g., mathematical reasoning, logical thought, spatial cognition, and narrative thought), they become increasingly more proficient within these cognitive domains. Case's [1][2] theory proposed that there are four main stages of cognitive development, each of which has three substages. He suggested that there is a recursive progression through the three substages and that integration of two cognitive structures used singly at the previous stage results in a consolidated structure that is used as the foundation for the subsequent stage of development (see Figure 1 in the previous page). Case's model suggests that at each stage of development there is an initial substage in which a new type of structure is assembled from two structures that operated independently in the previous stage but which can only be applied in isolation (unifocal coordination, working-memory requirement of 2). At the second substage, two of these structures can be applied in tandem but they cannot be successfully integrated (bifocal coordination, working-memory requirement of 3). At the third, and final substage, two or more units can be integrated and applied at the same time (elaborated coordination, working-memory requirement of 4). As previously mentioned, this results in the formation of a consolidated structure that provides the basis upon which subsequent levels are developed.

To depict this by way of example, consider McKeough's [19] model of narrative development. (See Table 1 for McKeough's [23] scheme for scoring narrative plot structure across ages 4 through 18 years).

According to Case, 4-year-olds are in substage 3 of the interrelational stage of cognitive development, and therefore, should be capable of handling only one consolidated interrelational unit at a time (working-memory requirement of 1). According to McKeough, by Age 4, children have consolidated an interrelational unit for a simple *action-based* narrative script in which external states, events, and actions are linked temporally and referentially. They have also consolidated an interrelational unit for identifying and describing affective states such as happy, sad, and so on. At 4 years of age, however, children are not yet able to integrate the two units, and so they tend to construct stories reflecting only the external or physical world of events and actions. In other words, their use of intentionality in stories is rudimentary, at best, although researchers in the child's-theory-of-mind school have shown that some understanding of mental states is normally in place [24]. The following provides an example: "Once upon

Table 1. McKeough's scheme for scoring narrative plot structure

Action	Does the story have a sequence of events that are temporally, causally, or referentially related and that occur exclusively in the physical world of action and events? (Note:”happily ever after” not scored as a mental state) **YES** ↓	**NO**-Level 0 Age ¡4 years
Intentional	Does the story include explicit or implicit reference to the mental states that motivate action in the physical world and is there a problem that is immediately resolved in the end? **YES** ↓	**NO**-Level 1 Age 4 years
	Does the story have a problem, a series of failed attempts or complications followed by a resolution (not necessarily solving the problem), such that additional mental states are mentioned or implied in the context of the story? **YES** ↓	**NO**-Level 2 Age 6 years
	Does one impediment or well developed a sub-plot have more significance than the others, thereby also broadening the characters' intentions/mental states? Is the impediment dealt with in the outcome, with the result that the resolution has a well-planned feeling? **YES** ↓	**NO**-Level 3 Age 8 years
Interpretive	Does the focus of the story shift from the characters' mental states to why particular mental states are held? Does a constellation of mental states or constellation of social circumstances create a psychological profile or character trait that is represented across time and situations? **YES** ↓	**NO**-Level 4 Age 10 years
	Are additional traits represented, such that a dialectic is created wherein the interaction of two states or traits lead to further psychologically oriented complications? **YES** ↓	**NO**-Level 5 Age 12 years
	Does the dialectical relation between states or traits act as an integrating device lending a greater sense of coherence to the story? **YES** ↓ Level 7 years-Age 18 years	**NO**-Level 6

a time there was a little girl who lived on a farm with a very good horse. And she always rode to the country on the horse and they always had a picnic together" [25].

By Age 6 children demonstrate the ability to coordinate the two abovementioned units. This results in a new narrative structure that includes external events as well as the mental states that underlie actions and accompany events. The following transcript provides an example: "Once upon a time there was a horse that wanted to be wise. And a little girl found him and she said, ¡¡Do you want to be wise?¿¿ And she teached him all the things that little horses are supposed to know. And so the little horse went to the farm and the little girl trained the little horse and the little horse had a happy life."[25]. Thus, by Age 6, children become capable of integrating the action/event schema and the mental schema. They begin to think of the stream of any familiar human activity as a coordinated sequence, always involving two components. These components have been referred to as a "landscape of action" (which is the behavioural element of any event sequence) and a "landscape of consciousness" (which is the internal or "intentional" element) [3]. Following Case [1], McKeough [19][22] posited that one way to understand the development within stages of narrative thought is to consider the development of working memory from 1 to 4 units. This results in a progression from pre-intentional (4 years) to uni-intentional (6 years) to bi-intentional (8 years) to integrated bi-intentional thought (10 years) [26]. Whereas 4-year-olds tend to use either one or the other of the two structures (i.e., a working-memory requirement of 1), six-year-olds are able to coordinate the two so that action sequences often contain reference to the mental states that underlie, or are associated with, them (i.e., a working-memory requirement of 2). Hence, a prototypic narrative from a 6-year-old has moved from a "script" to a "plot", centering around a problem and its resolution [19][22]. At eight years of age, further development of the intentional structure is evident. Eight-year-olds introduce a second focus in their stories, a complication of a series of unsuccessful attempts at resolution, prior to the solution. They demonstrate the ability to produce narratives with one or more additional sub-problems and associated mental states embedded within the main story [26], thereby demonstrating the ability to coordinate, at least in a simple fashion, two or more dual-landscape units, which requires a working-memory of 3, within Case's Model [1] (See Appendix A for a protypic example of an eight-year-old's narrative). By Age 10, children are typically able to integrate the complication into the resolution (requiring a working-memory capacity of 4), with the result that the story has a well-developed feeling and planned quality. Stories by ten-year-olds, therefore, have the tendency to become increasingly elaborate. The number of complicating events tends to increase and these tend to be better integrated with the original problem and its resolution [26]. Within Case's [1] framework, this structure is consolidated into one smoothly functioning unit and serves as the basic building block of the next stage of development (See Appendix A for a prototypic example of a 10-year-old's story). By adolescence a qualitative changes in narrative structure occur. Reasoning about social interaction takes a psychological turn.

Whereas most 10-year-olds focus on immediate, proximal intentions and mental states, by Age 12 a change in story focus and structure occurs such that immediate mental states and intentions are interpreted on another level in terms of enduring, transituational mental states and character traits. For example, a 12-year-old might describe one or more scenes in which a character is nervous when trying to make new friends, and then explain the character's nervousness as being due to his or her shyness or due to an unpleasant earlier experience. Reference to traits and prior experiences indicates that the narrative thought of 12-year-olds has an added structural dimension; 12-year-olds seem capable of holding in mind two or more immediate experiences (mental and physical) that occurred at different times or in different situations and then of extracting a higher-order unit of meaning-making, such as a personality trait or an enduring state, from the two lower-level units. The following excerpt from a 12-year-old's story illustrates the use of enduring states and traits (e.g., "very upset lately" and "usually a lot of fun to be around") as well as the interpretation of immediate experiences in terms of the past (e.g., "I'm starting to think that Sandy's parents gave her too much attention before the baby"). "My friend's mother has had a baby. It's been about a month or so since her mother came out of the hospital. My friend has been very upset lately because her mom has not been paying much attention to her, her dad is on a very long business trip to Dallas, Texas, but he should be coming home soon. so that also causes a problem, the main problem is that my friend Sandy isn't getting as much attention as she wants, and she is beginning to hate the baby. I try to tell her that the baby isn't the problem, the problem is that her mother doesn't realize that she is ignoring Sandy. I know for a fact, though, that Sandy's parents love her very much, and they treat her quite well. I'm starting to think that Sandy's parents gave her too much attention before the baby. I've told Sandy that she should talk to her mother about this problem, but she won't, she's scared thet her mom will get angry. ...Sandy is usually a lot of fun to be around, but lately she talks about how the baby is destroying her life. I think it would be neat to have a baby in the family, as you can tell, Sandy disagrees. ...". By Age 14, additional enduring states and traits enter the picture, often in the form of contrary tendencies within the same character that create an internal psychological conflict in addition to the outer story conflict. The following excerpts from a 14-year-old's story illustrate this structure. The teenage protagonist is not only in conflict with some of her friends, she is also torn between wanting to be with them to gain popularity and wanting to break away from them so she can be herself. "That's the problem with some of my friends. Popularity is such an important thing to them. If your not popular you're definitely not one of them. I don't think they ever really accepted me. Sometimes I hate the way they criticize and make fun of people who don't live up to their standards....You know when you think about it it's all very stupid. I mean it wasn't long ago when I would have done anything to get into their group, and now that I'm with them I'll do almost anything to get out. I've just got to break apart from them slowly and hopefully find people that know the meaning of friendship. I can't wait for the day to come. The day that I can be me and the day when someone will like me just for being myself." [23]. By Age 18, adolescents are able to resolve the dialectic of the internal and external struggles of their protagonists in a coordinated and coherent fashion, thus demonstrating the capacity to synthesize two higher-order units into a well-formed whole. But what of culture? At the outset of the paper, we presented narrative and narrative thought as a cultural artifact (e.g., [5]) and thus far, our discussion has centered on the individual's mental capacity. Moreover, as is no doubt evident, with development, the stories we presented increasingly reflect inc narrative conventions that are prevalent in Western literature. Some work has been done that addresses the cultural issue by investigating how youngsters come to use specific literary devices or conventions, namely, foreshadowing, surprise or trick endings, and flashback. Interestingly, a similar pattern of development in narrative structural complexity has been identified for the use of these devices [27][28]. On the basis of Case's [2]theory and McKeough's developmental analysis of plot structure [19][22][29], Case et al. [27] predicted that as children move through adolescence they should be able to comprehend and construct increasingly complex narrative structures involving not just a single complete story, but multiple stories or levels of meaning embedded within a larger story structure. In essence, they should become increasingly capable of dealing with stories within stories. Preliminary support for this hypothesis was obtained in a study in which students were asked to compose stories containing either foreshadowing or a trick ending. Both foreshadowing and trick endings are devices that can potentially be used to create double layers of story meaning; foreshadowing by hinting at a future direction or resolution of the story that has yet to be realized, and trick endings by providing unexpected information that can result in a dramatic reinterpretation of the entire story. To work effectively, both devices require coordination between the story told early on and the story unveiled later on in the narrative. Results of the study revealed a progression in complexity of the stories across the four age groups studied (10, 12, 14, and 18 years). On the trick ending task, 10-year-olds simply tacked on an ending that was unexpected with respect to the immediately preceding part of the story; they did not integrate the ending with the beginning of the story at all. An example is the story in which the protagonist was inexplicably transformed into a mosquito via a magic wand at the end of the tale. Twelve-year-olds used the beginning of the story to set up a clearly delineated expectation for the ending and then violated that expectation; they showed the first coordination of the beginning and ending of the story. For example, one story began with a fisherman hearing a blood-curdling scream. It went on to describe the fisherman's ill-fated attempt to find and rescue the person in trouble, the outcome of which was the fisherman's own death. The surprise ending was that the scream was actually that of a pig rather than a person. Fourteen- and 18-year-olds produced stories with double meanings throughout, stories that in retrospect could be read with both the expected and surprise meaning in mind; these students displayed the advanced ability to integrate two possible story lines into one coherent narrative. For example, in one 14-year-old's story, a duchess who is about to marry a newly crowned king goes through a series of meticulous wedding preparations in bathing, dressing, selecting jewellery, and so on. All the while she is eagerly anticipating the opulent life she will be leading as the queen. Each of these preparations and anticipations is thrown into stark relief when at the end of the story she discovers the king has magnanimously filled his court with smelly beggars, cripples, and lepers in need of his help. In an 18-year-old's story that demonstrates an advance in overall coherence, the protagonist recalls his youthful adventures with his teenage buddy, including swimming in the muddy old creek, meeting a few girls and staying out all night, making some noise and howling at the moon, getting into scraps, getting hauled away in the police van, and getting bailed out of jail by his "old man". Only at the very end do we find out that the protagonist is actually a dog rather than a human. Virtually every sentence in the story takes on a new flavor with this revelation in hand, indicating that the author carefully crafted each sentence with dual meanings in mind. McKeough and Genereux [28] have also demonstrated that adolescents are able to use flashback in their narratives. In using flashbacks, the characters' intentions are presented in a long-term context and character traits are established that endure across time and situations. The flashback causes events happening in the present to be interpreted differently. Current events are interpreted while taking into consideration past events. Flashbacks allow the reader to be shown that a character feels a certain way because past experiences have influenced the type of person he or she is. Therefore culturally-derived rhetorical devices, such as flashback, foreshadowing, and surprise endings, allow the reader to interpret the intentional states of story characters and mark a shift in the quality of the productions from intentional narratives to interpretive ones [26] (See Figure 2 for a model of narrative development).

How, then, does this neo-structural analysis fit with neurological research? Although the links, at this point, are tentative, on the level of the individual, we suggest that experience with stories, of both the literary and personal variety, can produce changes in local neurological circuits as local inhibitory and facilitory connections are formed [30]. For example, a young child experiences stories including characters that do something. This coupling of character and event might well also derive from their everyday living, as then observe people and animals in action, and moreover, in normative action (e.g., a kitten meows but does not bark). Neurological theory also proposes that changes in local neurological circuits occur in the context of a general cortical system in which, increasingly, local regions are differentiated and connected to more distant regions [31][32].With young children, we see a shift from rambling narratives with only local coherence and little or no global coherence to ones that have a clear topical focus by 5 years of age or 6 years of age [33]. This move toward temporally and causally organized narratives can be seen as indicative of a more general cortical system. The shift toward psychologically-oriented narratives with the onset of adolescence is another such example. These findings seem to support the position that, although the brain is clearly responsive to experiences that are encoded as a series of neural connections, functioning is also "highly con-

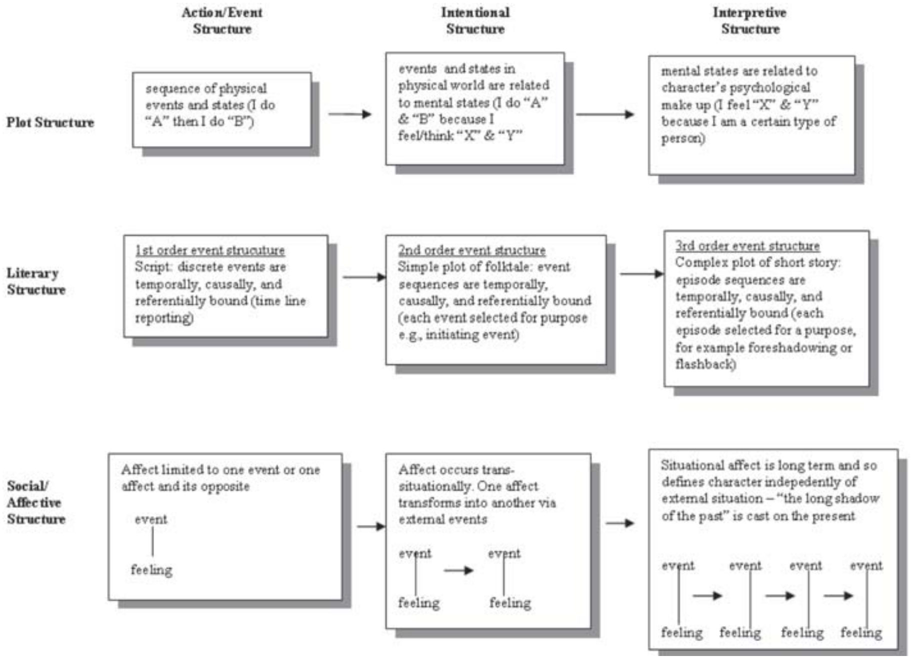

Action/Event Structure
Intentional Structure
Interpretive Structure
Plot Structure
sequence of physical events and states (I do "A" then I do "B")
events and states in physical world are related to mental states (I do "A" & "B" because I feel/think "X" & "Y"
mental states are related to character's psychological make up (I feel "X" & "Y" because I am a certain type of person)
Literary Structure
1st order event strucuture
Script: discrete events are temporally, causally, and referentially bound (time line reporting)
2nd order event structure
Simple plot of folktale: event sequences are temporally, causally, and referentially bound (each event selected for purpose e.g., initiating event)
3rd order event structure
Complex plot of short story: episode sequences are temporally, causally, and referentially bound (each episode selected for a purpose, for example foreshadowing or flashback)
Social/ Affective Structure
Affect limited to one event or one affect and its opposite
event
feeling
Affect occurs trans-situationally. One affect transforms into another via external events
event
feeling
event
feeling
Situational affect is long term and so defines character indepedently of external situation – "the long shadow of the past" is cast on the present
event
event
event
event
feeling
feeling
feeling
feeling

Fig. 2. Action, intentional and interpretive structure

strained by the general state of development of the cortex, which, in turn, is driven by maturation and general (as opposed to task-specific) experience"[34]. Finally, the position that frontal lobe development allows certain cognitive processes, such as working memory and reflective abstraction, which support the development of cortical integration, is at lease in part supported by our work. That is, age-related increases in working-memory capacity have been linked to increases in the structural complexity of stories [19][22]. And all of this growth occurs within the context of cultural experience, which appears, according to the analyses reported herein, to have a clear shaping effect of the "long-distant links" that are formed neurologically.

References

1. Case, R.: Intellectual development: Birth to adulthood. Academic Press New York (1985)
2. Case, R. (ed.): The mind's staircase: Exploring the conceptual underpinnings of children's thought and knowledge.Lawrence Erlbaum Associates Hillsdale, NJ (1992)
3. Bruner, J.: Actual minds, possible worlds.Harvard University Press Cambridge, MA (1986)
4. Bruner, J.: Life as narrative. In: Social Research, Vol.54. (1987) 11-32
5. Bruner, J.: Acts of meaning. Harvard University Press Cambridge, MA (1990)
6. Polkinghorne, D.E.: Narrative knowing and the human sciences. State University of New York Press New York (1988)

7. Mandler, J.M.: Stories, scripts, and scenes: Aspects of schema theory. Lawrence Erlbaum Associates Hillsdale, NJ. 1984
8. McCabe, A.: Editorial. In: Journal of Narrative and Life History, Vol. 1(1), (1991) 1-2
9. McCabe, A.: Chameleon readers: Teaching children to appreciate all kinds of good stories. McGraw-Hill New York, NY (1996)
10. McCabe, A.: Developmental and cross-cultural aspects of children's narration. In: Bamberg, M. (ed.): Narrative development: Six approaches. Lawrence Erlbaum Associates Mahwah, NJ (1997) 137-174
11. Yussen, S., Ozcan, N.: The development of knowledge about narratives. In: Education, Vol. 2(1), (1996) 1-68
12. Gergen, K., Gergen, M.: Narrative and the self as relationship. In: Advances in Experimental Social Psychology, Vol. 21, (1988) 17-56.
13. McAdams, D.P.: The stories we live by: Personal myths and the making of the self.William Morrow and Company New York (1992)
14. Singer, J., Salovey, P.: The remembered self: Emotion and memory in personality. Free Press New York (1993)
15. Applebee, A.N.: The child's concept of story: Ages two to 17. University of Chicago Press Chicago (1978)
16. Peterson, C., McCabe, A.: Linking children's connective use and narrative macrostructure. In: McCabe A., Peterson C. (eds.): Developing narrative structure. Lawrence Erlbaum Associates Hillsdale, NJ (1991) 29-53
17. Stein, N.: The development of children's storytelling skill. In: Franklin M.B., Barten S. (eds.): Child language: A reader. Oxford University Press New York (1988) 262-279
18. Kemper, S.: The development of narrative skills: Explanations and entertainments. In:. Kuczaj, Stan, A, II (ed.): Discourse development: Progress in cognitive development research). Springer-Verlag New York (1984) 99-124
19. McKeough, A.: A neo-structural analysis of children's narrative and its development. In: Case, R.(ed.): The mind's staircase: Exploring the conceptual underpinnings of children's thought and knowledge. Erlbaum Hillsdale, NJ (1992a) 171-188
20. Stein, N., Albro, E.: Building complexity and coherence: Children's use of goal-structured knowledge in telling stories. In: Bamberg M. (ed.): Narrative Development: Six Approaches. Lawrence Erlbaum New Jersey (1997) 5-44
21. Stein, N., Glenn, C.: An analysis of story comprehension in elementary school children. In: Freedle R.O. (ed.): New directions in discourse processing; Vol. 2, Advances in discourse processes. Ablex Norwood, NJ (1979) 255-282
22. McKeough, A.: Testing for the presence of a central social structure: Use of the transfer paradigm. In: Case, R. (ed.), The mind's staircase: Exploring the conceptual underpinnings of children's thought and knowledge.Erlbaum Hillsdale, NJ (1992b) 171-188
23. McKeough, A.: Narrative knowledge and its development: Toward an integrative framework. In: Issues in Education: Contributions for Educational Psychology, Vol. 2(1), (1996) 73-81
24. Astington, J., Olson, D., Harris, P.: Developing theories of mind. Cambridge University Press Cambridge, MA (1988)
25. McKeough, A., Sanderson, A., Martens, M., Salter, D.: Narrative development in late childhood and early adolescence: Evidence from fictional compositions, family stories, and moral dilemmas. Paper presented at the American Educational Research Association annual meeting, New York (1996)

26. McKeough, A.: Using the notion of a central conceptual structure to explain the development of children's understanding of human behaviour. Paper presented at the biennial meeting of the Society for Research in Child Development (1993)
27. Case, R., Bleiker, C., Henderson, B., Krohn, C., Bushey, B.: The development of central conceptual structures in adolescence. In: Case, R. (ed.): The role of central conceptual structures in the development of children's numerical, literary, and spatial thought. University of Toronto, Institute for Child Study Toronto (1993) 104-128
28. McKeough, A., Genereux, R.: Transformation in narrative thought during adolescence: The structure and content of story compositions. Manuscript submitted for publication. (2003)
29. McKeough, A.: Building on the oral tradition: How story composition and comprehension develop. In: Astington J.W. (ed.): Minds in the making: Essays in honor of David R. Olson. Blackwell Publishers Malden, MA (2000) 98-114
30. Taylor, M.J.: The role of event-related potentials in the study of normal and abnormal cognitive development. In: Boller, F., Grafman, J.(Series eds.) & Johnson R., Jr, Baron J. C.V.E. (Vol. eds.): Handbook of neuropsychology. Libr. 10. Section 14: Event-related brain potentials and cognition. Section 15: Positron emission tomography and neurobehavior. Elsevier Amsterdam (1995)
31. Thatcher, R.W.: Cyclical cortical reorganization during early childhood. Brain and Cognition, Vol. 20, (1992) 24-5
32. Thatcher, R.W.: Human frontal lobe development: A theory of cyclical cortical reorganization. In: Krasnegor, N.A.,. Lyon, G.R., Goldman-Rakic, P.S. (eds.): Development of the prefrontal cortex: Evolution, neurobiology, and behavior. Paul H. Brookes Baltimore MD (1997) 85-113
33. Sutton-Smith, B.: The importance of the storytaker: An investigation of the imaginative life. In: Urban Review, Vol. 8, (1975) 82-95
34. Case, R., Mueller, M.: Differentiation, integration, and covariance mapping as fundamental processes in cognitive and neurological growth. In:. McClelland J.L., Siegler R.S. (eds.): Mechanisms of cognitive development: Behavioral and neural perspectives. Erlbaum Mahwah, NJ (2001) 185-219

Appendix A

Prototypic example of an 8-year-old's story In the following example, earning money is the sub-problem that must be resolved prior to the resolution of the main goal of owning a horse. “Once upon a time a little girl was walking down the street and she seen a happy horse that she wanted to buy. But she had not enough money. So she ran home to tell her dad that if she could buy the horse and earn all the - and earn some money. And her dad said yeah. So she kept - so she was living in a cottage and was chopping down some trees. And on Monday she went back to that store and she seen the horse and she bought it and she went home and she went for a ride on it to the river.”[23]. Prototypic example of a 10-year-old's story: The story to follow provides an example of an integrated plot. “Sandy was a beautiful and kind girl. She lived with a loving family, but Sandy had a problem, she was deaf. She was born with it. She is 10 now and her hearing had got a little better but not moch. She hated it when other people made fun of her. Sandy kept telling herself that somehow she would get better, somehow. One day she was reading the newspaper and she saw a place where she could be trained to read lips. Quickly she showed the ad to her mom. Her mom told her she couldn't go through all that yet.When Sandy heard that she ran to her room and cried. Even though her mother told her she couldn't go, she was determined to somehow. She decided that she could tell her father and if he said no she wasn't going to communicate with her parents ever again.. Next day she asked her friend (named Caroline) what she thought about the idea. Caroline loved the idea because she was also deaf. That night Sandy asked her father what he thought about it. Her father also liked the idea. So finally her parents decided to talk it over while she went to her room. Sandy waited anxiously in her room. Finally it was time to come out. Her parents decided that she could go to the place where she could learn to read lips. Sandy hugged her parents and thanked them. She danced across the room to get to her bedroom. She nelt down in front of her bed and said a prayer of thanks to God.. Two years later when she and Caroline finished the school they could talk to people and read lips easily. Sandy thought she was the happiest person on earth because she had solved her problem [23].

Degrees of Similarity in Knowledge Transfer

Alessandro Antonietti and Claudio Maretti

Cognitive Psychology Laboratory
Department of Psychology
Catholic University of Sacred Heart, Largo Gemelli 1, 20123 Milano, Italy
alessandro.antonietti@unicatt.it

Abstract. In analogical problem solving individuals infer an abstract solution schema from the source and apply it to the target. The schema is composed by the relevant elements which can be matched onto the target. It is assumed that stories sharing a high number of elements are perceived as more similar than stories having a lower number of common elements. An analysis carried out by means of a Kohonen map supported this assumption: the pattern of similarities represented by the Kohonen map corresponded to the similarity judgements produced by human subjects.

1 Analogical Thinking

When we have to face a new situation, we may retrieve knowledge that we have acquired previously about a different situation which is similar to the present one. This occurs since we realise that the new situation is structurally isomorphic to that we have experienced in the past, even though the superficial features of the two situations are quite different. In these cases we reason through analogies. In fact, analogical thinking is based on the transfer of ideas from a familiar situation to a novel situation, so to extend some information, principles, or insights from a common set of knowledge and experiences to an unfamiliar domain. It has been noticed that in many disciplines the development of a novel theory or perspective depends on applying an analogy drawn from a different domain of knowledge. This is a process of which artistic intuition, historical reconstruction, scientific discovery, and technological invention take advantage. Political and social discourses and judgements are often based on analogies; legal and medical decision-making is sometimes grounded on analogical reasoning; moreover, in ordinary language people make use of analogies when they try to explain something or to express their thoughts and emotions. Finally, school instruction benefits by analogies because they allow to clarify concepts which are difficult to understand and are a potential mechanism for the development of creativity [1]. A field where analogical thinking is often applied is problem-solving: to reach the solution of a new problem (*target*), persons retrieve a familiar situation (*source*) which is analogous to that problem and which can suggest a relevant response. In analogical problem-solving three phases can be identified: - the encoding of the source; - the source access (individuals recall the source relevant to the target); - the mapping of the source onto the target. As far as the last phase is concerned,

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 338–347, 2003.

the process is usually conceived as based on the induction of a schema from the source and on its subsequent transfer onto the target, so to suggest to apply the solution strategy described in the source to the target problem. The schema which is abstracted from the source is composed by the relevant elements which can be matched onto the target, that is, the elements that have corresponding counterparts in the target. It is generally assumed that the higher is the number of elements shared by the source and the target, the easier is the transfer of the schema from the source to the target. This leads to predict the rate of transfer on the basis of the number of the source-target correspondences.

2 Experiments about Analogical Thinking

The experimental procedure employed to investigate analogical problem-solving consists in presenting a target problem, preceded by a source - describing a situation structurally similar to the target - that has been previously solved by means of a set of strategies which can be applied also to the target problem. In experiments about analogical problem-solving Duncker's radiation problem has been often employed. Such a problem describes the situation of a patient with a tumour within his body; high-intensity x rays can destroy the tumour, but they destroy also the surrounding healthy tissue; low-intensity rays are harmless to the healthy tissue, but they fail to destroy the tumour. The problem points to find a procedure to destroy the tumour with the x rays without damaging the healthy tissue. According to the dispersion-concentration strategy, the solution of the radiation problem is achieved by sending several weak rays from different directions toward the tumour, so that they produce no harm in the healthy tissue and, by simultaneously converging, produce a high-intensity effect which destroys the tumour. This is a recent version of the radiation problem and of its solution:

The Surgeon
A surgeon was trying to destroy a cancer which was situated in the central region of a patient's brain, by using a type of ray. He needed to use these rays at a high intensity in order to destroy the cancerous tissue. However, at such an intensity the healthy brain tissue will also be destroyed. After considerable thought, he knew just what to do. He divided the rays up into batches of low-intensity rays, and by sending them, simultaneously, from a number of different directions, the converged on the cancer, making up a sufficiently high intensity to destroy it [2].

In the experiments on analogical problem-solving, in order to suggest the dispersion-concentration strategy, the target was preceded by an isomorphic story. These are examples of isomorphic stories that have been employed:

The Light bulb
In a physics lab at a major university, a very expensive light bulb which would emit precisely controlled quantities of light was being used in some experiments.

Ruth was the research assistant responsible for operating the sensitive light bulb. One morning she came into the lab and found to her dismay that the light bulb no longer worked. She realised that she had forgotten to turn it off the previous night. As a result the light bulb overheated, and the filament inside the bulb had broken into two part. The surrounding glass bulb was completely sealed, so there was no way to open it. Ruth knew that the light bulb could be repaired if a brief, high-intensity laser beam could be used to fuse the two parts of the filament into one. Furthermore, the lab had the necessary equipment to do the job. However, the laser only generated low-intensity beams that were not strong enough to fuse the filament. She needed a much more intense laser beam. She needed a much more intense laser beam. So it seemed that the light bulb could not be repaired, and a costly replacement would be required. Ruth was about to give up when she had an idea. She placed several lasers in a circle around the light bulb, and administrated low-intensity laser beams from several directions all at once. The beams all converged on the filament, where their combined effect was enough to fuse it. Ruth was greatly relieved that the light bulb was repaired, and she then went on to successfully complete the experiment [3].

The Missile
A general was trying to destroy an intercontinental ballistic missile, using a type of laser beam, which would be entering the atmosphere at great speeds. He needed to use a high-powered laser beam in order to destroy the incoming missile. However, such high-powered laser beam were inaccurate because they would heat up the air through which they passed, thus distorting the beam. After considerable thought, he knew what to do. He divided the high-powered laser beams into a number of low-powered laser beams and, by sending these low-intensity, simultaneously, from a number of different directions, they converged on the missile making up sufficiently powerful beam to destroy it [2].

The General
A small country was ruled from a strong fortress by a dictator. The fortress was situated in the middle of the country, surrounded by farms and villages. Many roads led to the fortress through the countryside. A rebel general vowed to capture the fortress. The general knew that an attack by his entire army would capture the fortress. He gathered his army at the head of one of the roads, ready to launch a full-scale direct attack. However, the general then learned that the dictator had planted mines on each of the roads. The mines were set so that small bodies of men could pass over them safely, since the dictator needed to move his troops and workers to and from the fortress. However, any large force would detonate the mines. Not only would this blow up the road, but it would destroy many neighbouring villages. It therefore seemed impossible to capture the fortress. However, the general devised a simple plan. He divided his army into small groups and dispatched each group to the head of a different road. When all was ready he gave the signal and each group

marched down a different road. Each group continued down its road to the fortress so that the entire army arrived together at the fortress at the same time. In this way, the general captured the fortress and overthrew the dictator[4].

The Parade
A small country was controlled by a dictator. The dictator ruled the country from a strong fortress. The fortress was situated in the middle of the country, surrounded by farms and villages. Many roads radiate outward from the fortress like spokes on a wheel. To celebrate the anniversary of his rise to power, the dictator ordered his general to conduct a full-scale military parade. On the morning of the anniversary, the general's troops were gathered at the head of one of the roads leading to the fortress, ready to march. However, a lieutenant brought the general a disturbing report. The dictator was demanding that his parade had to be more impressive than any previous parade. He wanted his army to be seen and heard at the same time in every region of the country. Furthermore, the dictator was threatening that if the parade was not sufficiently impressive he was going to strip the general of his medals and reduce him to the rank of private. But it seemed impossible to have a parade that could be seen throughout the whole country. The general, however, knew just what to do. He divided his army up into small groups and dispatched each group to the head of a different road. When all was ready he gave the signal, and each group marched down a different road. Each group continued down its road to the fortress, so that the entire army finally arrived together at the fortress at the same time. In this way, the general was able to have the parade seen and heard through the entire country at once, and thus please the dictator [5].

Red Adair
An oil well in Saudi Arabia exploded and caught fire. The result was a blazing inferno that consumed an enormous quantity of oil each day. After initial efforts to extinguish it failed, famed fire-fighter Red Adair was called in. Red knew that the fire could be put out if a huge amount of fire retardant foam could be dumped on the base of the well. There was enough foam available at the site to do the job. However, there was no hose large enough to put all the foam on the fire fast enough. The small hoses that were available could not shoot the foam quickly enough to do any good. It looked like there would have to be a costly delay before a serious attempt could be made. However, Red Adair knew just what to do. He stationed men in a circle all around the fire, with all of the available small hoses. When everyone was ready all the hoses were opened up and the foam was directed at the fire from all directions. In this way a huge amount of foam quickly struck the source of the fire. The blaze was extinguished, and the Saudis were satisfied that Red had earned his three million dollar fee [4].

The Fire Chief
One night a fire broke out in a wood shed full of timber on Mr. Johnson's place. As soon as he saw flames he sounded the alarm, and within minutes dozens

of neighbours were on the scene armed with buckets. The shed was already burning fiercely, and everyone was afraid that if it wasn't controlled quickly the house would go up next. Fortunately, the shed was right beside a lake, so there was plenty of water available. If a large volume of water could hit the fire at the same time, it would be extinguished. But with only small buckets to work with, it was hard to make any headway. The fire seemed to evaporate each bucket of water before it hit the wood. It looked like the house was doomed. Just then the fire chief arrived. He immediately took charge and organised everyone. He had everyone fill their bucket and then wait in a circle surrounding the burning shed. As soon as the last man was prepared, the chief gave a shout and everyone threw their bucket of water at the fire. The force of all the water together dampened the fire right down, and it was quickly brought under control. Mr Johnson was relieved that his house was saved, and the village council voted the fire chief a raise in pay [4].

The Artificial Lake
An engineer plans the construction of an artificial lake to produce electric energy. According to his first plan, a single wide canal collects water coming from a valley and conveys it into a lake. However, the engineer realises that during the flood periods the stream of water flowing along the canal may be too strong and may damage the surrounding area. He also realises that during the drought periods a single stream of water may be insufficient to feed the lake. In order to avoid these mishap, the engineer elaborates a second plan. According to this plan, the lake is fed by four small canals whose total flow is the same as the single wide canal previously planned. These small canals are placed around the lake so that they convey water coming from four different valleys. In this way only a small amounts of water can flow in each canal and thus during flood periods dangerous overflowing might not occur. At the same time, the lake is fed by water from various valleys, so that also during drought periods it is sufficiently fed [1].

3 Structure of the Analogical Stories

All these stories involve the same structure: a *goal*, consisting in sending a *tool* toward a central *location* in order to produce a certain *effect*, has to be achieved. However, some *constraints* impede the direct dispatch of the entire amount of the tool; the *solution* is reached by dividing the initial whole amount of the tool in sub-parts and by conveying them simultaneously onto the central point. This abstract structure - consisting of six entries (namely, goal, tool, location, effect, constraint, solution) - is differently implemented in each story. Even though there are some common elements (e.g., the tool is always something whose total amount can be divided into sub-amounts that can be reassembled; the convergence point of the tool is positioned in a central location; the amount of the tool which is sent toward the central point has to be enough to produce the desired effect; the solution requires locating the tool in different positions, simultaneous forwarding of the tool, and concentration of the tool in the central point), other elements differ from a story to another. For instance, in some stories the goal is to destroy what is located in the central point (the tumour or the fortress), whereas in other stories the goal is to allow something (the lightbulb or the artificial lake) to work. Table 1 reports the presence (= 1) or absence (= 0) of the various elements in each story. On the basis of the previously mentioned assumption, we can conjecture that stories sharing a high number of elements tend to be perceived as more similar than stories having a lower number of common elements. Thus, it is interesting to identify what is the whole pattern of similarities existing among the stories.

4 A Kohonen Map of the Analogical Stories

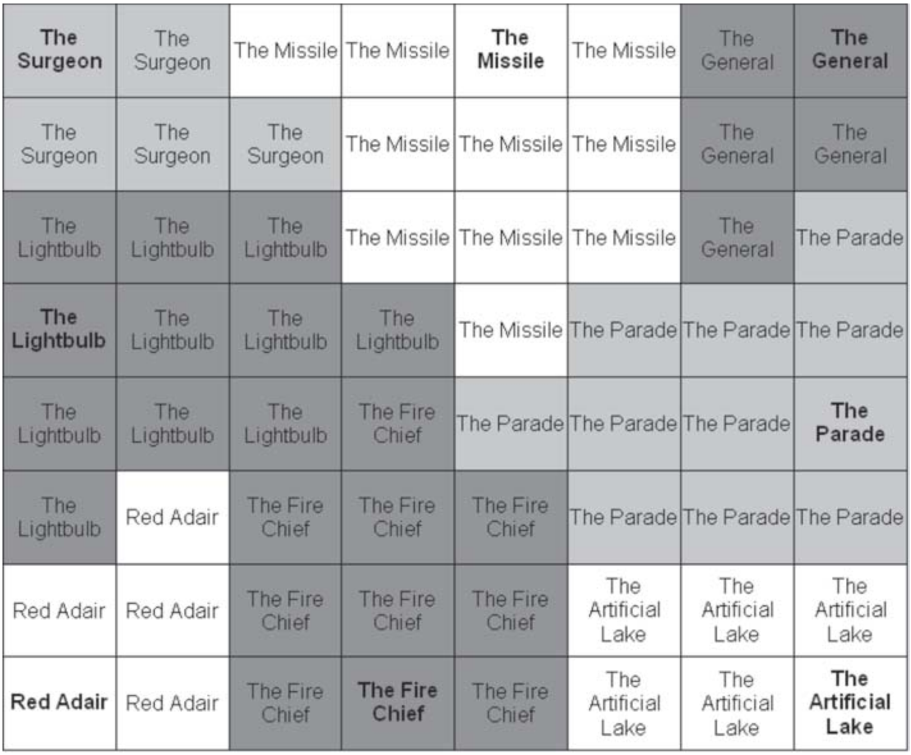
To do this, the matrix reported in Table 1 constituted the basis of the vectors (each vector corresponds to a story) of a neural network having 16 input nodes. The aim was to obtain a two-dimensional 8 X 8 map with 64 output nodes, each having 16 connections with the input nodes. A learning algorithm devised for the Kohonen maps was employed. In the first phase (ordering) of the learning procedure, the training set constituted by the vectors previously mentioned was presented for 1,000 epochs with the learning factor η equal to 1, the effective width σ equal to 8, and the temporal constant τ equal to 200. The parameters of the second phase (convergence) were as follows: epochs = 1,000; learning factor $\eta = 0.01$; effective width $\sigma = 0.01$ (τ was not used). At the end of the learning procedure, by presenting the vectors to the network, a specific predominant node, corresponding to a story, was found. Figure 1 depicts the surface of the map reporting the nodes specialised in recognising the different vectors (in bold) and the activation area around such nodes.

5 Psychological Reality of the Kohonen Map

We were interested in assessing if this pattern of similarities had any degree of psychological reality. In other words: persons perceive similarities among stories in the same manner as described by the Kohonen map? To test this, 20 adults were presented the stories reported above; they were asked to evaluate similarities among the stories and to write the titles of the stories on a paper sheet within a white square, by locating them so that the distance between each pair of stories corresponded to the perceived degree of similarity between them. A 8 X 8 grid was superimposed to the response sheets, so to obtain the co-ordinates (row number; column number: e.g., 5, 3) of each story. By assuming the story of the surgeon as pivot-story, the distance of each story from this story was computed by transforming the co-ordinates into Euclidean distances. Table 2 reports the distances between the story of the surgeon and each of the other stories computed on the Kohonen map previously described (first row) and the corresponding mean distance computed by considering the responses of the sample of adults (second row).

Table 1. Presence/absence of the elements of the schema in each story

Entry of schema	Element	Story							
		Surg	Lightb	Miss	Gener	Par	Red	Fire	Artif
Goal	-to produce no danger	1	1	1	1	1	0	1	1
	-to destroy or to conquest something	1	0	1	1	0	1	1	0
	-to allow something to work	0	1	0	0	1	1	0	1
Tool	-radiation (x-rays, laser beam, light)	1	1	1	0	0	0	0	0
	-liquid (water, foam	0	0	0	0	0	1	1	1
	-soldiers	0	0	0	1	1	0	0	0
Location	-directly available	0	0	0	1	1	1	1	1
Effect	-destruction or conquest	1	0	1	1	0	1	1	0
	-repairing	0	1	0	0	0	0	0	0
	-refilling	0	0	0	0	0	1	0	1
Constraints	-no danger along the path	1	1	1	1	0	0	0	1
	-no danger in the container	1	1	0	0	0	0	0	0
	the container can not be opened	1	1	0	0	0	0	0	0
	-the tool has to follow simulta-neously two or more paths	0	0	0	0	1	0	0	0
	-only small amounts of the tool can be conveyed	0	0	0	0	0	1	1	0
Solution	-dividing the tool	1	0	1	1	1	0	0	1



The Surgeon	The Surgeon	The Missile	The Missile	**The Missile**	The Missile	The General	**The General**
The Surgeon	The Surgeon	The Surgeon	The Missile	The Missile	The Missile	The General	The General
The Lightbulb	The Lightbulb	The Lightbulb	The Missile	The Missile	The Missile	The General	The Parade
The Lightbulb	The Lightbulb	The Lightbulb	The Lightbulb	The Missile	The Parade	The Parade	The Parade
The Lightbulb	The Lightbulb	The Lightbulb	The Fire Chief	The Parade	The Parade	The Parade	**The Parade**
The Lightbulb	Red Adair	The Fire Chief	The Fire Chief	The Fire Chief	The Parade	The Parade	The Parade
Red Adair	Red Adair	The Fire Chief	The Fire Chief	The Fire Chief	The Artificial Lake	The Artificial Lake	The Artificial Lake
Red Adair	Red Adair	The Fire Chief	**The Fire Chief**	The Fire Chief	The Artificial Lake	The Artificial Lake	**The Artificial Lake**

Fig. 1. Kohonen map of the analogical stories

Table 2. Distance between the Surgeon story and each of the other analogical stories in the Kohonen map and in adults' judgements

	The Lightbulb	The Missile	The General	The Parade	Red Adair	The Fire Chief	The Artificial Lake
Kohon. map	3.00	4.00	7.00	8.06	7.00	7.62	9.90
Human sub.	2.33	2.14	3.34	3.96	3.31	3.47	3.69

The correlation coefficient between the two kinds of measures resulted to be ρ = .92 (p = .04). So, the pattern of similarities produced through the Kohonen map tends to correspond to the similarity judgements produced by the human subjects. A further evidence of the psychological reality of the Kohonen map was provided by the cluster analysis carried out on the similarity ratings produced by subjects (see Figure 2).

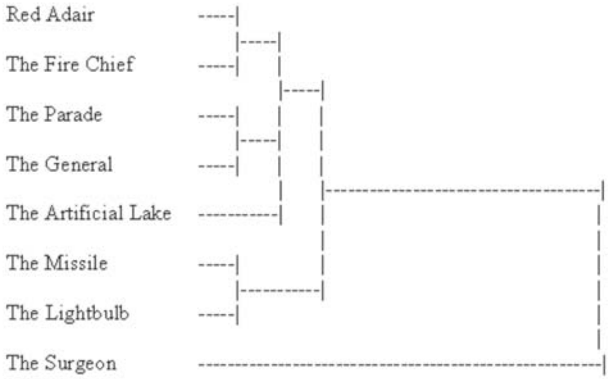


Fig. 2. Dendogram resulting from cluster analysis carried out considering adults' judgements of similarity among analogical stories

Finally, distance among the isomorphic stories measured on the Kohonen map were consistent with the rates of transfer recorded in experiments about analogical problem-solving. For example, stories - such as The Light bulb or the General - which resulted to be close to the surgeon problem, produced high percentages (respectively, 75% and 76%-100%) of dispersion-concentration solutions in the radiation problem when they were presented as sources; conversely stories located far from the surgeon problem induced low rates of analogical solutions (e.g., the Artificial Lake story suggested the dispersion-concentration solution to the radiation problem only to the 19% of subjects).

6 Conclusions

The analyses carried out showed that the application of a schema - consisting of entries corresponding to possible matching elements - to the set of source stories usually employed to investigate analogical problem solving succeeded in depicting an overall pattern of similarity among the stories. Such a pattern was produced by considering each story as a vector and by attributing to each story values according to the presence/absence of the various elements of the schema. The resulting matrix constituted a neural network which yielded, as outcome of a learning procedure, a map representing the distribution of the analogical stories according to their relative degrees of similarity. It is worth noticing that such a map appeared to mirror similarity judgements produced by individual who were asked to evaluate similarity among the stories and to predict the rates of transfer of the solution principle embedded in each source story to an analogous target problem.

References

1. Antonietti, A.: Analogical Discoveries. Carocci Roma (2002)
2. Keane, M: On Retrieving Analogues When Solving Problems. In: The Quarterly Journal of Experimental Psychology, Vol.39A (1987) 29-41
3. Holyoak, K.J., Koh, K.: Surface and Structural Similarity in Analogical Transfer. In: Memory & Cognition, Vol.15 (1987) 332-340
4. Gick, M.L., Holyoak, K.J.: Schema Induction and Analogical Transfer. In: Cognitive Psychology, Vol.15 (1983) 1-38
5. Gick, M.L., Holyoak, K.J.: Analogical Problem Solving. In: Cognitive Psychology, Vol. 12 (1980) 306-355

Formats and Languages of Knowledge: Models of Links for Learning Processes

Maria Rita Ciceri

Centre for Studies and Research in Communication Psychology
Department of Psychology, Catholic University of Sacred Heart
Largo Gemelli 1, 20123 Milano

Abstract. In this paper we will examine if communicability is a structural constraint on our information representation, acquisition and processing. In particular, first of all we will examine the description of the different formats of knowledge: declarative, propositional, procedural knowledge, narrative thinking, mental imagery. We will then consider the possible links between language and thought and especially between formats of knowledge, languages and the multimodal message. As a matter of fact the relationship between language and thought can be described by two events: a referential and a communicative one. Through the referential event the information we experience (perceptual, psychological, cultural) is organized into knowledge (knowledge of words, images, procedures); the communicative event requires the declension into languages through a medium, generating the multimodal message.

1 On Thought Communicability

In his work Barlow [1] points out that the specific property of neurons is their interconnection that is their way of conveying information. The functionality of our neural cells can be measured in terms of speed, simultaneity of information reception and transmission. Barlow suggests that *human brain works while communicating.* Paraphrasasing Barlow, we may ask ourselves if even *our mind works while communicating*; thus in this paper we will examine if communicability is a structural constraint on our information representation, acquisition and processing. According to this assumption, the extraordinary human faculty which is thought - that is our faculties of planning, categorizing, narrating just to mention a few- won't be examined as the charming mechanism of a "monad mind", considered as an unbelievable and isolated thinking machine. But, on the contrary the object of our analysis will be the activity of a "relational mind" within an unending exchange of information from inside to outside and among its components. In particular, we will examine: 1. the description of the different formats of knowledge 2. the possible links between language and thought; 3. the possible links between formats of knowledge, languages and the multimodal message (verbal, non-verbal, graphic...) which is the output of the act of communication.

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 348–357, 2003.

2 Different Formats of Knowledge: Words, Images, Procedures and Narratives

In the last 30 years psychology and in particular cognitive psychology has identified and described different formats of knowledge: declarative and propositional thinking, mental imagery and narrative thinking. They are different systems of organization and representation of the information.

2.1 Declarative Knowledge

Declarative knowledge is defined as the whole knowledge about the world which is permanently available in long term memory. It also includes the situational knowledge, that is knowledge about the present context and the new information that is continuously processed by working memory. Declarative knowledge is the map of reality and of all the experiences made by the subject during his life. As a matter of fact it is concerned with *what* is an object, an event,.. In this sense it involves both encyclopedic knowledge about objects and knowledge categorized in concepts, classes, sets. The nature of this kind of knowledge is given by the declaration: x is equivalent to y, j, k where x is defined by one or more defining features. Declarative knowledge carries on a referential and predicative function. It means that it enables us to denote and to predicate something about $x$and to put all these information in relation among themselves. Declarative knowledge can consist of propositional or spatial representations.

2.2 Propositional Knowledge

Propositional knowledge deals with facts and it may be generated even after a single experience. We can divide propositional knowledge into episodic and semantic knowledge. The first one refers to propositions about experience or episodes happened in past, in which the spatio-temporal coordinates are explicit (as for example in the proposition "Yesterday I went with my dog to the park"). The second one includes propositions where the spatio-temporal coordinates are no longer considered , as for example in the proposition "Dogs are taken by their owners to the park". Propositional structures organize and represent the world of experience through categories and elements (like person/object/time and place/action/manner) and define their function within the propositional structure (predicate, subject, adverbial, modifier) and the relationships among propositions (coordination, subordination, ...). More in detail, the propositional or phrasal structure is semantic and it doesn't depend on the translation into a specific language but it is given by the structure of predication, where the argument A is taken by the predicate p.

2.3 Mental Imagery

Mental images are representations that enable the experience of "seeing" something even without the real presence of the corresponding visual stimuli or of recognizing images. Like actual vision, mental imagery performs duties of recognition or of location or of reconnaissance. Following this approach, which has been defined as "*analog-computational*", Kosslyn and his collaborators have recently [2] defined mental images as temporary spatial displays in active memory that are generated from a "matrix", that is from more abstract representations in long term memory. According to the metaphor they employed, mental images might be like displays produced on a cathod-ray tube by a computer program operating on stored data. A great contribution to the analog-computational hypothesis comes from cognitive neuroscience. The discovery of topographically organized visual areas[1] has confirmed the hypothesis according to which mental images correspond to activation patterns at least in some topographically mapped areas of visual cortex.

2.4 Narrative Thinking

Only a short reference to narrative thinking will be presented here, because to this format of knowledge we will entirely dedicate the next paper. In this introductive section, we only intend to underline its differences and features compared to other formats. This format of thought has been a constant topic in psychological studies of the last 20 years, particularly within cognitive and interactionist psychology. Narrative thinking is now considered one of the modes of human mental functioning. The great number of studies dealing with it bring into relief two main aspects. The first one is related to its *interpretative dimension*: narrative thinking plays a mediating role between experience and who is narrating it. For that reason it is not bound to reality, but it offers a point of view on experienced and recreated reality. The second important aspect lies in its *episodic dimension*: narrative thinking deals with events, facts, episodes and therefore it has a spatio-temporal and causal organization, which is typical of narratives, as follows: *Event 1* → *Event 2* → *Event 3*.. [4] [5] [6] [7].

2.5 Procedural Knowledge

The last kind of knowledge we consider is related to action. Procedural knowledge is not concerned with "what to do" but with "how doing it". Usually its acquisition is slower, because it requires experience and exercise. It guides our acting in the world and therefore it involves the functional use of objects and the acquisition of effective action procedures. Procedural knowledge can be represented in the following form: If x, then y. Although the different formats of knowledge are characterized by functional specificity (they have functions of naming, depicting, narrating and coordinating actions), they are overlapping and often interdependent. The overlap is partly due to the sharing of perceptual experience. As a

[1] The first brain area that receives input from the eyes is V1area, known as primary visual cortex, striated cortex, OC and area 17. In 1986 Fox [3] used PET (Positron Emission Tomography) to demonstrate the existence of V1 area even in human beings.

matter of fact information related to the same event is organized in a specific way within the different formats (propositions, mental images, episodes, procedures). On the contrary interdependence is the rule when the subject acts in his field of experience. For example, when communicating we apply the procedure IF X, THEN Y to a proposition. Referring to the communicative action this formula becomes: "If the intention is that of undertaking to say *p*, then say *p*". In the same way, a narration can be declined in a propositional or a depictive way (sentences or images) although it doesn't correspond to either of them as regards the way of the organization of information.

3 How Language Helps Us to Think

Knowledge, which is organized in different formats of thought, is expressed through language. The relationship of interdependence between language and thought has been considered within a lot of different approaches.

3.1 Language as an Independent System

A first approach is the structuralist vision. According to it, language is independent of thought. Language is thus defined as an independent representational system. This approach, which is properly linguistic, is based on de Saussure [8] and Hjemslev's [9] theoretical hypotheses and finds a more recent statement in Rastier's [10] theory of difference. Language can be described as a system of internal relationships, a network of dependences, whose elements exist connected one to another and are completely independent of any determination from the outside. According to de Saussure, the first duty of linguistics exactly consists in isolating a restricted and self-sufficient entity, "the system", that is the language conceived as a whole and the principle of classification, from all the multiform phenomena that make up language (physic, psychic and social phenomena). Therefore the meaning of a word has to be searched within the linguistic system only, completely aside from any other factor: the perceived world, mental and conceptual reality and the medium chosen to communicate as well. Thus the meaning of a word (which is defined within the language) and the concept the word refers to (which belongs to the sphere of thought) are considered two different phenomena.

3.2 The Mediating Function of Language between Experience and Conceptualization

Unlike the structuralist perspective, the cognitive approach denies the autonomy of language from the conceptual structure and it identifies elements of interdependence at different levels. Attention shifts from the analysis of the structure of the language taken as a whole to the study of comprehension processes that lie below the system and its use. The study of meaning is integrated and considered strictly linked to the study of mental processes through which these contents are built up and learned. Explain how we understand is explaining how we mean. The meaning of the language can not be separated from our experience of the world, which is physic, perceptual, psychological, mental, cultural and social. The *conceptualization* (performed by thought) and the expression of meaning (performed through language) are two moments strictly interdependent, such as it is explained in Fillmore's [11] model, in Fodor [12] and Jackendoff's [13] hypotheses. According to Jackendoff, the conceptual structure is consistent with information coming from the different representational systems of our cognitive apparatus: linguistic, motor, visual, auditory, kinesical,... and constitutes the core, as it is showed in Figure 1.

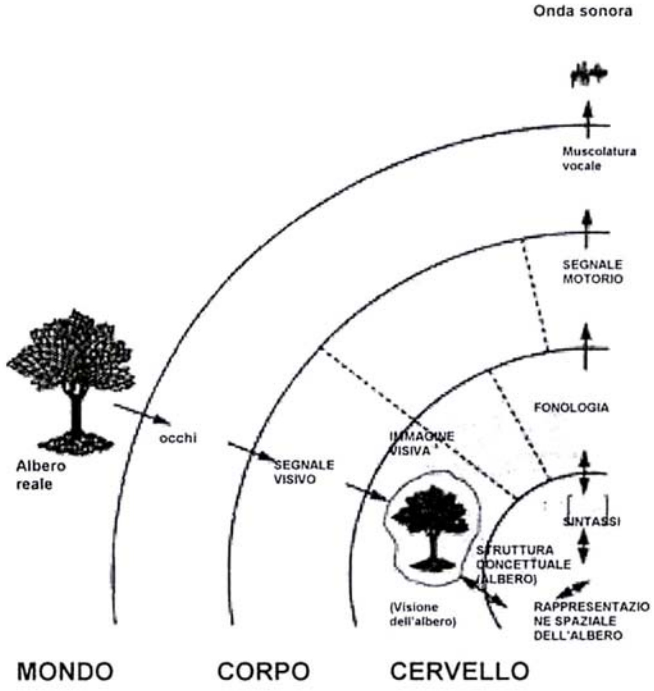


Fig. 1. Modular representation. The different interfaces and the central core of representation (taken from Jackendoff [13], partially changed)

All these systems support a process of modular representation, where they have the function of interfaces between the module and the central system. For that reason, the author says, we are able to speak about what we hear and see [13] operating the passage from senses to sense. In this model language represents the conscious part unlike the others mental processes of thought. Therefore language, which is a system specialized in communicating, is one of the most advanced functions of our mind and it is able to express and thus to make conscious both the data processed at a perceptual level and the contents of long term memory and their mental grammar.

3.3 "Dialogical Thinking"

The functional relationship among experience, conceptualization and expression is considered through different explicative criteria by the approach of cultural interactionism. Even in this case we are not making reference to an univocal theoretical perspective, but rather to a family of theories that have in common some basic assumptions. Within this approach we can mention authors like Bachtin [14], Vygotskij [15], Bruner [16], Fogel [17] and Searle [18]. The focus of attention shifts from mind to the subject-in-interaction with the world around him and with other subjects. Thought and language are not considered and analysed as faculties whose structure and potentialities are to be described, but they are considered in their happening, as actions, or better interactions within the continuous stream of communication. Fogel [17] believes that "mind's life is a dialogue, or better a verbal dialogue among imagined points of view continuously changing". Knowledge and memory are not encoded by cognition as representations of abstract physic properties of the objects, but rather as the form of the relationship between perception of the individual and his action upon the objects and with others individuals. Thus what becomes essential is the description of this gradual action of co-ordination and co-regulation between the individual and the world and among individuals themselves. Attention shifts from the instrument-system to the analysis of the act of communication. The strategies of co-regulation, the ways of segmentation of the communicative stream into *frames*, the analysis of all the system of representation and of verbal and nonverbal communication that speakers use become the new object of study. This approach gives a great importance to the role of culture and to the relationship between the interaction and Self-construction [19]. Thought is described in its narrative and interpretative dimension and in its inter-subjective function. Among its functions, it is considered not only the explanation of phenomena but even the understanding and interpretation of one's own and other people's human experience.

4 Referential and Communicative Event: The Multimodal Output

While the cognitive models we considered above bring into relief the relationships among language, thought and the other processes of perception and elaboration of information in the exploration of the world, the dialogic perspective suggests that language can not be reduced to the mediating function between conceptualization and experience, because language is even generator of and it is generated by acts of communication. Tomasello [20] sums up this assumption defining the relationship between language and thought as described by two events: a referential and a communicative one. Through the referential event the information we experience (perceptual, psychological, cultural) is organized into different formats of knowledge (propositional, procedural, narrative knowledge and mental imagery). In this way they are formulated according to the semantic and syntactic rules of a language into sentences, images, action procedures, narrative structures (see Figure 2).

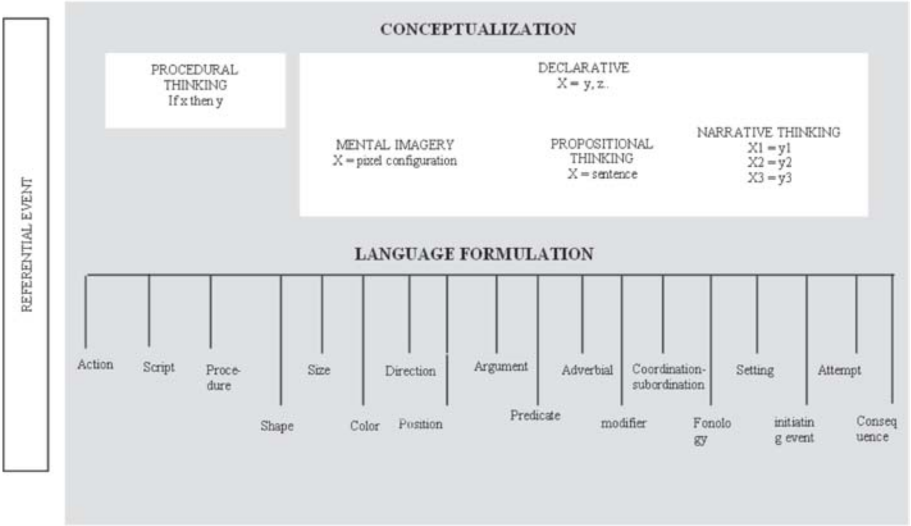


Fig. 2. The referential event: conceptualization and language formulation

The communicative event corresponds to the action of articulation and output of the message. It synergically activates the declension into languages through a medium, generating a multimodal interactive act/message as it is showed in Figure 3.

Referring to what happens in the referential event, Kosslyn describes how it is possible to store and convey the same information "the ball is on the box" through different formats of representation with specific semantic and syntactic structures. For example, the syntax of the propositional representation "on (ball, box)" consists of linguistic signs having function of arguments and predicates. On the other way semantics is constituted by the relationship symbolically expressed by those signs. Therefore a propositional representation is an abstract representation, because it can be referred to classes of objects or to abstracts entities in general and it is not bound to any perceptual modality. In the corresponding mental image syntax is constituted by *pixels* and by the space where they are placed. The rules of syntactical combination only require that pixels are spatial related one to another. On the other way semantics is given by the correspondence among parts and relations that the pixel configuration maintains with the original scene: this means that it is constituted by the relationship of similarity between representative and represented as regards shape, size, posi-

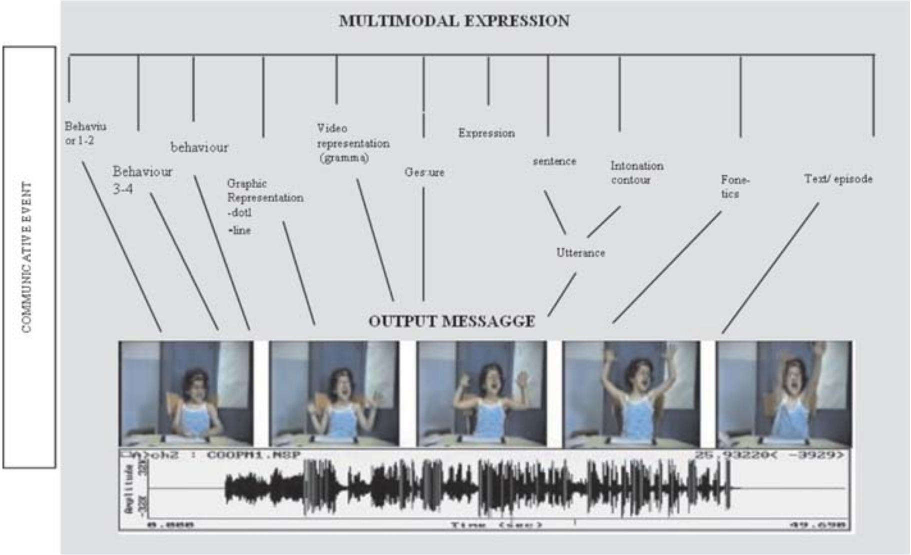


Fig. 3. The communicative event: multimodal expression and output message

tion. On the contrary in the communicative event mental imagery in its syntactic and semantic aspects is declined into a grammar by figurative languages (drawing, cinema, comics) where the minimal unit is not the phoneme, but the line, the pixel, the "gramma" [21]. Within these languages of communication, the image and its dimensions become the starting point, the material that is available to re-create portions of meaning and of reality and to communicate meanings. It's not only a matter of remembering images or of being able to reconstruct and recover them in one's mind, but of creating and connecting them to communicate. In the same way, the phrasal structure of propositional thinking is translated into audible sounds to communicate utterances. Moreover the multimodal communicative output can use facial expression and gestures that require the coordination of procedural knowledge, depictive representation and, in sign language, even phrasal structure, as showed in Figure 3. The perceptual action on the world mediated by our mental activity is the starting and arrival point for the construction of more and more advanced and formalized systems for the conceptualization, interpretation and communication synergically activated within the communicative event. The notion of communicative competence or faculty itself is completely changed: it is no longer seen as a cumulative set of abilities, but rather as the componential and processual production of symbolic and multimodal objects [22] [23] through the synergical concurrence of a range of formats of knowledge, of languages, of expressive modalities and of the competence in techniques of information transmission. It is clear the passage from the notion of language to that of languages, that is systems of representation and signalling. Therefore we refer not only to linguistic systems but to all semiotic systems (verbal, figurative, kinesics). In the same way there is evidence for the distinction and the continuous interaction among the different formats of knowledge, semiotic systems and media. The first ones – declarative knowledge, mental imagery, narrative and procedural thinking - are systems of mental representation. Languages are effective and actual codes, that can be more or less stable; they can deal with different fields such as music, numbers, gestures, words, prosody. Finally, media are all those instruments that have a communicative function and that consist in technologies to convey information, participate and share symbols [24]. The distinction between systems and medium is quite useful to detect the relations between communicative abilities and cognitive development, because it enables to detect not only the influence of the increasing ability of representation and symbolization on development itself but also the influence of the media that spread them. The medium is not an inert matter through which a message expressed by a code can be spread out. The choice of the medium, whether it is drawing, or a cinema motion picture or wrinting or a computer, involves different modalities of perception, it changes the format of knowledge involved and moulds the semiotic system itself. The importance of these issues for learning theories is great. Their investigation could in fact allow to come to a more complex definition of cognitive capabilities and to describe the complex and multiple connections among different fields of competence. The aim of this symposium is to provide a contribute to the discussion through the comparison among different formats of thought and knowledge: narrative thinking, the analogy, the comparison between propositional thinking and mental imagery, emotional competence.

References

1. Barlow, H.: Communication and Representation within the Brain. In: Mellor, D.H. (ed.): Ways of communicating. The Darwin College lectures. Cambridge University Press New York (1990)
2. Kosslyn, S.M., Rabin, C.: The Representation of left-right Orientation: A Dissociation between Imagery and Perceptual Recognition. In: Visual Cognition Vol. 6 (1999) 497-508
3. Fox, P.T., Mintun, M.A., Raichle, M.E., Meizen, F.M., Allman, J.M., Van Essen, D.C.: Mapping human visual Cortex with Positron emission Tomography. In Nature Vol. 323 (1986) 806-809
4. Labov, W., Waletsky, J.: Narrative Analysis: Oral Version of Personal Experience. In Helm, J. (ed.): Essays on the verbal and visual arts. University of Washington Press Seattle (1967)
5. Bruner, J.S.: Actual Mind, possible Word. Harvard Univ. Press Cambridge- London (1986)
6. Schank, R.C., Abelson, R.A.: Knowledge and Memory: the Real Story. In Wyer R.S., Srull T. (eds.): Advances in social cognition. Vol. 8 LEA Hillsdale (1995)
7. Bamberg, M.: Narrative development. Six Approches. LEA Mahwah (1997)
8. De Saussure, F. : Cours de Linguistique Générale. Payot Paris (1916)
9. Hjelmslev, L.: Prolegomena to a Theory of Language. University of Winsconsin Madison (1943)

10. Rastier, F.: Semantique et Recherches Cognitives PUF Paris (1991)
11. Fillmore, C.: Towards a descriptive Framework for spacial Deixis. In Jarvella R., Klein, W. (eds.): Speech, Place and Action. John Wiley & Sons London (1982)
12. Fodor, J.A.: The Language of Thougth. Harvard University Press Cambridge (1975)
13. Jackendoff, R.: The Architecture of the Language Faculty. MIT Press Cambridge, Mass. (1997)
14. Bachtin, M.: Il linguaggio come pratica sociale. Dedalo Libri Bari (1975)
15. Vygotskij, L.S.: Myülenie i Rec Socekgiz. Moska-Leningrad (1934)
16. Bruner, J.S.: The Culture of Education. Harvard University Press (1996)
17. Fogel, A.: Developing through Relationships: Origins of Communication, Self, and Culture. Harvester Wheatsheaf New York (1993)
18. Searle, J.R.: Mind, Language and Society. Basic Books New York (1998)
19. Duranti, A.: Linguistic Anthropology. Cambridge University Press Cambridge (1997)
20. Tomasello, M.: Language is not an Instinc. In: Cognitive Development, Vol. 10 (1998) 131-156
21. Deleuze, G.: L'immagine- Movimento. Unilibri Milano (1984)
22. Ciceri, R. (a cura di): Comunicare il pensiero, Omega, Torino (2001)
23. Sanders, R.E.: The Production of Symbolic Objects as Components of Larger Wholes. In: Greene, J.O. (ed.): Message Production: Advances in Communication Theory. Erlbaum Mahwah N.J (1997)
24. Olson, D.R., Torrance, N. (eds.): Modes of Thought. Cambridge University Press Cambridge (1996)

Pictures and Words: Analogies and Specificity in Knowledge Organization

Paola Colombo, Carla Antoniotti, and Maria Rita Ciceri

Centre for Studies and Research in Communication Psychology
Department of Psychology, Catholic University of Sacred Heart
Largo Gemelli 1, 20123 Milano

Abstract. In this paper we want to examine what kind of relationship exists between two semiotic systems, word and picture, and the formats of conceptual representation linked to them: propositional thinking and mental imagery. The specific aim of this research is to explore the characteristics of the representation and organization of knowledge referring to the meaning of an input-word ("elephant"), which we have considered in both types of communication: verbalization and graphic representation. The current study analyzes the verbalizations and the graphic representations of 75 pre-school children when using the input-word "elephant". Data seem to match the hypothesis of differentiation, according to which language plays the role of a filter, so that the choice of a language (verbal or iconic) activates preferential connections to the format of representation more similar to it.

1 Introduction

When processing new information and organizing it into knowledge, the mind uses different formats of representation. In the last decades cognitive psychology has described several *formats of knowledge*, as for instance declarative knowledge, narrative thinking, mental imagery, procedural knowledge, propositional thinking, just to mention a few [1–3]. The current studies agree in supporting the hypothesis of their functional specificity (categorization vs narration vs imagery) and their structural specificity (symbolic vs space-analog). These different formats of representation of knowledge also have different neuro-physiological and neuro-psychological bases: modern neuro-psychology has described specific cerebral structures and circuits for any of the above mentioned forms of representation. What we call representations, or mental symbols, are but patterns of neuron activation [4, 5], which can contain different kinds of information to be processed in specific ways, such as sense-perceptual, conceptual and linguistic [6–8].

In particular, in this paper we want to examine what kind of relationship exists between two semiotic systems, word and picture, and the formats of conceptual representation linked to them: propositional thinking and mental imagery. Propositional thinking has been examined as one of the representational forms of declarative knowledge. It controls the use of verbal language in a process that

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 358–366, 2003.

goes from the conceptual to the linguistic representation of syntactic categories to come to the actual articulation of an utterance [9]. This sequence emphasizes the important distinction between the referential event, which is related to mental representative structures, and the communicative event, which is related to the communicative act through the use of one or more semiotic systems [10]. The current studies in the field of cognitive and functionalist linguistics go beyond the dichotomy between generativist and interactionist theories and suggest the existence of a natural language constituted by general linguistic structures that operate at a categorial level and are strictly connected with the experiential dimension. In particular, those theories identify universal and recursive structures that control the use of all languages and are constituted by abstract schemes and common recursive syntactic categories such as predicates, subjects and their links [10–17]. Mental imagery is conceived as a particular space representational system of declarative knowledge. According to the analog-computational approach, it is a kind of mental representation that enables the experience of "seeing" something even without the real presence of the corresponding visual stimuli. It has perceptual roots but it is the product of a mental reconstruction that combines it with the information already stored in memory. It is analog because it includes the space and topological features of the represented object, such as shape, size, color, direction, relations among its parts.

It is of great interest for cognitive psychology and especially for learning sciences to analyse the links between the referential and the communicative event and therefore relationship between language and the forms of mental organization of information (formats of knowledge). Next to the model of language or mental grammar [18], which supports the function of interface and of mediator between perceptual and conceptual processes of mental language (from senses to sense), it has been suggested the hypothesis according to which language plays the role of a filter, so that the choice of a language (verbal or iconic) activates preferential connections to the format of representation more similar to it, as showed in figure 1 [1, 3]. All the issues presented above seem to refuse the existence of an univocal link between a communicative event mediated by a certain language and the corresponding conceptual representation, and to suggest the hypothesis of a network of preferential activations that link the information according to specific paths of meaning, as it is showed in the diagram below.

Considering the above premises, this study aims at exploring the specific modalities through which a word collects and organizes knowledge when two different semiotic systems are to be used, namely verbal language and graphic language. We consider here the word in its function of 'pointer', as able to activate in the mind different nets of information as soon as it is 'launched', depending on the selected organizing system [19, 20]. Starting from the hypothesis that the two forms of representation of knowledge do have different structural features, which may to various extent influence the amount and type of knowledge communicated, the specific aim of this research is to explore the characteristics of the representation and organization of knowledge referring to the meaning of an input-word (*elephant*), which we have considered in both types of com-

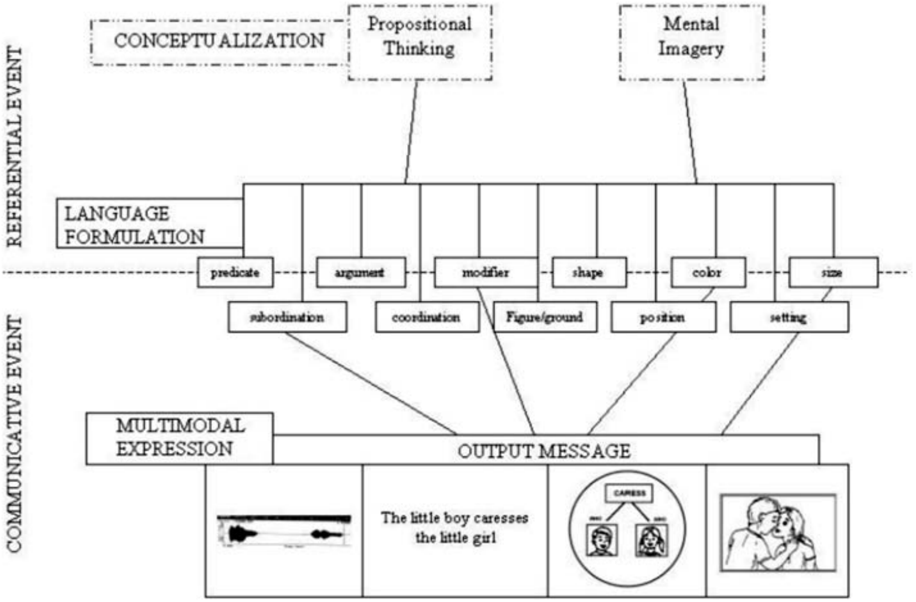


Fig. 1. Referential and communicative event: relationship between multimodal expression and the formats of knowledge (propositional thinking and mental imagery)

munication: *verbalization* and *graphic representation*. We therefore intend: a) to analyze the specificities of either communicative modality; b) to verify the possible correlation and complementariness between the two different communicative modalities.

2 Method

The current study analyzes the verbalizations and the graphic representations of 75 pre-school children when using the input-word *elephant*. The group was balanced as for age (3 ,4, 5 year-old) and sex. The task has been divided up into two phases: verbal launch and graphic launch. The order of presentation of both tasks has been balanced in the whole group.

Criteria of Analysis. The productions have been analyzed taking into consideration common and specific indicators for the two communicative modalities.
Verbal Launch: a) *Lexical items*: number of verbalized items; b) *Encyclopedical items*: number of activated items of information; c) *Level of Meaning Organization* (L.M.O.): knowledge can be organized in different ways, each characterized by a higher level of generalization and abstraction, with reference to the subjective experience [21]: - *episode*: the reference is to one's own experience, and the contest plays a major role, - *event*: the level of generalization is higher, the contest is less important, - *elements*: a list of traits and features which contribute to

Table 1. Mean of the number of lexical and of encyclopedic items for each age level

			L.M.O.			Sentence Structure					
	L.I.	E.I.	Ep	Ev	El	Pred	Arg	Modif	Adv	Typ	Ex.R.
3 ys	4,48	1,36	1,00	10,00	8,00	0,48	0,20	0,52	0,08	0,04	0,04
4 ys	10,36	3,12	0,00	12,00	14,00	0,92	0,36	1,56	0,08	0,00	0,20
5 ys	13,52	4,20	0,00	19,00	21,00	1,36	1,16	1,48	0,16	0,00	0,12
Total	10,43	2,89	1,00	41,00	43,00	0,92	0,57	0,71	0,11	0,01	0,12

the definition and classification of a word; d) *Semantic Structure*: - *predicates*: number of actions, - *modifies*: number of adjectives, - *adverbials*, - *arguments*: number of subjects and objects; e) *Superordinate or Subordinate level* [22]: - kind of elephants, - external relationships: class.

Graphic Launch: a) *Graphical items*: number of drawn items regarding figure and ground; b) *Encyclopedical items*: number of such parts of the drawing describing and defining figure, a stereotyped ground and a pertinent ground; c) *Number of elements* with form and content; d) *Action*: presence or absence of a movement or an action performed by the elephant; e) *Shape*: - animal, - antropomorphic image, - other.

Common Indicators: a) number of lexical items / number of graphical items; b) knowledge: number of expressed pieces of information; c) external relationships/ shape; d) predicate/action; e) arguments/number of elements.

3 Data Analysis

3.1 Analysis of Specific Indices: The Verbal Launch

Number of Lexical Items (L.I.) and Encyclopedical Items (E.I.). As far as the total mean are concerned, we notice a gradual and progressive increase in the amount of information depending on the age of the subjet (see table 1). Data have been statistically analysed, taking into account the variations of scores depending on age (3, 4, 5). Statistical analysis highlights the significance of the observed differences as regards any single factors as well as their interaction (Lexical Item: df=2; F=13,74;p¡0,01; Encyclopedic Items: df=2; F= 11,56; p¡0,01).

Sentence Structure. The analysis of the average scores shown in Table 1 highlights that arguments, predicates and modifiers, do prevail, though to different extents according to the subjects' age. Statistical analysis (ANOVA) gives evidence of the significance of the principal effect of a factor within subjects (sentence structure: F=74,84; p¡0,01; df=3), between subjects (age: F=21,16; p¡0,01; df=2) and of their interaction (sentence structure*age: F=3,65; p¡0,01; df=6).

Level of Meaning Organization (L.M.O.). When observing the scores referring to frequencies in the three levels, it is possible to point out that subjects generally tend to organize those items of information recorded in Events (Ev.) or in "Lists of traits" (Elements - El.), whereas Episodes (Ep.) do generally show very low

Table 2. Means of the number of graphical and encyclopedical items and frequences distribution of Action and Shape for each age levels

	Graph. items		Encyclop. items			Action		Shape		
	Eleph	Cont	Ele.	p.c.	st.c.	No	Yes	An.	Ant	Oth
3 ys	12,80	7,32	3,88	0,12	0,24	19,00	6,00	11,00	6,00	8,00
4 ys	14,92	13,16	6,68	0,28	1,64	18,00	7,00	20,00	3,00	2,00
5 ys	13,60	34,20	8,36	0,92	3,84	11,00	14,00	25,00	0,00	0,00
Total	13,77	18,23	6,31	0,44	1,91	48,00	27,00	56,00	9,00	10,00

frequencies. Statistical analysis (log-linear model) confirms the significance of the differences between the three levels of the factor (LOS: chi^2=14,99; p¡0,01; df=2) and of its interaction with the age factor (age*LOS: chi^2=10,29; p¡0,01; df=4).

3.2 Analysis of Specific Indices: The Graphical Launch

Number of Graphical and Encyclopedical Items. The chart shows the average number of details drawn, distinguishing between the subject of the drawing (elephant) and the context. Data highlight a few interesting differences: the group of 3 and 4-year-old subjects tend to reproduce a higher number of elements of the elephant and a lower number of the context, whereas 5-year-old children do the opposite. As for encyclopedical items, it increases with age, whereas in both contexts we see very few elements pertinent to the typical context of the input-word. Statistical analysis confirms the significance of the observed differences, with the exception of the factor within subjects as far as vocabulary index is concerned (Graphical items - Figure/ground: Df = 1; F=3,41; p=n.s.; Age: Df = 2; F=14,04; p¡ 0,01; Fig/ground*age:Df = 2; F=11,42; p¡ 0,01. Encyclopedical Items - Figure/ground: Df = 2; F=296,11; p¡ 0,01; Age: Df = 2; F=328,59; p¡ 0,01; Fig/ground*age: Df = 2; F=25,284;p ¡ 0,01).

Kinds of Information: Shape and Action. The chart shows the distribution of frequencies of drawings showing an action, according to the three types of representation (animal, anthropomorphic, other). Prevalence of drawings where the subject is not shown in action is to be noticed. However, *active elephants* do appear mostly in the drawings made by 5-year-old children. Furthermore, the frequency of drawings where we clearly recognize an animal representation is generally high (56/75). In particular, a discriminating factor of such a competence seems to be represented by age: the number of drawings belonging to the "animal" category icreases with age, while the other types of representations decrease accordingly. Statistical analysis (chart 7) confirms the significance of the differences between the frequencies of the levels of the factors within subjects (Shape: chi^2 =5,94; p¡0,05; df=1; Action: chi^2 =17,16; p¡0,01; df=2), though not of the factor age. The interaction between factors within subjects and factor age appears to be significant in both cases (Shape*age: chi^2 = 6,53; p¡0,05; df=2; Action*age: chi^2 = 6,30; p¡0,05; df=2).

3.3 Analysis of Common Indices: A Comparison between Verbal and Graphic Indices

Vocabulary and Information. The mean number of both vocabulary and encyclopedic items is altogether higher in drawings than in verbalizations (see graph 2). The mean data referring to the drawings are always higher when the drawing activity follows verbalization, though the difference is here less relevant. For both indices statistical analysis confirms the significance of the differences in average for a factor within subjects, with reference to the kind of task given (number of lexical and graphical items: F=158,286 p¡0,01 and df=1; encyclopedical items: F=158,29; p¡0,01; df=1). In relation to the factor between subject, the factor 'age' is meaningful in both cases (number of lexical and graphical items: F=27,488, p¡0,01; df=2; encyclopedical items: F=27, 49; p¡0,01; df=2), whereas the order of presentation of the tasks proves to be irrelevant.

Complexity: Articulation of Information. The chart shows data concerning three kinds of information in both assignments (verbal and graphic), taking into account the order of presentation (see tabel 3).As far as the external relationship and shape information is concerned, we can notice a more relevant presence in the drawings. Moreover, considering the order of the two tasks for each subject, an interesting datum emerges: the number of subjects spontaneously expressing such information is always higher whenever the graphic task comes before the verbal one. On the contrary, the pieces of information concerning predicates tend to be more verbalized. What's more, in all age groups the presence of a figure shown in action is generally followed by a corresponding verbal explanation, but the opposite is not true. As for arguments, the data reported in the chart show a higher number of them in the drawings, once more confirming the tendency already described for the other indicators: verbalization is richer in details if preceded by graphic representation. Statistical analyses applied to single indices of information (animal, predicates, arguments) generally show the significance of the differences observed between the two different types of assignment (Class/shape: F=7,56; p¡0,01; df=1; Predicates/actions: F=2,53; n.s.; df=1; Arguments: F=461,22; p¡0,01; df=1) and its interaction with the factor age (Class/shape: F=1,28, p=n.s.;df=2; Predicates/actions: F=7,89; p¡0,05; df=2; Arguments: F=25,65; p¡0,01; df=2), whereas neither the order of recording of competences nor the interaction between factors seems to be relevant.

4 Conclusions

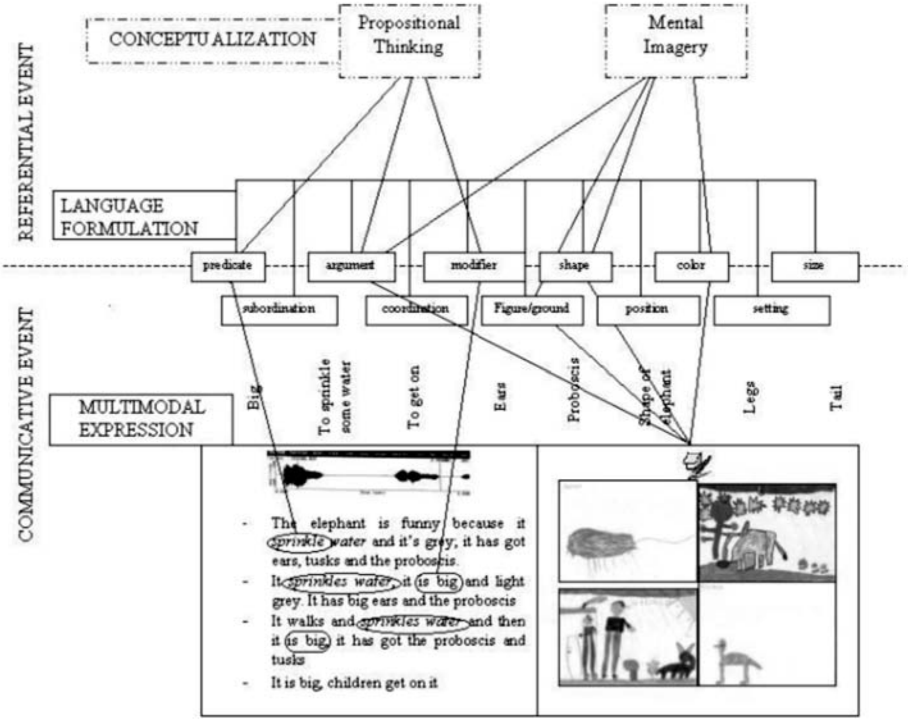
4.1 From Analogy to Specificity of the Formats of Representation

Considering the data presented above it is possible to draw some important conclusions. In general, our data show that for some aspects the two semiotic systems considered, word and image, overlap; as a matter of fact there are several information that occur in a similar way in the two forms of representation. However, data presented in this work seem to match the hypothesis of differentiation, according to which language plays the role of a filter, so that the choice of

Table 3. Number of lexical and encyclopedical items and frequences distribution of knowledge for each age level according to the presentation order

		N of items		Enc. Items		Class/shape		Predicates		Arguments	
		Verb	Dra	Verb	Dra	Verb	Dra	Verb	Dra	Verb	Dra
3 ys	V+D	3,58	25,17	1,08	4,33	0,00	6,00	5,00	3,00	0,33	4,25
	D+V	5,31	15,46	1,62	3,69	2,00	5,00	5,00	3,00	0,69	3,54
	Total	4,45	20,31	1,35	4,01	2,00	11,00	10,00	6,00	0,51	3,89
4 ys	V+D	11,83	35,25	3,25	7,5	2,00	11,00	5,00	4,00	1,50	7,17
	D+V	9,00	21,46	3,00	6,46	3,00	9,00	5,00	3,00	1,62	6,23
	Total	10,42	28,36	3,13	6,98	5,00	20,00	10,00	7,00	1,56	6,70
5 ys	V+D	13,67	58,42	4,17	9,92	0,00	12,00	11,00	9,00	1,42	8,67
	D+V	13,38	37,77	4,23	8,69	3,00	13,00	9,00	5,00	1,54	8,08
	Total	13,53	48,09	4,20	9,3	3,00	25,00	20,00	14,00	1,48	8,37
	Total	9,46	32,25	2,89	6,77	10,00	56,00	40,00	27,00	1,18	6,32

a language (verbal or iconic) activates preferential connections to the format of representation more similar to it [1, 3]. Results indeed show that, starting from one input-word, the search for meaning follows different patterns depending on the request made, whether to use verbal or graphic language. On a quantitative level, beyond the differences between the three ages considered, it is possible to say that a higher number of items of information (vocabulary and encyclopaedic knowledge) is provided by graphic representations. The relationship between the two forms of communication varies according to age: a sort of "predominance" of image over word is to be observed in younger children. As age increases, verbal and graphic forms become more and more complementary: data show that neither of the two forms add any significant items of information to the ones already expressed by 5-year-old children, no matter the order of presentation of both assignments. A closer analysis of data highlights several qualitative differences between the kinds of information produced by the two languages. We notice that drawings try to reproduce a clear image of the represented object; in fact, drawings generally present at least a correct shape of the object [3]. Verbalization shows both semantic and encyclopaedic information [23]. It is in fact possible to distinguish between different ways of organization of meaning, from the narration of autobiographical events to the classification of experience [21]. Data evidence the evolution from a contest-centred and ego-centred mode (episode) to higher degrees of generalization (events), to finally reach the skill of abstraction and classification of information (elements). Furthermore, verbal assignment makes it possible to understand that certain kinds of information are not spontaneously activated (i.e. adverbials, types and external relations), but must be activated on purpose, through a specific request. On the contrary, the same kind of information appears spontaneously in a graphic representation. For instance, information about a category (what an elephant is) is hardly ever verbalized, especially by younger children, whereas drawings show that those children do have a clear image of the elephant as an animal, and they draw it in a horizontal position, with four legs, a tail,...



- The elephant is funny because it sprinkle water and it's grey, it has got ears, tusks and the proboscis.
- It sprinkles water, it is big and light grey. It has big ears and the proboscis
- It walks and sprinkles water and then it is big, it has got the proboscis and tusks
- It is big, children get on it

Fig. 2. Relationship between output message and the formats of knowledge: examples taken from our data

A better focus on some characteristics of the input-word seems to be helped by the drawing activity coming before verbalization: information on the category (the elephant is an animal) is more easily expressed if the child has firstly done his drawing. Similarly, specific verbalized information is likely to appear in a drawing done afterwards. This is particularly evident when we come to consider actions: while verbalizations are generally rich in predicates, only a small number of drawings show an elephant in action (for example, an elephant spraying water, walking,. . .). To sum up, the two forms help each other in a complementary manner, since each deals with specific information. It is possible to identify a sort of selective filter directing the communication of particular information to a peculiar communicative mode. The following chart (fig. 2) is an attempt of synthesis of the phenomena described by data recorded during the experiment. All the information filed in the mind is activated in a different way according to the semiotic system involved in the communicative process. When verbalization is concerned, such categories as predicates, modifiers and arguments ere predominant. As for graphic representation, information regarding space, shape and relation between parts are more common. Drawing seems to represent a tool of a better focusing on a clear precise image of the word, especially with younger children, who gain support in the following verbalization.

References

1. Fodor, J.A.: The modularity of mind. An Essay on faculty psychology, Mit Press, Cambridge, Mass (1983)
2. Kosslyn,S.M.: Ghosts in the mind's machine. Creating and using images on the brain W.Wnorton & C., New York (1983)
3. Kosslyn, S.M.: Image and brain: the resolution of the imagery debate, MIT Press, Cambridge, Mass (1995)
4. Perner, J.: Understanding the Representational Mind, MIT Press,Cambridge, Mass (1991)
5. Eggermont, J.J.: Is there a neural code?.In: Neuroscience and Biobehavioral Reviews, Vol. 22, (1998) 355-370
6. Pinker, S.: How the Mind Works, W.W. Norton, New York (1997)
7. Posner, M.I.: Foundation on Cognitive Science, MIT Press, Cambridge, Mass (1990)
8. Edelman, G. M., Bright Air, Brilliant Fire: On the Matter of the Mind, Basic Books, New York, (1992)
9. Levelt, W.J.M.: The Structure of Messages, In: W.J.M. Levelt (ed.), Speaking: From Intention to Articulation, MIT Press,Cambridge, Mass (1989)
10. Tomasello, M.: The new Psychology of Language, Erlbaum, Mahwah, N.J. (1998)
11. Pullum, G.: Hyperlearning, Complexity, Learnability, and Stimulus Poverty. In: Invited presentation at the Parasession on 'The Role of Learnability in Grammatical Theory', 22nd Annual Meeting of the Berkeley Linguistics Society, University of California, Berkeley, California (1996)
12. Tomasello, M.: Language is not an Instinct, In: Cognitive Development, Vol. 10, (1995) 131-156
13. Camarata, S.M.: Connecting Speech and Language. Clinical Application. In: R. Paul, (ed.) Exploring the Speech-Language Connection, Paul H. Brookes Publishing Co., Baltimore (1998)
14. Lakoff, G: The Invariance Hypothesis; Is abstract reason based in image schemas?, In: Cognitive Linguistics, Vol. 1, (1990) 39-74
15. Castelfranchi, C., Parisi, D.: Linguaggio, conoscenza e scopi, Il Mulino, Bologna (1980)
16. Antoniotti, C. (ed.): La didattica del pensiero, Omega, Torino (1994)
17. Antoniotti, C. (ed.): Il Prisma dell'Io narrante, autore-lettore, Omega, Torino (1998)
18. Jackendoff, R.: The Architecture of the Language Faculty, MIT Press, Cambridge Mass (1997)
19. Eco U.: Trattato di semiotica generale, Bompiani, Milano (1975)
20. Ciceri M.R., Bagarozza G.: La costruzione dei significati. In:Ciceri R. (ed.): Comunicare il pensiero. Procedure, immagini, parole, Omega, Torino (2001)
21. Nelson, K.: Explorations in the Development of a Functional Semantic System. In: W.A. Collins (ed.), Children's Language and Communication, Hillsdale, Erlbaum (1979)
22. Rosch, E.: Priciples of Categorization. In: E. Rosch, B.Lioyd (eds.), Cognition and Categorization, Erlbaum, Hillsdale (1978)
23. Tulving, E.: Episodic and Semantic Memory. In E. Tulving, M. Donaldson (eds.), Organization of Memory, Academic Press, New York (1972)

Emotional and Metaemotional Competence: A Developmental View*

Ilaria Grazzani Gavazzi

Università degli Studi di Milano Bicocca
Facoltà di Scienze della Formazione
Piazza dell'Ateneo 1, 20126 Milano
ilaria.grazzani@unimib.it

Abstract. Starting from LeDoux's studies on amygdala [1], that show how emotions are the result of a complex interplay of conscious and unconscious processes, we focuse on the recent psychological concept of emotional intelligence ad its developmental implications. Following Salovey e Mayer [2] emotional intelligence is a set of aware abilities that people develop and learn to manage emotions in everyday life. From a developmental point of view, we describe qualitative important changes regarding children's emotional competence, what some authors recently call 'metaemotion' [3]. We suggest that the development of metaemotion could be related to quantitative important neurological changes on one hand, and to maturation of neural connections among different brain systems (e.g., language, memory) on the other hand.

1 Introduction

Over the last few decades, psychological research traditionally interested to the study of highest processes, such as memory and language, has focused on emotions. This is in part related to neuroscientific contribute that has studied the nervous basis, the localizations and the brain mechanism of some aspects of emotions [4]. In this perspective, the research on amygdala developed by LeDoux [1] throws light on many aspects regarding the 'emotional brain'. The author showed how amygdala has a direct impact on several cortical areas; how, through amygdala the arousal of several brain systems is produced (a network which controls behavioral responses, autonomic responses, and endocrine responses); and how the body, eventually, reacts to the central elaboration of information. In this perspective, following LeDoux emotions would have developed as brain states and body responses. Conscious feelings would be just the result of awareness of emotional brain system, a system that is for the author the most important dimension of emotions. Talking about consciousness is effectively a problem for neuroscientists, a problem that psychological research looks into. LeDoux himself thinks that even if the influence of neurological circuits from amygdala to cortex

* I wish to thank Prof. Ottavia Albanese and Dr.Carla Antoniotti for their precious contribute to the discussion of the issue presented in this paper.

B. Apolloni et al. (Eds.): WIRN VIETRI 2003, LNCS 2859, pp. 367–373, 2003.

is actually dominant, it could be the case that *connections* between amygdala and cortex will be formed along the evolution, producing the control of cortex by amygdala. "It could be... that human beings will become able to control their own emotions" [1]. On the basis of this brief introduction, I will refer to the meaning of emotional intelligence to present the developmentalists' contribute to the knowledge of emotional competence. I will focus on research that shows how starting from early infancy children develop the comprehension of nature, causes and control of emotions, an awareness of feelings that some authors call 'metaemotion'. I will end the paper suggesting briefly how psychological findings could relate to neuroscience.

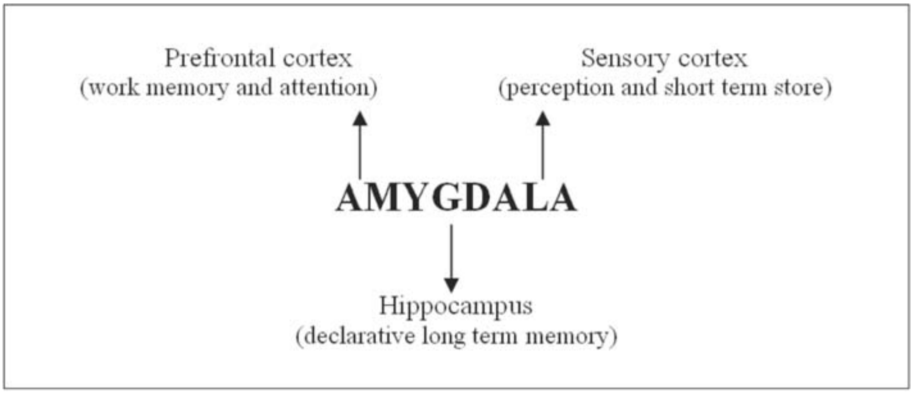

Prefrontal cortex
(work memory and attention)
Sensory cortex
(perception and short term store)
AMYGDALA
Hippocampus
(declarative long term memory)

Fig. 1. Some cortical projections of amygdala and their functions [1]

2 The Cognitive Revolution: From Appraisal of Emotional Events to Appraisal of Emotions

The cognitive revolution with its attention to the unconscious dimension of psychological processes has influenced the study of emotion as well. Despite of differences, various Authors [5] [6] [7] emphasize cognitive appraisal in emotional process, that is formed by both conscious and unconscious components. Going beyond, Salovey e Mayer [2] reviewing studies and research scattered in different scientific fields have more recently tried to conceptualize the idea of 'emotional intelligence'. They showed the existence of several studies, different from the traditional research on the interaction of cognition and affect, that shared the examination of how people appraisal and comunicate emotion, and how they use that emotion in solving problems (see Figure 2). The components presented by the Authors are interconnected among themselves, but they are also independent and differ as a function of individual specificity, as a century-long tradition among clinicians recognized. On the basis of this conceptualization the discussion moves from the role of conscious and unconscious components of emotions to the analysis of knowledge and experiences people have, develop and learn about their own emotions. In this context, Salovey and Mayer [2] refer to *meta-knowledge and meta-experience* of personal feelings and emotions, underlying that although several aspects, for instance, of emotional regulation, happen automatically other aspects are conscious, can be learned, and deserve to be investigated.

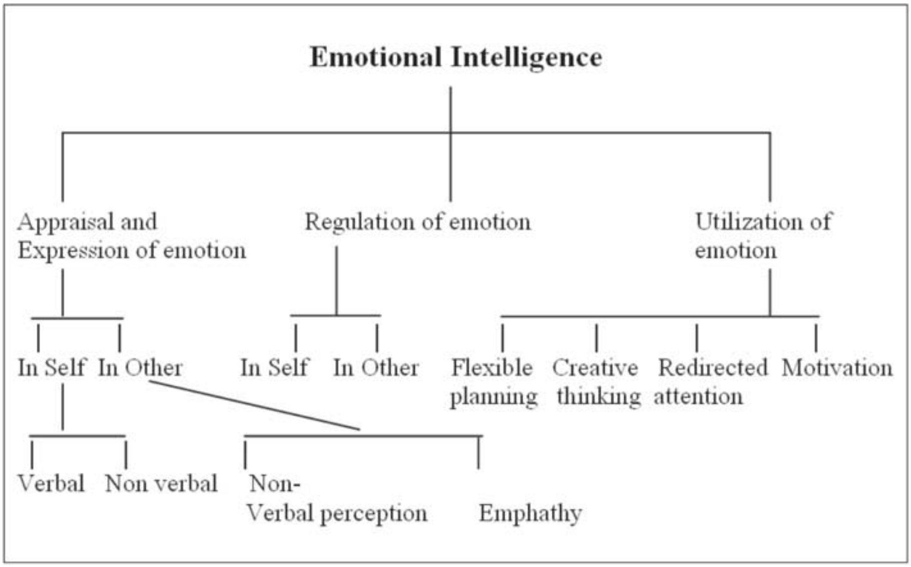


Fig. 2. Conceptualization of emotional intelligence [2]

3 Children's Emotional Competence: Metaemotion

From the developmental point of view, together with the issue of the origin of the emotions it is extremely interesting to understand how we become emotionally competent. As Harris recognizes, children early develop a theory of emotional mind [8] [9]. More recently, the author has used the term 'metaemotion' referring to it as a new paradigm of research [10] [3] with a double meaning: (1) the conscious understanding that a subject has of his or her emotions - for instance, understanding that emotions can be hidden, understanding the relation between moral or display rules on one hand, and the emotions on the other hand; (2) the conscious and non-conscious ability to regulate the expression and the experience of emotions. Here, I will focus on the first meaning of metaemotion. Metaemotion is, first of all, *comprehension of nature, causes, and possibility of control of emotions* (see Figure 3).

The Authors present a model of metaemotion development which contains nine components grouped in three categories, respectively related to understanding the nature of emotions (1-2), understanding the causes or antecedents of emotions (3-7), and understanding the possibility to control emotions (8-9). I am going to illustrate these categories and their components.

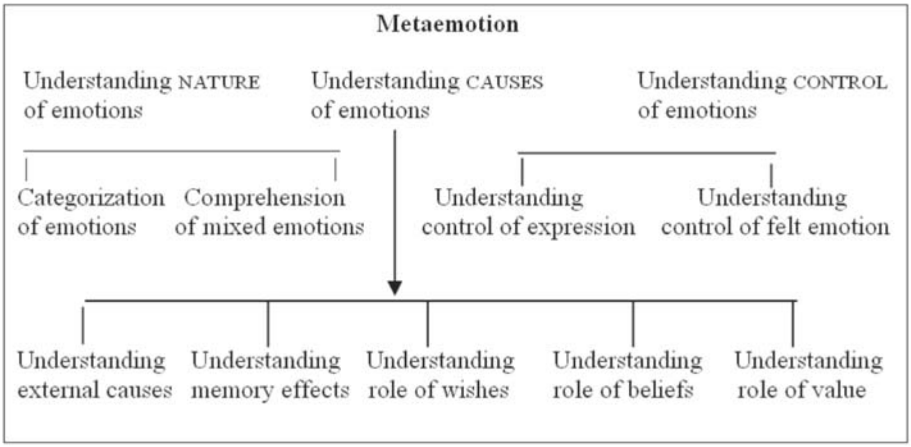

Metaemotion
Understanding NATURE of emotions
Understanding CAUSES of emotions
Understanding CONTROL of emotions
Categorization of emotions
Comprehension of mixed emotions
Understanding control of expression
Understanding control of felt emotion
Understanding external causes
Understanding memory effects
Understanding role of wishes
Understanding role of beliefs
Understanding role of value

Fig. 3. Metaemotion components following Pons, Doudin, Harris, DeRosnay (2002)[7]

- UNDERSTANDING NATURE OF EMOTIONS
  (1) Categorization is a component that emerges at the age of 18-24 months, when children start to use the emotional lexicon referring to emotions in self and in others. At the age of 3-4 years children are able to associate the adequate labels of basic emotions (happiness, fear, sadness, anger) to correspondent facial expressions. So doing, not only they recognize an actual emotion, but they also compare it with a recalled one. On the basis of these first types of categorization, children become able 'to think emotions' and to predict them.
  (2) The second component is the ability to understand mixed emotions, an ability that children begin to show when they are about 7-year-old. They recognize that a person can feel, at the same time, two different, even opposed, emotions, for instance feeling happy for a gift but also sad because it cannot be used.

- UNDERSTANDING THE CAUSES OF EMOTIONS
  (3) Children 3-year-old start to understand that emotions are related to external causes. They are able to link emotion observed in others to their antecedents (e.g., happiness because daddy has arrived; sadness because mummy went away). Starting from the age of 4-5 years, they can solve experimental tasks (e.g., listening to a short story, watching pictures) where prototypical situations of emotion are represented.
  (4) At about the age of 4, children start to understand the role and influence of memory on emotions. Particularly, they understand that the intensity of an emotional experience, both positive and negative, diminish day after day. At the age of 5 years, children recognize that some actual elements can produce the recall of past events (e.g., watching the picture of a fearful event can cause a feeling of fear). The comprehension of the role of recollections

of external events on actual emotions could facilitate the comprehension of the difference between external and internal antecedents of emotions.
(5) At the age of 3 years, children start to understand that desires can be important internal causes of emotions. Between the age of 3 and 5, they understand that a same situational antecedent could produce different emotional reactions in two different persons.
(6) At about the age of 5-6, children understand the role of beliefs on emotions, both of true and false beliefs. So, they understand that a child would feel happy believing to find sweets in a box, even if they are not there.
(7) Eight-year-old children begin to catch the role of moral rules in relation to feelings. Precisely, they can easily relate deprecable behaviors to negative emotions, such as guilt or shame, and merit actions to positive feelings such as pride and satisfaction.

- UNDERSTANDING THE CONTROL OF EMOTIONS
  (8) Before the age of 6, children's judgements of emotions are function of facial expressions. The control of the expression of emotion shows up at the age of 6-7 years, when children undertsand the difference between expression and feeling. For instance, they can understand that a person can smile even if he is not happy, since he doesn't wish to show his internal feelings.
  (9) Children come also to understand the possibility to control a felt emotion. At about the age of 6-7, they start to develop this component of metaemotion, at the beginning utilizing and recognizing the role of behavioral strategies (e.g, playing ad so avoiding negative thoughts). At the age of 10/11 years, children's use of strategies gets more sophisticated forms, involving psychological strategies, such us thinking pleasant episodes to move away unpleasant feelings; talking about emotions and sharing them with others to control their intensity).

  In Table 1 children's metaemotion is presented pointing out the developmental trajectory. Thanks to the emerging of children's theory of mind on one hand, and to the reaching of the operational mental functioning on the other hand, children improve their emotional competence and their knowledge of different aspects of emotion.

4 Conclusion

We suggest that the development of children's understanding of emotions is linked to the maturation and development of neurological and psychological systems (e.g. language, memory). We could speculate that in relation to the acquisition of new emotional competences (e.g., the ability to distinguish between expressed and felt emotion) neurological changes are produced as well, and new brain connections show up. Particularly, a higher level of cortical control could emerge, and a higher number of connections between different components of

Table 1. Some stages or levels of metaemotion development

age (years)	understanding nature of emotion	understanding cause of emotion	understanding control of emotion
2	Emotional lexicon	?	?
3/5	Categorization of joy, fear, sadness, ager	Understanding external cause of emotion	
		Understanding memory effects	
7	Categorization of complex emotions	Understanding role of desires	Understanding distinction apparent/felt emotions
		Understanding role of beliefs	
9/11	Categorization of mixed emotions	Understanding role of merits	Understanding control of emotional experience
12/14		?	?
Late adolescence		?	?

metaemotion (e.g, control of emotion and regulation of emotion) could grow up. Eventually, on the basis of a preliminary research [11] we do believe that during the first and late adolescence, in relation to important brain and ormonal changes, a farther development allow people to potentially become emotionally intelligent, engaging successfully in social relationships and using emotions as a tool of creativity.

References

1. LeDoux, J.: The emotional brain, (1996)
2. Salovey, P., Mayer, J.D.: Emotional intelligence. In: Imagination, Cognition and Personality, Vol. 9(3) (1990) 185-211
3. Pons, F., Harris, P., Doudin, P.A.: Teaching emotion understanding. In: European Journal of Psychology of Education, Vol. 17(3) 293-304
4. Oliverio, A.: Prima lezione di neuroscienze. Laterza Roma-Bari (2002)
5. Frijda, N.: Emotions. Cambridge University Press Cambridge (1986)
6. Scherer, K.: Studying the emotion-antecedent appraisal process: the expert system approach. In: Cognition and Emotion, Vol. 7 (1993) 325-355
7. Oatley, K., Johnson-Laird, P.: Towards a theory of emotions. In: Cognition and Emotion, Vol. 1(1) (1987) 29-50

8. Astington, J., Harris, P., Olson, D.R. (Eds.): Developing theories of mind. Cambridge University Press Cambridge (1988)
9. Saarni, C.: The development of emotional competence. The Guilford Press New York (1999)
10. Pons, F., Doudin, P.A., Harris, P., DeRosnay, M.: Métaémotion et intégration scolaire. In : Lafortune, L., Mongeau, P.: L'affectivité dans l'apprentissage. Presses de l'Université du Québec Sate-Foy (2002) 7-28
11. Grazzani Gavazzi, I., Antoniotti, C.: Consapevolezza emotiva in soggetti preadolescenti, adolescenti e giovani adulti. XVII Congresso Nazionale della Sezione di Psicologia dello Sviluppo, Bari, Settembre 2003

Author Index