Advances in Pattern Recognition

Springer
London
Berlin
Heidelberg
New York
Barcelona
Hong Kong
Milan
Paris
Singapore
Tokyo

Advances in Pattern Recognition is a series of books which brings together current developments in all areas of this multi-disciplinary topic. It covers both theoretical and applied aspects of pattern recognition, and provides texts for students and senior researchers.

Springer also publishes a related journal, **Pattern Analysis and Applications**. For more details see: http://link.springer.de

The book series and journal are both edited by Professor Sameer Singh of Exeter University, UK.

Also in this series:

Principles of Visual Information Retrieval
Michael S. Lew (Ed.)
1-85233-381-2

Advanced Algorithmic Approaches to Medical Image Segmentation
Jasjit Suri, Kamaledin Setarehdan and Sameer Singh (Eds)
1-85233-389-8

Šarūnas Raudys

Statistical and Neural Classifiers

An Integrated Approach to Design

With 68 Figures

Springer

Šarūnas Raudys
Data Analysis Department, Institute of Mathematics and Informatics,
Akademijos 4, Vilnius 2600, Lithuania

ISBN 1-85233-297-2 Springer-Verlag London Berlin Heidelberg

British Library Cataloguing in Publication Data
Raudys, Šarūnas
Statistical and neural classifiers : an integrated approach
to design. - (Advances in pattern recognition)
1. Pattern recognition systems 2. Neural networks (Computer
science)
I. Title
006.4
ISBN 1852332972

Library of Congress Cataloging-in-Publication Data
A catalog record for this book is available from the Library of Congress

Typesetting: Camera-ready by author
Printed and bound at the Athenæum Press Ltd., Gateshead, Tyne and Wear
34/3830-543210 Printed on acid-free paper SPIN 10762565

In memory of my Father
and in memory of my mathematics teacher –
who was like a real father to me at school.

Foreword

Automatic (machine) recognition, description, classification, and groupings of patterns are important problems in a variety of engineering and scientific disciplines such as biology, psychology, medicine, marketing, computer vision, artificial intelligence, and remote sensing. Given a pattern, its recognition/classification may consist of one of the following two tasks: (1) supervised classification (also called discriminant analysis); the input pattern is assigned to one of several predefined classes, (2) unsupervised classification (also called clustering); no pattern classes are defined *a priori* and patterns are grouped into clusters based on their similarity. Interest in the area of pattern recognition has been renewed recently due to emerging applications which are not only challenging but also computationally more demanding (e.g., bioinformatics, data mining, document classification, and multimedia database retrieval).

Among the various frameworks in which pattern recognition has been traditionally formulated, the statistical approach has been most intensively studied and used in practice. More recently, neural network techniques and methods imported from statistical learning theory have received increased attention. Neural networks and statistical pattern recognition are two closely related disciplines which share several common research issues. Neural networks have not only provided a variety of novel or supplementary approaches for pattern recognition tasks, but have also offered architectures on which many well-known statistical pattern recognition algorithms can be mapped for efficient (hardware) implementation. On the other hand, neural networks can derive benefit from some well-known results in statistical pattern recognition. Issues related to the training and test sample sizes, feature space dimensionality, error rate estimation, and the discriminatory power of different classifiers have been extensively studied in the statistical pattern recognition literature. It often appears that some of the neural network researchers attempting to solve pattern recognition problems are not aware of these results.

Professor Raudys' book is a timely addition to the literature on pattern recognition. Professor Raudys is eminently qualified to write a monograph which presents a balanced view of classifier design and promotes an integration of statistical pattern recognition and neural network approaches. Even though his early work, published in Russian, was not easily accessible to the pattern recognition community, he is now well-known and recognised for his early contributions on the topic of "curse of dimensionality" and its practical implications in designing a pattern recognition system. His intuition and

knowledge of the performance of various classifiers has enabled him to show the interrelationships among them. The book contains detailed descriptions of various classifiers which, to my knowledge, are not readily available in other textbooks. In addition to deriving analytical results concerning the relationship between sample size, dimensionality and model parameters, he also reports simulations results. These results are not available in other well-known textbooks on pattern recognition and neural networks.

In most pattern recognition books and review papers, it is the statistical approach which is used to compare and analyse classification and regression methods based on neural networks. Šarūnas Raudys was the first to show that ANN based classifiers and regression evolve into, and realise, a number of popular statistical pattern recognition methods. These findings, together with his earlier results, enable him to propose a way to utilise positive attributes of both approaches simultaneously in classifier design. His extensive coverage of the curse of dimensionality and related topics, together with a new approach to introduce statistical methodology into the neural networks design process, constitute the novelty of this book.

In summary, this book is an excellent addition to the literature on statistical pattern recognition and neural networks. It will serve as a valuable reference to other excellent books by Duda, Hart, and Stork, Ripley, Bishop and Haykin.

September 19th, 2000, East Lansing, Michigan Anil K. Jain

Preface

– Les hommes de chez toi, dit le petit prince,
cultivent cinq mille roses dans un même jardin...
et ils n'y trouvent pas ce qu'ils cherchent...
– Ils ne le trouvent pas, répondis-je...
Et cependant ce qu'ils cherchent pourrait être
trouvé dans une seule rose ou un peu d'eau...
– Bien sûr, répondis-je.
Et le petit prince ajouta:
– Mais les yeux sont aveugles. Il faut chercher
avec le coeur.

Antoine de Saint-Exupery "Le Petit Prince",
Chapitre XXV

In his book Antoine de Saint-Exupery wrote: "... In your country, people plant five thousand roses in the same garden ... and they do not find what they are searching for. Meanwhile, they could find everything they are seeking for in a single rose, or in a drop of water... However, eyes are blind. You have to seek by your heart." I am fond of Antoine de Saint Exupery. I like his books and I believe in these words.

When, after 25 years of research work in multivariate statistical analysis and statistical pattern recognition, I became interested in Artificial Neural Networks (ANN), I remembered de Saint-Exupery's words about the single rose. Instead of investigating multilayer perceptrons with thousands of neurones, I at first began to use statistical methods to analyse a single neurone – a single layer perceptron (SLP) and plain back-propagation (BP) training algorithm. The SLP and BP training are simplified mathematical models of complex information processing phenomena that take place in nature. I discovered that a single neurone can explain much about complex brain-learning behaviour.

After several years of research work, I learned that during the training phase, the single layer perceptron's weights are increasing and, therefore, the statistical properties of the cost function that is minimised during the training process are also changing. In its dynamical evolution, the SLP classifier can actually become one of several statistical classifiers that differ in their complexity.

At first the SLP classifier behaves as a simple Euclidean distance classifier. In this situation each pattern class is characterised by a sample mean vector. In further training, the SLP classifier begins to evaluate correlations and variances of features and almost becomes the standard linear Fisher classifier. Further, the SLP begins to behave as a robust classifier that ignores atypical training patterns. Later on in the training phase the SLP behaves as a classifier that minimizes the number of incorrectly classified training patterns. If there are no training errors, the SLP maximizes the margin between the decision boundary and the closest training-pattern vectors. Thus, the decision boundary is halfway between the training patterns and behaves as a support vector classifier where a fixed number of training patterns determine a position of the decision boundary. Thus, the SLP classifier begins as the simplest possible statistical classifier designed under the assumption of Gaussian pattern classes and ends as a complex classifier that can perform well with non-Gaussian pattern classes.

One more interesting peculiarity of the SLP classifier is that the performance (the generalization error) of the perceptron depends on the initial conditions. If the starting perceptron weight vector is almost optimal, the SLP classifier initially contains much useful information. This information can be utilized to determine the final weight vector. To preserve and use the information in the initial weights, one must not overtrain the SLP. This is a very valuable property of adaptively trained neural networks.

The more I work in the ANN discipline, the more I marvel at the remarkable qualities of neural networks. Specifically, I am amazed that the SLP classifier dynamics progress from the simplest algorithms to the most complex algorithms in a natural manner. Statisticians and engineers have long understood that in designing decision-making algorithms from experimental data one needs to progress from simple algorithms to complex algorithms. The artificial neurone accomplishes this complexity progression in a natural manner. Statisticians required several decades to develop a number of statistical classification and regression rules: Fisher (1936) proposed his linear discriminant function more than six decades ago and Vapnik devised his support vector machine (Vapnik, 1995) only recently. The neurone, however, implements this algorithm design progression in a logical sequence. One can think of this progression as nature's method.

There is a plethora of useful information currently available in the field of statistical pattern recognition. The main element of statistical pattern recognition is the assumption that the pattern vectors are random vectors that can be modelled as observations from multivariate distributions. In the parametric approach, the classifier designer assumes he knows this multivariate distribution precisely. In order to determine the classification algorithm, one needs to estimate the unknown multivariate density function parameters using the training data. Researchers have found that the more complex the multivariate density model assumed, or equivalently, the greater the number of parameters to be estimated, the greater the number of training vectors that must be employed to adequately determine the classifier. Therefore, a large number of parsimonious multivariate distribution densities have been formulated for this purpose.

One of the most interesting and important facts utilized in parametric classification is that if some of the pattern-class densities' parameters are in common, these parameters have a negligible influence on the increase in the generalization error. This is a very favourable property of the statistical parametric classifier approach. On the other hand, incorrect assumptions about the type of the probabilistic distribution assumed for the pattern vectors lead to an increase in the classification error. This is one of the main shortcomings of the parametric statistical classifier approach.

Being interested in both statistical pattern recognition and artificial neural network theory, I perceived a certain conflict between these two classification paradigms, and sometimes even a dissonance among proponents of these two classification methods. Statisticians generally have good mathematical backgrounds with which to analyse decision-making algorithms theoretically. They have proven many rigorous results concerning the optimality of statistical classification algorithms. However, they often pay little or no attention to the applicability of their own theoretical results and generally do not heed practical or even theoretical results obtained by ANN researchers.

ANN scientists advocate that one should make no assumptions concerning the multivariate densities assumed for the pattern classes. They, instead, propose that one should assume only the structure of the decision-making rules, for example a linear discriminant function in the space of original or transformed features, and then estimate the unknown rule coefficients (weights) directly from the training data. For this they suggest one minimize the number of errors incurred while classifying the training vectors (empirical error). Many such algorithms have been suggested to solve practical problems. Some of these algorithms have a theoretical justification and some have no theoretical elucidation yet.

Known properties and deficiencies of both statistical and neural classification algorithms hints that one should *integrate* the two classifier design strategies and utilize their good qualities. There are three key aspects of this integration. The first key is the fact that the correct initial weights of the perceptron contain information that can be saved for use in future training processes. Thus, we can utilize pattern classifiers based on statistical methods to define the initial perceptron weight vector. The second key is the fact that, during training, the perceptron changes its statistical properties and evolves from simple classification algorithms to more complex classification algorithms. The third key is the fact that, one can use the diversity of statistical methods and the multivariate models to perform different whitening data transformations, where the input variables are decorrelated and scaled in order to have the same variances. Then while training the perceptron in the transformed feature space, we can obtain the Euclidean distance classifier after the very first iteration. In the original feature space, the weight vector of this classifier is equivalent to the decision making rule found by utilizing the statistical methods and the multivariate models just mentioned. Further training can diminish the negative aspects of approximately correct or incorrect statistical assumptions.

Thus, it is possible to merge the statistical and neural approaches. Specifically, instead of using statistical methods and the multivariate models directly to design the classifier, we can use them to whiten the data. We can then train the perceptron paying special attention to the optimal stopping time. The data whitening reduces

the generalisation error and simultaneously speeds up the training process. This approach merges the qualities of both statistical and neural classification algorithm design strategies. Investigations of the visual cortex in biological systems, however, have shown that the input decorrelation technique is already realized in natural systems. This is one more piece of evidence that the data decorrelation and scaling technique performed prior to perceptron training is a natural method of information processing.

An objective of this book is to explain the details necessary to understand and utilise the integrated statistical and neural net approach to design the classification rules. We, therefore, present a discussion of the diversity of linear and non-linear statistical pattern classification algorithms that can be utilised in an advanced neural network analysis. Special attention is paid to the assumptions used to design the algorithm, the generalisation error, and the training-set size relationships. Knowledge of these relationships allows one to analyse and compare the amount of information obtained from the training data, the assumptions, or from the educated guesses utilised to construct the decision-making algorithm. This is perhaps the central question that arises in machine learning and classifier design theory.

Performance, complexity, and training-set size relationships in the nonparametric neural net approach have been discussed in a number of books (Vapnik, 1982, 1995; Wolpert, 1995; Cherkassky and Mulier, 1996; Vidyasagar, 1997, etc.). According to Wolpert, "the statistical and neural net approaches have their own jargon, their own mathematical models, their own concern, and their own results. And, for the most part, they don't interact". This book primarily takes a statistical point of view but does not ignore other approaches. Alternative texts are Raudys (1976), Aivazian *et al.* (1988), Fukunaga (1990), McLachlan (1992), Devroye, Gyorfi and Lugosi (1996), Duda, Hart, and Stork (2000). The present book, however, is more focused on the integration of statistical and neural approaches to design the classification algorithms. In order to focus on performance, complexity, and design set size relationships more deeply in this book, I employ a simple formulation of the pattern recognition problem. For more general formulations of the pattern recognition problem and related questions I refer interested readers to broader texts such as Fukunaga's book. To make the book accessible to more readers, I adopt Fukunaga's notation.

The book is targeted to graduate students and research workers in data modelling, pattern recognition, and artificial neural networks. No special background beyond a good working knowledge of probability and statistics, elements of linear algebra, and calculus at the undergraduate level is required. However, one will benefit by having a popular pattern recognition or neural networks book (e.g., Fukunaga (1990) or Haykin (1998)) close at hand.

The book is organized somewhat like a reference book. At the same time I pay particular attention to the ideas used to design and analyse statistical classification algorithms that can be useful for understanding artificial neural network classifiers. For analysis of neural networks and statistical algorithms, the most important aspect is assumptions utilised in the algorithm design process. Therefore, in order to have a comprehensive point of view of the problem, I omit a part of the details concerning the estimation of parameters of well known statistical algorithms that

can be found in the popular literature. To make understanding the main ideas easier, I provide a number of simple illustrative examples.

In the first chapter, I present the main definitions and terms and review the effects of finite training-set size on classifiers. In the second chapter, I review principles of the statistical decision theory, review important statistical multivariate data models, and give a taxonomy of pattern classification algorithms that can be obtained or improved while training ANN classification systems. In the third chapter, I present known results concerning the performance and generalisation error relationships for a number of parametric and nonparametric classification algorithms. In the fourth chapter, I consider training peculiarities and the generalisation and complexity of neural classifiers. In the fifth chapter, I explain the integration of the statistical and neural classification approaches. In the sixth and final chapter, I consider the topic of model selection, paying special attention to the accuracy of solutions to this important topic.

The main contents of this book crystallised during my work as a tenured researcher at the Institute of Mathematics and Informatics in Vilnius. However, other parts of the book were generated while visiting and collaborating with researchers, professors and graduate students at several institutions. These include the Department of Computer Science at Michigan State University; the Department of Information Systems at the Hankamer School of Business, Baylor University, USA; the Departments of Electrical Engineering and Applied Physics, Delft University of Technology, The Netherlands; LAFORIA, Institute Blaise Pascal, University Paris 6; Department of Electrical and Electronic Engineering, Bogazici (Bosforus) University, Istanbul; Laboratory of Information Representation, RIKEN, Tokyo; Ford Motor Company Scientific Research Laboratories. Many ideas and solutions were developed while closely collaborating with researchers of A. N. Kolmogorov Laboratory of Statistical Methods in Moscow State University. In particular I wish to thank Yurij Blagoveschenskij, Lev Meshalkin, Vladimir Vapnik, Gennadij Lbov, Anil Jain, Dean Young, Eric Backer, Bob Duin, Françoise Fogelman-Soulie, Patrick Gallinari, Bulent Sankur, Shun-ichi Amari, Andrzej Cichotski, Gintaras Puškorius and many others for useful and encouraging discussions and their hospitality and aid.

I would also like to thank my colleagues and former and present graduate students at the Institute of Mathematics and Informatics (Vilnius), Kaunas Vytautas Magnus University and Vilnius University for their challenging discussions and for their assistance in simulation studies. Special acknowledgement is expressed to Professor Laimutis Telksnys, Vitalijus Pikelis, Marina Skurichina, Tautvydas Cibas, Valdas Dičiūnas, Aistis Raudys, Aušra Saudargienė and Arūnas Janeliūnas. Exceptional thanks are expressed to Roy Davies, Edvardas Povilonis, Dean Young and Laura Thompson for their useful discussions and their aid in editing this book.

The author acknowledges the financial support from the Lithuanian State Science and Studies Foundation.

Vilnius, August 2000 *Šarūnas Raudys*

Contents

Abbreviations and Notations

ANN	artificial neural network
BP	back-propagation
DF	discriminant function
GCCM	Gaussian with common covariance matrix
EDC	Euclidean distance classifier
FS	feature space
k-NN	*k*-nearest neighbour
LDF	linear discriminant function
LOOM	leaving-one-out method
LVQ	learning vector quantisation
MEE	minimum empirical error
MLP	multilayer perceptron
PMC	probability of misclassification
PW	Parzen window
QDF	quadratic discriminant function
RBF	radial basis function
RDA	regularised discriminant analysis
RM	resubstitution method to estimate a training-set (empirical) error
SLP	single layer perceptron
SG	spherical Gaussian
SV	support vector
VC	Vapnik-Chervonenkis
ZEE	zero empirical error

$\boldsymbol{X} = (x_1, x_2, \ldots, x_n)^T$ is an n-variate vector to be classified; a subscript ^{T} denotes a transpose operation

n is a dimensionality of the feature vector $\boldsymbol{X}$

L is a number of pattern classes (categories, populations), $\omega_1, \omega_2, \ldots, \omega_L$

$p_i(\boldsymbol{X})$ is the class conditional probability density function (PDF) of vector $\boldsymbol{X}$ belonging to class ω_i

P_i is the a priori probability that observation $\boldsymbol{X}$ belongs to class ω_i

Ω is a feature space

One part of the design set is called *a training (learning) set*, while the other one is called *a validation set*. A set used to evaluate performance of the final variant is called *the test set*

N_i is the number of training vectors from class ω_r

$N = N_1+N_2$ if $N_2=N_1$, we denote $\overline{N} = N_1=N_2= N/2$

$\boldsymbol{M}_r$ is a mean vector, of the pattern class ω_r

$\hat{\boldsymbol{M}}_r$ is an arithmetic mean of the training-set of the class ω_r

$$\hat{\boldsymbol{M}}_i = \frac{1}{N_i} \sum_{j=1}^{N_i} \boldsymbol{X}_j^{(i)}$$

$\boldsymbol{X}_j^{(i)}$ is the j-th training set observation from ω_i

$\hat{\boldsymbol{M}} = ½ (\hat{\boldsymbol{M}}_1 + \hat{\boldsymbol{M}}_2)$ is a centre of the training set (if $N_2=N_1$)

$\boldsymbol{\Sigma}_i$ is an $n \times n$ covariance matrix (CM) of the of category ω_i. It is a function of n variances of all n features and $n(n\text{-}1)$ correlations between them. When both pattern classes share the same CM, we denote it by $\overline{\boldsymbol{\Sigma}}$

$$\begin{bmatrix} \boldsymbol{\Sigma}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_2 & \cdots & \mathbf{0} \\ \cdots & \cdots & \cdots & \cdots \\ \mathbf{0} & \mathbf{0} & \cdots & \boldsymbol{\Sigma}_H \end{bmatrix}$$ is the block diagonal covariance matrix

$$\hat{\boldsymbol{\Sigma}}_i = \frac{1}{N_i - 1} \sum_{j=1}^{N_i} (\boldsymbol{X}_j^{(i)} - \hat{\boldsymbol{M}}_i)(\boldsymbol{X}_j^{(i)} - \hat{\boldsymbol{M}}_i)^T$$ is a sample estimate of $\boldsymbol{\Sigma}_i$

$\hat{\overline{\boldsymbol{\Sigma}}}$ is a sample estimate of $\overline{\boldsymbol{\Sigma}}$. $\hat{\overline{\boldsymbol{\Sigma}}} = N_1/(N_1+N_2)\,\hat{\boldsymbol{\Sigma}}_1 + N_2/(N_1+N_2)\,\hat{\boldsymbol{\Sigma}}_2$

$\hat{\Sigma} = \Phi\Lambda\Phi^T$ is a singular value decomposition of matrix $\hat{\Sigma}$

$\hat{\Sigma}^+ = \Phi \begin{bmatrix} \lambda^{-1} & 0 \\ 0 & 0 \end{bmatrix} \Phi^T$ is a pseudoinversion of $\hat{\Sigma}$

$\hat{\Sigma} + \lambda\mathbf{I} = \Phi(\Lambda + \lambda\mathbf{I})\Phi^T$ is a ridge estimate of the sample covariance matrix $\hat{\Sigma}$

$\delta^2 = (\boldsymbol{M}_1 - \boldsymbol{M}_2)^T \bar{\Sigma}^{-1} (\boldsymbol{M}_1 - \boldsymbol{M}_2)$ is a squared generalised (Mahalanobis) distance between two pattern classes

$n^* = \dfrac{(\boldsymbol{M}^T\boldsymbol{M})^2 (tr\,\bar{\Sigma}^2)}{(\boldsymbol{M}^T\bar{\Sigma}\boldsymbol{M})^2}$ is an effective dimensionality of EDC for GCCM data

$\|\boldsymbol{X} - \hat{\boldsymbol{M}}_r\|^2 = (\boldsymbol{X} - \hat{\boldsymbol{M}}_r)^T(\boldsymbol{X} - \hat{\boldsymbol{M}}_r)$ is an Euclidean distance between $\boldsymbol{X}$ and $\hat{\boldsymbol{M}}_r$

$h(X) = \boldsymbol{X}^T\boldsymbol{V} + v_0$ is a linear discriminant function (DF)

v_0, $\boldsymbol{V}^T = (v_1, v_2, \ldots, v_n)$ are weights of the discriminant function

$N_X(\boldsymbol{M}_i, \boldsymbol{\Sigma}_i)$ is an n-dimensional Gaussian distribution density

$N_X(\boldsymbol{X}, \boldsymbol{M}_i, \boldsymbol{\Sigma}_i) = (2\pi)^{-n/2} |\boldsymbol{\Sigma}_i|^{-1/2} e^{-1/2 (\boldsymbol{X} - \boldsymbol{M}_i)^T \boldsymbol{\Sigma}_i^{-1} (\boldsymbol{X} - \boldsymbol{M}_i)}$

$cost_t = \dfrac{1}{N_1 + N_2} \sum_{i=1}^{2} \sum_{j=1}^{N_i} (t_j^{(i)} - f(\boldsymbol{V}^T \boldsymbol{X}_j^{(i)} + v_0))^2$ is a cost function

$f(\boldsymbol{V}^T \boldsymbol{X}_j^{(i)} + v_0)$ is an activation function

$t_j^{(i)}$ is a desired output (a target) for $\boldsymbol{X}_j^{(i)}$, the j-th training vector from ω_i

$\Phi\{a\} = \int_{-\infty}^{a} (2\pi)^{-1/2} \sigma^{-1} \exp\{-\tfrac{1}{2} t^2/\sigma^2\} dt$ is a standard Gaussian cumulative distribution function

$\varepsilon_B = \Phi\{-\tfrac{1}{2}\delta\}$ is a Bayes error for two Gaussian populations with common CM

ε_∞^A is an asymptotic error for the classifier A

ε_N^A is a conditional probability of misclassification (PMC), a conditional generalisation error

$\bar{\varepsilon}_N^A$ is an expected probability of misclassification or simply an expected generalisation error

tr is a trace of the matrix (a sum of diagonal elements)

$\mathbf{I}_r$ is an $r \times r$ identity matrix (ones on the diagonal and zeros outside)