SILESIAN UNIVERSITY OF TECHNOLOGY

Department for Strength of Materials and Computational Mechanics
Department of Fundamentals of Machinery Design

POLISH ASSOCIATION FOR COMPUTATIONAL MECHANICS

METHODS OF ARTIFICIAL INTELLIGENCE

Editors:
T. Burczyński, W. Cholewa, W. Moczulski

AI-METH Series, Gliwice, 2004

The papers in this volume have been qualified for publication on the basis of reviews of their draft versions. All papers have been included in the volume exactly as received, on authors' own responsibility.

SUGGESTED REFERENCING:

```
@INPROCEEDINGS{key,
  author  =      "",
  title =        "",
  pages =        "",
  booktitle =    "{M}ethods of {A}rtificial {I}ntelligence",
  editor =       "Burczy\'nski, T. and Cholewa, W. and Moczulski, W.",
  publisher =    "AI-METH Series",
  year =         "2004",
  address =      "Gliwice",
  month =        "November"
}
```

ISBN 83–921605–0–9

TYPESETTING

Marek Wyleżoł
(materials submitted by authors)

COVER DESIGN

Mirosław Dziewoński

PUBLISHERS

AI-METH Series issued by:

Department for Strength of Materials and Computational Mechanics, Silesian University of Technology

Department of Fundamentals of Machinery Design, Silesian University of Technology

http://www.ai-meth.polsl.pl

Printed in
Drukarnia cyfrowa D&D sp. z o.o. ul. Moniuszki 6, 44-100 Gliwice

**PROCEDINGS OF THE SYMPOSIUM
ON METHODS OF ARTIFICIAL INTELLIGENCE *AI-METH 2004*
AND THE WORKSHOP ON KNOWLEDGE ACQUISITION
IN MECHANICAL ENGINEERING
17-19 NOVEMBER, GLIWICE, POLAND**

UNDER AUSPICES OF:

Ministry of Scientific Research and Information Technology
Division IV, Technical Sciences, of the Polish Academy of Sciences

SCIENTIFIC COMMITEE

Marek BALAZIŃSKI	-	École Polytechnique de Montréal, Canada
Adam BORKOWSKI	-	Polish Academy of Sciences, Warsaw, Poland
Tadeusz BURCZYŃSKI	-	Silesian University of Technology, Gliwice, Poland
Wojciech CHOLEWA	-	Silesian University of Technology, Gliwice, Poland
Carlos COTTA	-	University of Málaga, Spain
Vytautas CYRAS	-	Vilnius University, Lithuania
Roman GALAR	-	Wrocław University of Technology, Poland
Avelino J. GONZALEZ	-	University of Central Florida, Orlando, USA
Salvatore GRECO	-	University of Catania, Italy
Zdzisław S. HIPPE	-	University of Information Technology and Management, Rzeszów, Poland
Janusz KACPRZYK	-	Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland
Jan KICIŃSKI	-	Institute of Fluid Flow Machinery, Polish Academy of Sciences, Gdańsk, Poland
Michał KLEIBER	-	Institute of Fundamental Technological Research, Polish Academy of Sciences
Józef KORBICZ	-	University of Zielona Góra, Poland
Witold KOSIŃSKI	-	Polish-Japanese Institute of Information Technologies, Warsaw, Poland
Jan Maciej KOŚCIELNY	-	Warsaw University of Technology, Poland
Jacek ŁĘSKI	-	Silesian University of Technology, Gliwice, Poland
John C. MILES	-	Cardiff University, Wales, UK
Wojciech MOCZULSKI	-	Silesian University of Technology, Gliwice, Poland
Edward NAWARECKI	-	AGH University of Science and Technology, Cracow, Poland
Antoni NIEDERLIŃSKI	-	Silesian University of Technology, Gliwice, Poland
Eugenio OÑATE	-	Technical University of Catalonia, Barcelona, Spain
Janusz ORKISZ	-	Cracow University of Technology, Poland
Maria E. ORŁOWSKA	-	University of Queensland, Australia
Manolis PAPADRAKAKIS	-	National Technical University, Athens, Greece
Witold PEDRYCZ	-	University of Alberta, Edmonton, Canada
Jacques PERIAUX	-	Pôle Scientifique, Dassault-Aviation/University ParisVI, France
James F. PETERS	-	University of Manitoba, Canada
Jerzy POKOJSKI	-	Warsaw University of Technology, Poland
Bob RANDALL	-	University of New South Wales, Sydney, Australia
Zbigniew RAŚ	-	University of North Carolina, Charlotte, USA
Ryszard ROHATYŃSKI	-	University of Zielona Góra, Poland
Leszek RUTKOWSKI	-	Technical University of Częstochowa, Poland
Robert SCHAEFER	-	Jagiellonian University, Cracow, Poland
Raimar SCHERER	-	Dresden University of Technology, Germany
Ménad SIDAHMED	-	University of Technology, Compiégne, France
Ian SMITH	-	Swiss Federal Institute of Technology, Lausanne, Switzerland
Roman SŁOWIŃSKI	-	Poznań University of Technology, Poland
Ryszard TADEUSIEWICZ	-	AGH University of Science and Technology, Cracow, Poland
Tadeusz UHL	-	AGH University of Science and Technology, Cracow, Poland
Kurt VARMUZA	-	Vienna University of Technology, Austria
Zenon WASZCZYSZYN	-	Cracow University of Technology, Poland
Jan WĘGLARZ	-	Poznań University of Technology, Poland
Zygmunt WRÓBEL	-	University of Silesia, Katowice, Poland

PROGRAMME COMMITTEE CHAIRS

Tadeusz BURCZYŃSKI	-	Silesian University of Technology, Gliwice, Poland
Wojciech CHOLEWA	-	Silesian University of Technology, Gliwice, Poland
Witold PEDRYCZ	-	University of Alberta, Edmonton, Canada

ORGANIZING COMMITTEE (Silesian University of Technology)

Marek ADAMCZYK
Krzysztof CIUPKE
Mirosław DZIEWOŃSKI
Wojciech MOCZULSKI – chair
Ewa OPOKA
Piotr PRZYSTAŁKA
Krzysztof PSIUK
Anna TIMOFIEJCZUK
Bronisława ULEJCZYK
Ryszard WYCZÓŁKOWSKI
Marek WYLEŻOŁ

Contents

Probabilistic PCA and neural networks in search of representative features for some yeast genome data

Anna Bartkowiak
University of Wrocław, Inst. of Computer Science,
ul. Przesmyckiego 20, 51-151 Wrocław, Poland
e-mail: aba@ii.uni.wroc.pl

Stanisław Cebrat and Paweł Mackiewicz
University of Wrocław, Inst. of Genomics and Microbiology,
Przybyszewskiego 63/77, 51-148 Wrocław, PL
e-mail: {cebrat,pamac}@microb.uni.wroc.pl

Keywords: reduction of dimensionality, yeast genome, latent structure, probabilistic PCA, multi-layer perceptron

1. Introduction, the data and the problem

We consider data characterizing $N = 3300$ yeast genes, each characterized by $d = 13$ variables (traits). The data will be in the following called 'the yeast genome' data. A more detailed description of the data may be found in [1, 2] or [6]. The gathered variables have a quite clear interpretation and some of them are fairly dependent. Attempt to simply omit some of the variables is not working: the eventually omitted variables (by use of the *idep* procedure) can not be explained in a satisfactory manner by the retained variables. None the less, at least some of the recorded variables are linearly interdependent. This may be stated, when analyzing the eigenvalues (of their correlation matrix), exhibited in Figure 1.

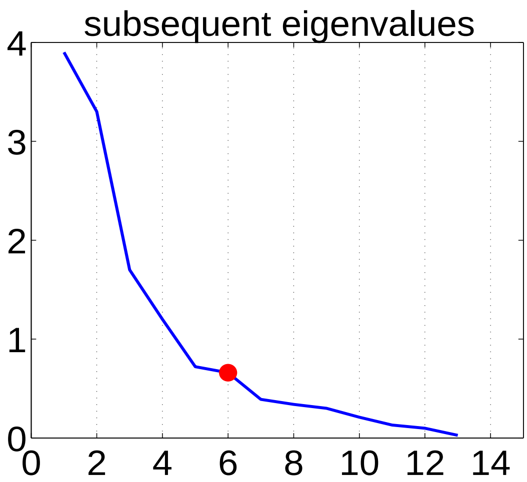

Fig. 1. Scree graph exhibiting eigenvalues of correlation matrix calculated from $N = 3300$ genes. The decay of subsequent eigenvalues is shown. It seems that $h = 6$ is the right number of latent variables. To the right of the 6th eigenvalue – marked by a big filled circle – the decay exhibits a linear pattern, which means that no more common factors can be extracted

Our problem is: Could the observed variables be transformed to a reduced set, containing $h < d$ new, derived features – without loosing not too much of total inertia (variance) of the entire set.

We apply for our task 3 methods: (1) traditional principal components, (2) probabilistic principal components, (3) neural networks using multilayer perceptrons in the 13-6-13 layout. It seems, that the data may be explained by $h = 6$ derived latent variables. Thus, in further analysis we were seeking for 6 new, derived features, called also latent variables. We think, we were quite successful: The traditional PCA and the NN models explain, when using 6 factors, about 88% of total variablility of the data; however these methods do not provide any generative model of the data. Probabilistic principal components (Bishop, Tipping 1999) permit to find $h = 6$ features, which are able to reproduce 78.53 % of total variance of the data.

This result is interesting for several reasons: (1), it is confirmed, that *principal components* extract too much of total variance of the data set (which means, that they account some random effects as systematic effects). (2), it was interesting to state, that *neural networks* using perceptrons behave similarly as principal components and yield similarly overestimated approximation of hidden factors. This is opposed to the recent paper by Nicole [5], where some doubts were expressed, whether neural networks are suitable for a broad application in biological systems. (3), the *new features (latent variables)*, derived from the observed variables, have a very clear and interesting interpretation: The set of 12 variables (representing 3 legs of the spider-plots [6]) has split into 3 double factors, each factor expressed by 2 latent variables.

In the following we explain briefly the methods and show some results obtained when using the chosen methods.

2. Traditional PCA and Probabilistic PCA

Traditional PCA is well explained in the books by Jolliffe (2002) or Krzanowski (2000). PCA is a purely mathematical technique, working with available data. No underlying generative model of the data is considered. The predictions of the target data are heuristic, based on the data sample on which the predictions were evaluated. The method reproduces the entire data set (or, its covariance matrix), by rank one matrices.

However, the principal components – based only on the gathered data – do not provide any generative model of the data, and no generalization can be done, neither no statistical tests of significance.

A more general approach is by introducing a generative model of the data, which is valid also in the context of neural neutworks, considered as a tool for data analysis. Nabney [4] writes: "The goal of training a network is to model the underlying generator of the data in order to make the best possible predictions when new input data is presented. The most general information about the target vector $\mathbf{t}$ for inputs $\mathbf{x}$ is given by the conditional density $p(\mathbf{t}|\,\mathbf{x})$".

Tipping and Bishop (see, e.g., [7]) have introduced *probabilistic principal components* working with a generative data model. The following basic model is assumed:

$$\mathbf{t} = \mathbf{W}\mathbf{x} + \boldsymbol{\mu} + \boldsymbol{\epsilon} \tag{1}$$

Here $\mathbf{t}$ and $\mathbf{x}$ denote the observational and latent variables, and $\boldsymbol{\epsilon}$ – Gaussian noise $\sigma^2\mathbf{I}$.

The observed values $\mathbf{t}$ in d variables are supposed to be generated by $q < d$ hidden (latent) variables $\mathbf{x}$ distributed normally with isotropic variance.

Under the assumed model [1] the observed vector $\mathbf{t}$ is distributed normally .

$$\mathbf{t} \sim N_d(\boldsymbol{\mu}, \mathbf{W}\mathbf{W}^T + \sigma^2\mathbf{I}). \tag{2}$$

The unknown parameters of the model [2] are: $\mathbf{W}$ and σ^2. They may be estimated either directly from the log-likelihood or by the EM algorithm. Corresponding formulae may be found in the paper by Tipping and Bishop [7].

3. Multi-layer perceptron

Neural networks have developed a special type of learning (Hebbian learning) to capture the essential characteristics (main directions) of the data. Quite a lot of research was needed to find out, what really the Hebbian learning is yielding.

Generally, artificial neural networks are considered as semi-parametric or non-parametric models for data analysis, see e.g., Gaudart et al. [3], and the references therein. Realization of the method of principal components in the framework of Hebbian learning was the subject of many investigations, (see, e.g., the papers by Oja, Sanger et others). Recently, a critical discussion of the approaches has been published by Nicole [5].

Instead of the traditional Hebbian approach we have formulated the task in terms of approximation of the data. Thus the network has as target the data presented at the input. The number of neurons in the hidden layer was put equal to h, the number of the desired hidden factors (in our case this was $h = 6$).

For our yeast genome data we have used a multi-layer perceptron with 2 hidden layers. Its layout was: 13 – 6 – 13. This means, there were 13 inputs, the first hidden layer with $h = 6$ neurons has being condensing the inputs to 6 derived variables. The derived 6 variables $z_1, \ldots, z_6$ acted as input to the second hidden layer who's task was to reproduce from the z's the target, which was again the input vector.

The implementation in Netlab puts in the first layer as obligatory the 'tanh' activation function, which makes that all z's are contained in the interval (-1,1). The second hidden layer has used the 'linear' activation function.

The network needed about 3000 epochs (presentations of the data matrix) to get stabilized parameters.

It was a big surprise to us obtaining, by such a standard and simple tool, results very similar to those, obtained by probabilistic PCA with rotation varimax.

Table 1. Matrix $\mathbf{W}$ expressing 6 latent variables for the yeast genome data. The presented matrix was obtained from rotated matrix $\mathbf{U}\sqrt{(\boldsymbol{\Lambda} - \sigma^2\mathbf{I})}$.

	1.leg	1.leg	3.leg	2.leg	3.leg	2.leg	%
ang1	-.08	**.84**	.07	-.18	-.21	.14	.81
ang2	.03	-.10	-.08	**.83**	.13	.17	.76
ang3	.00	-.06	**-.85**	.06	.11	.02	.74
x1	**.72**	**-.37**	-.09	.28	.15	-.28	.84
y1	**.58**	**.67**	.06	-.02	-.27	.02	.85
x2	.30	-.17	-.04	**.69**	.16	**-.42**	.80
y2	-.21	.08	-.17	.20	.14	**.82**	.81
x3	-.04	-.27	-.20	.24	**.74**	-.01	.71
y3	.05	-.04	**-.79**	.05	.27	.17	.73
lgth	.65	-.01	-.08	-.06	-.14	-.50	.70
rho1	**.85**	.21	.04	.14	-.16	-.24	.88
rho2	.29	-.10	.09	.21	-.05	**-.83**	.84
rho3	.15	.12	.24	-.05	**-.77**	-.19	.73

Table 2. Results from training a perceptron with layout 13-6-13 using the yeast genome data. Weights connecting the hidden layer with neurons of the input layer are shown. All weights were multiplied by 10. To be comparable with results from Table 1, some columns should be permuted.

	3.leg	2.leg	2.leg	1.leg	1.leg	3.leg
ang1	-.10	.03	-.22	**-.76**	-.26	-.43
ang2	.41	-.32	**-.64**	.21	-.34	.06
ang3	**.96**	.04	.37	**-.40**	-.15	**.61**
x1	.17	-.11	.06	**.43**	**.51**	-.20
y1	.06	.02	-.31	**-.37**	.31	-.54
x2	.27	**-.55**	-.24	.07	-.29	-.02
y2	.27	**.40**	**-.44**	.26	.05	-.20
x3	-.08	-.16	.32	.20	-.25	**-.75**
y3	**.82**	.13	.36	-.26	-.05	.19
lgth	.15	-.13	.24	-.09	.39	.01
rho1	.12	-.05	-.18	-.00	**.52**	-.40
rho2	-.05	**-.55**	.20	-.23	-.25	.18
rho3	.12	-.03	**-.39**	-.11	.22	**.92**

References

[1] Cebrat S., Dudek M.R. *The effect of DNA phase structure on DNA walks.* The European Physical Journal B., **3** (1998), 271–276.

[2] Cebrat S., Mackiewicz P., Dudek M.R. *The role of the genetic code in generating new coding sequences inside existing genes.* Biosystems, **45** (2) (1988), 165–176.

[3] Gaudart J., Giusiano, B., Huiart L., *Comparison of the performance of multi-layer perceptron and linear regression for epidemiological data.* Computational Statistics & Data Analysis, **44** (2004), 547–570.

[4] Nabney I., *Netlab: Algorithms for Pattern Recognition.* Springer, 2002.

[5] Nicole S., *Feedforward neural networks for principal components extraction.* Computational Statistics & Data Analysis, **33**(2000), 425–437.

[6] Smorfland: http://smorfland.microb.uni.wroc.pl/

[7] Tipping M.E., Bishop C.M, *Probabilistic principal component analysis.* J. Roy. Statist. Soc., B, **61** (1999), 611-622.