PATTERN RECOGNITION WITH DISCRETE AND MIXED DATA: THEORY AND PRACTICE

PATTERN RECOGNITION WITH DISCRETE AND MIXED DATA: THEORY AND PRACTICE

PROEFSCHRIFT

TER VERKRIJGING VAN DE GRAAD VAN DOCTOR
AAN DE ERASMUS UNIVERSITEIT ROTTERDAM
OP GEZAG VAN DE RECTOR MAGNIFICUS
PROF. DR. A. H. G. RINNOOY KAN
EN VOLGENS BESLUIT VAN HET COLLEGE VAN DEKANEN.
DE OPENBARE VERDEDIGING ZAL PLAATSVINDEN OP
WOENSDAG 7 SEPTEMBER 1988 OM 13.45 UUR

DOOR

CARLOS EDUARDO DE AMORIM QUEIROS
GEBOREN TE LISBOA

PROMOTIECOMMISSIE

PROMOTOR: PROF. DR. E. S. GELSEMA

PROMOTOR: PROF. DR. IR. J. H. VAN BEMMEL

OVERIGE LEDEN: PROF. DR. J. D. F. HABBEMA
DR. R. P. W. DUIN

to my parents

to Isabel, Joana and Teresa

CONTENTS

CHAPTER 1

INTRODUCTION

This thesis is devoted to aspects related to the analysis of medical data bases in the context of pattern recognition. It contains both theoretical aspects and practical applications and its scope includes questions and problems that arise when applying pattern recognition methods and techniques to this type of data.

As an introduction to this thesis, this chapter contains a general discussion and a presentation of the contents and organization of the thesis. Also, the main points of notation and terminology are introduced and defined.

1.1. Medical data and pattern recognition.

The goal of the application of statistical pattern recognition techniques to medical records, is the classification of the (disease) patterns that may be present in such records in terms of the information they contain. Typically, a medical record contains a description of history, symptoms, results from laboratory tests, signals, etc., all related to a given patient, i.e. all the information normally required by a physician when making a diagnosis and/or a prognosis. Pattern recognition may be used in order to obtain procedures (computer implemented algorithms) to assign diagnostic or prognostic classes to a given patient, on the basis of information also used by a physician. These procedures are not intended to replace but to assist the physician in the decision making process. The procedures are called classifiers or discriminants and the symptoms, signals, etc., are called features. Each individual record is termed an object, and a collection of objects with qualitatively and/or quantitatively similar characteristics, as established by an expert, is called a class. It should be clear that pattern recognition can be applied to a wide variety of areas and problems, of which (computer-aided) medical decision making is just an example.

In order to arrive at a classifier and restricting ourselves to what is called supervised learning, a set of objects known a-priori to belong to two or more classes (depending on the problem at hand) is needed. In this set, each object must be represented by a group of features and have a class assigned to it. The role of medical data bases is now clear: they are the set of objects required for supervised learning.

Since an object is described by features, we first of all concentrate on the typical nature of some of the features that may be found in medical data bases.

An example of a symptom is the presence or absence of pain at some anatomical location. In order to describe this symptom, a variable that can only assume two values (categories) should be used. Since the symptom cannot be measured quantitatively because it is essentially of a qualitative nature, the way in which the two categories are coded is immaterial, provided that they are different. Variables that take their values or categories from a domain which is discontinuous and which does not include any continuous sub-domain, are called discrete variables. Discrete features have often to be considered when coping with medical data. In this particular example, and since only two values are possible, we speak of a binary discrete feature.

A further example of a symptom is the following: is the pain located in the chest, the left shoulder or the right shoulder ? This is similar to the previous case with the exception that there are three possible values (categories) that the symptom may assume.

Another example of a symptom is the following: is the pain mild, moderate or strong ? Again there is something qualitative that requires a discrete variable (three categories) in order to describe it. Nevertheless, there is an intrinsic difference between this example and the previous ones. In this symptom, an order relation exists between its categories (mild is less than moderate which in turn is less than strong) whereas in the other examples, there is no such relation. In order to distinguish between these two cases, discrete variables are either ordered discrete variables or non ordered discrete variables.

Laboratory tests generally yield results that assume values in a domain with a large (infinite) number of elements. Therefore, continuous variables are particularly suited to describe them.

There are also features (e.g. the age of the patient in years) that can only assume a finite number of different values, but that are nevertheless better described by continuous variables. This approach can be justified as follows. Firstly, they are quantitative measures. Secondly, their discrete nature stems from rounding off values in an infinite domain. Finally, the number of different values they can assume is relatively large, implying that errors due to their treatment as continuous variables, will not be significant.

This brief analysis of the various types of features, which may be found in medical data bases, indicates that mixed data types (discrete and continuous) will have to be dealt with. We may think of three approaches to cope with such data:

i) The use of methods that can handle mixed data sets.

ii) The discretization of continuous features according to some criterion followed by the use of techniques suitable for discrete data only.

iii) The assignment of quantitative values, according to some criterion, to discrete categories and the subsequent use of techniques suitable for continuous data.

Intuitively, the first approach appears to be the most logical one. Models can be found in the literature that handle mixed data. However, these models may be (computationally) fairly complex to apply and may require a large number of parameters to estimate. As for the second approach, there are models and techniques which are very appealing. Furthermore, the framework of discrete feature spaces (i.e. spaces where all features are assumed to be discrete) presents the researcher with a fruitful ground to develop and test general procedures that can also be applied in other spaces. Nevertheless, the required discretization of continuous features, necessarily brings a certain loss of information. Finally, the third approach has an important drawback: even if some values can be found which quantify categories of a feature such as 'yes' or 'no', it is nevertheless fully incorrect to subsequently treat them as continuous variables.

A major problem that arises in medical data bases, is the unfortunate occurrence of missing values. It is usual to find records that are incomplete, i.e. records where not all the information required has been collected. Such missing values present problems at two levels: at the level of analysis of the medical data base and at the level of the application, i.e. at the level of applying classifiers in computer-aided medical decision making.

A literature search which covered some of the most representative journals, conference proceedings, textbooks, etc., showed that the problems just mentioned, have not been sufficiently studied and analyzed by what can be called, 'the pattern recognition community'. On the other hand, a substantially large amount of papers and textbooks can be found in the more traditional statistical literature, although not always directly related to the specific problems associated with pattern recognition and discrimination.

This thesis attempts to look into some of the problems of applying pattern recognition techniques to the analysis of medical data bases.

1.2. Contents and organization of this study.

Since this thesis concentrates on problems arising when applying pattern recognition techniques to medical data bases, two concepts form a central theme:

1 - The occurrence of discrete features, either in isolation (discrete feature spaces) or in combination with continuous features (mixed features spaces).

2 - The occurrence of missing values.

Both concepts present problems. Classical pattern recognition techniques have been developed and theoretically analyzed, mainly for continuous feature spaces. This applies not only to classification techniques, but also to techniques for feature selection and error estimation. Also, in the classical pattern recognition literature, the concept of missing values is only superficially treated.

Therefore, in this thesis it is attempted to treat these two aspects in pattern recognition in a systematic way. In doing so, it soon became clear that it was desirable to develop the treatment of both areas further than was available in the literature. This gave rise to the subjects presented in the following chapters:

Chapter 2: theoretical work leading to a new error bound. It is valid for a certain type of classifier. Since the bound is an expected value with respect to all possible training sets of a given size, it takes into account sample fluctuations.

Chapter 3: the development of new methods for feature selection. Specifically, these new methods are built around the concept of structure selection as opposed to the concept of feature selection. A simulation study, presenting results in a well controlled environment, is also included. Finally, a theoretical exercise is presented which shows a situation where an incorrect but simpler model for the conditional probability of an object given its class, yields a classifier which is better than a classifier based on a correct model.

Chapter 4: an analysis of the problems associated with the occurrence of missing values. Also the presentation of new approaches to estimate missing values and a simulation study of the design and use of a classifier in the case of incomplete data. Both discrete and continuous data are considered.

Chapter 5: an overview and discussion of existing techniques for discrete and/or mixed feature spaces as a background for the study of two medical data sets.

Chapter 6: the analysis of two different medical data bases, where some of the techniques described or proposed, were applied. The two medical data sets that were analyzed were on acute chest pain and severely head injured patients.

Briefly, it can be said that, given a major general problem (in our case, the application of pattern recognition techniques to medical data bases), specific problems were identified and solutions were devised and/or proposed.

The main conclusions, discussion and identification of new problems are presented in chapter 7. The somewhat unusual decision to put a review chapter

(chapter 5) next to the end of the thesis, is justified by the fact that chapter 5 was mainly intended as a background for two studies presented in the following chapter. In chapter 5, methods which were not used in the analysis presented in chapter 6 were nevertheless also reviewed, for the sake of completeness, and in order to make this thesis as much as possible self-contained.

1.3. Software developments.

Closely associated with this thesis and, in a way, work which is also to be considered part of it, is the extension of the interactive package for pattern recognition ISPAHAN (see GELS80) with a versatile set of procedures for discrete and mixed data and for incomplete data sets. For this development, some existing methods were implemented with or without modifications thought to be relevant. Also, a set of new methods was developed and implemented. A brief description of this work is given in appendix F.

The integration of the various algorithms into ISPAHAN, proved its value in the analysis of two medical data sets, reported in chapter 6.

1.4. Basic notations.

In this section, the main points of notation are defined. An arbitrary object $\underline{x}$, characterized by p features, will be denoted by a p-dimensional vector $\underline{x} = (x_1,x_2,...,x_p)^T$ in which the superscript ^{T} indicates the transpose. Also, the notation $\underline{x}_{1,2,...,p}$ will be used to denote an arbitrary object. A realization of a discrete feature will be denoted by x_{i_j} where i_j is the category assumed by the j-th discrete feature. The number of categories that the j-th discrete feature can assume, will be denoted by I_j.

Classes will be denoted either by a discrete variable w_i, or by an uppercase character (e.g. A, B, etc.). Whenever the class membership of an object is known, the object will be indicated by the symbol $\underline{x}$ with a superscript indicating the class (e.g. $\underline{x}^A$).

The probability of object $\underline{x}$, given class w_i, is written as Prob($\underline{x}|w_i$) or P($\underline{x}|w_i$). The joint probability functions of object and class are written as Prob($\underline{x},w_i$) or P($\underline{x},w_i$). A-priori class probabilities are written as Prob(w_i) or P(w_i). Since a Bayes framework is generally assumed throughout this thesis, discriminant functions become comparisons between a-posteriori probabilities of a class given an object (indicated as Prob($w_i|\underline{x}$) or P($w_i|\underline{x}$)), or comparisons between joint probabilities of object and class.

Both the symbols Prob and P, relate to true values. Estimated values are written as Prob^ or P^ (e.g. P^($\underline{x}$|A) is the estimated value of the conditional probability of object $\underline{x}$ given class A).

The number of objects available in a data set will be denoted by the symbol n. The symbol n^i will indicate the number of objects available from class w_i. Considering p discrete features, the symbol $n_{i1,i2,...ip}$, will indicate the number of objects available that scored simultaneously in the i_1-th category of feature 1, the i_2-th category of feature 2, ..., and the i_p-th category of feature p. Also, and in order to simplify the notation, sometimes the elements of a discrete feature space will be indicated by a variable j assuming integer values. In these cases, the symbol n_j will denote the number of objects available in element j of the discrete feature space. The symbols $n_{+,...,+,ij,+,...,+,ik,+,...,+}$ or $n_{j,k}$ will be used to denote the number of objects available that scored simultaneously in the i_j-th category of feature j and the i_k-th category of feature k, the remaining features not being considered. With all these symbols and whenever the object's class is known, a superscript indicating the class will be added to the symbol.

It may happen that minor deviations from this notation will appear now and then. This will be clearly indicated, whenever appropriate. Also, this basic notation will be recalled at the proper places, in order to facilitate the understanding of the text.

CHAPTER 2

UPPER BOUNDS

In this chapter, upper bounds for the expected classification error and for the expected non-optimal error component (expectation with respect to the training sets), are derived. They are applicable in the general measurement space (see HUGH68) in a two class classification problem. The training sets are assumed to be randomly sampled from the universe of objects.

2.1. Introduction.

Several upper bounds for the classification error have been proposed with the aim of providing easily computable expressions for the error. As pointed out by Duin (see DUIN77), they generally do not take into account sampling fluctuations and therefore are of no use for answering questions such as, 'How many objects are needed to guarantee a given accuracy ?'

In DUIN77 (see also DUIN78) a different approach is taken. For the case of the general measurement space (see HUGH68) it leads to upper bounds on the expected value of the classification error and the non-optimal error component (defined later on), that are dependent on the sample size (n), the measurement complexity [1] (m) and the Bayes error (ε_1).

In this chapter, steps are taken in deriving bounds for the expected values of the classification and non-optimal error components, that have as parameters n and m (upper bound on the non-optimal error component), and n, m and ε_1 (upper bound on the classification error). The theory developed here is applicable whenever we have:

[1] The measurement complexity, as introduced by Hughes (see HUGH68) for the general measurement space, is the number of different elements in that space. More generally, it refers to the number of parameters in a classifier that need to be estimated. This is consistent with the Hughes's case. It should not be confused with the term dimensionality which refers to the number of features.

i) a general measurement space with two classes.

ii) a classifier based on the Bayes rule, and estimated with the help of a training set obtained by randomly sampling from the universe of objects.

iii) a classifier which, in the case of equal estimates, randomly assigns an object to a class.

This chapter is organized as follows. In section 2, notations and some basic concepts are introduced and/or recalled. In section 3 the bounds are presented together with the proofs. Finally section 4 presents concluding remarks and a discussion. All (tedious) proofs are presented in appendices.

2.2. The expected non-optimal error component.

Let $\underline{x}$ be a p-dimensional vector representing an observed object pattern. Let w be a random variable representing class membership. The joint probability density function (p.d.f.) of $\underline{x}$ and w will be represented by

$$P(\underline{x},A) = P(\underline{x}|A)\ P(w{=}A)$$

$$P(\underline{x},B) = P(\underline{x}|B)\ P(w{=}B) \qquad (2.1)$$

where A and B are the two possible occurrences of w (discrimination between two classes only is considered here). In order to design a classifier, a set of objects is randomly selected from the universe of objects. Based on this sample, estimates are obtained for $P(\underline{x},A)$ and $P(\underline{x},B)$. Let $P^\wedge(\underline{x},A)$ and $P^\wedge(\underline{x},B)$ designate the estimates. The error of a Bayes classifier that makes use of $P^\wedge(\underline{x},A)$ and $P^\wedge(\underline{x},B)$ is

$$\varepsilon = \int_{P^\wedge(\underline{x},A)\leq P^\wedge(\underline{x},B)} P(\underline{x},A)\ d\underline{x} \quad + \quad \int_{P^\wedge(\underline{x},A)>P^\wedge(\underline{x},B)} P(\underline{x},B)\ d\underline{x} \qquad (2.2)$$

or

$$\varepsilon = \int_{P^\wedge(\underline{x},A)<P^\wedge(\underline{x},B)} P(\underline{x},A)\ d\underline{x} \quad + \quad \int_{P^\wedge(\underline{x},A)>P^\wedge(\underline{x},B)} P(\underline{x},B)\ d\underline{x} \quad +$$

$$.5 \int_{P^\wedge(\underline{x},A)=P^\wedge(\underline{x},B)} (P(\underline{x},A) + P(\underline{x},B))\ d\underline{x} \qquad (2.3)$$

depending on the way decisions are made when $P^\wedge(\underline{x},A) = P^\wedge(\underline{x},B)$: always choose the same class (expression 2.2), or randomly choose one of the two classes (expression 2.3). Only the latter case is considered. Following Duin (DUIN78), the error may be expressed in two components:

$$\varepsilon = \varepsilon_1 + \varepsilon_2 \tag{2.4}$$

where ε_1 is the Bayes error and ε_2 is the classifier non-optimal error component (ε_2 is the component of the error that, in a specific design situation, may be reduced with the help of pattern recognition methods and techniques). It is defined as

$$\begin{aligned}\varepsilon_2 = & \int\limits_{\substack{P(\underline{x},A)\leq P(\underline{x},B)\\ P^\wedge(\underline{x},A)>P^\wedge(\underline{x},B)}} (P(\underline{x},B) - P(\underline{x},A))\, d\underline{x} \quad + \int\limits_{\substack{P(\underline{x},A)>P(\underline{x},B)\\ P^\wedge(\underline{x},A)<P^\wedge(\underline{x},B)}} (P(\underline{x},A) - P(\underline{x},B))d\underline{x} \quad + \\ & .5 \quad \Bigg(\int\limits_{\substack{P(\underline{x},A)\leq P(\underline{x},B)\\ P^\wedge(\underline{x},A)=P^\wedge(\underline{x},B)}} (P(\underline{x},B) - P(\underline{x},A))\, d\underline{x} \quad + \int\limits_{\substack{P(\underline{x},A)>P(\underline{x},B)\\ P^\wedge(\underline{x},A)=P^\wedge(\underline{x},B)}} (P(\underline{x},A) - P(\underline{x},B))d\underline{x} \quad \Bigg)\end{aligned} \tag{2.5}$$

In a specific problem, the expected value of ε with respect to all training sets is

$$E_{\underline{x}}(\varepsilon) = \varepsilon_1 + E_{\underline{x}}(\varepsilon_2) \tag{2.6}$$

where

$$\begin{aligned}E_{\underline{x}}(\varepsilon_2) = & \int\limits_{P(\underline{x},A)\leq P(\underline{x},B)} (P(\underline{x},B) - P(\underline{x},A)) \;\; \text{Prob}\,(P^\wedge(\underline{x},A) > P^\wedge(\underline{x},B))\; d\underline{x} \quad + \\ & \int\limits_{P(\underline{x},A)>P(\underline{x},B)} (P(\underline{x},A) - P(\underline{x},B)) \;\; \text{Prob}\,(P^\wedge(\underline{x},A) < P^\wedge(\underline{x},B))\; d\underline{x} \quad + \\ & .5 \int\limits_{P(\underline{x},A)\leq P(\underline{x},B)} (P(\underline{x},B) - P(\underline{x},A)) \;\; \text{Prob}\,(P^\wedge(\underline{x},A) = P^\wedge(\underline{x},B))\; d\underline{x} \quad +\end{aligned}$$

$$.5 \int_{P(\underline{x},A)>P(\underline{x},B)} (P(\underline{x},A) - P(\underline{x},B)) \text{ Prob } (P\hat{~}(\underline{x},A) = P\hat{~}(\underline{x},B)) \, d\underline{x} \qquad (2.7)$$

This is the expected non-optimal error component of a classifier. It is clear that ε_2 is the only component of the error that depends on the composition of the training set.

In a discrete feature space, each vector $\underline{x}$ may be uniquely mapped to a scalar j, representing each possible occurrence of $\underline{x}$ and assuming all integer values in the interval [1 , m], where m is the measurement complexity ($m = I_1 * I_2 * ... * I_p$). Expression (2.7) may then be rewritten as

$$\begin{aligned} E_{\underline{x}}(\varepsilon_2) = & \sum_{j=1}^{m_1} (P(j,B) - P(j,A)) \text{ Prob } (P\hat{~}(j,A) > P\hat{~}(j,B)) \quad + \\ & \sum_{j=m_1+1}^{m} (P(j,A) - P(j,B)) \text{ Prob } (P\hat{~}(j,A) < P\hat{~}(j,B)) \quad + \\ & .5 \sum_{j=1}^{m_1} (P(j,B) - P(j,A)) \text{ Prob } (P\hat{~}(j,A) = P\hat{~}(j,B)) \quad + \\ & .5 \sum_{j=m_1+1}^{m} (P(j,A) - P(j,B)) \text{ Prob } (P\hat{~}(j,A) = P\hat{~}(j,B)) \end{aligned} \qquad (2.8)$$

where it is assumed that $P(j,A) \leq P(j,B)$, for $1 \leq j \leq m_1$, and $P(j,A) > P(j,B)$ for $m_1 + 1 \leq j \leq m$. This assumption brings no loss of generality.

2.3. The bounds.

In order to derive the bound on the expected non-optimal error component, a procedure is followed that is basically the maximization of equation 2.8 with respect to P(j,A) and P(j,B) for all j, under the constraints

$$\sum_{j=1}^{m} P(j,A) + \sum_{j=1}^{m} P(j,B) = 1$$

$$\sum_{j=1}^{m_1} P(j,A) + \sum_{j=m_1+1}^{m} P(j,B) = \varepsilon_1 \qquad (2.9)$$

Further on, these constraints are going to be relaxed.

A method is needed in order to estimate the parameters of the classifier. The maximum likelihood technique is selected. This does not necessarily restrict the area of application of the bounds to the case of classifiers estimated with this technique (e.g. for the case of a Bayesian estimator which assumes the parameters of the classifier to be uniformly distributed, the bound still holds). The maximum likelihoods estimates of P(j,A) and P(j,B) are

$$P\hat{}(j,A) = n^{A}{}_{j} / n \qquad P\hat{}(j,B) = n^{B}{}_{j} / n \qquad (2.10)$$

where $n^{A}{}_{j}$ ($n^{B}{}_{j}$) is the number of objects from class A (class B) present in the training set that represent the j-th possible occurrence of object $\underline{x}$.

The sampling distribution can be expressed by a multinomial with parameters

$$n,\ P(1,A),\ \ldots,\ P(m,A),\ P(1,B),\ \ldots,\ P(m,B) \qquad (2.11)$$

Using 2.8, 2.10 and 2.11 the expression to maximize can be written as

$$\sum_{j=1}^{m_1} (P(j,B) - P(j,A)) \qquad ($$

$$\sum_{i=0}^{n} \sum_{t=0}^{\min(i,n-i)} n! / (i!\ t!\ (n-t-i)!)\ P(j,A)^{i}\, P(j,B)^{t}\, (1-P(j,A)-P(j,B))^{n-i-t}\ -$$

$$.5 \sum_{i=0}^{f(n)} n! / (i!\ i!\ (n-2i)!)\ P(j,A)^{i}\, P(j,B)^{i}\, (1-P(j,A)-P(j,B))^{n-2i} \qquad)$$

$$+ \sum_{j=m_1+1}^{m} (P(j,A) - P(j,B)) \qquad ($$

$$\sum_{i=0}^{n} \sum_{t=0}^{\min(i,n-i)} n! / (i!\; t!\; (n-t-i)!)\; P(j,B)^{i}\, P(j,A)^{t}\, (1-P(j,B)-P(j,A))^{n-i-t} \; -$$

$$.5 \sum_{i=0}^{f(n)} n! / (i!\; i!\; (n-2i)!)\; P(j,B)^{i}\, P(j,A)^{i}\, (1-P(j,B)-P(j,A))^{n-2i} \qquad) \tag{2.12}$$

where

$$f(n) = n / 2 \qquad \text{if n is even}$$

$$f(n) = ((n + 1) / 2) - 1 \qquad \text{if n is odd} \tag{2.13}$$

Unfortunately, the direct maximization of 2.12 under the constraints 2.9, is a complex problem. This arises from the fact that 2.12 as a function of P(j,A) (P(j,B)) is not a 'well behaved' function, i.e. the second derivative with respect to P(j,A) (P(j,B)) can be positive in the domain of P(j,A) (P(j,B)). In order to be able to proceed, a change of variables is required. Let

$$k(j) = P(j,A) + P(j,B) \qquad \text{for } 1 \leq j \leq m$$

$$q(j) = |P(j,A) - P(j,B)| \qquad \text{for } 1 \leq j \leq m \tag{2.14}$$

It is clear that $q(j) \leq k(j)$. The constraints in 2.9 can be rewritten as

$$\sum_{j=1}^{m} k(j) = 1 \qquad \sum_{j=1}^{m} q(j) = 1 - 2\,\varepsilon_1 \tag{2.15}$$

The problems concerning the maximization still remain (equation 2.12 as a function of q(j) is not 'well behaved'). To overcome this, two new constraints are imposed.

$$k(1) = 1 \tag{2.16}$$

$$k(j) = q(j) \qquad \text{for every } j \neq 1 \tag{2.17}$$

These constraints increase the value of

$$K = \sum_{j=1}^{m} k(j) \tag{2.18}$$

so that it is guaranteed that K is larger than 1 (2.16) and is as close as possible to 1 (2.17).

It is now proved that the maximum of 2.12 under constraints 2.16 and 2.17 is larger than or equal to the maximum of 2.12 under constraints 2.15. It is shown in appendix A that 2.12 is an increasing function of k(j). Therefore it is also an increasing function of K. Consequently, a maximum of 2.12 under constraint 2.15 is also larger than or equal to the maximum under constraints 2.9. Constraint 2.17 does introduce a relation between the values of q(j) and k(j). However, since 2.16 guarantees that $1 \leq K$, the maximum of 2.12 under these constraints is necessarily larger than or equal to the maximum when the constraints are those expressed in 2.15. Introducing 2.16 and 2.17 in 2.12, we are left with

$$.5 \sum_{j=2}^{m} q(j)\,(1-q(j))^{n} \quad +$$

$$q(1) \quad (\quad \sum_{i=f(n)+1}^{n} n!\,/\,(i!\,(n-i)!)\,(.5\,(1-q(1)))^{i}\,(.5\,(1+q(1)))^{n-i} \quad +$$

$$.5\,n!\,/\,(f(n)!\,f(n)!)\,(.5\,(1-q(1)))^{f(n)}\,(.5\,(1+q(1)))^{f(n)} \quad)$$

for n even, and

$$.5 \sum_{j=2}^{m} q(j)\,(1-q(j))^{n} \quad +$$

$$q(1) \sum_{i=f(n)+1}^{n} n!\,/\,(i!\,(n-i)!)\,(.5\,(1-q(1)))^{i}\,(.5\,(1+q(1)))^{n-i} \tag{2.19}$$

for n odd. The values that maximize the terms under the summation over the j's are

$$q(j) = 1\,/\,(n+1) \qquad \text{for } 2 \leq j \leq m \tag{2.20}$$

The proof is straightforward. In what concerns the remaining terms in both expressions in 2.19 (the maximization with respect to q(1)), the result is derived in appendix B. It is sufficient to say here that a gaussian approximation is used. The maximum is

$$.17 / \sqrt{n} \tag{2.21}$$

The bound is now obtained by replacing 2.20 and 2.21 into 2.19.

Proposition: ε_d is an upper bound for the expected non-optimal error component.

$$E_{\underline{x}} (\varepsilon_2) \leq \varepsilon_d = .5 (m - 1) / (n + 1) (1 - 1 / (n + 1))^n +.17 / \sqrt{n} \tag{2.22}$$

The proof has just been presented.

Corollary: ε_c is an upper bound for the expected classification error.

$$E_{\underline{x}} (\varepsilon) \leq \varepsilon_c = .5 (m - 1) / (n + 1) (1 - 1 / (n + 1))^n +.17 / \sqrt{n} + \varepsilon_1 \tag{2.23}$$

The proof is immediate. Both 2.22 and 2.23 hold only for a feature space, classifier and training set as defined in the introduction.

Two limits are of interest. When an infinite number of objects is available and the measurement complexity is finite, ε_c and ε_d are equal to

$$\lim_{n \to \infty} \varepsilon_d = 0$$

$$\lim_{n \to \infty} \varepsilon_c = \varepsilon_1 \tag{2.24}$$

Therefore, when n → ∞, ε_d and ε_c are equal to the quantities they bound.

2.4. Discussion.

The bound just presented (only the bound on the expected non-optimal error component is considered in this discussion) has three important advantages. First, it is easy to compute. Secondly, none of its parameters (n and m) require an estimate. Thirdly, for $m < n$ and for a given admissible expected non-optimal error component, the number of objects required in the training set compares favorably to m (e.g. for a maximum admissible expected non-optimal error component equal to .05 and a measurement complexity of 100, the bound indicates that approximately 420 objects from both classes should be obtained).

The first two advantages are self evident. In what concerns the third and in order to have an idea about the number of objects required to guarantee a given expected non-optimal error component, a graph is presented in fig. 2.1 that depicts the evolution of ε_d as a function of the number of objects available in the training set (x axis), parametrized by the measurement complexity m. Only situations such that $m < n$, are shown. This is the region of interest anyhow.

There are also negative points. First, its field of application is restricted to classification problems which satisfy the conditions enumerated in the introduction. Secondly, the use of a gaussian approximation seems to bring with it the usual requirement of a large n. Thirdly, the methodology followed, i.e. a worst case approach, together with the relaxing of important constraints, raises questions over the tightness of the bound in a particular situation.

The first point is inherent. As for the second, it does not apply here (see appendix B). As for the tightness, it depends on the particular situation.

Finally, the bounds do provide an answer to the question formulated in the introduction. Moreover, the relation obtained between m, n, and ε_d (m, n, ε_1 and ε_c) is a reasonable one (for $m < n$).

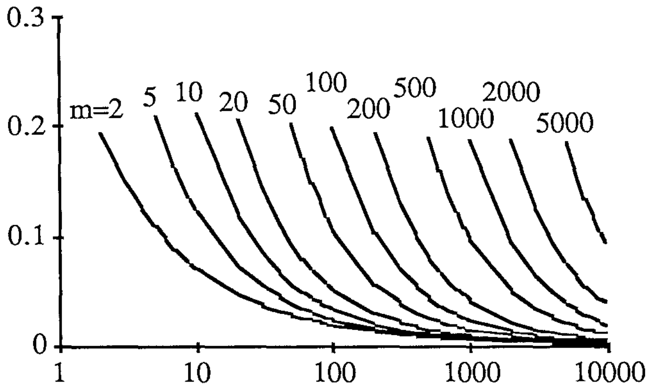


Fig. 2.1: Each curve relates the upper bound on the non-optimal error component (y axis) with the number of objects available in the training set (x axis) for a given measurement complexity (m), for n greater than or equal to m.

CHAPTER 3

FEATURE SELECTION

In this chapter, the main topic is feature selection. We shall elaborate on the importance of search strategies allowing for structure changes, by making use of qualitative reasoning as well as by presenting the results of sampling experiments. These strategies, popular in regression analysis, have generally not been used by the pattern recognition community. The overall topic will allow us to digress into other areas (e.g. error estimation), where some points are also discussed.

3.1. Introduction.

Several arguments may be given for the need of feature selection when the practitioner is faced with the design of a pattern recognition system. The most important is the possibility of occurrence of the peaking phenomenon, i.e. the existence of an optimal complexity [1] beyond which the performance of the pattern recognition system starts to deteriorate. Quoting a nicely formulated sentence by Jain and Chandrasekaran (JAIN82)

> 'In some sense the errors caused by the non-optimal use of added information, outweights the advantages of extra information.'

In order to avoid peaking, the number of features to be used might have to be restricted, in which case the best ones should be kept. Hence, there is a need for a feature selection strategy and feature selection algorithms.

Feature selection algorithms are characterized by a search procedure, a selection criterion and a stopping rule. These may be roughly defined as the strategy for moving in the domain of all possible combinations of features, the quantity used to measure the quality of a given feature set, and the set of rules that, if met, will cause the algorithm to stop.

[1] The term complexity, in a general sense, refers to the number of parameters in a classifier that need to be estimated. See footnote 1 in chapter 2.

Generally, the selection criterion is an (implicit or explicit) function of probability functions involving features and/or classes (e.g. error rate for a given classification problem). Since these probability functions are generally not known, models have to be assumed, a structure selected [2], and parameters estimated.

Usually a model and a structure are chosen a-priori and maintained throughout the selection. For instance, if the gaussian model is assumed together with a structure of non-independence, neither the model nor the structure are changed (e.g. to a non gaussian model and/or to independent features) throughout the selection process. This is not satisfactory. In this chapter it will be shown that it is important to be more flexible as far as the structure is concerned. The following example explains why.

Assume that the selection algorithm uses a forward search (at each step a new feature is selected and added: this is the type of search considered throughout this chapter), and that the selection criterion is a function of joint probability functions of features and class membership.

i) Let no structure at all be imposed on the probability functions involved. All possible interactions between features are then taken into account and the discriminatory ability available in such interactions may be retained. However, only a relatively small number of features may be expected to be reliably selected since the set of objects available for estimation has a finite size and the number of parameters required may be large even for a small number of features. The resulting restricted set may not be adequate to achieve a reasonable discrimination between classes.

ii) Let a stronger probability structure be assumed. For instance, conditional independence of the features given the class. In this case, the number of parameters that needs to be estimated is drastically reduced and the number of features that may be selected may not be as limited as in case i). However, neglecting the interactions between features (this is the consequence of having assumed independence) may cause loss of discriminating power even in stages of the search where interactions could be safely taken into account (it is assumed that a structure of independence is also used in the final design of the classifier).

[2] Given a model for a multidimensional feature space, its structure are the relations of independence assumed between the features. A strong probability structure is one where all features are independent whereas a weak probability structure (or no structure at all) is one where there are few (or no) relations of independence or conditional independence between features.

These two cases are two extreme situations. Nevertheless, they show how the performance of a feature selection algorithm may be affected and how this may affect the final classifier.

The importance of a more flexible approach allowing for structure changes, will be stressed here, i.e. given a convenient general model, at each step, test the quality of a feature not only in one but in various settings considering different interactions with the set of features so far selected. Selecting a feature, implies the acceptance and subsequent estimation of a number of unknown parameters. For a given model, allowing for structure changes amounts to a more refined analysis of the unknown parameters that each feature brings into play. Instead of feature selection, the term parameter selection is more appropriate.

It is very difficult (if not impossible) to present conclusive theoretical results (the same holds for all non-optimal search strategies). Consequently, computer experiments had to be performed. Only discrete feature spaces were considered. A general pattern for the various experiments may now be given.

Various feature spaces defined by a number of binary features were established. For each space, two classes were defined with equal a-priori probabilities, assumed to be known. Each experiment consisted of the execution of various feature selection algorithms. These can be grouped in three sets, according to the overall search strategy (and/or model) used (assumed). They are:

i) Algorithms I (Independence).

The algorithms that use a stepwise forward search and that assume that initially and throughout the selection the features are independent given the class, i.e. no structure changes being allowed. These algorithms will be referred to as algorithms I.

ii) Algorithms F (Full).

The algorithms that use a stepwise forward search and that assume initially and throughout the selection a full multinomial model for the probability functions of the features given the class, i.e. no structure changes being allowed. These algorithms will be referred to as algorithms F.

iii) Algorithms C (Change).

The algorithms that assume the basic model (see chapter 5) as the general model and that allow structure changes during the selection. This search strategy will be called the augmented stepwise forward search. The structures allowed are restricted only to those admitting independence between sets of features given the class. This set of structures includes both the cases of full independence and full multinomial. However, it does not include every possible structure. These algorithms will be referred to as algorithms C.

Therefore, two search strategies were used that adopt a fixed structure and one that allows for structure changes.

The performance of the various algorithms was compared by means of criteria to be defined later on.

The experiments themselves may be grouped into two sets. In one case, it was required to randomly generate artificial data. In the other case, a more theoretical approach was followed, which did not require data to be randomly generated.

In the following sections, the various experiments are presented. Also the selection criteria used and the question of estimating the parameters of a classifier, given its model and structure (the selection criteria used require that classifiers are estimated), are dealt with. Since the latter applies to both sets of experiments, it will be presented first. The notation used is the one presented in chapter 1, unless otherwise stated.

3.2. Estimation of classifiers.

For every experiment, three estimators were used. Let us assume k sets of independent features given the class [3]. Let n^J be the number of samples available from class J, m the complexity of the space defined by all features, m_i the complexity of the subspace defined by a set of non-independent features, $P^{\wedge J}_{i1,\dots,ik}$ the estimated probability given class J (the indicator i_j may assume all the integer values in the interval $[1,m_j]$). Note that $m = q\ m_1 \dots m_k$ where q does not need to be equal to 1 (if not all the features are used, then $q > 1$). Let $n^J_{+,\dots,+,il,+,\dots+}$ be the number of samples available from class J that score in element i_l of the corresponding subspace. The three estimators are then:

i) Maximum likelihood (MLK)

$$P^{\wedge J}_{i1,\dots,ik} = (n^J_{i1,+,\dots,+} / n^J) * \dots * (n^J_{+,\dots,+,ik} / n^J) \tag{3.1}$$

[3] When all features are assumed to be fully independent, k is equal to the number of features. When independence is not assumed (the probability function given the class is best represented by a full multinomial), then k is equal to 1.

ii) Bayesian estimate (BAY)

$$P^{\wedge J}_{i1,\ldots,ik} = (n^J_{i1,+,\ldots,+} + 1) / (n^J + m_1) * \ldots * (n^J_{+,\ldots,+,ik} + 1) / (n^J + m_k) \tag{3.2}$$

iii) Another Bayesian estimate (ANB)

$$\begin{aligned} P^{\wedge J}_{i1,\ldots,ik} = {} & (n^J_{i1,+,\ldots,+} + m/m_1) / (n^J + m) * \ldots \\ & * (n^J_{+,\ldots,+,ik} + m/m_k) / (n^J + m) \end{aligned} \tag{3.3}$$

A few more words concerning the estimators BAY and ANB. The first one is a commonly used Bayesian estimator and is derived by assuming that the probabilities of a subspace defined by a set of non-independent features (i.e. the parameters of the subspace) have an uniform distribution. If this is true, the estimator is Bayesian optimal. As for the second, each marginal estimate is derived by assuming a two class problem and that the probabilities for the full dimensionality (not for subspaces as in the case of the BAY estimator) are again uniformly distributed. If this is true and if a classifier is designed using only one subset of non-independent features, then the classifier is Bayesian optimal. If more then one set of non-independent features are used, then it is not optimal anymore.

3.3. Specific feature spaces.

The first set of experiments uses three different discrete feature spaces for the discrimination between two classes. In a simulation experiment, for each feature space, 30 different training sets of a given size were generated, sampling independently per class. Each set of data contained 32 (in some cases 64) objects per class.

For every space and training set, the three basic search strategies were applied with various selection criteria, based on three different ways of estimating the error rate. Also, the three classifier estimators described in the previous section were applied in order to obtain classifiers as required by the error rate estimators. The selection was stopped either whenever all features were selected or when two consecutive iterations produced error estimates larger than the current minimum. It was decided not to stop when the error did not decrease (a more obvious rule) because in discrete feature spaces, very often the error does not change when a single feature is added, whereas if a second feature is added a substantial decrease may occur. The smallest set of features that gave the minimum error estimate was the set selected.

The outcome of each selection process is a set of features and a structure for the classifier (either imposed a-priori or selected). The performance of each selection process was assessed by computing the actual error of the classifier designed in accordance with the results of the selection procedure. This quantity could be computed since the underlying distributions were fully known. The error average over the 30 training sets is an estimate of the expected error with respect to the training sets and therefore was used to assess the quality of each selection technique for a particular discrimination problem and object set size.

In the next two subsections, both the feature spaces used as well as the different ways of estimating the error rates are discussed in detail.

3.3.1. The feature spaces.

The first feature space used is defined by 10 binary features. Given the class, these are assumed to be independent. A log-linear model (see chapter 5) was used to represent each class-conditional probability function. The non-zero coefficients of the model are given in table 3.1. This table shows that only the differences between ten parameters are important for discrimination. The Bayes errors for each individual feature as well as for the ten features taken together are given in table 3.2. It is thus a highly structured discrimination problem. This space will be referred to as the H (highly structured) space.

The second feature space is again defined by 10 binary features. Given the class, six features and the set composed of the remaining four, are assumed to be independent. Again a log-linear model (see chapter 5) was used to represent each class conditional probability function. The non-zero coefficients of the model are given in table 3.3. This table shows that only the differences between 20 parameters are important for discrimination. All but the fourth order factor are non-zero for the four features assumed to be non-independent. The interactions between these four features are important for discrimination. Table 3.4 presents the Bayes errors for each feature taken individually as well as for the ten features taken together. We thus have a moderately, leaning to highly, structured discrimination problem. This space shall be referred to as the M (moderately structured) space.

The third feature space used is defined by 5 binary features. In this case, the log-linear model was not used. Instead, a pseudo-random generator was used to obtain preliminary values for each individual probability. Final values were obtained by applying the constraints that the probabilities have to sum up to one, and that the Bayes error for the full dimensionality should be equal to 0.1. Table 3.5, presents the Bayes error when every feature is considered individually and when all possible combinations of four features are considered. The 'poor' performance of all combinations of four features does suggest that the class conditional probability functions are highly non-structured and that the discriminatory power lies in the

Coef.	Class A	Class B
C (1)	.0793	-.0793
C (2)	.1477	-.1477
C (3)	.2079	-.2079
C (4)	.2616	-.2616
C (5)	.3100	-.3100
C (6)	.3543	-.3543
C (7)	.3949	-.3949
C (8)	.4324	-.4324
C (9)	.4674	-.4674
C (10)	.5000	-.5000

Table 3.1: Non-zero coefficients of the log-linear model used to define feature space H.

Features	Bayes error
Feat. 1	.46
Feat. 2	.43
Feat. 3	.40
Feat. 4	.37
Feat. 5	.35
Feat. 6	.33
Feat. 7	.31
Feat. 8	.30
Feat. 9	.28
Feat. 10	.27
All	.14

Table 3.2: Bayes error for each individual feature and for the full dimensionality, in the case of feature space H.

Coefficients	Class A	Class B
C (1)	.5	-.5
C (2)	.4	-.4
C (3)	.3	-.3
C (4)	.2	-.2
C (5)	.1	-.1
C (6)	.05	-.05
C (7)	.05	-.05
C (8)	.05	-.05
C (9)	.05	-.05
C (10)	.05	-.05

Table 3.3: Non-zero coefficients of the log-linear model used to define feature space M (see continuation).

Coefficients	Class A	Class B
C (7,8)	.2	-.2
C (7,9)	.3	-.3
C (8,9)	.3	-.3
C (7,10)	.4	-.4
C (8,10)	.4	-.4
C (9,10)	.4	-.4
C (7,8,9)	.3	-.3
C (7,8,10)	.4	-.4
C (7,9,10)	.4	-.4
C (8,9,10)	.4	-.4

Table 3.3: Continuation.

Features	Bayes error
Feat. 1	.27
Feat. 2	.31
Feat. 3	.35
Feat. 4	.40
Feat. 5	.45
Feat. 6	.47
Feat. 7	.36
Feat. 8	.36
Feat. 9	.36
Feat. 10	.35
All	.075

Table 3.4: Bayes error for each individual feature and for the full dimensionality, in the case of feature space M.

Features	Bayes error
Feat. 1	.43
Feat. 2	.50
Feat. 3	.46
Feat. 4	.47
Feat. 5	.36
4 Features	.25
4 Features	.28
4 Features	.22
4 Features	.27
4 Features	.21
All	.10

Table 3.5: Bayes error for each individual feature, for every combination of 4 features and for the full dimensionality, in the case of feature space L.

interactions between the features. It can also be inferred that the differences between a minimum of 15 (2^4 -1) and a maximum of 31 (2^5 - 1) parameters are important for discrimination. This space will be referred to as the L (low structured) space.

The justification for using these three feature spaces now follows. Firstly, it was decided to consider feature spaces with varying degrees of structure. If a scale measuring structure was constructed, space H, space M and space L would be in this order in that scale, from high to low. Secondly, unfavourable spaces for algorithm C as compared to one of the other competing strategies, were sought. This is the case for space H and space L. In these two situations, a-priori knowledge about the true structure is present in one search strategy: for space H, strategy I, and for space L, strategy F. A better performance for strategy C was not expected but it was important to compare it with the performance of an optimum strategy. Thirdly, the feature spaces were tailored in such a way that no single feature (or, e.g. pairs of features) could give errors close to the optimum. In this way, more detailed analyses were required from the selection algorithms. Finally, and in particular for space L, the need to obtain reasonably stable classifiers, has led us to consider 5 binary features only.

3.3.2. The selection criterion.

A key parameter in the overall performance of a feature selection algorithm is the selection criterion. It was decided to use the error rate. Two reasons may be given for this choice:

i) The error rate is the natural choice for comparing classifier designs (and that is what feature selection is about).

ii) The experiments with feature selection provided a good opportunity to evaluate some error estimation techniques [4].

Three different error estimators were used. They are presented in the following.

[4] A good estimator is one that gives estimates close to the true values. However, for the specific purpose of feature selection, this definition can be loosened: good estimators are those that give estimates that rank in the same order as the true values. That is, if the actual error associated with a set of features is larger than the one associated with another set of features, then the estimated value of the former should be larger than the estimated value of the latter. In other words, the need for a good estimator in the general sense, is a sufficient although not a necessary condition for good performance of the feature selection algorithm.

3.3.2.1. Error estimation.

Due to the fact that the number of objects available was small, rotation techniques were used. A general framework for these kind of techniques is as follows. Given a set of objects, the entire data is partitioned a number of times into a training and a test set. For each partition, a classifier is designed using the objects in the training set, and is applied to the objects in the test set. The error is then estimated by counting the number of misclassifications and averaging over all partitions.

Therefore, in order to apply rotation methods, and given a set of data, both the size of one set (e.g. training set) and the number of partitions to be used have to be defined. If n objects are available and it is decided to use test sets of size k ($k < n$), the number of ways to partition the data available is equal to number of combinations of k elements in a set containing n elements. For reasons of reliability, the three estimators adopted all require that more than one partition be used.

3.3.2.2. Leave-one-out.

If k is equal to 1 and all possible partitions are explored, we have the first rotation technique used. This is the leave-one-out method [5] (LACH67). This is a well known technique, its main properties are:

i) it yields an almost unbiased estimate (LACH67).

ii) for discrete feature spaces it has a large variance (GLIC78).

Although it requires the design of several classifiers, it can be implemented in such a way that in the case of discrete feature spaces, computing time remains acceptable. Throughout this chapter, the leave-one-out technique will be referred to as estimator 1.

[5] This method is often referred to as a jackknife method. This is not correct. Although both methods partition the objects available and make use of different partitions, for each partition, the leave-one-out uses all the objects either as members of the training set or as members of the test set, whereas a jackknifing technique does not use, in any way, the objects left out (see MILL74).

3.3.2.3. Other ratios for partitioning.

Other partitioning ratios may be used although most likely not all possible partitions can be explored. Indeed, their number may be enormous and the computer time required absurd. Still, a given number may be randomly selected, the number being such that computer time remains manageable.

Error estimators number two and three can now be defined:

- Second estimator: rotation using test sets with 3 samples from each class and 25 different randomly generated partitions.

- Third estimator: rotation using test sets with 16 samples from each class (i.e. half the data available) and 7 different partitions, again randomly generated.

Since a-priori probabilities were assumed to be known and an equal number of objects per class was available, the generation of each training set was subject to the condition that it should contain an equal number of objects from both classes. These two estimators will be referred to throughout this chapter as estimators 2 and 3, respectively.

The size of each the test set used in the second estimator, was imposed to be approximately 10% of the total number of objects available (32 per class). The number of partitions used was established taking into account the computer time required. As for the third estimator, which Toussaint (TOUS74) calls the modified hold-out method, each test set contained 50% of the total number of objects available and the number of partitions was fixed by the requirement that estimators 2 and 3 should have a similar chance. A good measure of equality is, in this case, the computer time required. In arriving at the number of 7, it was assumed that the time required for one classification was equal to the time the program requires to make use of one object in the estimation of the classifier and that the design of an 'optimal' program would have to be different. It may be remarked that every error estimate requires 150 classifications when estimator 2 is used, and 224 classifications when estimator 3 is used.

The properties of such estimators, to our knowledge, have not yet been studied. However, some conclusions may be made from qualitative reasoning about the properties.

i) Bias:

Each partition in each estimator, yields an error estimate that is known to be pessimistically biased. Furthermore, the bias will increase when the number of objects assigned to each training set is decreased.Therefore, it may be concluded that both estimators are pessimistically biased. Moreover, this bias is likely to be more pronounced in the case of the third estimator since fewer objects are used to design the classifiers.

ii) Variance:

Two effects have to be considered:

1 - The number of different values that, in each partition, the estimated error may assume in the interval [0.,1.], is larger for estimator 3.

2 - More partitions are used in estimator 2.

The first effect leads us to expect that estimator 3 has a smaller variance (following a reasoning similar to Glick's (see GLIC78) when discussing the variance of the leave-one-out). The second effect leads us to expect that estimator 2 has smaller variance. Which of these effects is dominant and how the variances compare to the variance of the leave-one-out method, theoretically it is not clear.

3.3.2.4. Breaking ties. Insensitivity of the error.

However good the estimator may be, the error rate will perform poorly as a selection criterion if its insensitivity is not taken into account. Specifically, and in the context of a forward search, several candidate features may yield the same error when combined with the set of features already selected, in spite of the fact that individually some might have a good discrimination ability and others might not be able at all to discriminate. A random choice is then required and it may happen that the feature selected is one with no discriminatory ability at all. This poses problems: a useless feature is selected and the performance of the algorithm in the next steps of the search, is handicapped. Duin (DUIN78) has shown this effect for the case of independent binary features. Ben-Bassat (BEN80) has addressed the same problem by allowing more general models and introducing the question of insensitivity with respect to a-priori probabilities. Both have assumed that the relevant probabilities are known. It is clear that the problem remains when estimates are used. The case studied by Duin is of special significance here since independence models are explicitly assumed. To correct for the insensitivity, a solution proposed by Ben-Bassat (BEN80) has been used. In case of equal error estimates, the tie is broken by selecting the feature and structure for which the variance of the error over the current candidate space is largest. This variance is estimated either by estimator 1, or 2, or 3, if the estimator used for the error is 1, or 2, or 3, respectively.

Finally, in the case of a tie both in the error as well as the variance, the tie-breaking rule is then to select the feature and structure that require the smaller number of parameters.

3.3.2.5. Remark.

Three different error estimators (estimators 1, 2 and 3) were applied. Since all of them require that classifiers are estimated and since three different estimators were used for this purpose (MLK, BAY and ANB), a total of 9 different selection criteria were used.

3.3.3. Results.

The results of the experiments are presented in tables 3.6 through 3.14. Each table relates to a particular feature space and a given method of estimating the parameters of a classifier (e.g. table 3.6 relates to space H and the MLK estimation technique). Each row in a table is related to a given search technique (i.e. C, F or I defined above) and a given error estimation method (i.e. estimators 1, 2 and 3 defined above), and presents the estimated mean and standard deviation (between parentheses) of the actual error (i.e. the true classification error), the estimated error, the number of features selected and the number of parameters per class that had to be estimated. All these entries relate to the 'best' classifier as chosen by a feature selection algorithm.

In the following, an analysis and discussion of the results with respect to the search strategies and the error estimators, is presented.

3.3.3.1. Search strategy.

Considering first the highly structured space H (i.e. independence between features; tables 3.6 - 3.8), it appears that the best strategy is I followed by C. 'Best' is here assessed by the true errors of the resulting classifiers. This is not surprising in this space since search I always makes correct assumptions as far as the structure of the space is concerned. Nevertheless, and except for the combination of Bayesian designs and the leave-one-out error estimator as the selection criterion, the performances achieved are similar for these two strategies. As for strategy F, the assumption of interactions between features, which has no meaning in this space, implies that generally fewer features are selected which in turn implies that the results are the worst. Nevertheless, for the combination of maximum likelihood design and the leave-one-out error estimator, the performance of search F comes close to the C search. For this specific combination, an experiment involving again 30 different sets with 64 objects per class, was performed. The results are shown in the first three rows of table 3.15. The improvement achieved when going from search F to search C, is clearly seen.

		Actual error	Estimated error	Num. of feat.	Num. of param.
Estimator 1	C	22.6 (3.4)	10.6 (3.6)	5.3 (1.9)	10.1 (6.0)
	F	24.8 (3.6)	11.7 (3.0)	3.9 (0.6)	15.3 (7.1)
	I	21.3 (2.6)	12.2 (3.6)	5.0 (1.6)	5. (1.6)
Estimator 2	C	24.5 (3.5)	11.6 (3.9)	5.2 (1.9)	9.4 (4.4)
	F	26.8 (3.8)	14.7 (5.2)	3.1 (1.0)	10.9 (7.3)
	I	22.6 (3.7)	15.9 (6.1)	5.1 (1.7)	5.1 (1.7)
Estimator 3	C	22.3 (3.6)	14.8 (3.6)	5.6 (2.1)	9.0 (4.6)
	F	26.7 (3.6)	19.3 (3.7)	2.6 (1.2)	7.2 (6.5)
	I	22.3 (4.7)	16.5 (4.4)	5.1 (2.7)	5.1 (2.7)

Table 3.6: Results of the experiments described in section 3.3, for space H and classifiers estimated using the MLK estimator. Each entry in the table shows an estimated mean and between parentheses, an estimated standard deviation. All values have been multiplied by 100. For more details, see subsection 3.3.3.

		Actual error	Estimated error	Num. of feat.	Num. of param.
Estimator 1	C	25.6 (8.2)	8.2 (5.1)	6.2 (1.9)	77.0 (155.3)
	F	38.1 (5.2)	1.0 (1.6)	7.6 (1.0)	240 (166.1)
	I	21.6 (3.0)	12.7 (3.7)	5.1 (1.6)	5.1 (1.6)
Estimator 2	C	22.9 (3.3)	10.3 (3.8)	5.9 (2.0)	12.1 (6.0)
	F	26.8 (3.8)	14.7 (5.2)	3.1 (1.0)	10.0 (7.3)
	I	22.3 (3.8)	12.0 (3.6)	5.1 (1.7)	5.1 (1.7)
Estimator 3	C	21.9 (3.6)	13.7 (4.3)	6.3 (2.4)	12.5 (6.1)
	F	26.7 (3.6)	19.3 (3.7)	2.6 (1.2)	7.2 (6.5)
	I	21.7 (5.6)	15.9 (4.5)	5.7 (2.7)	5.7 (2.7)

Table 3.7: Results of the experiments described in section 3.3, for space H and classifiers estimated using the BAY estimator. All values have been multiplied by 100. For more details see both subsection 3.3.3 and the caption of table 3.6.

		Actual error	Estimated error	Num. of feat.	Num. of param.
Estimator 1	C	24.4 (7.3)	8.2 (5.1)	6.1 (1.9)	59.4 (132.5)
	F	38.1 (5.2)	1.0 (1.6)	7.6 (1.0)	240 (166.1)
	I	20.6 (3.0)	11.8 (3.3)	5.5 (2.2)	5.5 (2.2)
Estimator 2	C	22.9 (3.9)	9.7 (3.7)	5.7 (2.1)	12.2 (12.6)
	F	26.8 (3.8)	14.7 (5.2)	3.1 (1.0)	10.0 (7.3)
	I	21.2 (3.1)	12.1 (3.5)	5.5 (1.7)	5.5 (1.7)
Estimator 3	C	22.1 (3.0)	13.4 (3.7)	5.4 (2.0)	9.3 (4.3)
	F	26.7 (3.5)	19.3 (3.7)	2.6 (1.2)	7.2 (6.5)
	I	21.5 (5.1)	15.3 (4.8)	5.5 (2.3)	5.5 (2.3)

Table 3.8: Results of the experiments described in section 3.3, for space H and classifiers estimated using the ANB estimator. All values have been multiplied by 100. For more details see both subsection 3.3.3 and the caption of table 3.6.

Turning now to the M feature space (tables 3.9 - 3.11), algorithms using search C yield generally the best results, the exception being the combination of Bayesian designs and leave-one-out error estimators (search I is then the best). It may be remarked that this combination tends to yield classifiers (for the C and F searches) that require more parameters than the number of objects available which explains why search I is the best [6]. This behavior occurred also in the H feature space and denotes a failure of the stopping criterion. For the combination of maximum likelihood design and leave-one-out error estimator, the performance of search I comes close to that of search C. For this specific combination, an experiment involving again 30 different sets with 64 objects per class, was performed. The results are shown in the last three rows of table 3.15. The improvement achieved when going from search I to search C, is clearly seen. It is interesting to observe that search F replaces search I as the second best. This can be attributed to the larger number of objects used.

Finally, turning now to the low structured space (tables 3.12 through 3.14), the best search is F, followed by search C. This is a feature space where the discriminating power is mainly concentrated in high-order interactions between features making search F specially attractive for this space. Search I yields a very poor performance and this may be traced to the assumption of independence, which implies the omission of exactly those significant interactions between features. Also, the small number of features selected by this strategy, is not surprising.

In general, it may be said that search C yields better results. Either it is the best or close to it. The latter occurs when the knowledge available in one of the competing strategies matches with the space structure. The good performance of search C is a direct cause of its flexibility with respect to the structure of the space.

3.3.3.2. Error estimates.

Comparing the performance of the three error estimators in terms of the achieved classifier performance, it can be said that they yielded similar results. It may be added that the leave-one-out method, in the case of the structured spaces (spaces H and M) and Bayesian designs, yielded the worst results whereas this combination performed better in the L space.

[6] It is certainly an incorrect practice to allow for more parameters than the number of objects available. However, this is an experiment and it is important to identify both the positive as well as the negative aspects of the various parts of the feature selection algorithms tested.

		Actual error	Estimated error	Num. of feat.	Num. of param.
Estimator 1	C	20.2 (4.9)	9.1 (3.6)	5.6 (1.7)	12.4 (5.1)
	F	23.5 (5.2)	11.5 (3.4)	3.9 (0.6)	15.8 (9.9)
	I	21.4 (2.5)	14.2 (4.0)	4.7 (1.6)	4.7 (1.6)
Estimator 2	C	20.5 (4.4)	8.8 (3.9)	6.6 (1.8)	13.9 (6.7)
	F	24.2 (5.4)	13.7 (4.7)	3.6 (1.0)	14.3 (9.5)
	I	24.1 (4.3)	14.4 (5.2)	4.5 (2.1)	4.5 (2.1)
Estimator 3	C	21.2 (4.7)	13.4 (4.4)	6.4 (2.4)	10.8 (4.7)
	F	28.9 (4.8)	19.3 (3.7)	2.4 (1.2)	6.0 (4.9)
	I	24.2 (4.2)	17.6 (3.9)	4.1 (2.1)	4.1 (2.1)

Table 3.9: Results of the experiments described in section 3.3, for space M and classifiers estimated using the MLK estimator. All values have been multiplied by 100. For more details see both subsection 3.3.3 and the caption of table 3.6.

		Actual error	Estimated error	Num. of feat.	Num. of param.
Estimator 1	C	23.9 (9.7)	5.4 (6.0)	6.5 (1.7)	83.1 (189.6)
	F	33.4 (7.8)	0.2 (0.6)	7.0 (1.3)	203.8 (253.9)
	I	22.3 (2.8)	14.5 (4.2)	5.6 (5.4)	5.6 (5.4)
Estimator 2	C	20.6 (4.9)	8.5 (4.3)	5.7 (1.9)	11.5 (5.2)
	F	24.2 (5.4)	13.7 (4.7)	3.6 (1.0)	14.3 (9.5)
	I	24.2 (4.1)	14.2 (5.6)	4.6 (2.0)	4.5 (2.0)
Estimator 3	C	18.7 (5.2)	12.4 (4.3)	6.5 (2.4)	17.2 (9.8)
	F	28.9 (4.8)	19.3 (3.7)	2.4 (1.2)	6.0 (4.9)
	I	24.2 (4.1)	17.2 (4.4)	4.7 (2.3)	4.7 (2.3)

Table 3.10: Results of the experiments described in section 3.3, for space M and classifiers estimated using the BAY estimator. All values have been multiplied by 100. For more details see both subsection 3.3.3 and the caption of table 3.6.

		Actual error	Estimated error	Num. of feat.	Num. of param.
Estimator 1	C	27.9 (10.6)	3.4 (4.7)	6.6 (1.6)	122.1 (204.6)
	F	33.4 (7.8)	0.2 (0.6)	7.0 (1.3)	203.8 (253.9)
	I	24.1 (2.5)	15.2 (3.4)	4.5 (1.6)	4.5 (1.6)
Estimator 2	C	21.7 (7.0)	8.4 (4.9)	5.5 (1.8)	16.5 (11.0)
	F	24.2 (5.4)	13.7 (4.7)	3.6 (1.0)	14.3 (9.5)
	I	24.8 (3.8)	15.7 (5.6)	4.5 (2.0)	4.5 (2.0)
Estimator 3	C	19.7 (6.4)	12.1 (5.3)	5.8 (1.9)	14.5 (9.0)
	F	28.9 (4.8)	19.3 (3.7)	2.4 (1.2)	6.0 (4.9)
	I	25.9 (4.4)	18.2 (5.5)	4.0 (2.1)	4.0 (2.1)

Table 3.11: Results of the experiments described in section 3.3, for space M and classifiers estimated using the ANB estimator. All values have been multiplied by 100. For more details see both subsection 3.3.3 and the caption of table 3.6.

		Actual error	Estimated error	Num. of feat.	Num. of param.
Estimator 1	C	23.8 (6.3)	18.2 (5.1)	4.3 (0.8)	18.7 (11.7)
	F	22.0 (5.9)	17.0 (4.0)	4.5 (0.7)	23.3 (9.3)
	I	38.1 (4.4)	32.6 (6.3)	1.9 (1.2)	1.9 (1.2)
Estimator 2	C	26.7 (8.3)	20.8 (6.8)	4.1 (1.0)	12.5 (9.6)
	F	24.6 (8.1)	19.7 (5.5)	4.0 (1.4)	20.6 (12.0)
	I	39.0 (5.1)	32.3 (6.0)	1.3 (1.0)	1.3 (1.0)
Estimator 3	C	27.8 (8.0)	25.6 (6.4)	4.1 (1.3)	13.9 (10.6)
	F	27.1 (7.8)	26.3 (5.9)	3.5 (1.5)	15.4 (11.5)
	I	39.1 (4.8)	34.1 (6.3)	1.4 (0.8)	1.4 (0.8)

Table 3.12: Results of the experiments described in section 3.3, for space L and classifiers estimated using the MLK estimator. All values have been multiplied by 100. For more details see both subsection 3.3.3 and the caption of table 3.6.

		Actual error	Estimated error	Num. of feat.	Num. of param.
Estimator 1	C	22.7 (6.2)	14.3 (7.6)	4.5 (0.8)	21.6 (12.1)
	F	19.7 (4.6)	10.9 (5.2)	4.9 (0.4)	29.1 (5.8)
	I	38.1 (4.1)	32.5 (6.0)	1.9 (1.3)	1.9 (1.3)
Estimator 2	C	27.3 (8.7)	21.7 (7.1)	4.1 (1.5)	12.0 (9.3)
	F	24.6 (8.1)	19.7 (5.5)	4.0 (1.4)	20.6 (12.0)
	I	38.5 (4.7)	32.4 (6.1)	1.3 (1.0)	1.3 (1.0)
Estimator 3	C	27.8 (8.1)	25.9 (5.9)	3.8 (1.3)	13.3 (10.9)
	F	27.1 (7.8)	26.3 (5.9)	3.5 (1.5)	15.4 (11.5)
	I	38.8 (4.5)	34.2 (6.3)	1.4 (0.8)	1.4 (0.8)

Table 3.13: Results of the experiments described in section 3.3, for space L and classifiers estimated using the BAY estimator. All values have been multiplied by 100. For more details see both subsection 3.3.3 and the caption of table 3.6.

		Actual error	Estimated error	Num. of feat.	Num. of param.
Estimator 1	C	23.2 (6.7)	14.7 (7.9)	4.4 (1.0)	21.4 (12.3)
	F	19.7 (4.6)	10.9 (5.2)	4.9 (0.4)	29.1 (5.8)
	I	38.7 (3.9)	33.2 (5.3)	1.7 (1.2)	1.7 (1.2)
Estimator 2	C	27.6 (8.7)	21.7 (6.9)	3.9 (1.4)	11.8 (9.5)
	F	24.6 (8.1)	19.7 (5.5)	4.0 (1.4)	20.6 (12.0)
	I	39.2 (4.9)	32.2 (6.2)	1.3 (1.0)	1.3 (1.0)
Estimator 3	C	28.0 (7.6)	26.6 (6.3)	3.8 (1.4)	13.1 (10.4)
	F	27.1 (7.8)	26.3 (5.9)	3.5 (1.5)	15.4 (11.5)
	I	39.1 (4.8)	34.3 (6.3)	1.4 (0.8)	1.4 (0.8)

Table 3.14: Results of the experiments described in section 3.3, for space L and classifiers estimated using the ANB estimator. All values have been multiplied by 100. For more details see both subsection 3.3.3 and the caption of table 3.6.

		Actual error	Number of features	Number of param.
H	C	19.3 (2.9)	6.3 (1.9)	12.1 (6.6)
	F	23.4 (2.5)	4.8 (0.5)	27.8 (9.8)
	I	17.5 (2.1)	6.5 (2.0)	6.5 (2.0)
M	C	16.3 (4.3)	5.9 (1.9)	16.2 (8.5)
	F	18.6 (4.5)	4.6 (0.6)	24.6 (10.8)
	I	19.9 (2.2)	5.8 (2.2)	5.8 (2.2)

Table 3.15: Results of the experiments described in subsection 3.3.3, when data sets of 64 objects per class were used. The upper part relates to the H space and the lower part to the M space. In both cases, the selection criterion was the leave-one-out error estimator and the classifiers required were designed using the MLK technique.

As for the estimates themselves, it is clear that a common feature seems to hold for all error estimates of the 'best' classifiers: all are substantially optimistic. This contradicts the expected behavior of the estimators described above. An explanation follows. The properties presented above pertain to the estimate of the error rate of one particular classifier. The error estimate of the 'best' classifier is the estimate of the minimum error over a set of classifiers by means of a set of objects that essentially remains the same for every evaluation. Thus, it is a completely different statistic. The importance of this aspect may better be seen by considering the case where only one feature is to be selected. In table 3.16 the results of a simple experiment are compiled. Using intermediate results from the experiments described above, mean values and standard deviations were estimated for the estimated values of the error of the single best feature (actual best feature) and for the single 'best' feature as selected by the selection criteria. Here again the latter is optimistic for the three estimators considered, whereas the relation between the number of parameters (in this case one) and the number of objects available (32 objects per class) is a reasonable one although not for the minimum error over a set, as shown by the results.

Briefly the three estimators failed to cope with the new circumstances introduced by the selection and produced a misleading final estimate. If part of the samples available (e.g. half) had been set aside and used exclusively for the estimation of the error of the 'best' classifier as produced by the feature selection, a more realistic value would have been attained. Nevertheless, in order to achieve better performance with feature selection, it is important that efforts are directed towards the study of the properties of the various available selection criteria (be it the error rate or others ones), when the circumstances of their application involves an optimization procedure.

3.4. Families of feature spaces.

In the previous section, three specific feature spaces have been used in order to show the importance of allowing structure changes during the selection. Similar experiments but involving feature selection over families of feature spaces (and all training sets of a given size) will now be presented. The search strategies used were the three techniques mentioned in section 3.1. The selection criterion was the mean recognition accuracy, a concept first introduced by Hughes (see HUGH68) and used by this author and others (e.g. CHAN71, CHAN74, CHAN75, etc.), to study the peaking phenomenon. It is the expectation of the probability of correct recognition over all training sets and over an assumed family of feature spaces. That is:

$$E_\theta (E_\chi (P_{cr})) \qquad (3.4)$$

		Error estimator 1	Error estimator 2	Error estimator 3
Selected best	H	22.9 (2.9)	22.3 (3.3)	21.9 (3.5)
	M	24.4 (3.5)	21.8 (4.1)	23.4 (4.0)
	L	34.3 (5.6)	32.6 (6.0)	34.7 (6.6)
Actual best	H	25.5 (4.3)	27.4 (6.1)	25.5 (4.5)
	M	27.1 (4.6)	27.0 (6.7)	26.9 (5.8)
	L	35.8 (6.7)	36.0 (6.6)	37.0 (8.6)

Table 3.16: This table shows the results of the experiments described in subsection 3.3.3.2. Each entry in this table shows the estimated means and standard deviations (between parentheses). Column-wise, the results relate to the three error estimators used. Row-wise, the results relate to the three feature spaces and to the selected 'best' feature and the actual best feature. The upper part of this table, is associated with estimates for the 'best' feature as indicated by the selection criteria. The lower part is associated with error estimates for the actual best feature.

The symbol P_{cr} indicates the probability of correct recognition, the symbol E_χ indicates expectation with respect to all training sets of a given size and the symbol E_θ indicates expectation with respect to the parameters of a family of feature spaces.

It should be remarked that this criterion could be computed exactly, and therefore did not require an estimate. Nevertheless, and in order to apply it, classifiers had to be estimated. Three different estimators (MLK, BAY, ANB) were again used. The stopping criterion was taken to be a decrease in the value of the mean recognition accuracy.

Two different families were considered. Each feature space in each family was assumed to be defined by p binary features, yielding a space composed of $m = 2^p$ different elements. Furthermore, two classes, called A and B, were assumed with equal a-priori probabilities. In the following subsections, the two families and the results obtained are presented.

3.4.1. First family of feature spaces.

In order to completely define a family of feature spaces, a probability distribution for the parameters of the spaces composing the family, has to be defined. Let $P(u_i)$ and $P(v_i)$ be the probability of a parameter of the space for class A and class B, respectively. A general measurement space is assumed (see HUGH68) and therefore each parameter is the joint probability of an element of the space and a class. It is assumed that:

$$P\ (u_1,...,u_m,v_1,...,v_m) = P\ (u_1,...,u_m)\ P\ (v_1,...,v_m) \tag{3.5}$$

Furthermore, it is also assumed that $P(u_1,...,u_m)$ is uniform for $0 \leq u_1 \leq \alpha \leq 1$ and zero otherwise. That is:

$$\begin{aligned} P\ (u_1,...,u_m) &= k_1\ du_1\ ...\ du_m \qquad \text{for } 0 \leq u_1 \leq \alpha \leq 1 \\ &= 0 \qquad\qquad\qquad\qquad \text{otherwise} \end{aligned} \tag{3.6}$$

where k_1 is equal to

$$k_1 = (m-1)!\ /\ (1 - (1-\alpha)^{m-1}) \tag{3.7}$$

This guarantees that the probabilities do sum up to one. As for class B, it is assumed that $P(v_1,...,v_m)$ is uniform for $0 \leq \alpha \leq v_1 \leq 1$ and zero otherwise. That is:

$$P\ (v_1,...,v_m) = k_2\ dv_1\ ...\ dv_m \quad \text{for } 0 \leq \alpha \leq v_1 \leq 1$$
$$= 0 \quad \text{otherwise} \tag{3.8}$$

where k_2 is equal to

$$k_2 = (m\text{-}1)!\ /\ (1\text{-}\alpha)^{m\text{-}1} \tag{3.9}$$

With respect to this family, the three estimation techniques used in the design of classifiers, are all non-optimal (it would be trivial if it were not so). In the first place, an infinite number of objects in the training set was not considered and hence the conditions of optimality of the MLK estimator are not met. Secondly, the probabilities $P(u_1,...,u_m)$ and $P(v_1,...,v_m)$ do not coincide with the ones assumed for the derivation of the Bayesian estimators, and hence the conditions of optimality for the two Bayesian estimators are also not met. This situation reflects a common problem associated with the Bayesian design of classifiers, where it is very difficult to known and/or to obtain estimates of the a-priori distribution of the parameters.

For this family and for most of the spaces that belong to it, the condition of independence (or quasi-independence) between features or sets of features does not hold. For the case of 2 binary features and for $\alpha = 0.25$, the probability of quasi-independence is equal to $\beta * 0.3415$ (class A) and $\beta * 0.7172$ (class B). Variable β quantifies the quasi-independence and its significance can better be perceived by remarking that the probability of quasi-independence between features is, e.g. for class A,

$$\int_0^{\alpha} du_1 \int_0^{1\text{-}u_1} du_2 \int_{b_0}^{b_1} du_3 = \beta\ k$$

where

$$b_0 = (1 - \beta/2)\ u_1\ (1\text{-}u_1\text{-}u_2)\ /\ (u_1 + u_2)$$

$$b_1 = (1 + \beta/2)\ u_1\ (1\text{-}u_1\text{-}u_2)\ /\ (u_1 + u_2) \tag{3.10}$$

The symbol k in the expression above, is a constant that guarantees that:

$$\left(\int_0^{\alpha} du_1 \int_0^{1-u_1} du_2 \int_0^{1-u_1-u_2} du_3 \right) / k = 1 \tag{3.11}$$

For reasonable values of β, the probability is indeed small. Thus, a second type of error (the first type of error is related to the estimation techniques used and was already indicated above) affected the design of classifiers that assumed independence between features. The error is itself this assumption.

For this family and in expectation, it should be remarked that all combinations of the same number of features have the same discriminatory power. So, the selection of features was, in this case, the selection of a structure and the selection of a number of features.

The mean recognition accuracy is equal to

$$n^A!\, n^B!\, (m-1)!\, (m-1)! \,/\, (1-\alpha)^{m-1} \,(1 - (1-\alpha)^{m-1}) \;*$$
$$\left(F_1(n^A{}_1,n^B{}_1,n^A,n^B,m,\alpha) + \sum_{i=2}^{m} F_2(n^A{}_1,n^A{}_i,n^B{}_1,n^B{}_i,n^A,n^B,m,\alpha) \right) \tag{3.12}$$

Functions $F_1(n^A{}_1,n^B{}_1,n^A,n^B,m,\alpha)$ and $F_2(n^A{}_1,n^A{}_i,n^B{}_1,n^B{}_i,n^A,n^B,m,\alpha)$ are defined in appendix C.

The expression above can be greatly simplified if the classifier assumes full multinomials for the conditional probability functions (the expectation is then the same for all m-1 elements other than element 1). Unfortunately, this is not the case when independence is assumed. As such, and in order to avoid excessive computation times, only small set sizes and spaces defined by a small number of features, were considered. Specifically, spaces defined by two, three and four binary features, assuming a number of objects per class ranging from 1 to 3, have been dealt with. Also due to excessive computation times, with four binary features the number of objects considered per class was either 1 or 2. As for α, it was taken to be 1/m.

The results of the experiments are presented in tables 3.17 through 3.19. Each entry in a given table represents the value of the mean recognition accuracy obtained and the structure chosen (the structure is coded in terms of the number of different elements in each marginal).

		Search I	Search C	Search F
MLK	1,1	.5895 (2-2)	.5895 (2-2)	.5868 (4)
	2,2	.6279 (2-2)	.6292 (4)	.6292 (4)
	3,3	.6449 (2-2)	.6542 (4)	.6542 (4)
BAY	1,1	.5908 (2-2)	.5908 (2-2)	.5868 (4)
	2,2	.6279 (2-2)	.6292 (4)	.6292 (4)
	3,3	.6435 (2-2)	.6542 (4)	.6542 (4)
ANB	1,1	.5868 (2-2)	.5868 (2-2)	.5868 (4)
	2,2	.6254 (2-2)	.6292 (4)	.6292 (4)
	3,3	.6457 (2-2)	.6542 (4)	.6542 (4)

Table 3.17: This table shows the results of the experiments described in subsection 3.4.1. (first family of feature spaces) when two binary features are considered. Each entry in the table shows the mean recognition accuracy and, between brackets, the structure selected (e.g. the upper left entry indicates that 2 binary features, assumed to be independent, were selected). The first column indicates the estimator used for the classifiers, the second column indicates the number of objects considered and finally, the first row indicates the search strategy.

		Search I	Search C	Search F
MLK	1,1	.5514 (2-2-2)	.5514 (2-2-2)	.5494 (8)
	2,2	.5777 (2-2-2)	.5829 (8)	.5829 (8)
	3,3	.5923 (2-2-2)	.6069 (8)	.6069 (8)
BAY	1,1	.5514 (2-2-2)	.5514 (2-2-2)	.5494 (8)
	2,2	.5778 (2-2-2)	.5829 (8)	.5829 (8)
	3,3	.5906 (2-2-2)	.6069 (8)	.6069 (8)
ANB	1,1	.5494 (2-2-2)	.5494 (4-2,8)	.5494 (8)
	2,2	.5777 (2-2-2)	.5829 (8)	.5829 (8)
	3,3	.5934 (2-2-2)	.6069 (8)	.6069 (8)

Table 3.18: This table shows the results of the experiments described in subsection 3.4.1. (first family of feature spaces) when three binary features are considered. See also the caption in table 3.17.

		Search I	Search C	Search F
MLK	1,1	.5281 (2-2-2-2)	.5281 (2-2-2-2)	.5273 (16)
	2,2	.5449 (2-2-2-2)	.5494 (16)	.5494 (16)
BAY	1,1	.5282 (2-2-2-2)	.5282 (4-2-2)	.5273 (16)
	2,2	.5449 (2-2-2-2)	.5494 (16)	.5494 (16)
ANB	1,1	.5273 (2-2-2-2)	.5273 (8-2)	.5273 (16)
	2,2	.5455 (2-2-2-2)	.5494 (16)	.5494 (16)

Table 3.19: This table shows the results of the experiments described in subsection 3.4.1. (first family of feature spaces) when four binary features are considered. See also the caption in table 3.17.

3.4.2. Second family of feature spaces.

As in the previous subsection, the probability distribution of the members of this family of feature spaces had to be defined. It was assumed that:

$$P\ (u_1,...,u_m,v_1,...,v_m) = P(u_1,...,u_m)\ P(v_1,...,v_m) \tag{3.13}$$

$P(v_1,...,v_m)$ is the same as $P(u_1,...,u_m)$ in the previous section. As for class A, $P(u_1,...,u_m)$ is uniform over the permissible range of $u_1,...,u_m$. That is:

$$P(u_1,...,u_m) = (m-1)!\ du_1\ ...\ du_m \tag{3.14}$$

Thus, this is a family similar to the one in the previous subsection. The three estimation techniques used in the design of classifiers are again non-optimal although the probability function given class B is optimally estimated when estimator ANB is used.

The probability of spaces where quasi-independence between features holds, was again quantified for the case of two binary features. For $\alpha = 0.25$, the probability of quasi-independence is equal to $\beta * 0.5$ (class A) and $\beta * 0.3415$ (class B). Similarly as with the first family, two types of errors affect the design of classifiers when independence is assumed, and one type of error when independence is not assumed.

Also for this family and in expectation, all combinations of the same number of features have the same discriminatory power. So, the selection of features is again the selection of a structure and the selection of a number of features.

The mean recognition accuracy is now equal to

$$n^A!\ n^B!\ (m-1)!\ (m-1)!\ /\ (\ (1-(1-\alpha)^{m-1})\ (n_A+m-1)!\ \ )\ \ (\\ G_1(n^A{}_1,n^B{}_1,n^A,n^B,m,\alpha) + \sum_{i=2}^{m} G_2(n^A{}_i,n^B{}_1,n^B{}_i,n^A,n^B,m,\alpha)\ \ ) \tag{3.15}$$

Functions $G_1(n^A{}_1,n^B{}_1,n^A,n^B,m,\alpha)$ and $G_2(n^A{}_i,n^B{}_1,n^B{}_i,n^A,n^B,m,\alpha)$ are defined in appendix D. Expression 3.15 may again be greatly simplified when, for a given

		Search I	Search C	Search F
MLK	1,1	.6351 (2-2)	.6351 (2-2)	.6171 (4)
	2,2	.6881 (2-2)	.6881 (2-2)	.6715 (4)
	3,3	.7144 (2-2)	.7144 (2-2)	.7041 (4)
BAY	1,1	.6441 (2-2)	.6441 (2-2)	.6171 (4)
	2,2	.6918 (2-2)	.6918 (2-2)	.6715 (4)
	3,3	.7112 (2-2)	.7112 (2-2)	.7041 (4)
ANB	1,1	.6171 (2-2)	.6171 (2-2)	.6171 (4)
	2,2	.6777 (2-2)	.6777 (2-2)	.6715 (4)
	3,3	.7106 (2-2)	.7106 (2-2)	.7041 (4)

Table 3.20: This table shows the results of the experiments described in subsection 3.4.2. (second family of feature spaces) when two binary features are considered. See also the caption in table 3.17.

		Search I	Search C	Search F
MLK	1,1	.6353 (2-2-2)	.6353 (2-2-2)	.5946 (4)
	2,2	.6842 (2-2-2)	.6842 (2-2-2)	.6441 (8)
	3,3	.7097 (2-2-2)	.7097 (2-2-2)	.6825 (8)
BAY	1,1	.6353 (2-2-2)	.6353 (2-2-2)	.5946 (4)
	2,2	.6909 (2-2-2)	.6909 (2-2-2)	.6441 (8)
	3,3	.7066 (2-2-2)	.7127 (2-4)	.6825 (8)
ANB	1,1	.5946 (2-2-2)	.5946 (4-2)	.5946 (4)
	2,2	.6594 (2-2-2)	.6594 (2-2-2)	.6441 (8)
	3,3	.6991 (2-2-2)	.6991 (2-2-2)	.6825 (8)

Table 3.21: This table shows the results of the experiments described in subsection 3.4.2. (second family of feature spaces) when three binary features are considered. See also the caption in table 3.17.

		Search I	Search C	Search F
MLK	1,1	.6276 (2-2-2-2)	.6276 (2-2-2-2)	.5815 (2)
	2,2	.6689 (2-2-2-2)	.6689 (2-2-2-2)	.6142 (2)
BAY	1,1	.6313 (2-2-2-2)	.6322 (4-2-2)	.5815 (2)
	2,2	.6753 (2-2-2-2)	.6753 (2-2-2-2)	.6142 (2)
ANB	1,1	.5815 (2)	.5815 (2)	.5815 (2)
	2,2	.6293 (2-2-2)	.6293 (2-2-2)	.6142 (2)

Table 3.22: This table shows the results of the experiments described in subsection 3.4.2. (second family of feature spaces) when four binary features are considered. See also the caption in table 3.17.

problem and training set, the classification of an object depends only on the number of samples available of the same type. If structure is assumed, this is not the case. This implies large computation times and consequently, the experiments performed are again limited. Specifically the same cases as in the previous section, have been dealt with. The results of the selection are presented in tables 3.20 through 3.22, where each entry presents the mean recognition accuracy obtained and the structure selected.

3.4.3. Results.

Looking at tables 3.17 through 3.22, it is clear that search C is either the best or the co-best. If, for the families of spaces defined by two binary features this may be considered trivial (in the sense that all possible types of classifiers are tested), the same does not hold for other cases. Nevertheless, it is fair to say that the mere increase in search space always gave search C an advantage, since the selection criterion is not an estimate (and therefore is not affected by sampling variations).

The better performance of search C is the result of permitting structure changes which allowed it to select more structured classifiers when the ratio between sample size and number of independent estimates required per classifier was small, and less structured classifiers when the ratio increased.

This theoretical exercise also shows how classifiers assuming incorrect structures (more precisely, structures with a small probability of occurrence), may perform better than classifiers that assume correct structures (more precisely, structures with a large probability of occurrence), but requiring a larger number of estimates. In this case, this was due to the use of non-optimal estimation techniques.

3.5. Search space.

A search strategy allowing for structure changes, implies an increase in the search space and consequently in computer time. Hence it is important to quantify the size of the search space [7]. Assuming that it is required to select n out of N features, the size of the search space is:

- For a forward search, selecting one feature at each iteration and not allowing structure changes

[7] The size of the search space is the number of feature sets and/or structures that are tested, when it is required to select n out of N features.

$$S = n\,N - .5\,n^2 + .5\,n \tag{3.16}$$

- For search strategy C

$$N_{min} \leq S \leq N_{max}$$

$$N_{min} = N + 2\,(\,(n-1)\,(N-1) - .5\,(n-1)^2 + .5\,(n-1)\,)$$

$$N_{max} = (N+1)\,(nN - n^2/2 + n/2) - (\,N\,(N+1)\,(2N+1) - (N-n)\,(N-n+1)\,(2(N-n)+1)\,)\,/\,6 \tag{3.17}$$

where S is the size of the search space. In expression 3.17 the lower limit relates to the situation where at each step, the full multinomial model is selected. The upper limit relates to the situation where at each step, structures of complete independence are selected.

The increase in the search space is indeed significant, especially if the selection criterion has to be computed from scratch for every classifier tried (this is the case of the error rate). A question has now to be raised: how does this search compare with others that adopt a fixed structure, but, at each step, add more than one feature and/or allow forward and backward tracking ? Briefly, how does search C compare with other search strategies that do not allow for structure changes but test a comparable number of solutions ? The answer to this question is very difficult to give. On the one hand a better result is to be expected from the fixed structure search, since combinations of features are tested instead of single features. On the other hand, the basic problems associated with fixed structure algorithms will generally be magnified. Which one of these effects will prove to be dominant is difficult to predict. It is nevertheless tempting to suggest that for small sample sizes, the second aspect is, most likely, the dominant one.

3.6. Concluding remarks.

The results of computer experiments concerning feature selection algorithms using various selection criteria and various ways of estimating the parameters of a classifier and involving specific feature spaces and families of feature spaces, have been presented. To these experiments, can be added those reported by Queiros et al. (QUEI84), which are similar to the ones presented here, the main differences concerning the selection criterion used (test of the model of independence between the random variable representing class membership and the random variable representing the candidate feature given the model and structure so far selected; this criterion is reviewed in chapter 5) and the fact that more structures were allowed. All the results support the qualitative reasoning given in the first section of this chapter.

Whenever a new feature is selected, it brings with it the requirement that one or more parameters will have to be estimated. It may happen that not all these parameters are important for discrimination. If nothing is known a-priori about the discriminatory ability of some of these parameters, it is important to make a more refined analysis of the discriminatory power of a feature. This can be achieved with feature selection algorithms incorporating structure selection.

All the experiments presented and/or referenced here, involved only binary features. There are no apparent reasons that may prevent the results to be generalized for spaces defined by non-binary features.

In the examples presented above, the set of admissible structures did not cover all possible ones. If the computational burden is manageable (for instance if the number of features is small or the selection criterion does not need to be recomputed every time), it might be worthwhile to increase the number of admissible structures. If this is not the case, then the set has to be restricted and/or the selection criterion simplified.

There is a need for further studies concerning some of the selection criteria used in feature selection algorithms. In particular those that are directly or indirectly related to the error rate. What is known about some properties of some criteria (e.g. bias), is not valid when these criteria are used in an optimization procedure.

CHAPTER 4

INCOMPLETE DATA SETS

In this chapter, the main topic is incomplete data sets. This subject is analyzed and some methods for estimating missing values are presented. Furthermore, the results of a simulation experiment are presented.

4.1. Introduction.

The design and use of statistical classifiers is sometimes rendered difficult by the occurrence of objects with missing data values. That is, the description of some objects may not be complete, due to the fact that some of the measurements required were not performed or recorded. In other words, some objects may have missing values.

There may be various reasons for the occurrence of missing values in a data set. A few examples are:

- Human forgetfulness to take or record the values.

- Sudden malfunction of the equipment when the value of a feature is to be measured.

- Deliberate omission (e.g. a patient may refuse to answer a question).

- The feeling that the most important data is already present, so that the rest of the data may be omitted.

- Urgency to make a decision implying that time consuming measurements have to be disregarded.

If for some causes remedies can and should be found, it is clear that this may not be the case for others. Thus, the occurrence of missing values may have to be taken into account, both in the design of a classifier, which implies estimation from an incomplete training set, as well as when the classifier is applied in practice. Throughout this chapter, it will be assumed that missing values occur at random.

Sections 4.2 and 4.3 are devoted to an analysis of methods that have been proposed in order to cope with this problem, starting with the case of a classifier applied in practice. Section 4.4 presents some methods of estimating missing values and in section 4.5 an experiment is described involving the design of classifiers with sets of objects that are incomplete. Finally, section 4.6 contains concluding remarks.

4.2. Classification of an object with missing values.

Three basic methods have been proposed for the classification of incomplete objects. These are indicated with the terms: 'subspace classification', 'estimate and replace' and 'delete'.

In the first method, classification is done in the subspace defined by the features that are known for the object under consideration.

The second method estimates values for the features missing and then proceeds to classification as if the object were complete.

The 'delete' method simply amounts to the refusal to classify.

From a theoretical point of view, 'subspace classification' is the best approach. In the first place, Sebestyen (SEBE62) has proved that when using an optimal classifier, no useful purpose can be achieved by estimating and replacing missing values, as compared to making the decision in the subspace defined by those features with no missing values. Therefore, at the theoretical level, the first method always yields results that are equal or better than those yielded by the second method. Secondly, a refusal to classify has to be considered as an error and therefore, 'subspace classification' is also theoretically better than the 'delete' method.

In a practical application, 'subspace classification' may cause problems if Bayes classifiers are used and continuous features are involved. Complex integrations are then required which may prove to be too time consuming to perform (Monte Carlo methods may be needed). Solutions for this problem have been proposed that may be divided into two groups:

- Brute force methods that store decision functions for all possible combinations of missing values (see HAND74).

- Use of decision functions that avoid complex integrations either by omitting terms (see HAND76) or can be easily modified to cope with missing values (see KITT78).

Both groups have drawbacks. For the first group, it is sufficient to point out that if p features define the complete space, there are $2^p - 1$ different ways in which one or more missing values may occur. If p is large, this leads to an unreasonably large number of decision functions to be estimated and saved. As for the second, the point has to be raised whether the limitations imposed on the domain of the decision functions (so that integrations can be avoided), do or do not lead to a classifier that is far from optimal.

In discrete feature spaces, the practical problem just mentioned does not arise since an integration in a continuous space is a summation in a discrete feature space.

From a theoretical point of view, the 'estimate and replace' technique is equal or worse than the 'subspace classification' (as indicated above) but equal or better than the 'delete' technique. Given an object with missing values for classification, the 'delete' technique always yields an error whereas the use of the 'estimate and replace' method may or may not cause an error.

The problems associated with the 'estimate and replace' approach are directly related to the complexity required for the estimators of the missing values. If in some cases a very simple technique (e.g. use as an estimate the estimated mean value) may not seriously damage the performance, in other situations more sophisticated procedures may be required (e.g. an estimate is obtained as a linear combination of the values of the known features). In such cases, a considerable number of different estimators may be required (e.g. one for every combination of missing features) and features not needed for classification may have to be measured in order to estimate missing values.

The major advantage of the 'delete' technique is its speed and simplicity. It is particularly attractive if the probability of occurrence of objects with missing values is low, since the effects on the classifier error will not be significant. However, situations may arise where the method cannot be applied at all (e.g. a patient in a critical condition requiring an urgent diagnosis on part of the data normally available).

These are some of the advantages and drawbacks of the three methods mentioned above. Depending on the practical circumstances of the problem at hand, either just one or combinations of the three, may prove to be the best.

4.3. Design of a classifier using an incomplete data set.

The design of a classifier on the basis of a set of labeled objects that contains missing values, presents problems that are conceptually different from the case introduced in the previous section. In the following, the various techniques proposed by various authors are presented and discussed.

When the training set contains missing values, the simplest approach is to ignore objects and/or features that are incomplete. This is the 'delete' technique, analogous to the one mentioned in section 4.2. The advantage of this method is its inherent simplicity, its major drawback being the loss of information caused by the deletion of objects and/or features. This may be significant if objects and/or features have a small percentage of missing values. On the other hand, if a given object (or feature) has a large percentage of missing values, the loss of information that results from excluding it from further analysis is most likely not significant.

A second method is the 'estimate and replace'. In spite of the difficulties that may be associated with the estimation, the method is also inherently simple since it provides data sets that may subsequently be manipulated as if they were complete, making them suitable to be analyzed by any available software package that assumes the data to be complete. No information is lost but the question has to be raised how the errors in the estimates affect the design of a classifier. A definite answer cannot be provided but nevertheless, two aspects can be mentioned suggesting that these errors may not be significant:

1 - It is not so uncommon to have data sets with a large number of features, some of them correlated. This is particularly relevant with medical data bases.

2 - Objects tend to cluster in terms of their classes. Otherwise, good results can not be expected from any classifier.

If both conditions are met, it can be expected that a reasonable missing value estimator will provide good estimates. Nevertheless, if an object (or feature) has a large number of missing values, any attempt at estimation is bound to lead to poor results.

For the case of discrete feature spaces, a third technique assumes that missing values are a new possible value for the feature in which they occur. This method has appealing aspects such as its simplicity and the fact that, at least technically, no information is lost nor is the data corrupted. On the other hand, the increase in the domain of a feature implies that more parameters need to be estimated. Moreover, if a feature is ordered, the order relation is lost with the introduction of the new values. Also, there may be strong reasons (e.g. the reduction in the number of features) to suggest that the occurrence of missing values may be controlled and consequently the addition of a 'missing' category bears no meaning.

A fourth technique is to treat the data as it is, i.e. using those values that are known and ignoring those that are not. This is only practical in some restricted cases because of computational problems among others.

The first two techniques have been further analyzed and applied. In section 4.5 simulation experiments are presented that 'quantify' their relative merits in two specific discrimination problems. In chapter 6, they are jointly applied in two practical cases. This combination of 'delete' and 'estimate and replace' is appealing in view of the following reasons. If a given object (or feature) has a large number of missing values, the loss of information that results from its exclusion from further analysis is not significant, whereas the estimation of a missing value will most probably result in a significant error. Hence the 'delete' technique is a better choice. On the other hand, if a given object (or feature) has a small number of missing values, then the loss of information is significant if the 'delete' technique is applied whereas good estimates may be expected for the values missing. Hence the 'estimate and replace' technique is a better choice. Therefore, these two techniques combined together can better handle cases where there are both a large and a small number of missing values in the data.

4.4. Estimation of missing values.

In this section, some methods for the estimation of missing values, required by the 'estimate and replace' technique, are analyzed. Some modifications for existing methods are introduced and a new approach is proposed for the estimation of missing values in the case of discrete feature spaces.

4.4.1. Continuous feature spaces.

For continuous feature spaces, two methods will be described here:

1 - The first method uses an iterative procedure. Assuming a set of n objects in a p dimensional space and supposing that m objects have no values recorded on the first q features, then:

Step 1 - For the first q features compute the mean values using the complete objects and assign them to the missing values.

Step 2 - Compute the mean vector and covariance matrix using both complete and completed objects. Proceed to a multivariate regression of the first q variables on the last p-q variables.

Step 3 - Using the regression coefficients and the mean values estimated in the previous step, obtain new estimates for the values originally missing.

Step 4 - If there is no appreciable change or if the maximum number of iterations allowed is reached, stop and use the estimates obtained in step 3 of the current iteration. Otherwise goto step 2.

The assumptions made above do not represent a loss of generality; the algorithm may just have to be repeated for other combinations of missing values.

This procedure is the one reported in BEAL75.

2 - The second method uses a nearest neighbours approach, using a Euclidian distance metric. As with all procedures based on a Euclidian metric, the data set is first scaled, such that the standard deviations for all features are equal. The objects are then processed sequentially. If an object (object i) with one or more missing values is encountered, the following procedure is activated for each unspecified feature f separately:

> Denoting the subspace spanned by the known features by S and the subspace spanned by the known features plus feature f by S^+, a search in S is made for all objects completely defined in S^+. Among these objects, the k objects nearest to object i are identified and the missing value of object i is replaced by the average value of f calculated from the k nearest neighbours.

This algorithm is a modified version of a method originally introduced by Dixon (see DIXO79). The modification, introduced in this thesis, consists in the way the set of candidate nearest neighbours is formed. In Dixon's case, this set is composed of all objects that do not have a missing value in the feature under consideration. Hence a heuristic metric was required in order to cope with objects that have arbitrary patterns of missing values. This modified version avoids the use of the heuristic metric. Also, all information available is used in a way that agrees with Sebestyen conclusions (SEBE62).

4.4.2. Discrete feature spaces.

In order to estimate missing values in discrete feature spaces, the following new technique is proposed.

Let the vector $(x_1,x_2,...,x_p)$ represent an object with known components $x_2, ...,x_p$, and for which the first component x_1 is missing and may take one of I_1 possible values. The proposed technique selects as an estimate for x_1 the value x_{k1} for which

$$P\ (x_{j1}|x_2,...,x_p) \leq P\ (x_{k1}|x_2,...,x_p) \tag{4.1}$$

for all $1 \leq j \leq I_1$.

The justification for this approach is as follows. The estimation of a missing value of a discrete feature is equivalent to the assignment of a class label to an object. In both cases, a discrete variable representing either a feature or class membership, is assigned a value on the basis of other variables. Thus, in discrete feature spaces, the estimation of missing values may be treated as a classification problem. From general pattern recognition theory, it is known that if x_1 is selected according to rule 4.1, the average probability of error is minimized. Therefore, the proposed technique is optimal.

Rule 4.1 is equivalent to the following rule:

Choose x_{k1} such that

$$P\ (x_{j1},x_2,...,x_p) \leq P\ (x_{k1},x_2,...,x_p) \tag{4.2}$$

for all $1 \leq j \leq I_1$.

If more than one value is missing in an object, e.g. when features 1, ..., q with I_1, ..., I_q possible values are missing, rules 4.1 and 4.2 take the form:

Choose from all I_1*...*I_q possible combinations for x_1, ..., x_q that combination x_{k1}, ..., x_{sq} such that:

$$P\ (x_{j1},...,x_{rq}|x_{q+1},...,x_p) \leq P\ (x_{k1},...,x_{sq}|x_{q+1},...,x_p) \tag{4.3}$$

for all $1 \leq j \leq I_1$, ..., $1 \leq r \leq I_q$, or equivalently:

$$P\ (x_{j1},...,x_{rq},x_{q+1},...,x_p) \leq P\ (x_{k1},...,x_{sq},x_{q+1},...,x_p) \tag{4.4}$$

for all $1 \leq j \leq I_1$, ..., $1 \leq r \leq I_q$. Rules 4.3 and 4.4 amount to the classification into one of I_1*...*I_q classes and are also optimal.

The probabilities in rules 4.1, 4.2, 4.3 and 4.4, are generally not known and have to be estimated using a limited number of objects (all or part of the objects available in the data set). In particular for rule 4.3, the following assumption of independence may have to be made in order to obtain reliable estimates.

$$P\ (x_1,\ldots,x_q | x_{q+1},\ldots,x_p) = \prod_{k=1}^{q} P\ (x_k | x_{q+1},\ldots,x_p) \tag{4.5}$$

The estimation of $x_1,\ldots,x_q$ is therefore decomposed in q independent classification problems, thereby simplifying the estimation.

So far the discussion has been general, showing the feasibility and optimal properties of the 'classifier-estimator' approach. In the following, two specific algorithms are introduced.

> Algorithm 1: The corresponding 'classifier-estimator' follows a Bayes approach which assumes that features without missing values are independent given a feature where a missing value occurs. The assignment of an estimate to a missing value is then done according to rule 4.1. In the case of a tie, the most represented value among all the objects in the data set is selected. If an object has more than one missing value, each related feature is treated independently. For each missing value, the 'classifier-estimator' is estimated using all the objects in the data set that do not have a missing value in the feature under consideration and which have known values in the subspace defined by the features known in the object under consideration.
>
> Algorithm 2: The corresponding 'classifier-estimator' is a nearest neighbour classifier that uses the following metric: the distance between two objects is equal to the number of features in which they differ. The estimate for the value missing is then the most represented value among the nearest neighbours. In the case of a tie, the most represented value among all objects in the data set is selected. If an object has more than one missing value, each related feature is treated independently. The search for the nearest neighbours is done among all the objects in the data set that do not have a missing value in the feature under consideration and which have known values in the subspace defined by the features known in the object under consideration.

Algorithm 1 corresponds to a parametric design whereas algorithm 2 corresponds to a non-parametric design. Since the number of objects generally available may be small and the number of features may be large, it was decided to use a simplified model for the 'classifier-estimator'. Both algorithms apply Sebestyen conclusions (SEBE62): the classification is done within the subspace defined by the features known in the object under consideration.

4.5. Experiments.

This section presents a set of experiments that simulate a conventional pattern recognition design problem, involving both the design of a classifier and the estimation of its error (using the hold-out estimator). They were performed in order to quantitatively assess the effects of missing values and to evaluate some of the techniques which have been proposed to cope with them. The techniques analyzed were the 'delete' and versions of the 'replace and estimate' using the estimators described in the previous section. Both a discrete and a continuous feature spaces were considered.

4.5.1. Setup.

In all simulations a two-class discrimination problem in a four dimensional space was considered. The a-priori probabilities of the classes were assumed to be known and equal.

From each of the two distributions, 50 objects were sampled at random. These objects were divided in a training and test set of equal sizes. A classifier was then designed using the training set and evaluated using the test set.

Having thus defined and evaluated a classifier for the complete data set, missing values were then inserted in the objects, using a pseudo random number generator for the selection of the position of missing values in the data matrix. The constraint that every object should have at least two features with known values was imposed on the process.

Two incomplete data sets were thus generated, containing 15% and 25% of missing values, respectively. The incomplete data sets were then processed, using five different approaches:

DEL - Ignore all objects with missing values in the training and in the test sets.

PAR1 - Estimate and replace missing values treating the training and the test sets separately. For the continuous case, the regression approach technique was used; in the discrete data set, the missing values were estimated using the Bayes classifier described above; in both cases, the objects were pooled from the two classes.

PAR2 - Same as PAR1 but treating the classes in the training set separately.

NN1 - Estimate and replace missing values using the nearest neighbours methods in the continuous and in the discrete spaces. In both cases, the objects from the two classes were pooled, and the training and test sets were treated separately.

NN2 - Same as in NN1, but treating the classes in the training set separately.

Thus, besides the 'delete' method, four different versions of the 'estimate and replace' method were defined, differing in the estimator adopted and in whether or not classes in the training set were pooled. The training and test sets were always treated separately so that the independence between these sets was not compromised.

The data sets thus obtained were then treated as complete sets and classifiers were designed and evaluated as for the complete data set, as described above.

The overall experimental set up was as follows:

1 - Generate a complete data set (50 times) from the conditional distributions. For each complete data set design and evaluate a classifier.

2 - For each of the 50 data sets, generate the incomplete data sets both with 15% and with 25% of missing values.

3 - For each of the incomplete data sets, reconstruct a 'completed' set according to the 5 methods described above and design and evaluate a classifier.

The performances of the classifiers designed with the complete data sets and with the five versions of the completed data sets, were evaluated on the basis of the mean true error. This was estimated as follows. Each time the experiment was repeated, the true errors of the classifiers were either computed exactly (discrete features) or estimated from a sample containing 1500 objects (continuous features). The mean true error was then estimated by taking averages over the 50 times that the experiment was repeated.

Wilcoxon's two-sided sign rank test (see, e.g. SIEG76) was then applied to the estimated mean true errors in order to test if the differences between them were significant or due to chance.

The effects of missing values on the estimation of the error rate (test set error) was evaluated by computing averages of error estimates over the 50 times that the experiment was repeated. Standard deviations were also estimated.

4.5.2. Continuous features.

The generating distributions for the two classes were assumed to be Gaussian, with mean vectors

$$\underline{\mu}_1 = [\ 0.5,\ 1.0,\ 1.5,\ 2.0]^T$$

$$\underline{\mu}_2 = [\ 0.6,\ 1.2,\ 1.8,\ 2.4]^T \qquad (4.6)$$

and with a common covariance matrix:

$$\Sigma = \Sigma_1 = \Sigma_2 = \begin{matrix} .004 & .002 & .002 & .002 \\ .002 & .004 & .002 & .002 \\ .002 & .002 & .004 & .002 \\ .002 & .002 & .002 & .004 \end{matrix} \qquad (4.7)$$

For each data set, maximum likelihood estimates $\underline{\mu}\hat{}_1$, $\underline{\mu}\hat{}_2$ and $\Sigma\hat{}$ were obtained, pooling the objects from both classes in the computation of $\Sigma\hat{}$. The following decision rule was then applied:

Assign object $\underline{x}$ to class 1 if:

$$(\underline{x}-\underline{\mu}\hat{}_1)^T\ \Sigma\hat{}^{-1}\ (\underline{x}-\underline{\mu}\hat{}_1) < (\underline{x}-\underline{\mu}\hat{}_2)^T\ \Sigma\hat{}^{-1}\ (\underline{x}-\underline{\mu}\hat{}_2)$$

and to class 2 otherwise.

(4.8)

This is the Bayes optimal decision rule for normally distributed features when the classes have the same covariance matrix and when the a-priori probabilities are equal.

The results of the experiments are presented in tables 4.1 (15% of missing values) and 4.2 (25% of missing values).

The tables are organized as follows. The fist column applies to the complete data set, the last columns refer to the five different versions of the completed sets. In the first three lines, the values of the mean estimated error, the corresponding standard deviation and the mean true error are given. The lower part of the tables contain the matrix of Wilcoxon's two-sided rank test probabilities applied to the corresponding differences in the values of the mean true error. Since this matrix is by definition symmetric, only the lower left part is given.

	ALL	DEL	PAR1	PAR2	NN1	NN2
Mean est. err	.148	.176	.174	.184	.185	.193
S.d. est. err	.045	.067	.044	.047	.054	.055
Mean true err	.145	.161	.150	.155	.148	.148
ALL	-	-	-	-	-	-
DEL	.5e-5	-	-	-	-	-
PAR1	.5e-2	.2e-3	-	-	-	-
PAR2	.3e-4	.1	.2e-3	-	-	-
NN1	.108	.5e-3	.264	.3e-3	-	-
NN2	.7e-1	.7e-4	.306	.7e-5	.630	-

Table 4.1: Results of the simulation experiment described in subsection 4.5.2 (continuous feature space). The incomplete data sets contained 15% of missing values.

	ALL	DEL	PAR1	PAR2	NN1	NN2
Mean est. err	.148	.180	.199	.210	.201	.198
S.d. est. err	.045	.109	.060	.064	.058	.051
Mean true err	.145	.186	.154	.170	.152	.151
ALL	-	-	-	-	-	-
DEL	<.1e-6	-	-	-	-	-
PAR1	.2e-1	.2e-5	-	-	-	-
PAR2	.9e-6	.1e-1	.2e-3	-	-	-
NN1	.3e-1	.1e-5	.874	.1e-2	-	-
NN2	.3e-1	<.1e-6	.418	.6e-4	.585	-

Table 4.2: Results of the simulation experiment described in subsection 4.5.2 (continuous feature space). The incomplete data sets contained 25% of missing values.

As expected, the best classifier was obtained with the complete data set. From the Wilcoxon's probability matrix it may be seen that the differences with the performances of the classifiers based on the completed data sets were in most cases significant.

Nevertheless, the 'estimate and replace' methods generally yielded classifiers with mean true errors closer to the one for the complete set. The 'delete' approach always yielded the lowest classifier performance.

The pooling of classes did not make a significant difference in the case of the nearest neighbour estimation. This may be explained as follows. Objects clustered in terms of the two classes, also in the subspaces where the selection of the nearest neighbours was done. Therefore, with the nearest neighbours estimation and with the classes pooled, the nearest neighbours selected also tended to be objects from the same class as the object under consideration. Hence, even with the classes pooled it can be said that class information was implicitly used.

With the regression technique, the pooling of classes did make a difference. Paradoxically, the best results were obtained with the classes pooled. The paradox stems from the fact that better results were expected with the classes not pooled, since estimates are obtained based on a mean vector and a covariance matrix and therefore, with the classes pooled only global information is used. The behaviour observed may be related to the estimation of the covariance matrices required throughout the regression, and the number of objects used (larger with the classes pooled) in its estimation.

Comparing the mean true errors in tables 4.1 and 4.2, it can be concluded, as expected, that the performance of a classifier deteriorates with an increase in the number of missing values.

The error estimates as obtained by the hold-out technique, had large standard deviations. This may be explained both by the small number of objects used and the relatively large true errors of the classifiers. The combination of these two factors was particularly important and damaging for the 'delete' technique.

The mean estimated errors were generally pessimistic when compared to the mean actual errors. This may be explained by the following reasons. Firstly, the hold-out estimator is known to be pessimistically biased. Secondly, the mean true errors were computed without considering the possibility of occurrence of missing values (the true error is larger or equal when missing values occur) whereas in this simulation test sets were used that contained missing values.

4.5.3. Discrete features.

For the discrete case, binary features were considered with class definitions such that an optimal classification error of 0.1 could be obtained for the full dimensionality. The model for the classes was generated in the same way as the low structured space (space L) described in chapter 3. Classifiers were designed assuming full multinomial models for the probability of an object given its class. Maximum likelihood was used to estimate the required parameters.

The results of this set of experiments are presented in tables 4.3 (15% of missing values) and 4.4 (25% of missing values). These tables are organized in a way similar to tables 4.1 and 4.2.

The best classifiers were obtained when all objects were complete, the differences being generally significant according to Wilcoxon's test.

The 'delete' technique generally yielded classifiers with the highest mean true error with two exceptions, both when there were 15% of missing values and an 'estimate and replace' technique was used with the data pooled over the two classes.

Among the versions of the 'estimate and replace', the nearest neighbours estimators performed worse than their Bayes counterparts. This may be explained by the poor metric adopted for the nearest neighbours estimator.

The pooling of classes always made a significant difference according to Wilcoxon's test: better results were obtained with the classes not pooled for both classifier-estimators. In what concerns the nearest neighbours estimator, this behaviour may be related to the poor metric adopted and the low structure of the feature space considered. As for the Bayes classifier, both the low structure of the space and the fact that global information is used in the estimation when classes are pooled, may explain the differences found.

The overall best method to deal with missing values was the version that uses a Bayes classifier and applies it separately per class (the results of Wilcoxon's test suggest that the differences found between this and the other methods, were significant).

Comparing the mean true errors in tables 4.3 and 4.4, it can be concluded that the performance of a classifier deteriorates with an increase in the number of missing values.

The standard deviations and mean estimated errors showed a behaviour similar to the case of the continuous features. The discussion presented above also applies here.

	ALL	DEL	PAR1	PAR2	NN1	NN2
Mean est. err	.143	.191	.288	.259	.288	.266
S.d. est. err	.076	.097	.087	.089	.090	.087
Mean true err	.143	.196	.201	.172	.213	.187
ALL	-	-	-	-	-	-
DEL	.1e-5	-	-	-	-	-
PAR1	.3e-5	.485	-	-	-	-
PAR2	.1e-2	.2e-2	.2e-3	-	-	-
NN1	.7e-6	.5e-1	.227	.1e-4	-	-
NN2	.4e-4	.296	.6e-1	.6e-1	.4e-2	-

Table 4.3: Results of the simulation experiment described in subsection 4.5.3 (discrete feature space). The incomplete data sets contained 15% of missing values.

	ALL	DEL	PAR1	PAR2	NN1	NN2
Mean est. err	.143	.310	.358	.340	.378	.356
S.d. est. err	.076	.132	.066	.076	.073	.084
Mean true err	.143	.295	.271	.216	.291	.248
ALL	-	-	-	-	-	-
DEL	<.1e-6	-	-	-	-	-
PAR1	<.1e-6	.6e-1	-	-	-	-
PAR2	<.1e-6	.5e-6	.8e-5	-	-	-
NN1	<.1e-6	.861	.137	.1e-5	-	-
NN2	<.1e-6	.4e-3	.2e-1	.5e-2	.2e-2	-

Table 4.4: Results of the simulation experiment described in subsection 4.5.3 (discrete feature space). The incomplete data sets contained 25% of missing values.

4.6. Concluding remarks.

An analysis has been presented of methods to cope with the occurrence of missing values in the design and application of statistical classifiers. Several techniques were introduced for the estimation of missing values which were used in an experiment that compared the 'estimate and replace' and the 'delete' methods, in the design of classifiers. A new approach has been presented for the estimation of missing values in discrete feature spaces.

In terms of the performance of classifiers, the simulation study has shown that missing values are a nuisance and that every effort should be directed at their elimination. Both in terms of classifiers and of error estimates, the performance decreased with an increase in the percentage of missing values.

The simulation study has also shown that the 'estimate and replace' techniques were able to cope with the occurrence of missing values showing that the errors in the estimation may be manageable as compared to an outright disregard of incomplete objects and/or features. Nevertheless, if a data set has objects (or features) with large percentages of missing values and objects (or features) with small percentages of missing values, a combination of 'delete' and 'estimate and replace' is appealing as explained in section 4.3.

If a selection has to be made among the 'estimate and replace' versions used, the results of the simulation presented in the previous section may be used in spite of the drawback that it was a limited set of experiments. For the discrete feature space, the Bayes classifier applied without pooling the classes was the best approach and therefore is the one selected. For continuous features, the selection is the nearest neighbour approach with the classes not pooled. The results of the simulation are not sufficiently clear in order to justify this selection. Two other reasons should be given. Firstly, the way the regression and the nearest neighbours algorithms were implemented, allows the information available to be used more efficiently by the nearest neighbours estimator. Secondly, class membership information has to be used in situations like the one presented in fig. 4.1 (showing objects from two classes in a two dimensional space).

Finally, given an incomplete data set, it may appear attractive to concentrate incomplete objects either in the training set or in the test set. This approach is to be avoided since either the selection of classifier or the estimation of its error is compromised, and the rule of random partitioning is broken.

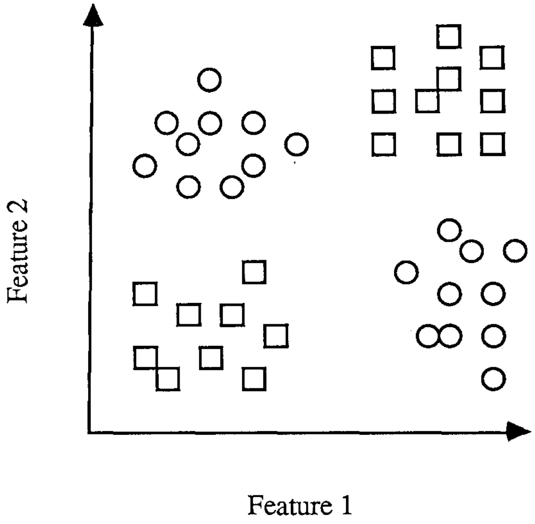


Fig. 4.1: Objects from two classes in a two dimensional space.

CHAPTER 5

REVIEW OF TECHNIQUES

In this chapter, procedures and techniques for pattern recognition with discrete and/or mixed data types, are reviewed and discussed. The subjects discussed include classifiers as well as procedures normally required in the design of a pattern recognition system, such as feature selection, error estimation and mapping techniques.

5.1. Introduction.

The methods described hereafter are applicable to situations where one or more of the features used in discrimination are of a discrete nature. A discrete feature is one that assumes only a finite number of values.

The aims this chapter tries to achieve, are, on the one hand, to provide a review and compilation of methods, and, on the other hand, to present the procedures applied in the analysis of two medical data sets. The results of this analysis are presented in chapter 6 of this thesis. This review is by no means complete. Further details will have to be found in the literature duly referenced.

The organization of this chapter is as follows. Section 5.2 deals with classifiers. Some of these can only be applied with discrete or discretized features, whereas others can cope with mixed data types (i.e. mixtures of discrete and continuous features). Section 5.3 is dedicated to the presentation of feature selection procedures. It is, in a way, an extension of the material of chapter 3 with an emphasis on the selection criteria. Section 5.4 presents a technique for mapping, i.e. correspondence analysis. Section 5.5 treats error estimation. Finally, section 5.6 presents concluding remarks.

5.2. Classifiers.

Assuming a Bayes framework, the selection of a classifier is translated into the selection of a suitable model (or models) to represent the underlying probabilities of interest [1], and the subsequent estimation of the parameters that define the model. Therefore, in a Bayes framework, in this review models are discussed that have been proposed in the literature for discrete and/or mixed feature spaces. As it will become clear, Bayes classifiers will occupy us mostly. Nevertheless, other types will be duly referenced.

5.2.1. Basic models.

An object described by p discrete random variables x_j (variable x_j assuming I_j values), can assume one of $I_1*I_2*... *I_p$ distinct realizations. Assuming two classes, w_1 and w_2, a Bayes classifier is defined as:

$$\begin{array}{ll} \text{Choose } w_1 & \text{if } P(w_1)\, P(x_1,...,x_p|w_1) \geq P(w_2)\, P(x_1,...,x_p|w_2) \\ \text{Choose } w_2 & \text{otherwise.} \end{array} \qquad (5.1)$$

A simple model for the conditional probabilities $P(x_1,...,x_p|w_i)$, is the full multinomial. Under this model, the probability of an element of the feature space (an element is one of the possible realizations an object can assume) given its class, is assumed to be independent of the probabilities of all the other elements in the feature space, the only restriction being that their sum be equal to 1. In other words, $Prob(x_1,...,x_p|w_i)$ is a function of $I_1*I_2*...*I_p$-1 independent variables (where each variable is the probability of occurrence of an element). This model is called a non-structured model.

With a classifier based on such model, all possible interactions between variables, whether or not relevant for classification, are accounted for. However, such a classifier requires the estimation of $I_1*I_2*...*I_p$-1 parameters.

The large number of parameters that is required, and the resulting estimation errors leading to either large classifier design errors or large training sets if those errors are to be contained, suggest the assumption of relations of independence between sets of features or conditional independence given combinations of features. This reduces the number of parameters to be estimated. Introducing this kind of assumptions is equivalent to giving more structure to the model. The 'most' structured model is the one that assumes all features to be independent given the class:

[1] By probabilities of interest it is meant either the a-posteriori probability of a class given a set of measurements, or the joint probability of class and object, or finally, the conditional probability of an object, given its class.

$$\mathrm{Prob}(x_1, ..., x_p|w_1) = \mathrm{Prob}(x_1|w_1)*...*\mathrm{Prob}(x_p|w_1) \tag{5.2}$$

for class w_1, and similarly for class w_2. It can be easily seen, that in this case, a total of $(I_1\text{-}1)+...+(I_p\text{-}1)$ parameters needs to be estimated.

Between the complete independence and the full multinomial, there is a whole range of structured models, corresponding to independence between sets of features and/or conditional independence given certain combinations of features. There are also cases that cannot be expressed in terms of relations of independence and which will be more conveniently introduced as a loglinear model (see next subsection).

The large number of models that can be constructed assuming different relations of independence, provide a largely flexible way of constructing decision functions. This ability together with readily obtained estimates (both maximum likelihood and Bayesian estimates were already introduced in chapter 3), makes the basic models attractive.

However, they can only be applied to discrete or discretized feature spaces. The latter develop if some of the features are continuous variables, in which case their range is divided into a finite number of intervals and the probabilities in each interval estimated, implying a loss of information. Moreover, order relations in ordered discrete features or discretized continuous features are lost. Order relations are important because they can be seen as constraints, with direct implications on the estimation of parameters.

So far, only conditional probabilities have been considered. However, this set of models can also be applied to joint probabilities of class and object, provided that class membership is not assumed independent of any feature (the assumption of independence between the variable expressing class membership and a feature is equivalent to the assumption that the feature has no discriminatory power and therefore is useless).

5.2.2. Loglinear models.

Instead of probabilities, their logarithms can also be considered. Following Bishop (BISH75), the logarithm can be decomposed as:

$$\ln P(x_1, ..., x_p|w_i) = \alpha_0(w_i) + \sum_{j=1}^{p} \alpha_{x_j}^{j}(w_i) +$$

$$+ \sum_{j<k}^{j,k} \alpha_{xj,xk}(w_i) + \ldots + \alpha^{1,..,p}_{x1,\ldots,xp}(w_i)$$

(5.3)

where the various α's satisfy the following relations:

$$\sum_{x_j=1}^{I_j} \alpha^{j}_{xj}(w_i) = \sum_{x_j=1}^{I_j} \alpha^{j,k}_{xj,xk}(w_i) = \sum_{x_k=1}^{I_k} \alpha^{j,k}_{xj,xk}(w_i) = \ldots = 0$$

(5.4)

and represent the effects of the various features and interactions between features on the logarithm of the probability. The superscripts denote the features involved and the subscripts denote the values assumed by the features involved. The dependence on w_i indicates dependence on the class. The term α_0 is called the overall mean, the terms α^{j}_{xj} are called the main effects, the terms $\alpha^{j,k}_{xj,xk}$ the first order effects (interactions), the terms $\alpha^{j,k,l}_{xj,xk,xl}$ the second order effects (interactions), etc.. This representation is called the log-linear model. If all possible α's are assumed different from zero, then the model is called saturated (BISH75) and it is equivalent to the full multinomial.

A Bayes classifier can be easily derived (two classes, w_1 and w_2, are assumed).

Decide for w_1 if

$$\ln (P(w_1)) - \ln (P(w_2)) + \alpha_0(w_1) - \alpha_0(w_2) + \sum_{j=1}^{p} (\alpha^{j}_{xj}(w_1) - \alpha^{j}_{xj}(w_2)) +$$

$$\sum_{j<k} (\alpha^{j,k}_{xj,xk}(w_1) - \alpha^{j,k}_{xj,xk}(w_2)) + \ldots + \alpha^{1,\ldots,p}_{x1,\ldots,xp}(w_1) - \alpha^{1,\ldots,p}_{x1,\ldots,xp}(w_2) \geq 0$$

Decide for class w_2 otherwise.

(5.5)

Structured models can also be expressed by the so called (BISH75) non-saturated hierarchical models [2] in which some of the α's are assumed to be equal to zero. For example,

$$\ln P(x_1, \ldots, x_p | w_i) = \alpha_0(w_i) + \sum_{j=1}^{p} \alpha_{xj}^{j}(w_i) \tag{5.6}$$

In this case, only the main effects are non-zero. This model corresponds to the case of independence of the features given the class. The corresponding classifier is easily established.

There are cases where a log-linear model cannot be interpreted in terms of independence (this also applies to the basic model). For instance, in a space defined by 3 features in which the second order interaction term is assumed to be zero, a given probability cannot be expressed in terms of the product of the marginals probabilities (BISH75 sets the conditions where these cases occur; further back, BIRC63 also discusses these points). In such cases, the maximum likelihood estimates have to be obtained by a special iterative algorithm known as the Deming-Stephen algorithm (see BISH75 and FINB70). This algorithm is able to obtain maximum likelihood estimates in the first iteration when the probabilities can be expressed in terms of the products of the marginal probabilities.

Summarizing, for every basic model there is a corresponding log-linear model. This is a one to one correspondence. The differences between the standardized coefficients α for the different classes, are directly related to the discriminating power.

Finally, if a suitable modification is introduced in the model, order relations can be taken into account (see EVER77). In this case, the one to one correspondence mentioned above is broken. As with the basic models, continuous feature must be discretized before the model can be applied. However, if the model handles order relations, implicit in continuous features, then at least this aspect is kept.

[2] A model is hierarchical if the following rule applies: given an α with superscript S that is assumed to be different from zero, every α for which the superscript is contained in S is also assumed to be non-zero.

5.2.3. Logistic model.

Somewhat unusual in a Bayes probabilistic context, the logistic model is applied directly to the a-posteriori probabilities of a class given an object. This model has been thoroughly studied by Anderson (see ANDE72, ANDE74, ANDE79, ANDE82). An earlier reference is DAY67. More recently Lesaffre has also analyzed several aspects of the model (LESA86).

Assuming 2 classes the logistic form of the posteriori probabilities is (see ANDE72)

$$\text{Prob}\ (w_1|\underline{x}) = \exp\ (\beta_0 + \underline{\beta}^T\underline{x})\ /\ (1 + \exp\ (\beta_0 + \underline{\beta}^T\underline{x}))$$

$$\text{Prob}\ (w_2|\underline{x}) = 1\ /\ (1 + \exp\ (\beta_0 + \underline{\beta}^T\underline{x}) \tag{5.7}$$

Since a-posteriori class probabilities are used, the decision function is immediate and involves only the computation of a linear function.

In a k class problem, the logistic form for the a-posteriori probabilities are

$$\text{Prob}\ (w_i|\underline{x}) = \exp\ (\beta_0(w_i) + \underline{\beta}^T(w_i)\underline{x})\ /\ (1 + \sum_{s=1}^{k-1} \exp\ (\beta_0(w_s) + \underline{\beta}^T(w_s)\underline{x}))$$

for $1 \leq i \leq k - 1$, and

$$\text{Prob}\ (w_k|\underline{x}) = 1\ /\ (1 + \sum_{s=1}^{k-1} \exp\ (\beta_0(w_s) + \underline{\beta}^T(w_s)\underline{x})) \tag{5.8}$$

In order to fully explain the meaning of these expressions, both the meaning of $\underline{x}$ as well as that of the $\underline{\beta}$, have to be analyzed.

Following the usual notation, the symbol $\underline{x}$ indicates a vector representing an object, whereas the symbol $\underline{\beta}$ is a vector indicating the coefficients (parameters) of the distribution. Let the features be binary and coded as either 0 or 1. According to the formulation of the logistic model presented in ANDE72, both $\underline{\beta}$ and $\underline{x}$ are p*1 vectors, where p is the number of features. Each element in $\underline{x}$ represents a variable and each element in $\underline{\beta}$ is associated with a variable.

In order to cope with variables assuming more than two values, two situations may be distinguished:

- if the variable is an ordered variable, it may be represented in $\underline{x}$ and $\underline{\beta}$ by one element.

- if the variable is categorical, in order to remove the ordering implied by the model, it must be replaced by I_j-1 dummy binary variables, where I_j is the number of different values the original variable may assume.

This clarifies the remark above about the effect of an order relationship: it reduces the number of parameters required by the model, resulting in more robust estimates.

The logistic model shown in expressions 5.7 and 5.8, includes distributions such as multivariate discrete distributions following a log-linear model with equal (not necessarily zero) interactions between variables. In other words, it is equivalent to a Bayes classifier where the basic model is applied assuming the features to be independent given the class.

The model can be further extended by introducing relations between variables. In the so-called quadratic logistic discrimination (see ANDE82), first order interactions between variables are introduced. In the model above and for binary discrete variables, this amounts to code a vector $\underline{x}$ with p (representing the main effects of the variables) plus p*(p+1)/2 elements (representing interactions between variables). That is

$$\underline{x}^T = (x_1,...,x_p,x_1*x_1,...,x_p*x_p,x_1*x_2,...,x_{p-1}*x_p) \tag{5.9}$$

where x_i is coded as one or zero. Correspondingly, $\underline{\beta}$ is a vector of the same dimension, where each element represents the contribution of a main effect or an interaction. In this case, the logistic model includes models such as the log-linear model with unequal main effect and first order interactions and equal (not necessarily zero) higher order interactions.

Clearly, more complex models involving higher order interactions may be similarly constructed. More complex models also involve more parameters to estimate. With respect to this, Lesaffre (LESA86) has shown the importance of deciding upon the optimal base class (that is class w_k above) since there may exist one such class, for which the number of parameters required is minimal.

The discussion so far has concentrated on discrete variables. The logistic model can also be applied with continuous variables. If all features are continuous, the logistic model may include distributions such as multivariate normal distributions with equal covariance matrices (in this case each feature is represented by one element in $\underline{x}$). The quadratic logistic model may also include multivariate normal distributions with unequal covariance matrices. Naturally, joint distributions of continuous and discrete variables following the distributions referenced above, can also be modeled in the logistic form.

Anderson (ANDE72) developed maximum likelihood estimators for the parameters of the logistic model, in case of mixture sampling (i.e. sampling from the universe of objects), separate sampling (i.e. sampling per class) and conditional sampling (i.e. sampling at an object). Questions arise when considering separate sampling with continuous variables (see ANDE82). In order to obtain maximum likelihood estimates, Anderson (ANDE72) proposed an iterative procedure based on the Newton-Raphson procedure (quasi-Newton procedures have also been proposed). Our own experience with the Newton-Raphson approach suggests that convergence is obtained in a few iteration steps. Nevertheless, there are cases where the algorithm does not converge, or, more precisely, where the maximum likelihood estimate occurs at infinity. Anderson identified these cases as being those of complete separation between the training objects from different classes (ANDE72), and those of zero marginal proportions (ANDE74).

Summarizing, logistic models appear to be quite powerful. They can handle continuous and discrete variables and are able to model a wide range of distributions. Specifically, for every logistic model, there is an equivalent log-linear model which includes all interactions and main effects included in the logistic model, plus all other implied lower order terms. Possibly, the major deficiencies of this model, arise in the estimation of its parameters. The iterative nature of the method proposed and two cases where the maxima are at infinity have already been observed. It should be added, that the use of the Newton-Raphson procedure requires the inversion of a ((k-1)*(p+1))*((k-1)*(p+1)) matrix [3], and this for the case where only main effects are taken into account.

5.2.4. Probit model.

Mathematically, the probit model is very similar to the logistic model. It also models directly a-posteriori probability distributions and the rationale for the model may be put as follows (see TALL75, ALBE81, ALBE81).

Consider a discrimination problem which involves two classes. Further, let y be a variable which takes values in a continuous interval, and is a function of $\underline{x}$ (y=f($\underline{x}$)). Each value of y may be thought of as a measurement in a scale, on which

[3] k is the number of classes and p is the number of features.

every object (represented by $\underline{x}$) in the feature space can be plotted. If the two classes represent two quantitatively but not qualitatively distinct groups (see ALBE81), a possible definition of these classes in terms of y, is

$$\text{Class } w_1 = \text{set of all } \underline{x} \text{ such that } y = f(\underline{x}) \leq z$$

$$\text{Class } w_2 = \text{set of all } \underline{x} \text{ such that } y = f(\underline{x}) > z \tag{5.10}$$

As an illustration, consider the following example. Let an object be a student in a given school. Each student is described by the marks obtained. The variable y represents the performance in some way expressed in terms of the marks. The definition of two classes, i.e. good and bad students, amounts to the selection of a threshold z on the scale of y.

The posteriori probabilities of a class given an object are:

$$\text{Prob}(w_1|\underline{x}) = \text{Prob}(y \leq z|\underline{x})$$

$$\text{Prob}(w_2|\underline{x}) = \text{Prob}(y > z|\underline{x}) \tag{5.11}$$

and the decision function follows immediately.

To complete the definition of the probit model, it remains to define the function f and distribution of y given $\underline{x}$. In the probit model, y is a linear combination of the elements of $\underline{x}$:

$$y = f(\underline{x}) = \alpha_0 + \alpha_1 x_1 + \ldots + \alpha_p x_p = \underline{\alpha}^T \underline{x}^1$$

where

$$\underline{x}^1 = [1, x_1, \ldots , x_p]^T \tag{5.12}$$

and the distribution of y given $\underline{x}$ is the normal distribution:

$$\text{Prob}(y \leq z|\underline{x}) = \text{Prob}(w_1|\underline{x}) = \phi\ (\underline{\alpha}^T \underline{x}^1\ ) \text{ and}$$

$$\text{Prob}(y > z|\underline{x}) = \text{Prob}(w_2|\underline{x}) = 1 - \phi\ (\underline{\alpha}^T \underline{x}^1\ ) \tag{5.13}$$

This model is applicable to both continuous and/or discrete features and maximum likelihood estimates have been obtained both for the cases of sampling from the mixture and separate sampling. In both cases, Newton-Raphson procedures are required.

Since (see ALBE81)

$$\exp(1.6*t) / (1 + \exp(1.6*t)) \tag{5.14}$$

is a good approximation to $\phi(t)$, the probit and logistic models lead to numerically closely related functions.

Finally, to our knowledge, there are no extensions yet of the probit model, to enable it to be applied in cases involving discrimination between more than two classes.

5.2.5. Location model.

The location model, is applicable in discrimination problems involving both continuous and discrete features. In a traditional framework, the model is applied to conditional probabilities given the class.

Let an object be described by p features, the first m being continuous variables and the remaining p-m being discrete variables. The probability of an object given its class, can then be decomposed as

$$\mathrm{Prob}(\underline{x}|w_i) = \mathrm{Prob}(\underline{x}_c,\underline{x}_d|w_i) = \mathrm{Prob}(\underline{x}_c|\underline{x}_d,w_i)\ \mathrm{Prob}(\underline{x}_d|w_i)$$

where

$$\underline{x}_c{}^T = (x_1, ..., x_m) \text{ and } \underline{x}_d{}^T = (x_{m+1}, ..., x_p) \tag{5.15}$$

$\mathrm{Prob}(\underline{x}_c|\underline{x}_d,w_i)$ is a multivariate probability conditional on both the class and the element of the feature space as defined by the values assumed by the discrete features. The location model assumes that the $\mathrm{Prob}(\underline{x}_c|\underline{x}_d,w_i)$ is a multivariate gaussian probability density function, with the same covariance matrix for each class and every combination of the discrete features, and mean dependent on the class and the values of the continuous and the discrete features. As for $\mathrm{Prob}(\underline{x}_d|w_i)$, the basic model is assumed. The location model is presented in KRZA75, KRZA79 and KRZA80, where only binary discrete variables are assumed.

Assuming for the time being that a full multinomial is considered for the discrete features and that there are only binary discrete features coded as 0 or 1, the mean vector for class w_i is expressed as

$$\underline{\mu}_i = \underline{v}_i + \sum_{j=m+1}^{p} \underline{\alpha}^i_j x_j + \sum^{p}\sum^{p}_{m<j<k} \underline{\beta}^i_{j,k} x_j x_k + \ldots$$

$$\ldots. + \underline{\gamma}^i_{m+1,\ldots.p}\, x_{m+1} \cdots x_p \tag{5.16}$$

The vectors $\underline{\alpha}^i_j$ (dimension m*1) relate to main effects, the vectors $\underline{\beta}^i_{j,k}$ (dimension m*1) to first order interactions, and so on. The vector $\underline{v}_i$ is the mean vector taking only into consideration the continuous features. The overall mean given values for the discrete features ($\underline{\mu}_i$), can be obtained just by inserting the values of the discrete variables.

If models other than the multinomial are assumed for the discrete features, then the terms accounting for interactions that are not needed, are dropped from the expression above. Specifically, if the discrete features are assumed to be independent, then the last expression reduces to

$$\underline{\mu}_i = \underline{v}_i + \sum_{j=m+1}^{p} \underline{\alpha}^i_j x_j \tag{5.17}$$

Expressions for the estimation of the means and the pooled covariance matrix of $\mathrm{Prob}(\underline{x}_c|\underline{x}_d,w_i)$ can be found in, e.g. KRZA75. They involve the inversion of a matrix which can be large (its dimension depends on the number of discrete features and the complexity of the model assumed for the discrete space). However, an iterative procedure is not required. For the estimation of $\mathrm{Prob}(\underline{x}_d|w_i)$, the techniques developed for the basic models apply.

The location model can be heavy in the sense that it may require a large number of estimates. Furthermore, if marginal probabilities have a zero maximum likelihood estimate, problems will arise in the estimation of the model parameters (the matrix which is to be inverted is singular whenever this situation occurs).

The location model can be easily extended in order to cope with variables assuming more than two values. If such a variable is not ordered, dummy binary variables can be used and the model applied directly. If a variable is ordered, the definition of dummy variables is not needed and the order relation may be kept.

5.2.6. Other models.

The number of models which can be found in the literature, is not limited to the ones just presented. For completeness, some other models as well as other classification schemes, not necessarily based on Bayesian approaches, are briefly introduced here.

Hills (HILL67) proposed a nearest neighbours approach. Essentially, a distance metric suitable for a discrete feature space is used (the distance between two objects is equal to the number of features in which they differ). Due to the intrinsic differences between continuous and discrete feature spaces, the decision function has to be suitably adapted. Specifically, the concept of nearest neighbour has to be modified. A neighbour region has to be defined and the set of objects within the neighbour region is the neighbour set. The decision function takes into account the number of objects from each class in the neighbour set, possibly weighted by the total number of objects per class in the reference set. Also, other distance metrics can be defined, that more or less can take into account order relations if ordered discrete features are used.

Extensions of the Parzen's estimators to discrete feature spaces can also be found in the literature. Aitchison et al. (AITC76) present the method and propose a kernel of the form

$$k(\underline{y}|\underline{x},\lambda) = \lambda^{p-d(\underline{x},\underline{y})} (1-\lambda)^{d(\underline{x},\underline{y})} \quad \text{for} \quad 0.5 \leq \lambda \leq 1 \tag{5.18}$$

where λ is a smoothing parameter, p is the dimension of the feature space, and $d(\underline{x},\underline{y})$ is the distance between $\underline{x}$ and $\underline{y}$. The distance is the same as used in HILL67.

Habbema et al. (HABB78) have applied the variable kernel method. The authors introduced several kernels suitable for discrete (either ordered or non ordered) and continuous variables. This makes the variable kernel approach applicable to mixed data types.

Orthogonal expansions have also been used. Martin et al. (MART72) developed a model for multinomial distributions, on a space defined by a set of binary features, where the conditional probabilities are of the form

$$\text{Prob}\ (\underline{x}|w_i) = \text{Prob}\ (\underline{x})\ (1 + h(\underline{a}_{wi},\underline{x}))$$

$$h(\underline{a}_{wi},\underline{x}) = \sum_{j}^{i} a_j \, \phi_j(\underline{x})$$

$$\phi_0(\underline{x}) = 1 \qquad \phi_i(\underline{x}) = 2x_i - 1 \quad \text{for} \quad i = 1,\dots,p$$

$$\phi_{\underline{\gamma}}(\underline{x}) = \prod_{j=1}^{s} \phi_{\gamma j}(\underline{x}) \quad \underline{\gamma} = (\gamma_1,\dots,\gamma_2) \quad \text{for} \quad s = 2,\dots,p \quad \text{and} \quad \gamma_s \,\varepsilon\, [1,\dots,p] \tag{5.19}$$

where the ϕ_j form a set of orthogonal polynomials and the $\underline{a}_{wi,j}$ are coefficients specific to a given class. In a Bayesian context, and for a k class discrimination problem, the decision function is

Choose class w_i if $d_i(\underline{x}) = \min (d_1(\underline{x}),\dots,d_k(\underline{x}))$

where

$$d_j(\underline{x}) = \text{Prob } (w_j) \ (1 + h(\underline{a}_{wj},\underline{x})) \tag{5.20}$$

The polynomial $h(\underline{a}_{wi},\underline{x})$ is a least square approximation to $(\text{Prob}(w_i|\underline{x})/\text{Prob}(w_i) - 1)$. Depending on the number of terms retained in $h(\underline{a}_{wi},\underline{x})$, full or reduced models can be adopted.

Ott et al. (OTT76) have also estimated probability functions by using an orthogonal expansion. The set of orthogonal functions adopted was

$$\phi_{\underline{r}}(\underline{x}) = (-1)^{\underline{x}'\underline{r}} \tag{5.21}$$

where $\underline{x}'$ is the transpose of $\underline{x}$ (coded with 0's and 1's) and $\underline{r}$ is a binary index that numbers all the different objects in the feature space. The parameters of the expansion are computed according to a mean least square criterion. Again, terms may or may not be deleted from the expansion.

Bahadur (BAHA61) has also a contribution to the modelization of discrete probability spaces. Assuming a feature space defined by a set of binary features (coded as 1 or 0), Bahadur's model expresses a conditional probability distribution in the form

$$p(\underline{x}|w_i) = f(\underline{x}|w_i) \, f_1(\underline{x}|w_i)$$

where, dropping the class indicator,

$$f(\underline{x}) = 1 + \sum_{i<j} r_{ij} z_i z_j + \sum_{i<j<k} r_{ijk} z_i z_j z_k + \ldots + r_{1\ldots p} z_1 \ldots z_p$$

$$f_1(\underline{x}) = \prod_{i=1}^{p} \mathrm{Prob}(x_i = 1)^{xi} \, (1 - \mathrm{Prob}(x_i = 1))^{1-xi}$$

and

$$z_i = (x_i - \mathrm{Prob}(x_i = 1)) \,/ \sqrt{} \, (\mathrm{Prob}(x_i = 1)\,(1 - \mathrm{Prob}(x_i = 1)))$$

$$r_{ij} = E\,(z_i z_j) \qquad r_{1\ldots p} = E\,(z_1 \ldots z_p) \tag{5.22}$$

and the symbol E indicates expectation. As in previous models, parcels in $f(\underline{x})$ may be assumed to be zero, i.e. high order correlations may be assumed to be zero. In practical cases and with a limited number of training objects, this is what should be done. Possibly, the appeal of this reparametrization, was the ability to represent multinomial distributions in terms of means, correlations, etc.. However, this may result in negative estimates of probability.

Chow et al. (CHOW68) proposed the modelling of a discrete probability distribution by means of a product approximation in which only dependencies of a variable on another one are used. Specifically

$$\mathrm{Prob}(\underline{x}) = \prod_{i=1}^{p} \mathrm{Prob}\,(x_{mi} \mid x_{mj(i)}) \qquad 0 \leq j(i) \leq i \tag{5.23}$$

where $(m_1,\ldots,m_p)$ is an unknown permutation of the integers 1,....,p. This is a special case of the basic model presented above. Also, only p-1 (of a total of p*(p-1)/2) dependencies between the p variables are considered. The contribution of Chow is a method to select the most suitable set of dependencies.

The methods just presented may be grouped as follows. Firstly, methods which have their counterparts in continuous spaces (e.g. kernel and nearest neighbours), and therefore can be thought of as extensions of methods developed for continuous spaces. Secondly, methods which are finite series expansions of the multinomial space (Bahadur, Martin, Ott). Thirdly, procedures which are optimization techniques to select a convenient model among those offered by the basic model (Chow). If a fourth group is added comprising the methods presented in previous sections, a classification is established which covers the lines of action of the work on discrete/mixed classifiers. Missing from this classification scheme, are the work of, e.g. Nakache (NAKA80), based on correspondence analysis, Young et al. (YOUN81), with the A distribution, Dillon et al. (DILL78), with the distributional distance, Stoffel (STOF74), with the prime event theory, Gleser (GLES72) with sequential decision trees, and possibly others.

Finally, and at the level of application, examples can be found where the specific nature of discrete features was simply ignored. Procedures developed for continuous features (e.g. Fisher's linear discriminant) are sometimes used without any explicit adaptation, provided that 'unreasonable' things do not occur (e.g. divisions by zero).

5.2.7. Sampling experiments.

In this section some sampling studies are presented where several classification procedures were compared.

Gilbert (GILB68) tested five types of classifiers:

1 - A Bayes classifier based on a full multinomial model, with parameters estimated by maximum likelihood.

2 - Similar to 1) but with the features assumed to be independent given the class.

3 - Linear logistic model with maximum likelihood estimates of the parameters.

4 - Similar to 3) but with minimum chi-square estimates.

5 - Fisher's linear discriminant.

All classifiers except classifier 1, are or may be put into the form of linear procedures in the variables. Several populations were used, all of them defined by 6 binary features, and arising from a model where only main effects and first order interactions were assumed. All experiments involved discrimination between two classes. The various classifiers were compared based on their actual error rate and the mean correlation coefficient (see GILB68 for a definition of the correlation coefficient). For each population (15 in total), 100 data sets with 100 objects and 100 data sets with 500 objects, were generated. In view of the structure chosen for the populations, the sample size and the number of parameters required by the various classifiers, a similar behaviour of the four linear procedures was found. All of these were superior to the full multinomial approach.

Moore (MOOR73) has also evaluated five discrimination techniques. They were:

1 - A Bayes classifier based on a full multinomial model, with parameters estimated by maximum likelihood.

2 - Classifier based on a first order Bahadur model.

3 - Classifier based on a second order Bahadur model.

4 - Quadratic classifier.

5 - Fisher's linear discriminant.

Classifiers 2 and 5 are linear procedures in the variables. Classifiers 4 and 5 are suited for continuous feature spaces. Populations defined by six binary variables were used. However, and differently from Gilbert (GILB68), Bahadur models were used in order to generate the populations under study. Three population groups were considered: independence between features (group I), only one nonzero correlation coefficient (group II), and all correlations positive (group III). As in GILB68, the criteria to compare performances, were the actual error and the correlation coefficient. In group I classifiers 2 and 5 performed better and the worst was the full multinomial. For group II, the relative performances remained similar, whereas in group III, the second order Bahadur model performed better.

Dillon et al. (DILL78, GOLD78) also undertook a simulation study which is similar to the studies just presented. Again feature spaces defined by 6 binary variables and with two classes were considered. A Bahadur second order approximation was used to model the populations under simulation. Three population groups were considered. Group I is similar to Moore's group II (only one correlation in one class is assumed to be different from zero). Group II and III were similar to Moore's group III (group II is the case where correlations are different in the two classes, whereas in group III the correlations are the same in both classes). The classifiers tested were:

1 - A Bayes classifier based on a full multinomial model, with parameters estimated by maximum likelihood.

2 - Similar to 1) but with the features assumed to be independent given the class.

3 - Fisher's linear discriminant.

4 - A classifier based on the distributional distance (see DILL78).

5 - A classifier based on a first order Bahadur model.

6 - A complete model of the Martin-Bradley expansion.

7 - A model based on the Martin-Bradley expansion but taking only main effects into account.

8 - A model based on the Martin-Bradley expansion taking into account main effects and first order interactions.

With sample sizes of 200 and 400 (100 trials) their conclusions were as follows. For group I, the linear procedures were comparatively better except for large values of the correlation coefficient and larger sample sizes. This pattern was the same for group II, and group III.

In Schmitz et al (SCHM83), another simulation study of four classification procedures is presented. They are:

1 - Fisher's discriminant.

2 - Quadratic discriminant.

3 - Classifier based on the linear logistic model.

4 - Classifier based on the kernel model.

Several feature spaces defined by three discrete features and one continuous feature were used. The data was generated from several four dimensional gaussian distributions (with various mean vectors and different covariance structures) followed by discretization of three features. As a result, feature vectors were obtained composed of one continuous variable, an (assumed) ordered discrete feature (4 categories), a (assumed) non-ordered discrete feature (3 categories) and a binary feature. All experiments involved discrimination between two classes. The various classifiers were compared by means of a rank-order analysis of four performance measure: the error rate, the quadratic scoring rule, the modified logarithmic scoring rule and a doubt-based scoring rule (see SCHM83). These performance measures were estimated using test sets independent of the training sets and with the same number of objects. With sample sizes of 50 objects per class (16 trials for 45 different spaces) and 100 objects per class (9 trials for 36 different spaces) their conclusions were as follows. There were small differences in performance between training sample sizes of 50 and 100. The major factor in deviation from an overall average performance was the covariance structure. The linear discriminants (1 and 3 above) had similar performance. With unequal covariance structure the kernel method was the best and the linear discriminators were the worst. This ranking was reversed in the case of equal covariance structures.

Schmitz et al (SCHM85) undertook another simulation study of five classification procedures. They were:

1 - Fisher's discriminant.

2 - Quadratic discriminant.

3 - Classifier based on the linear logistic model.

4 - Classifier based on the kernel model.

5 - Classifier based on an independence model.

As in SCHM83, mixed data was considered. Several feature spaces defined by three binary features and three continuous feature were used. The data was generated according to the location model. Several correlation structures, various distances between classes and interactions between the binary variables were selected, yielding various feature spaces. All experiments involved discrimination between two classes. The various classifiers were compared by means of two performance measures: the error rate and the modified logarithmic scoring rule. These performance measures were estimated using test sets independent of the training sets and with either 200 or 500 objects per class. Training set sizes of 500 objects per class, 100 objects per class, 25 objects per class and 25 objects for one class and 100 for the other class, were used. Their conclusions were as follows. When the variables are independently distributed, the linear classifiers (1, 3 and 5 above) have similar performance and perform better than the non-linear procedures. For equal interaction structures and absence of non-linearities (induced by the conditional distribution of the continuous variables), classifiers 1 and 3 were the best. For non-linearities the non-linear procedures (2 and 4) performed better. The performance of classifiers 2 and 4 was more dependent on the size of the training set.

These experiments mainly show the importance of the relation between the size of the training sample, the complexity (number of parameters for which estimates are required) of the classifier adopted, and the structure of the feature space under consideration. Classifiers did not behave better because they were, e.g. basic model classifiers or Bahadur classifiers, but because they were linear when the sample size was small, or took into account a sufficient number of higher order interactions between features, when the training sample was larger and differences in these interactions among the various classes were significant.

5.2.8. Remarks.

The material just presented indicates the large variety of classifiers that can be found in the literature. Among the various methods presented, a justification is given for the three models which were applied in chapter 6. These models were thoroughly discussed in the previous subsections [4]. This justification will proceed through various steps.

Firstly, Bayes classifiers were selected. This stems directly from the fact that at the theoretical level in a statistical framework, they are optimum.

Secondly, parametric classifiers were preferred to non-parametric classifiers. There are two reasons for this. Generally, and in 'real' life, the size of training samples is small. Secondly, some of the parametric models selected also give very flexible ways to model the feature space.

Thirdly, some of the models originate from a set of hypotheses which induce at least curiosity in their use.

These are a set of basic reasons which are satisfied in total or in part by the models selected. In the following each of the classifiers selected is briefly recalled.

The basic model (or its log-linear version) is simple, naive, and yet powerful, and estimates are easily obtained.

The logistic model has the very interesting aspect of dealing directly with a-posteriori distributions. It can cope with mixed data types in contrast to the basic model.

The location model was mainly selected for its ability to deal with mixed data types, and its naive approach: it separates discrete from continuous features.

Finally, it should be remarked that some of the models yield equivalent discriminants.

5.3. Feature selection techniques.

Feature selection, in discrete or in continuous feature spaces, is basically the same. In both spaces, feature selection algorithms require the choice of a search strategy, a selection criterion and a stopping criterion.

Differences between comparable strategies in continuous and discrete feature spaces are mainly of a computational nature. In other words, in discrete feature spaces there is the possibility to apply strategies that would be impractical in a continuous feature space due to difficulties in computing the selection criteria in a

[4] Although the probit model was also thoroughly discussed, it was not used because of its similarities with the logistic model.

continuous feature space, e.g. a summation in a discrete feature space is an integration in a continuous feature space.

Selection criteria have to take into account the nature of the discrete space (i.e. if they involve probability functions, then these have to be modeled by discrete probability functions). This also applies to the stopping criteria. Exceptions to this are the heuristic criteria.

Following a pattern similar to the previous section on classifiers, methods used in chapter 6 are first dealt with. Other approaches found in the literature as feature selection procedures in discrete feature spaces, are then presented. Some final remarks conclude this section.

5.3.1. Algorithms used.

Instead of describing the algorithms one by one, we will rather concentrate on the search strategies, and on the selection and stopping criteria used. Each particular algorithm used is then a combination of a search strategy, a selection criterion and a stopping criterion.

5.3.1.1. The search strategies.

Forward sequential search strategies were used. Two variants to be called B1 and B2 were applied. Both variants allowed for structure changes during the search. That is, at each step of the selection a new feature is chosen, and a 'best' structure is identified. Variants B1 and B2 differ in the set of structures allowed. Variant B1 allows for independence between sets of features. Variant B2 allows for independence between sets of features as well as conditional independence between sets of features, given the values of other features. Thus, variant B1 is a restricted version of variant B2. A restriction of this type was required for some selection criteria in order to avoid excessive computation time. Further comments on the search strategies, may be found in chapter 3.

5.3.1.2. The selection criteria and the corresponding stopping criteria.

The selection criteria used are either based directly on the error rate or on measures associated with the error rate. Three different selection criteria and three associated stopping criteria were used. They will be referred to as criteria A, B and C.

Criterion A is an estimate of the error rate according to the leave-one-out technique with a correction for insensitivity as proposed by Ben-Bassat (see chapter 3). Since the leave-one-out technique implies a classifier, Bayes classifiers were used with the basic model assumed for the conditional probabilities given the class. Naturally, and according to the search strategy, various structures and variations on the model can be assumed. As for the estimation of the parameters of the classifiers, two techniques, i.e. maximum likelihood estimation and Bayesian estimation were used (see chapter 3, and expressions 3.1 and 3.2, respectively). Depending on the way a particular training sample was obtained (either sampling from the universe of objects or sampling per class), a-priori probabilities were either estimated with the help of the training sample, or given a value which somehow reflected expert knowledge. The stopping criterion was an increase in the error rate as estimated by the leave-one-out method.

Criterion B is also calculated using a leave-one-out technique. It works as follows: for every object in the training set, a-posteriori probabilities are estimated using all the other objects in the training set. A penalty score is then computed. For a c class classification problem, if object $\underline{x}_i$ belongs to class w_j, the corresponding penalty score is

$$s_i = \sum_{k=1}^{c} (P^\wedge(w_k|\underline{x}_i) - P^\wedge(w_j|\underline{x}_i)) \quad =$$

$$\sum_{k=1}^{c} P^\wedge(w_k|\underline{x}_i) - \sum_{k=1}^{c} P^\wedge(w_j|\underline{x}_i) \quad =$$

$$1 - c * P^\wedge(w_j|\underline{x}_i) \qquad (5.24)$$

and the criterion is

$$S = \sum_{i=1}^{n} s_i / n \qquad (5.25)$$

where n is the total number of objects in the training set [5]. In a two class problem, a correct classification (using a classifier based on the Bayes rule) gives a negative value to the corresponding s_i whereas an incorrect classification results in a positive value. With more than 2 classes, incorrect classifications give either

[5] S in expression 5.25 is a scoring rule. In HILD78, the concepts of proper and strictly proper scoring rules are introduced. S is a non-proper scoring rule.

positive or negative values to s_i but a correct classification gives s_i the largest negative value. The larger the difference between the a-posteriori probabilities, the larger the 'reward' (or 'punishment') for the classification.

The a-posteriori probabilities required by the selection criterion were computed in accordance with the structure and estimation technique being tested and were estimated using all the objects but the one being classified. The basic model and maximum likelihood and a Bayesian estimation method were used. Criterion B is a risk averaging technique which is directly related to the error rate. The stopping criterion is defined in terms of an increase in the value of S.

Criterion C involves a test statistic which is given by the log-likelihood ratio test (GOOD70). This criterion is computed testing the model of independence between the random variable expressing class membership and the random variable representing the candidate feature, given the features and structures so far adopted. Under the null hypothesis (independence), the test statistic is distributed as a chi-square with a given number of degrees of freedom (GOOD70). The basic model is used for the various probabilities needed.

Suppose that at some step, features x_j, x_k and x_l have been selected together with the structure (for the joint probability function)

$$\mathrm{Prob}(w_i,x_j,x_k,x_l) = \mathrm{Prob}(w_i,x_j,x_k)\ \mathrm{Prob}(w_i,x_j,x_l)\ /\ \mathrm{Prob}(w_i,x_j) \tag{5.26}$$

where $\mathrm{Prob}(w_i,x_j,\ldots,x_z)$ is the generic representation of the joint probability function of class w_i and features $x_j,\ldots,x_z$. If a candidate feature x_r is subsequently assessed and, e.g. it is assumed that feature x_r is independent of features x_j, x_k and x_l, the following two hypotheses are constructed:

$$\mathrm{Prob}(w_i,x_j,x_k,x_l,x_r) = \mathrm{Prob}(w_i,x_j,x_k)\ \mathrm{Prob}(w_i,x_j,x_l)\ \mathrm{Prob}(w_i,x_r)\ /\ \mathrm{Prob}(w_i,x_j)\ /\ \mathrm{Prob}(w_i) \tag{5.27}$$

and

$$\mathrm{Prob}(w_i,x_j,x_k,x_l,x_r) = \mathrm{Prob}(w_i,x_j,x_k)\ \mathrm{Prob}(w_i,x_j,x_l)\ \mathrm{Prob}(x_r)\ /\ \mathrm{Prob}(w_i,x_j) \tag{5.28}$$

For the first hypothesis, the maximum likelihood estimate of $\mathrm{Prob}(w_i,x_j,x_k,x_l,x_r)$ is

$$P\wedge(w_i,x_j,x_k,x_l,x_r) = (n^i{}_{jk}\, n^i{}_{jl}\, n^i{}_r) \,/\, (n\, n^i\, n^i{}_j) \tag{5.29}$$

and

$$\chi^2{}_{(1)} = 2 \sum_{i,j,k,l,r} n^i{}_{jklr} \log_e [(n^i{}_{jklr}\, n^i{}_j\, n^i) \,/\, (n^i{}_r\, n^i{}_{jk}\, n^i{}_{jl})] \tag{5.30}$$

is distributed as a chi-square with the number of degrees of freedom equal to

$$c\, I_j\, I_k\, I_l\, I_r - c\, I_j\, I_k - c\, I_j\, I_l - c\, I_r + c\, I_j + c \tag{5.31}$$

For the second hypothesis, the maximum likelihood estimate of Prob(w_i,x_j,x_k,x_l,x_r) is

$$P\wedge(w_i,x_j,x_k,x_l,x_r) = (n^i{}_{jk}\, n^i{}_{jl}\, n_r) \,/\, (n^2\, n^i{}_j) \tag{5.32}$$

and

$$\chi^2{}_{(2)} = 2 \sum_{i,j,k,l,r} n^i{}_{jklr} \log_e [(n^i{}_{jklr}\, n^i{}_j\, n) \,/\, (n_r\, n^i{}_{jk}\, n^i{}_{jl})] \tag{5.33}$$

is distributed as a chi-square with the number of degrees of freedom equal to

$$c\, I_j\, I_k\, I_l\, I_r - c\, I_j\, I_k - c\, I_j\, I_l - I_r + c\, I_j + 1 \tag{5.34}$$

The difference between $\chi^2{}_{(2)}$ and $\chi^2{}_{(1)}$ is also distributed as a chi-square:

$$\chi^2{}_{(2)} - \chi^2{}_{(1)} = 2 \sum_{i,j,k,l,r} n^i{}_{jklr} \log_e (n^i{}_r\, n \,/\, n^i \,/\, n_r) \tag{5.35}$$

with

$$c\, I_r - c - I_r + 1 \tag{5.36}$$

degrees of freedom. The symbol $n^i{}_{jklr}$ indicates the number of objects from class w_i that score in categories j, ..,r of features x_j, ..., x_r, the symbol I_j indicates the number of categories of variable x_j and the symbol c indicates the number of classes. In this way, the quantity $\chi^2_{(2)} - \chi^2_{(1)}$ measures the change in the degree of association between the variable expressing class membership and the features, when feature x_r is added according to the structure assumed. It tests the model of independence between the random variable expressing class membership and the random variable representing the candidate feature, given the features and structure so far selected (null hypothesis). The feature for which Prob $(z > \chi^2_{(2)}-\chi^2_{(1)})$ is a minimum (z is a general variable) is conditionally selected. Structures that require more estimates or features with a larger value of categories (again implying more parameters to estimate), yield a larger number of degrees of freedom. For the same value of the test statistic, this implies a decrease in the confidence level for rejecting the null hypothesis. Features with higher discriminating power yield a larger value for the test statistic and therefore an increase in the confidence level for rejecting the null hypothesis. Thus this selection criterion evaluates the quality of a feature (in terms of discrimination ability) as well as its cost (in terms of the number of parameters). An expression similar to 5.35 is computed in the first step of the search, when each feature is analyzed individually. Therefore, if the algorithm keeps track of previous computations, they do not have to be repeated in later steps. This stems from the selection criterion itself, and the implicit assumption that the structure in class conditional probabilities is the same as the structure in joint probabilities (see TOUS74). This is a weak assumption and rarely verified. Nevertheless, it speeds up the feature selection. Finally, the stopping criterion is defined in terms of a limit on the confidence level of the test.

5.3.1.3. The algorithms.

The algorithms used in chapter 6, can be grouped as follows:

i) Search strategy B1 with selection criteria A an B and the corresponding stopping criteria.

ii) Search strategy B2 with selection criterion C and the corresponding stopping criterion.

Group i) includes four different feature selection algorithms since two estimation techniques were used in order to estimate the classifiers required by the selection criteria A and B. Group ii) includes one feature selection algorithm.

Search strategy B2 was not applied with selection criteria A and B, due to excessive computation times. For the same reason, two additional selection criteria indicated in chapter 3 were not used. Finally, the outcome of each selection is both a set of features as well as a structure for the class conditional probability function required in a Bayes classifier.

5.3.2. Other approaches.

For reasons of completeness, some feature selection algorithms, which appeared in the literature in the specific context of discrete feature spaces, will now be shortly reviewed. For each algorithm presented in chronological order, the search strategy, the selection and stopping criteria are given.

Raiffa (RAIF61) introduces two 'item selection procedures'. Two class discrimination problems and binary features are considered although the extensions to more than two classes and to features with more than two categories, are immediate. The search strategy is either forward sequential with s features added at each step or a restricted 'add s take away r' (with s > r). At each step, s features are conditionally selected. From these s features, r features are removed, at the end of each iteration. The selection criterion is the expected value of the Bayes risk plus the cost, suitably expressed, of measuring a feature. The selection stops when the cost of obtaining a new feature is larger than the decrease in the risk function. The cost associated with the features can be made dependent on the features selected at a given step enabling an easy and elegant way to introduce restrictions in the order of selection.

Hills (HILL67) proposes a forward sequential search procedure and two selection criteria. The first of these is the divergence, defined as:

$$\sum_{i=1}^{m} (\mathrm{Prob}(\underline{x}_i|w_1) - \mathrm{Prob}(\underline{x}_i|w_2)) \log_e(\mathrm{Prob}(\underline{x}_i|w_1)/\mathrm{Prob}(\underline{x}_i|w_2)) \tag{5.37}$$

The second criterion is an approximation to the divergence:

$$\sum_{i=1}^{m} (\mathrm{Prob}(\underline{x}_i|w_1) - \mathrm{Prob}(\underline{x}_i|w_2))^2 /(\mathrm{Prob}(\underline{x}_i|w_1)+\mathrm{Prob}(\underline{x}_i|w_2)) \tag{5.38}$$

The variable m indicates the number of different elements in the feature space. For the first criterion, both $\mathrm{Prob}(\underline{x}_i|w_1)$ and $\mathrm{Prob}(\underline{x}_i|w_2)$ must be always different from zero whereas for the second only their sum has to be always different from zero. Hills mainly considered two class discrimination problems and binary features. The extension to other features is straightforward whereas the extension to more than two classes requires, in the case of the divergence, a selection for every combination of two classes followed by a heuristic to combine the various results obtained. A generalization is suggested for the second criterion (see HILL67).

Lachin (see LACH73) describes a selection criterion which guides an 'add 1, take away 1' search strategy. At each step, a feature is selected and a feature may be removed depending on a given criterion. Lachin's selection criterion uses a test statistic which is an equivalent of the Pearson goodness of fit chi-square. Consider a two class problem and a full multinomial model for the class conditional probabilities. The test statistic

$$\chi^2_{(1)} = n\ (1 - \sum_{j\ldots k} (n^1_{j\ldots k}\, n^2_{j\ldots k}\, n) / (n^1\, n^2\, n_{j\ldots k})) \qquad (5.39)$$

is asymptotically distributed as chi-square with c_j-1 degrees of freedom (c_j is the complexity of the space at step j in the selection). In the above expression $n^i_{j\ldots k}$ is the number of samples from class w_i in categories j,...,k of features $x_j,\ldots,x_k$, $n_{j\ldots k} = n^1_{j\ldots k} + n^2_{j\ldots k}$ and n^i is the number of objects from class w_i in the training set. The null hypothesis of independence between the variable expressing class membership and the features can be tested with $\chi^2_{(1)}$. If a new feature is added, another hypothesis of independence can be tested with the new chi-square ($\chi^2_{(2)}$) having c_{j+1}-1 ($c_{j+1} > c_j$) degrees of freedom. Lachin shows that $\chi^2_{(3)} = \chi^2_{(2)} - \chi^2_{(1)}$ is also asymptotically distributed as a chi-square with $c_{j+1} - c_j$ degrees of freedom. The null hypothesis of conditional independence between the variable expressing class membership and the new feature, given the features already selected, can be tested by $\chi^2_{(3)}$. This statistic is a measure of the effectiveness of the new feature for discrimination, given the others already selected. The feature for which the probability Prob($z>\chi^2_{(3)}$) is a minimum (z is a general variable) is conditionally selected. If this probability is less than the minimum acceptable probability level (m.p.a.) for the addition of a new feature, the feature is accepted. Otherwise the procedure is discontinued. At the end of each iteration, each feature is tentatively removed from the selected set and a new set of values, one for each feature, of Prob($z>\chi^2_{(3)}$) is computed and compared with the minimum acceptable probability level for the deletion of a feature (m.p.d.). Features with Prob($z>\chi^2_{(3)}$) > m.p.d. are deleted. Both m.p.a. and m.p.d. are levels of significance and m.p.a. > m.p.d..

Goldstein et al. (GOLD75) propose feature selection algorithms in the context of a two class discrimination problem. Surprisingly an exhaustive search strategy is indicated, limiting the applicability of the algorithms to feature spaces with low dimensionality. Two selection criteria are indicated, both based on a distance measure (ranging from zero to one) between discriminant scores, large values being associated with better discrimination. The distance results from the work of Glick (GLIC73) and is defined as

$$\begin{aligned} &d = \min_i \mid \sqrt{\mathrm{Prob}(\underline{x}_i,w_1)} - \sqrt{\mathrm{Prob}(\underline{x}_i,w_2)} \mid \\ &\text{for} \quad \sqrt{\mathrm{Prob}(\underline{x}_i,w_1)} - \sqrt{\mathrm{Prob}(\underline{x}_i,w_2)} \neq 0 \end{aligned} \tag{5.40}$$

where i scans the feature space. Full multinomial models are assumed and an estimate of d is obtained by inserting the maximum likelihood estimates of the joint probabilities. The selection criteria proposed are then

$$\begin{aligned} &\min_i \mid (\sqrt{n^1_i} - \sqrt{n^2_i}) \,/\, \sqrt{r} \mid \\ &\text{for} \quad (\sqrt{n^1_i} - \sqrt{n^2_i}) \,/\, \sqrt{r} \neq 0 \end{aligned} \tag{5.41}$$

and

$$\begin{aligned} &\mathrm{average}_i \mid (\sqrt{n^1_i} - \sqrt{n^2_i}) \,/\, \sqrt{r} \mid \\ &\text{for} \quad (\sqrt{n^1_i} - \sqrt{n^2_i}) \,/\, \sqrt{r} \neq 0 \end{aligned} \tag{5.42}$$

The set of features selected is the one for which 5.41 (5.42) is maximum. In expressions 5.41 and 5.42, r is the complexity of the space defined by those features not yet selected and the index i scans the feature space defined by those features already selected and the candidate feature. The symbol n^i_j denotes the number of objects available in the training set from class w_i that score in element j of the feature space under consideration. The first criterion discards too much information whereas the second is expected to be more reliable at the cost of an increase in computer time. In short, both are somewhat rude approaches.

Nakache et al. (NAKA78) propose a technique analogous to stepwise regression with a forward sequential search strategy. The selection criterion adopted essentially borrows the mathematics of stepwise regression (see KEND66 vol 2, chap. 27). In order to apply stepwise regression, a coefficient of association between variables (i.e. variables representing features and class membership) is needed. Cramer's coefficient of association in contingency tables (see KEND66 vol 3, chap. 33) which relates two discrete variables, is adopted (it assumes values between 0, in the case of independence, and 1, in the case of complete association). A multiple association coefficient is then defined in terms of coefficients of partial association and coefficients of association, by an expression equivalent to the one that in regression relates multiple correlation with partial correlation and correlation. The variable expressing class membership is assumed to be the dependent variable whereas the features are the independent variables. The selection criterion is the increase in the coefficient of multiple association. The feature to which the largest increase is associated is selected, provided that the increase is larger than a minimum level. If no features can be found, the algorithm stops.

Habbema et al. (HABB81) describe a program that uses a forward sequential procedure for feature selection. Six selection criteria are indicated. They all are weighted (by the estimated a-priori probabilities) means of the average of penalty scores per class. A common expression is

$$Q = \sum_{i=1}^{c} P^\wedge(w_i)\,(n^i)^{-1} \sum_{j=1}^{n^i} q(\underline{x}^i_j) \tag{5.43}$$

where $q(\underline{x}^i_j)$ is an individual penalty assigned to object $\underline{x}_j$ from class w_i, in the training set. Six forms are proposed for the individual penalties yielding six selection criteria. They are

$$\begin{aligned}
q_1(\underline{x}^i_j) &= 0 \text{ when the object is correctly classified} \\
&= 1 \text{ when the object is incorrectly classified} \\[1ex]
q_2(\underline{x}^i_j) &= -\log_e P^\wedge\,(w_i \mid \underline{x}^i_j) \\[1ex]
q_3(\underline{x}^i_j) &= -\,[\,\log_e y_{i,j,i} + \varepsilon \sum_{i \neq k} \log_e (y_{i,j,k}/\,\varepsilon\,)\,] \\
&\quad \text{where } y_{i,j,k} = (1-\varepsilon)\,P^\wedge\,(w_k \mid \underline{x}^i_j) + \varepsilon \text{ and } \varepsilon \text{ is a small number} \\[1ex]
q_4(\underline{x}^i_j) &= (1 - P^\wedge\,(w_i \mid \underline{x}^i_j))^2 + \sum_{k \neq i} P^\wedge\,(w_k \mid \underline{x}^i_j)^2 \\[1ex]
q_5(\underline{x}^i_j) &= \sum_{k} (I(k \geq i) - \sum_{s=1}^{k} P^\wedge(w_s \mid \underline{x}^i_j))^2 \\
&\quad \text{where } I(k \geq i) = 0 \text{ if the object is correctly classified} \\
&\quad \text{and } I(k \geq i) = 1 \text{ if the object is incorrectly classified} \\[1ex]
q_6(\underline{x}^i_j) &= \text{loss incurred if object } \underline{x}_j \text{ from class } w_i \text{ is assigned to class k.}
\end{aligned} \tag{5.44}$$

The penalty score $q_1(\underline{x}^i_j)$ yields the error rate, $q_2(\underline{x}^i_j)$, $q_3(\underline{x}^i_j)$ and $q_4(\underline{x}^i_j)$ are monotonic functions of a-posteriori probabilities, $q_5(\underline{x}^i_j)$ takes into account a possible natural ordering of the classes, and finally $q_6(\underline{x}^i_j)$ results in a risk function. The a-posteriori probabilities required are estimated by the leave-one-out method and a feature is selected according to the minimum of Q. The selection stops when the decrease in Q is lower than a threshold value.

Other papers are those by, e.g. Leonard et al. (LEON74), where an algorithm is presented that selects binary features for use with linear classifiers, Bezdek et al. (BEZD77), where again binary features are considered and fuzzy sets are used, Goldstein et al. (GOLD78), where a procedure is reported that relies on the work of Kullback (KULL59), Anderson (ANDE82), where a likelihood ratio is tested and used for the selection of features in the logistic model, Li et al. (LI84) where a measure of association between binary vectors is proposed, etc..

Typically, the selection criteria are either estimates of the error or functions of it, statistical tests, etc.. This is the same for continuous feature spaces, the differences resulting from the nature of a discrete space and from computational aspects as indicated at the beginning of section 5.3. In particular, distance measures that assume the existence of a metric space, are not suitable with non-ordered discrete features.

5.3.3. Remarks.

As in the section on classifiers, the choice of the feature selection methods used in the applications in chapter 6 will now be justified. The selected methods were presented in section 5.3.1. In the following, the aspects of search strategies, selection criteria, models and estimates are discussed.

As for search strategies, forward sequential searches were used with structure changes at each step and throughout the search. The reasons for allowing structure changes have already been indicated in chapter 3. The results of simulation experiments were also given there.

Three selection criteria were used. Two are directly related to the error rate and one makes use of a statistical test. Since the error is the benchmark of a classifier its use as a selection criterion is a natural choice. Criterion C was used because, firstly it had been applied in a simulation experiment involving feature selection (QUEI84), and secondly, it is computational very efficient.

The basic model was used whenever probability functions were required. Its simplicity, its power for modelling complex structures, and the easy and efficient ways in which estimates (e.g. maximum likelihood) can be computed, were the main reasons.

Duin (DUIN78) has shown the importance of the choice of an estimation technique in the design of a Bayes classifier. Either a maximum likelihood estimate or a Bayesian one (with the exception of selection criterion C and this for obvious reasons) were used. Although these two approaches may not yield the 'best' estimates, the point is made that it is important to test more than one type of estimators.

Having selected features using one of the above procedures, a classification procedure may then be formulated on the basis of the basic model, the structure and the features selected. Whenever selection criteria A or B are used, the feature selection procedure is also an implied classifier selection procedure.

5.4. Mapping techniques.

Mapping techniques to be applied to sets of objects described by discrete features (either ordered or non ordered), have to take into account the uncomfortable fact that the original data is of a qualitative nature. For instance, let a discrete feature be 'colour of the eye', with 4 possible outcomes (black, blue, brown, other). It is not possible to apply directly a numeric transformation to such a set of data. First, it is necessary to suitably code the data before the transformation can be obtained and applied. Since the original data is qualitative and the results are projections on a metric space, mapping techniques can also be viewed as techniques to quantify qualitative data. This type of data occurs in a wide range of sciences (e.g. social sciences, psychology, etc.), and many efforts have been directed to the problem of quantification of discrete data. There exists a specialized literature (e.g. the journal Psychometrika from the Psychometric Society) covering this topic. A remarkable paper on the quantitative analysis of qualitative data is YOUN81.

Among the various techniques available, correspondence analysis has been used. In the following section, the method is introduced.

5.4.1. Correspondence analysis.

Correspondence analysis is a multivariate statistical technique which has been used for the analysis of multivariate discrete data. In the literature this, or closely related techniques, are also known under the names of reciprocal averaging, multiple correspondence analysis, homogeneity analysis. Various treatments and applications may be found in BENZ73, GELS82, GELS84, GIFI81, HILL73, HILL74, LEBA77, LORE77, NAKA77, NISH80, QUEI83, etc.. The technique may be approached from several angles and has been developed by various authors in attempts to solve a wide variety of problems. Here, it will be introduced as a mapping technique applicable to discrete data, following LEBA77.

Consider an I*J contingency table. Such a table may be used to group n objects according to two different characteristics. Each entry in the table (n_{ij}) is the number of objects having attribute i of the first characteristic and attribute j of the second. One characteristic may, e.g. represent a division in I classes of the universe of objects. The other one may, e.g. represent a set of properties which an object may or may not have. In this way, a class may be described by the distribution of its objects among the J properties. Similarly, a property may be described by the way objects having the property, are distributed among the I classes. Instead of the raw numbers n_{ij}, the table with general element given by

$$f_{ij} = n_{ij} / n \tag{5.45}$$

is considered. Here n is the total number of objects and the table with elements f_{ij} is called a frequency table. Applying correspondence analysis to such a set of data results in a mapping onto a lower dimensional space (defined by a set of orthonormal axes) in such a way that similarities between classes and between properties (i.e. rows and columns) are preserved as much as possible, according to a certain metric. The so called row and column profiles (BENZ73) are used to represent rows and columns respectively, and the metric used is the chi-square distance (BENZ73). In the following, the profiles and chi-square metric are presented and commented upon. Instead of identifying the profiles as row and column profiles, the terms class and property profiles shall be used in order to remain consistent with the (restricted) interpretation given above.

A class may be considered as a point in $\underline{R}^J$ with coordinates given by

$$p^c_{ij} = f_{ij} / (\sum_{j=1}^{J} f_{ij}) \tag{5.46}$$

This is the so-called class-profile. By choosing this form of representation, the similarity between classes is made independent of the total number of objects. The distinguishing characteristic of a class is the distribution of its objects among the various properties. Since the sum of the coordinates of a profile is equal to 1, the profiles may be thought of as to be contained in a subspace of $\underline{R}^J$ with a maximum possible dimension of J-1.

The same reasoning may be applied to the properties. Property profiles have as the general element

$$p^p_{ij} = f_{ij} / (\sum_{i=1}^{I} f_{ij}) \tag{5.47}$$

and may be thought of as to be contained in a sub-space of $\underline{R}^I$ with a maximum possible dimension of I-1.

The chi-square distance, as introduced by Benzecri (BENZ73) is defined as

i) Distance between classes i and i'.

$$d^2(i,i') = \sum_{j=1}^{J} \left(\sum_{i=1}^{I} f_{ij}\right)^{-1} \left(p^c_{ij} - p^c_{i'j} \right)^2$$

ii) Distance between properties j and j'.

$$d^2(j,j') = \sum_{i=1}^{I} \left(\sum_{j=1}^{J} f_{ij}\right)^{-1} \left(p^p_{ij} - p^p_{ij'} \right)^2 \qquad (5.48)$$

The chi-square distance differs from the conventional Euclidian distance by the fact that each square in the sum is weighted by a coefficient that may be different for different terms. Each coordinate is scaled by its mean value so as to give comparable weights to different coordinates. It guarantees a certain invariance of the results with respect to the divisions imposed on the data in terms of the definitions of the two characteristics, i.e. classes having the same profile may be merged without affecting the distances between property-profiles; reciprocally, properties having the same profile may be merged without affecting the distances between class-profiles. The chi-square metric necessarily implies a different definition of the norm and the scalar product in the sense that the contributions of each coordinate are also weighted.

Using the profile concept and the chi-square metric, the mapping axes (factorial axes) are obtained by maximizing the sum of the squares of the distances between the projections and the mean projection under constraints of orthonormality (orthonormality here is also to be interpreted in terms of the chi-square metric). In order to respect the real distribution of the population, each distance is weighted by the frequency of occurrence of the class (property) it relates to. Otherwise too much emphasis would be given to rare classes (properties).

The maximization can be done either with respect to the class profiles or with respect to the property profiles. In both cases it leads to an eigenvalue problem where the first eigenvalue (always equal to 1) and corresponding eigenvector are trivial. Subsequent eigenvectors may be used as mapping axes and the corresponding eigenvalues are equal to the variance of the projections on the axes. Ignoring the trivial solution, a maximum of I-1 or J-1 (whichever is the smaller of the two) eigenvectors may be obtained. At this point, a parallel with the technique of principal components, is easily formulated.

The two maximizations (either for class profiles or for property profiles) yield the same set of eigenvalues and different sets of eigenvectors, which are related one to the other. The so called transition formulas (see, e.g. LEBA77), relate the projections of both classes and properties and can be used to obtain the representation of classes and properties on the same plot (they also allow to map classes and properties not considered when solving the eigenvalue problem). Such a plot is called a simultaneous representation. In the common plot, the relative position of the elements of a set (either classes or properties) is still interpretable in terms of similarities between them. The projection of a class is, up to a constant, the weighted mean of the projection of the properties and vice-versa. If objects in a class tend to have a certain property, it may be expected that the projections of that class and of that property are close. Conversely, if the objects in a class rarely have a certain property, it can be expected that the corresponding projections come far apart. If the constant just mentioned would be equal to 1, the projection of a class would be the mean of the projections of the properties, each one weighted by the percentage of objects within the class having that property. The projection of a property would be the mean of the projections of the classes, weighted by the percentage of objects per class. This is a way to represent two different sets. Correspondence analysis tries to approach this situation, as well as possible, and in accordance with a criterion.

Two sets of coefficients - the absolute and relative contributions - can be computed and used to support the interpretation of the mappings. The absolute contribution of a row (or column) is the fraction of the total variance along a given factorial axis, which is due to that row (column). The relative contribution of a row (column) is the fraction of the total variance of a row (column) which is explained by a given factorial axis.

Correspondence analysis can also be applied to p-way contingency tables, by using incidence tables. It is then called multiple correspondence analysis, or first order correspondence analysis or also homogeneity analysis. If n objects described by p discrete features are available, an incidence table is constructed in the following way. For each feature x_j, a matrix $\underline{Z}_j$ (dimension equal to $n*I_j$) is defined where n is the total number of objects and I_j is the total number of categories that feature j assumes. Each entry is of the form

$z^j_{ik} = 1$

if object i assumes the k-th category of feature x_j

$z^j_{ik} = 0$

if object i does not assume the k-th category of feature x_j.

(5.49)

The p matrices resulting from the p features can be collected in a super matrix $\underline{Z}$:

$$\underline{Z} = (\underline{Z}_1, ..., \underline{Z}_p) \tag{5.50}$$

with n rows and the number of columns equal to the sum of the number of categories each feature has. Correspondence analysis may then be just applied to $\underline{Z}$.

In the application of correspondence analysis in chapter 6, an approach (the class-approach) was used which will now be explained. First an incidence table is constructed just as above, with each row of the table representing an object. Next, the rows representing objects from the same class are merged in a single row. The merging is done by summing the various row vectors. This results in a new table with the number of rows equal to the number of classes. At this point, correspondence analysis is applied to this new table. As a result, 2D plots are obtained with the projections of classes and features into two mapping axes. The eigenvectors vectors chosen are generally those associated with the largest eigenvalues (after ignoring the trivial eigenvector), i.e. those where the variance of the projections is larger and which show the major trends in the data.

In the factorial plots obtained, the distance between the projections of two classes is related to their similarity. A discrete feature is represented by a set of points, each one corresponding to one category of the feature. Closeness between the mapping of a class and certain categories of the features, indicates that the objects of that class tend to assume these feature values. Proximity to the origin implies an average profile. Therefore, rare profiles are projected away from the origin. This allows a simple characterization of the data.

By using classes instead of single objects in order to obtain the factorial axes, more concise plots are obtained. The first non-trivial eigenvector (the one associated with the largest non-trivial eigenvalue) is the axis that maximizes the separability (defined by the chi-square metric) between classes (represented by profiles), the second non-trivial eigenvector is the axis, orthogonal to the first eigenvector, that maximizes the separability between classes, and so on.

Single objects were also mapped into the mapping space, using the transition formulas. The robustness of the plots may be checked by mapping single objects not used in the construction of the table to which correspondence analysis was applied.

Finally, it should be stressed that correspondence analysis is derived using algebraic concepts in spite of the fact that, at some steps, the formulation might suggest a statistical component.

5.4.2. Remarks.

With incidence tables, an initial quantification of the discrete data is obtained. A category is coded as either a one or a zero depending on whether or not the object assumes a certain category in a feature. This implies that the projections of the categories on an axis, can be interpreted as a quantification of the categories. The quantification is such that when considering projections on the first non trivial factorial axis, the first eigenvalue of the corresponding correlation matrix, is maximized.

The incidence tables, as indicated above, are constructed in such a way that interactions between features are not preserved. As a result, the plots obtained do not reflect these interactions. This can be avoided by the use of incidence tables where the elementary matrices $\underline{Z}_j$ are not associated with a single feature but with combination of features. In practice, however, the number of objects available would soon be too small to guarantee stable mappings.

Since there are no appropriate restrictions and the initial coding of the data is not appropriate either, order relations, in ordered discrete features, are not used and may be lost in the plots.

5.5. Error estimation.

In chapter 6, and in order to design classifiers, the data sets available were divided evenly in a training and a test set. Leaving the test set aside in order not to compromise the error estimation, a classifier was selected, following the usual steps. At this point, its error was estimated by means of several estimation techniques.

The errors were generally estimated by three different techniques:

A) The hold-out technique.

B) The leave-one-out estimator applied to all data available.

C) Rotation technique applied to all data available and averaged over a given number of randomly generated partitions of the data, each one resulting in a training and a test set.

Also, an estimate of the variance of the error estimate, was obtained for method C. This is given by

$$(n-1)^{-1} [\sum_{i=1}^{n} \varepsilon_i^2 - \varepsilon^2] \qquad (5.51)$$

where ε is the average error estimate, ε_i is the estimate obtained with partition i, and n is the number of partitions.

The properties of the hold-out estimator are well known. Specifically, it yields pessimistically biased estimates.

As for the other two estimators, their properties have already been indicated and/or discussed in chapter 3. Nevertheless, the following aspect should be noted. They were applied to all data available. Half of this data was used for the selection of the ('best') classifier to be tested and therefore lower estimates of the error should be expected as compared to the case where all the data had not been used in order to select the 'best' classifier.

5.6. Final remarks.

This chapter has been devoted to the review and presentation of procedures and techniques for pattern recognition with discrete and/or mixed data types. The topics reviewed or presented, with more or less detail, were classifiers, feature selection, correspondence analysis and error estimation. Both the text and the references in it, should be fairly sufficient for anyone who has to design classifiers capable to handle discrete data.

Whenever a model or a technique, suited only for discrete types, is to be applied to mixed data, then the continuous features need to be discretized before use. The loss of information that is caused by the discretization, can be minimized by choosing the intervals in such a way that the number of objects falling in each one is approximately equal (LORE77).

The next chapter presents the application, of some the techniques reviewed and discussed here, to two medical data sets.

CHAPTER 6

APPLICATIONS TO TWO MEDICAL DATA SETS

This chapter presents two applications using the procedures discussed in previous chapters. The data sets studied include a set of records on head injured patients and a set of records on patients with acute chest pain. Both medical aspects and the performance of the various procedures used, are discussed.

6.1. Introduction.

Procedures of missing value estimation, of feature selection and of error estimation have been discussed in previous chapters. In this chapter their use in the analysis of two real data sets will be illustrated.

The two data sets apply to two different medical domains: severe head injury and acute chest pain. Although the domains are quite different from a technical point of view, the data sets share common characteristics, often found in medical data.

i) The occurrence of discrete features. In the first set all features are discrete; the second set is of mixed types.

ii) The occurrence of missing values. This is an unfortunate aspect in data analysis. As it will be seen, for some features, the percentage of missing values is quite high.

iii) The data sets were obtained from outside. The author was not involved in the data acquisition. This is also the real life situation. Too often the data analyst is involved in a phase where all data is already collected. In that sense, this set-up simulates an unfortunate experimental reality.

iv) For both data sets, the analysis presented here is more general than the analysis performed by the original investigator.

In the following, the technical terminology will be used, rather than the equivalent medical terminology. Each data set is a collection of medical records. A record will be referred to as a patient or an object. Each record contains information, hereafter referred to as features, considered to be relevant for diagnosis and/or prognosis. In both cases there are more than two diagnostic and/or prognostic classes possible. These will be simply referred to as classes.

6.2. Head injury.

This data set contains records of 990 head injured patients. It was supplied by Dr. Titterington [1], who has originally acquired the data and analyzed them (TITT81), in a retrospective study comparing several discriminators. The original data set contained 1000 patients, made available in the form of punched cards, 10 records per card. Apparently, during the transfer, one card was lost.

6.2.1. Material.

The data on patients with severe head injury was collected with the purpose of investigating the feasibility of predicting the recovery of individual patients (see TITT81). The ultimate clinical goal was to design procedures to be used for the allocation of limited intensive care resources to those patients with a fair chance of moderate or even good recovery.

Patients entered the study if in coma for at least 6 hours, i. e. if a minimum level of brain damage was present (see TITT81). The outcomes (the classes to assign the patients to) were expressed in terms of the patient's state, 6 months after injury. The outcomes considered were:

i) dead or vegetative (DEAD; 498)
ii) severe disability (SEV; 97).
iii) moderate disability or good recovery (MILD; 380).

Above, in brackets the mnemonics for each class and the number of patient data actually used are given. As indicated above, our version of the data set contains 990 records. However, for reasons to be given later on, 15 of these were disregarded. The sampling in this case may be considered as sampling from the universe, so that a priori class probabilities may be estimated from the number of patients per class.

Each patient record consists of six features (the mnemonics in capitals will be used throughout this text):

[1] We would like to express our gratitude to Prof. Jennett and Dr. Titterington, from the University of Glasgow (U.K.), for making this data set available to us.

1 - AGE: age of the patient;
2 - EMV: composite score of eye, motor and verbal response;
3 - MRP: motor response pattern;
4 - CHANGE: change in the neurological function over the first 24 hour;
5 - EYEIND: eye indicant;
6 - PUPIL: pupil reaction to light.

This feature set corresponds to subset III in TITT81. A brief description of these features, adapted from TITT81, follows. Feature AGE indicates the patient's age and is grouped into decades (0-9, 10-19, ..., 70+). EMV is a raw sum of three scores: the E, the M and the V scores. The E score indicates the eye opening to stimulation and is graded from 1 (nil) to 4 (normal). The M score indicates the motor response of the best limb to stimulation and is graded from 1 (nil) to 6 (normal). The V score indicates the verbal response to stimulation and is graded from 1 (nil) to 5 (normal). Feature EMV is grouped as 3, 4, 5, 6, 7, 8, 9-15. Feature MRP is an overall summary of the motor response in the four limbs and is graded from 1 (nil) to 7 (normal). Feature CHANGE is graded 1 (deteriorating), 2 (static) or 3 (improvement). Feature EYEIND is a summary of three other measures related to eye responses, and is graded 1 (bad), 2 (impaired) or 3 (good). Finally, feature PUPIL is graded from 1 (non-reacting) to 2 (reacting) and codes the pupil reaction to light. The values in the data set were the best from a series of successive tests done during the first 24 hours after the onset of coma.

All features but AGE and CHANGE, quantify a patient's reaction to stimuli. They are coded in such a way that the higher values indicate normal reactions. As for CHANGE, this feature is associated with a development towards normality and is coded such that a higher value (3), indicates improvement. All features may thus be regarded as ordered features, the coding of the categories reflecting their order.

6.2.2. Previous and present analysis.

In TITT81, a comparison of discriminant techniques applied to this data set is given. Restricting the discussion to what in TITT81 is called set III (our present data), a large number of discriminants was applied. Most of them were discriminants designed for discrete feature spaces (16). Three discriminants based on the assumption of normality were also defined, mainly for reasons of curiosity, as the author explains.

Three remarks must be made about the original analysis:

i) For this data set, no attempt was made at selecting features and testing discriminants in subspaces.

ii) No systematic procedure for uncovering the structure of the feature space was used.

iii) No attempts at multi-level discrimination were made.

In the present analysis, apart from visualizing the data by means of mappings, various single level and multi-level decision strategies were designed, with systematic feature selection and structure selection at the various branch points. Both the basic model and the logistic model, taking the ordering properties of the features into account, were used for discrimination.

6.2.3. Training and test sets. Missing values. A-priori probabilities.

Since the number of objects in the data set is fairly large and the number of categories in all the features are small enough considering the total number of objects, no compression of categories for any feature was done.

Feature	All data	Training set	Test set
Age	0.0 %	0.0 %	0.0 %
EMV	5.1 %	5.6 %	4.5 %
MRP	3.7 %	4.2 %	3.3 %
Change	26.3 %	26.4 %	26.1 %
Eyeind	20.7 %	22.0 %	19.4 %
Pupils	2.4 %	2.6 %	2.4 %

Table 6.1: Missing data for severely head injured patients.

The entire data set was divided into a training set and a test set with approximately the same number of objects per class in each set. The test set was used exclusively for error estimation. Both sets are the same as in TITT81 (this information was coded in the punched cards), with the exception of the 10 lost records.

There are missing values in the data set. Table 6.1 indicates the percentages, per feature, for the entire data, the training and the test sets. It can be seen that features CHANGE and EYEIND have, by far, the largest percentages of missing values. The missing values are approximately equally distributed among the training and test sets. All objects were kept for further analysis, except those in the training set with 3 or more values missing. A total of 15 objects were thus discarded. The remaining objects with missing values were completed by estimating values for those missing. The classification approach, with a Bayes classifier assuming independence between features (see chapter 4), was the estimation technique used. The estimates were obtained separately for the training and test sets. For objects in the training set, the estimation was done for each class separately. By separating training and test sets, the independence between these sets was preserved.

A priori class probabilities were estimated from the proportions of objects in the original training set.

6.2.4. Mappings.

Since the data set is fairly large and the number of features is comparatively small, it was decided to first inspect plots of the training set, taking all features into consideration.

Using correspondence analysis, factorial plots were obtained. Fig. 6.1 shows the projections of the three classes onto the plane spanned by the first two factorial axes, fig. 6.2 the projections of the features and fig. 6.3 the projections of the objects. The factorial axes were obtained using the class-approach (see chapter 5). Since there are three classes, there are only two non-trivial factorial axes; the first axis accounts for 93 % of the total variance.

Objects from class MILD tend to project on one side of the first axis. Objects from class DEAD tend to project on the other side and objects from class SEV tend to be in the middle. Nevertheless, the overlap is significant, even between classes MILD and DEAD.

In fig. 6.2, it can be seen that all features but AGE and CHANGE, project along the first axis in such a way that categories corresponding to normal measurements are close to objects from class MILD, and 'abnormal' categories are close to objects from class DEAD. The projection of feature AGE, indicates that young patients tend to be classified as MILD, whereas old patients tend to be classified as DEAD. As for feature CHANGE, the category 'deteriorating condition' projects more closely to the DEAD class, whereas the other two categories project closer to the MILD class.

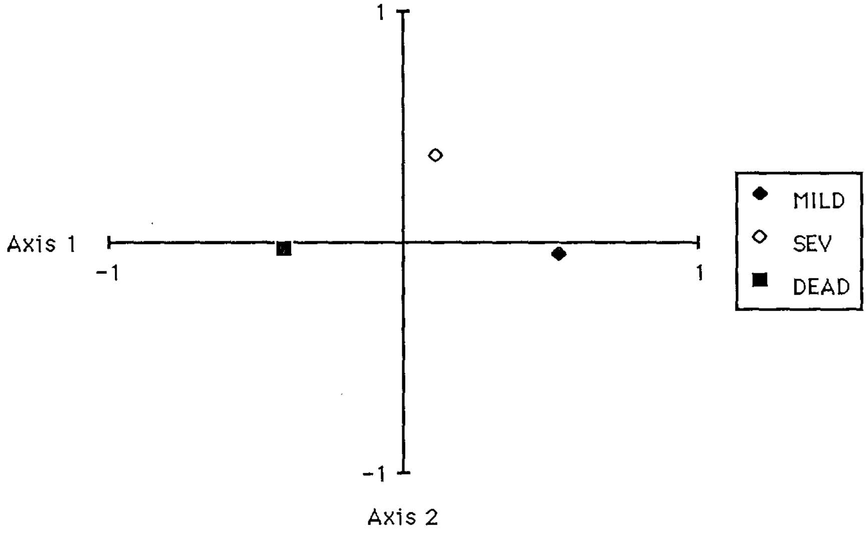


Fig. 6.1: Mapping of the three classes onto the two factorial axes.

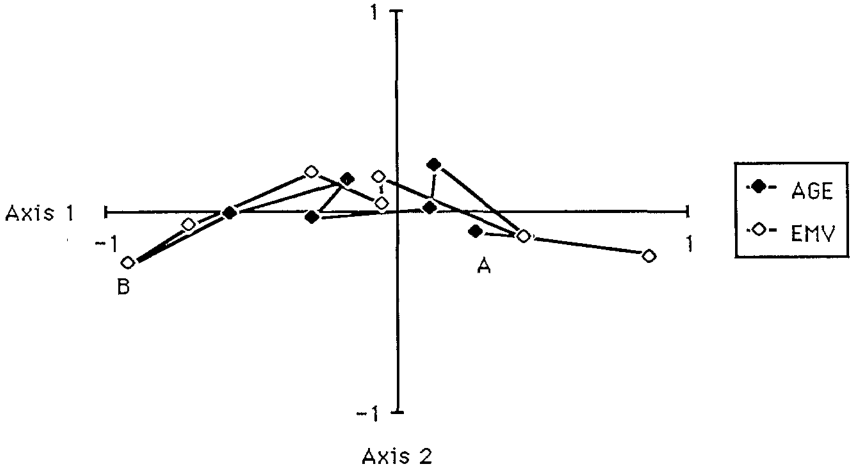


Fig. 6.2: Mapping of the six features onto the two factorial axes. Each feature is represented by a set of points (each one associated with a category), joined by straight lines. The point closer to character A, is related to the age range 0 - 9. By moving away from this point and along its associated segment, the age is spanned in an orderly manner. The same applies to the other features but now the movement is from 'abnormality' to 'normality'. The points associated with 'abnormality' are indicated by the characters B (EMV), C (MRP), D (CHANGE), E (EYEIND) and F (PUPILS). See also the two following pages.

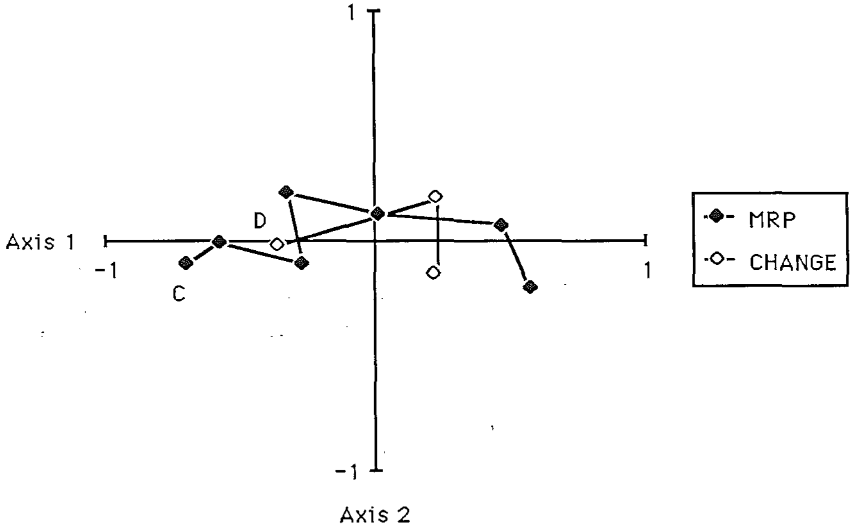


Fig. 6.2: Continuation.

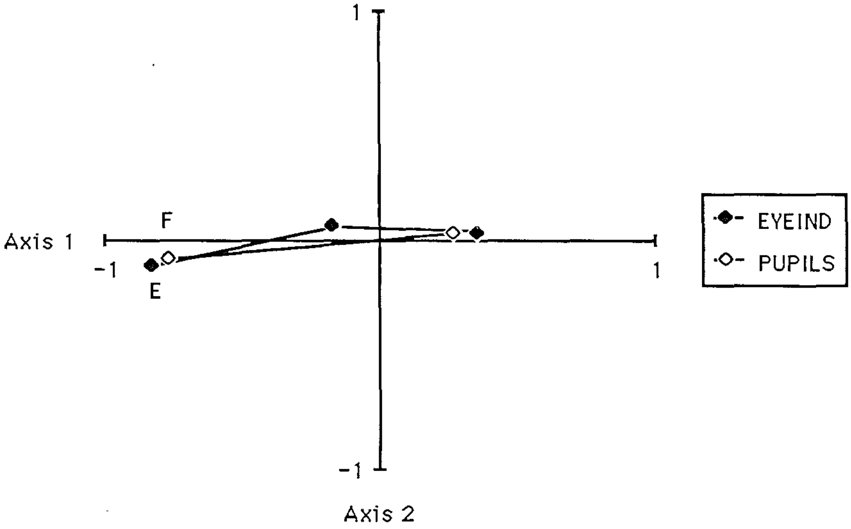


Fig. 6.2: Continuation.

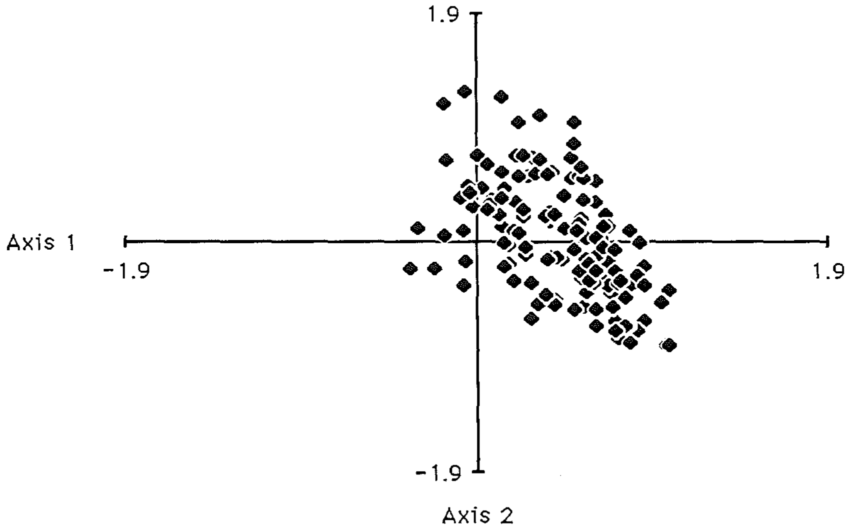


Fig. 6.3: Mapping of objects from class MILD onto the two factorial axes.

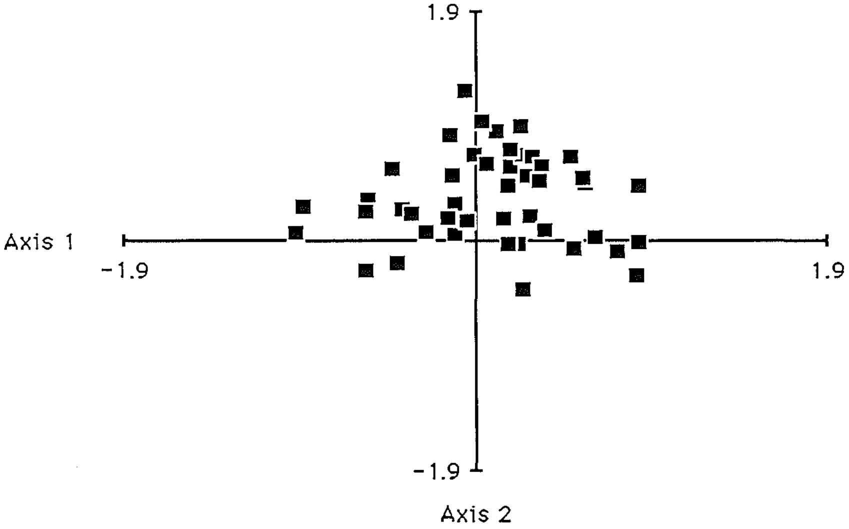


Fig. 6.3 (continuation): Mapping of objects from class SEV onto the two factorial axes.

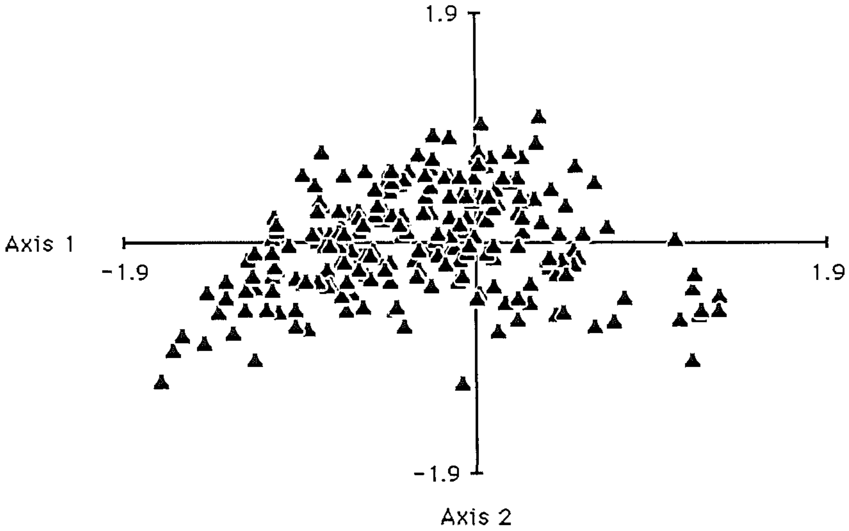


Fig. 6.3 (continuation): Mapping of objects from class DEAD onto the two factorial axes.

Order relations are not taken into account by correspondence analysis. Nevertheless, the projections of the categories onto the first axis, follow their natural ordering.

6.2.5. A single level classifier.

It was decided to try first a single level classification strategy. The corresponding decision tree is shown in fig. 6.4. In TITT81, the same decision tree is used, although without feature or structure selection. Therefore, a number of feature selection algorithms, able to test several structures, were applied. The results are presented in table 6.2. Both the ranking of the features and the structures selected, are shown.

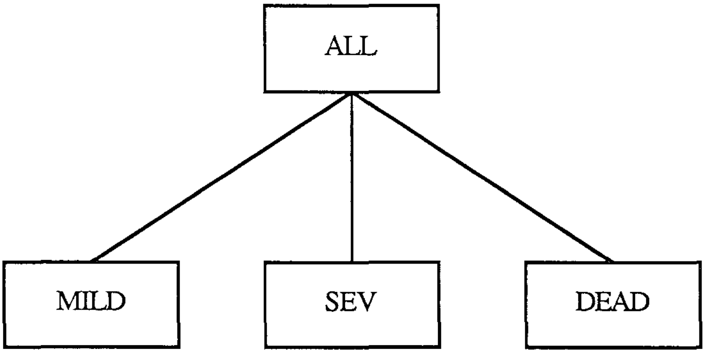


Fig. 6.4: A one level decision tree.

Except with algorithms that use selection criterion A, all 6 features were selected. The structures selected consisted of groups of at most two non-independent features. Nevertheless, some structures required a number of estimates that was large in view of the number of objects available in class SEV (this was not the case for the classes MILD and DEAD).

Even if only 3 features are considered, there are large differences in the feature ordering and structures selected (an exception are the results of the algorithms using selection criterion A). Since the number of objects available is large (except for class SEV; nevertheless, since the prior probability of this class is low, it should not influence the results too much), suggesting that robust estimates may

Algorithm	Ranking	Structure
Search strategy B1 Selection criterion A MLK estimates	1 - MRP 2 - PUPILS 3 - AGE	1 2 - 3
Search strategy B1 Selection criterion A Bayesian estimates	1 - MRP 2 - PUPILS 3 - AGE 4 - EYEIND 5 - EMV	1 - 4 2 - 3 5
Search strategy B1 Selection criterion B MLK estimates	1 - EYEIND 2 - AGE 3 - EMV 4 - PUPILS 5 - CHANGE 6 - MRP	1 - 2 3 - 5 4 6
Search strategy B1 Selection criterion B Bayesian estimates	1 - EYEIND 2 - MRP 3 - AGE 4 - PUPILS 5 - EMV 6 - CHANGE	1 2 3 4 5 - 6
Search strategy B2 Selection criterion C Conf. lim. 99 %	1 - EYEIND 2 - PUPILS 3 - EMV 4 - MRP 5 - AGE 6 - CHANGE	1 2 3 4 5 6

Table 6.2: This table presents the results of the feature selection algorithms (see chapter 5) for the decision tree in fig. 6.4. The first column indicates the algorithm used (see chapter 5), the second column indicates the feature ranking and the third column the structure chosen: e.g. the first algorithm selects a structure with two non independent sets of features (MRP, PUPILS and AGE).

be expected, we may conclude that the discriminating power of some of the features (e.g. MRP and EYEIND) is very similar.

In order to evaluate the feature combinations and the structures selected in the previous step, several discriminants were used. Also, several error estimators were applied. The discriminants are:

1) A Bayes classifier, assuming the basic model (see chapter 5) for the conditional probabilities, given the class, the parameters of which were estimated using either maximum likelihood or Bayesian approaches. Briefly, the classifiers and/or models used by the feature selection algorithms, were tested.

2) A classifier based on the logistic model, but adapted in such a way as to take the ordering properties of all features into account (see chapter 5). Such a model requires the estimation of $(N_c - 1) * (N_f + 1)$ parameters (N_c is the number of classes and N_f is the number of features), which makes it very attractive (see chapter 5).

Applying one type of classifier, the errors may in principle be estimated in different ways (see chapter 5):

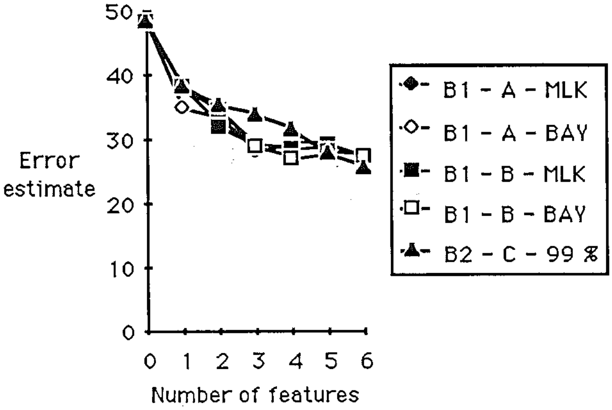
a) the hold-out estimator.

b) the leave-one-out estimator applied to all data available.

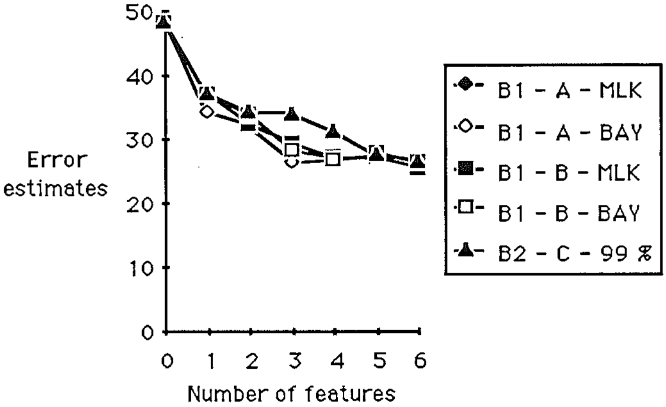
c) rotation technique applied to all data available and averaged over 20 randomly obtained partitions, each one resulting in a test set with approximately 1/3 of the total number of objects.

All these estimators were applied to the basic model. Using the discriminator based on the logistic model, only estimator a) was used. Computational, it would be very heavy if estimators b) and c) were used because an iterative procedure requiring inversion of matrices, is required in order to estimate the logistic model (see chapter 5).

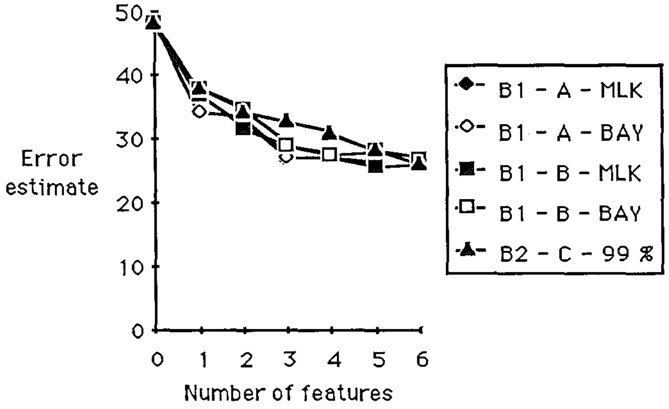
The results are presented in fig. 6.5, where the chart number and lettering reflects the combination of classifier and error estimation technique used. Fig. 6.5 shows the error estimates obtained as functions of the number of features used and according to the ranking, and in case of the basic model, to the structure selected. With 0 features, the estimates shown were obtained by using only the maximum likelihood estimates of the a-priori probabilities. Features were progressively added, in accordance with the ranking produced by the corresponding feature selection algorithm. Each chart has five curves, each one related to a feature selection algorithm.



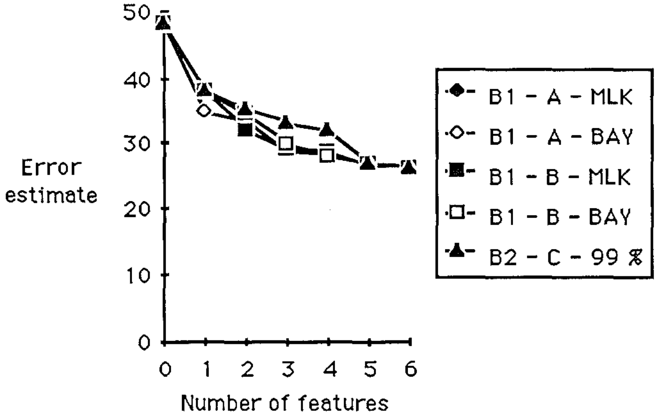
1.A (Fig. 6.5)



1.B (Fig. 6.5)



1.C



2.A

Fig. 6.5: Error estimates for the decision tree in fig. 6.4. In every chart, each error function is associated with a feature ranking produced by a feature selection algorithm. Each chart is related both to an error estimation technique and to a classifier design. The two symbols under each chart indicate the estimator (letter) and the classifier (number) used. See text, for the meaning of the symbols under each chart.

An analysis of fig. 6.5 reveals the following points:

i) At least one feature contributes significantly to a decrease in the error rate.

ii) The rate at which the error decreases, reduces with the number of features, and the error is more or less constant after the third feature, suggesting that 3 or 4 features are sufficient.

iii) The different error estimators yield very similar results. In view of the large number of objects, this behaviour is to be expected.

iv) Both the Bayes and logistic classifiers, yield similar results. Since the logistic classifier requires fewer estimates, it should be preferred.

The estimated standard deviation of the error estimates, obtained with the averaged rotation technique, ranges between 1.2% and 2.5%. The confusion matrices, reveal that very few objects from class SEV were correctly classified (see fig. 6.6 for an example of a confusion matrix).

	MILD	SEV	DEAD
MILD	166	7	25
SEV	24	2	22
DEAD	40	5	199

Fig. 6.6: Typical confusion matrix for the decision tree in fig. 6.4. The rows indicate the true class membership, whereas the columns indicate the computer classification.

Fig. 6.5 shows that in terms of results, all the algorithms yield a combination of features and structure that give similar results when tested. Nevertheless, the charts in fig. 6.5 show that the curves related to algorithm B2-C-99% are upper bounds to the other curves. More steps are required to achieve results similar to the others. The explanation of this behaviour must lie with the selection criterion used by this algorithm. It is not directly related to the error rate (as the other selection criteria are) and weak assumptions about the underlying probability functions (see chapter 5) are made. All stopping criteria but the ones used by algorithms B1-A-MLK and B2-C-99%, allowed the selection of features which were not necessary (see fig. 6.5), and structures that required a relatively large number of parameters.

Finally, the results presented in fig. 6.5 and those presented in TITT81, are very similar. The error rates (for the full dimensionality) presented in TITT81 range from 23.2% to 33.8% depending on the classifier used. The logistic model used here gave similar results to the one in TITT81. Nevertheless, it was able to achieve this with fewer parameters to estimate.

6.2.6. A two level decision tree.

In view of the results obtained so far, it was decided to test another classification strategy. The corresponding decision tree is shown in fig. 6.7.

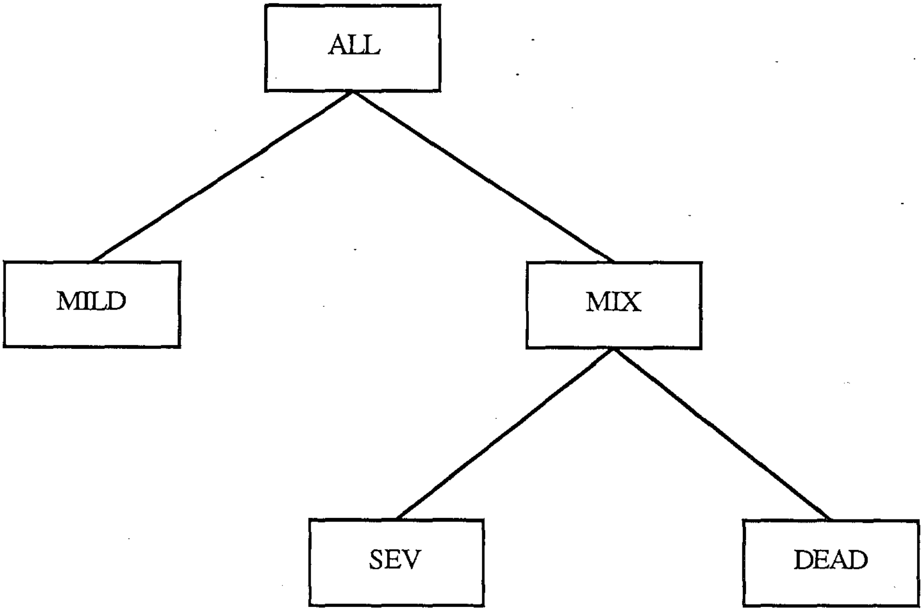


Fig. 6.7: A two levels decision tree.

As it may be seen, this is a two level decision tree, not explored by Titterington et al.. At the first level of decision, class MILD is separated from the other two

Algorithm	Ranking	Structure
Search strategy B1 Selection criterion A MLK estimates	1 - EMV 2 - CHANGE 3 - EYEIND 4 - PUPILS 5 - AGE	1 - 2 - 3 4 5
Search strategy B1 Selection criterion A Bayesian estimates	1 - EMV 2 - CHANGE 3 - AGE 4 - EYEIND 5 - PUPILS	1 - 2 3 4 - 5
Search strategy B1 Selection criterion B MLK estimates	1 - EMV 2 - AGE 3 - PUPILS 4 - EYEIND 5 - MRP 6 - CHANGE	1 - 2 3 4 5 - 6
Search strategy B1 Selection criterion B Bayesian estimates	1 - EMV 2 - MRP 3 - AGE 4 - PUPILS 5 - CHANGE 6 - EYEIND	1 - 5 2 3 4 6
Search strategy B2 Selection criterion C Conf. lim. 99 %	1 - EYEIND 2 - PUPILS 3 - EMV 4 - MRP 5 - AGE 6 - CHANGE	1 2 3 4 5 6

Table 6.3: This table shows the results of the feature selection algorithms when applied to the root node (node ALL) of the tree in fig. 6.7 (see also the caption in table 6.2).

(MIX). The reason for merging the two classes SEV and DEAD in the first level is based on the mappings shown in figs. 6.1 and 6.3. At a second level, the separation between classes SEV and DEAD is attempted. The two decision levels were analyzed separately.

Feature selection algorithms were applied to node ALL yielding the results shown in table 6.3. Just as with the one level decision tree, there are large differences between the rankings and structures selected. The comments written above, apply here as well. Feature EMV, was always ranked first by the algorithms using an error related selection criterion. At least in one case (B1-A-MLK), a structure with a group of three non-independent features, was selected.

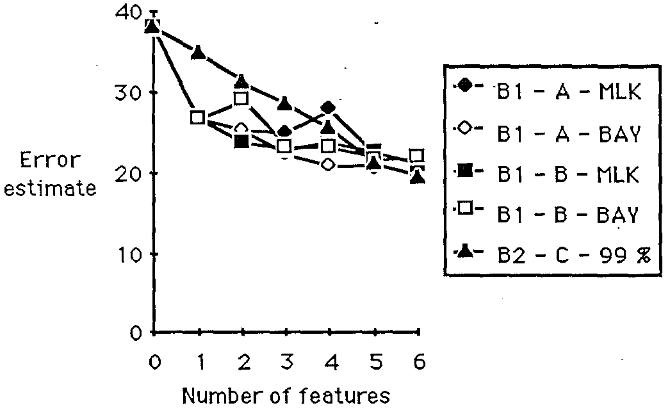
For error evaluation, Bayes classifiers assuming the basic model and logistic classifiers were used. The error estimators used were the same as those applied to the single level decision tree. The results are shown in fig. 6.8.

The charts in fig. 6.8 show again that the curves related to algorithm B2-C-99% are upper bounds to the other curves. This has already been commented on, in another case (see above). It is also worthwhile noting that this algorithm makes a bad selection already at the first step.

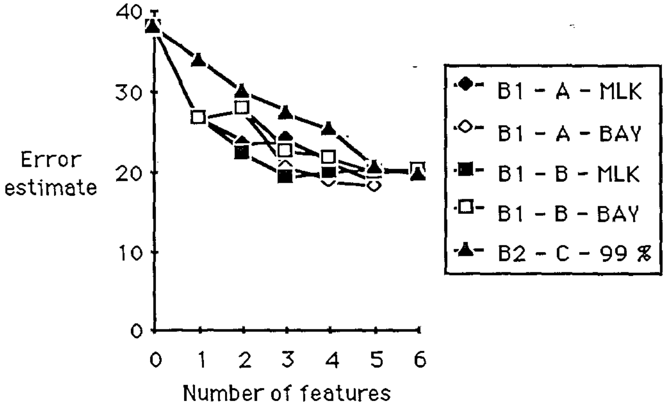
With feature EMV alone, an error rate of 26.5 % is achieved. The error tends to stabilize after the third feature at a value of 21 %. This value is thus a lower bound for the total tree. This lower bound will be actually attained if an error free classifier can be designed at node MIX. The estimated standard deviation of the error estimates obtained with the averaged rotation technique, ranges between 1.3% and 2.4%. The performances of the Bayes classifiers based on the basic model and of the logistic classifiers, are again similar.

Next, the classification problem at node MIX was analyzed using the same procedures as for node ALL. The results (not shown here), are as follows. Feature EYEIND always ranks first and the stopping criteria of all but algorithm B1-A-MLK, allow structures requiring a large number of parameters, to be selected. There is a clear failure to discriminate between classes SEV and DEAD. There is no decrease in the error rate if features are used. An analysis of the confusion matrices shows that either all objects are classified as DEAD (in which case the minimum error is attained, since this corresponds to assignment according to the highest prior probabilities alone), or there are errors both in the classification of objects from SEV and from DEAD.

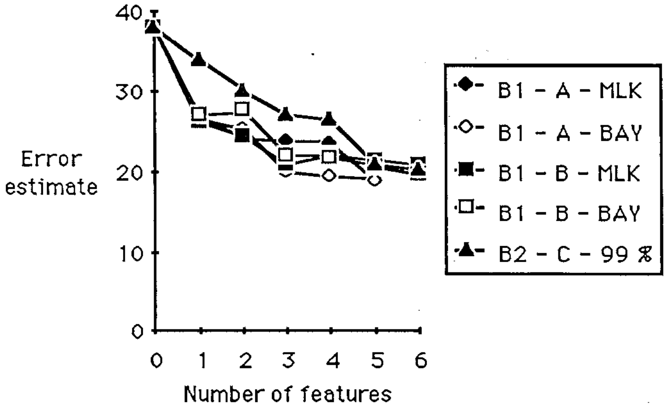
An estimate of the error rate associated with the decision tree in fig. 6.7, can be readily composed by summing the error rate at node ALL (21 %) with the error rate at node MIX (16.7 %) multiplied by the sum of the a-priori probabilities of classes SEV and DEAD (.56). The result of these operations is approximately 30%. The one level decision tree performed better.



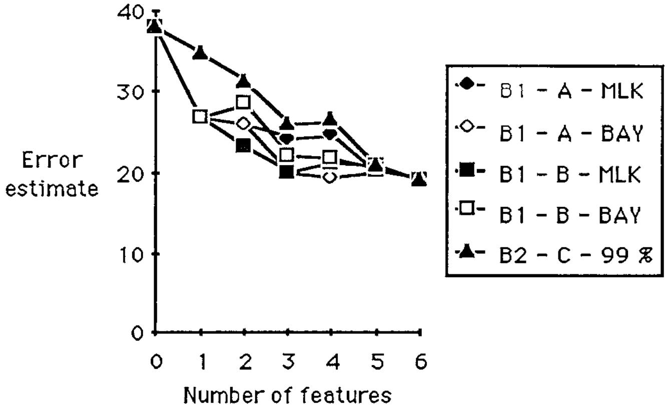
1.A (Fig. 6.8)



1.B (Fig. 6.8)



1.C



2.A

Fig. 6.8: Error estimates for node ALL of the decision tree in fig. 6.6. See also the caption in fig. 6.5.

6.2.7. Discussion.

The study in TITT81 on head injured patients, has been expanded here by the application of feature selection algorithms and by the use of classifiers not considered by Titterington. Also both a single level and a two levels decision tree were used. The performance of the classifiers has been assessed exclusively by error estimates. When comparing these results with those obtained by Titterington (TITT81), it can be concluded that they are similar.

The possibility of separation between the classes MILD and DEAD, and between the classes MILD and SEV, was also analyzed. As above, the research involved feature selection and error estimation. From the results obtained (not shown here) it can be concluded that:

- for MILD versus DEAD, an error rate slightly less than 20 %, can be achieved with 2 or 3 features.

- for MILD versus SEV, the best error rate occurs when the classification is based solely on a-priori probabilities.

MILD versus DEAD, MILD versus SEV and SEV versus DEAD, are all the possible combinations of two classes. The error rates obtained, indicate that SEV is strongly overlapped with both MILD and DEAD and that even the overlap between MILD and DEAD is significant. Taking into account the fact that the data set is fairly large, that the number of parameters required by most of the classifiers tested compare favourably with the number of objects used to estimate them (implying that the error estimates are stable and subsequent conclusions about overlaps between classes may be generalized), it can be concluded that the task of discriminating between the three classes with a smaller error, requires the definition and use of new features.

The various feature selection algorithms used yielded different rankings and structures for all decision levels studied. Whenever the number of objects was large (e.g. MILD versus MIX), these differences were interpreted in terms of features and structures with similar discriminating power. For other cases (e.g. SEV versus DEAD), estimation errors have also to be taken into account. The stopping criteria generally failed to act when they should. Features were selected beyond the point of stabilization of the error rate, as shown by the evaluation of some classifiers. Also, structures were selected requiring a large number of estimates. Since this behaviour also occurred with the acute chest pain data base, it will be analyzed later on in this chapter.

6.3. Acute chest pain.

This data set contains records of 300 patients with acute chest pain. It was supplied by Dr. F. T. de Dombal [2] who acquired and analyzed them (DOMB83).

6.3.1. Material.

The data on patients with acute chest pain was collected with the purpose of studying the possibilities of defining a set of procedures to assist a physician in making decisions based solely on clinical data.

Acute chest pain may originate from the heart, from the lungs, or from elsewhere. If the pain is of myocardial origin, it is important to predict whether or not the patient will suffer from further complications (possibly causing death), requiring treatment in an intensive care unit (see DOMB83).

Although the diagnosis generally presents little difficulty if ECG and enzyme data are available (see DOMB83), in remote locations (where an electrocardiograph and suitable laboratory equipment may not be available) or at an early stage (when the typical changes are not yet fully developed), a first diagnosis may have to be made based solely on the patient symptoms and signs (i.e. clinical data).

The data set consists of 300 records with the following (confirmed) diagnoses and outcomes (survival or death):

a) 100 cases with non-specific chest pain, i.e. with the source of the pain not cardiovascular.

b) 42 cases with myocardial infarction (M.I.) who survived with no further problems.

c) 42 cases with M.I. who survived but developed further problems (arrhythmias, 22 cases, pump failures and arrhythmias, 12 cases, pump failures, 8 cases).

[2] We would like to express our gratitude to Dr. F. T. de Dombal, from St. James's Hospital Leeds (U.K.), for making this data set available to us. Also, Ms. Biba Unwin, from Dr. de Dombal's group, who retrieved and organized the data, is hereby gratefully acknowledged.

d) 16 cases with M.I. who developed further problems (arrhythmias, 8 cases, pump failures and arrhythmias, 6 cases, pump failures, 2 cases), and died.

e) 90 cases with angina pectoris (A.P.) who survived with no further problems.

f) 10 cases with A.P. that survived but developed further problems (arrhythmias, 4 cases, pump failures and arrhythmias, 2 cases, pump failures, 2 cases, other problems, 2 cases).

Four classes were defined among these groups:

i) non-specific chest pain, a) above (NON; 100 cases; 0.25).

ii) M.I. with no further problems, b) above (MIN; 42 cases; 0.20).

iii) M.I. with further problems, c) and d) above (MIP; 58 cases; 0.30).

iv) A.P. with or without further problems, e) and f) above (AP; 100 cases; 0.25).

In parentheses the mnemonics for each class, the number of records in the data set and estimates of the prior probabilities (see below), are given.

The sampling may be considered as sampling per class, the classes being NON, AP and MI (MIN and MIP). The a-priori probabilities estimates shown above were indicated to us and reflect the situation at an emergency department in Leeds (U.K.).

Each patient record contains 34 features. The mnemonics (in capitals) that will be used throughout this text, and the meaning of the features are as follows [3]:

1 - ONSET; pain onset, either sudden or gradual.
2 - OCCUR; pain occurrence, either continuous or intermittent.
3 - SEV; pain severity, either moderate or severe.
4 - DURAT; pain duration.
5 - TYPE; type of the pain.
6 - SITE; site of the pain.
7 - RAD; radiation.
8 - PROG; progress.
9 - DISP; dispnoea.
10 - SPUTUM; sputum.

[3] Appendix E presents a description of these features.

11 - NAUSEA; nausea.
12 - VOMIT; vomiting.
13 - SWEAT; sweat.
14 - COLD; whether the patient feels cold/clammy.
15 - COUGH; cough.
16 - MOOD; the patient's mood.
17 - COLOUR; the colour of the patient.
18 - RMOV; respiratory movements.
19 - JVP; Jugular venous pressure.
20 - HSOUND; heart sounds.
21 - CSOUND; chest sounds.
22 - PPAIN; previous chest pain.
23 - PILL; previous relevant illness.
24 - NUMB; numbness.
25 - APPET; appetite.
26 - TENDER; tenderness.
27 - BOWELS; the functioning of the bowels.
28 - TEMP; temperature.
29 - PERC; percussion.
30 - AGE; age of the patient.
31 - PULSE; pulse rate.
32 - BPSYS; systolic blood pressure.
33 - BPDIA; diastolic blood pressure.
34 - RRATE; respiratory rate.

Some comments are now due on this set of features. Firstly, there are discrete as well as continuous features (features AGE, PULSE, BPSYS, BPDIA and RRATE, are the continuous features). Secondly, some of the discrete features have a large number of categories (e.g. TYPE, SITE, RAD, PILL). Thirdly, for some discrete features, there is a set of basic categories together with other categories which are combinations of some of these basic categories. Finally, not all records are complete, i.e. there are missing values.

6.3.2. Present analysis.

In DOMB83, results are presented concerning the analysis of a similar data set. Unfortunately the author does not specify the steps leading to the results. In the present analysis, various multi-level decision strategies were designed, with systematic feature selection and structure selection at the various branch points. Single level decision strategies were not used because of the natural hierarchy in this data. Faced with a patient with acute chest pain, the first and fundamental question to answer is whether the source of the pain is in the heart or elsewhere. Following this, urgent decisions concern only those patients where the source of the pain is located in the heart. In the following sub-sections the various steps of the analysis are indicated.

6.3.3. Initial preparation.

As indicated in section 6.3.1, for some of the discrete features, the number of categories is large as compared to the number of objects. If they are not compressed, i.e. merged into a smaller number of categories, the estimation of the selection criteria in feature selection algorithms, the estimation of classifiers, etc., is going to be seriously compromised in view of the limited number of objects available. On the other hand, the merging brings with it a loss of information. Nevertheless, since this information cannot be properly used (due to the limited number of objects), it was decided to merge some categories in some features. Depending on the feature, one or more heuristic criteria were used. Features with more than 9 categories, were considered for merging.

Feature TYPE, had initially 34 different categories. It was decided to keep the basic categories which have fairly large numbers of objects, merge into these categories others that were combinations involving them, and create another category (others) where several categories are merged. The result of this transformation was another version of feature TYPE with 5 categories, as follows; heavy, tight, sharp, ache and other.

Feature SITE had initially 12 different categories. The transformation applied here, resulted in a new feature SITE with 7 categories as follows; central, across, left side, across and left side, chest, central and left side, other. Essentially, the same criteria governed the merging here, with the exception that some of the categories that are combination of the basic categories were retained (across and left side, central and left side) in view of the relatively large number of objects in them.

Feature RAD had initially 41 different categories. Of these categories, one indicates the absence of radiation and the remaining 40 indicate to where the pain radiated. There is an unbalance between categories since two different entities were coded here: the occurrence and the location of radiation, the second one being conditional on the first one. In order to correct this unbalance and at the same time reduce the number of categories, it was decided to just consider the presence or absence of radiation.

Feature MOOD had initially 10 different categories. It was transformed into another feature MOOD, with 5 categories, as follows; normal, anxious, lethargic, distressed and other. The criteria used here, were essentially those applied to feature TYPE.

Feature PILL had initially 23 categories. Of these categories, one indicates the absence of any previous relevant illness, whereas the remaining 22 specify the relevant illnesses. After an analysis of the distribution of the objects among these categories, it was decided to merge the 22 categories specifying the relevant illnesses into a single category, so that only the information on presence or absence of previous illnesses was retained.

An analysis of the distribution of objects among the categories of some of the other discrete features shows, that for some features some of the categories have very few objects. This is an indication that the condition to which the category relates is rare. This implies that, if they are kept, these categories can have a disturbing effect in the selection of structures. Whenever a condition of non independence is established between two or more features, the number of parameters that require an estimate is equal to the product of the number of categories of the features, minus 1. Consequently, a rare category can bring a substantial increase in the estimates required, without any substantial gain in discrimination ability. Therefore, it was also decided to merge 'rare' categories into categories with a larger number of objects. The features affected by this transformation on account of the reasons just given, were features DURAT, COLOUR, BOWELS, TEMP and CSOUND. In feature DURAT, category 'more than 1 week' was merged with category 'between 24 hours and 1 week'. In feature COLOUR, categories 'cyanosed and pale' and 'cyanosed' were merged with category 'pale'. In feature BOWELS, categories 'diarrhoea' and 'constipated' were merged into a new category which may be called 'abnormal'. In feature TEMP, categories 'below normal' and 'above normal' were merged into an 'abnormal' category. In feature CSOUND, 6 categories indicating abnormal chest sounds were merged into a single category.

Also the number of categories in feature PROG was reduced from 4 to 3 by dropping category 'varied' (just one patient had its progress described as 'varied'). This was done by considering it to be a missing value since it was difficult to choose one of the remaining categories (better, same and worse) for merging. Finally the number of categories of feature COUGH was reduced from 5 to 4, after an error was found in the coding of the data.

Feature PERC was disregarded because an analysis of the distribution of objects among its categories, revealed that all but 5 objects were either missing (18 objects) or scored in one particular category. Hence, no discriminatory power could be expected from this feature.

After this initial transformation of the data, the entire data set was divided into a training set and a test set with the same number of objects for each class.

There are missing values in the data set. Table 6.4 indicates the percentages per feature for the entire data, the training and the test sets. Also indicated there are the type of variable (continuous or discrete) which best represents the feature it relates to, and the number of categories of the initial and transformed discrete features.

Looking at the percentages of missing values per feature, it can be seen that some of them are very large. Since good estimates for replacements are not to be expected when there is a large percentage of those missing, it was decided to

Feature	Type of Var.	Categories	Red. Cat.	Missing	Miss L	Miss T
Onset	D	2	2	13%	14%	12%
Occur	D	2	2	12%	11%	13%
Sev	D	2	2	16%	17%	15%
Durat	D	7	6	1%	1%	1%
Type	D	34	5	23%	25%	21%
Site	D	12	7	1%	1%	0%
Rad	D	41	2	4%	3%	5%
Prog	D	4	3	8%	8%	9%

Table 6.4: Missing data for acute chest pain patients. Also included are the feature's types (continuous or discrete) and the number of categories of the discrete features, both originally and after merging of categories. See continuation.

Table 6.4: Continuation.

Disp	D	5	5	3%	5%	1%
Sputum	D	2	2	21%	23%	18%
Nausea	D	2	2	18%	16%	19%
Vomit	D	2	2	10%	12%	7%
Sweat	D	2	2	7%	7%	7%
Cold	D	2	2	21%	21%	21%
Cough	D	5	4	14%	12%	12%
Mood	D	10	5	7%	7%	7%
Colour	D	5	3	6%	4%	4%

Rmov	D	2	2	3%	2%	2%
JVP	D	2	2	13%	12%	12%
Hsound	D	2	2	3%	4%	3%
Csound	D	7	2	1%	1%	2%
Ppain	D	2	2	3%	2%	5%
Pill	D	23	2	8%	5%	11%
Numb	D	2	2	35%	-	-
Appet	D	2	2	26%	-	-
Tender	D	2	2	82%	-	-

Table 6.4: Continuation.

Table 6.4: Continuation.

Bowels	D	3	2	26%	-	-
Temp	D	3	2	33%	-	-
Perc	D	3	3	6%	-	-
Age	C	-	-	1%	0%	1%
Pulse	C	-	-	7%	7%	7%
BPsys	C	-	-	9%	9%	9%
BPdia	C	-	-	9%	9%	10%
Rrate	C	-	-	66%	-	-

discard all features with more than 25% of missing values. This decision affected features NUMB, TENDER, RRATE, APPET, BOWELS and TEMP. All subjects were kept, since all of them had less than 50% of values missing.

For all remaining missing values, replacements were estimated. The estimation was done separately for the discrete and the continuous features. The classification approach with a Bayes classifier assuming independence between features (see chapter 4) was the estimation technique used for the discrete features. The nearest neighbours approach with 3 neighbours and the features normalized so as to have unit standard deviation (see chapter 4), was the technique used for the continuous features. In both cases, the estimates were obtained separately for the training and test sets. For objects in the training set, the estimation was done separately per class.

6.3.4. Feature selection.

After some features were eliminated due to the existence of a large number of missing values, there remain still 27 features, 23 discrete features and 4 continuous features. A strategy is required in order to deal with this large number of features and the complexity it introduces. In the following, the strategy is explained and justified.

Following a classical approach in pattern recognition, subsets of features were defined and feature selection was performed on these subsets. The partial results were then assembled into a 'final' feature selection.

The subsets defined were:

1 - a subset whose members are the continuous features, i.e. AGE, PULSE, BPSYS and BPDIA.

2 - a subset whose members are features DISP, SPUTUM, NAUSEA, VOMIT, SWEAT, COLD, RMOV, JVP, HSOUND, COUGH, MOOD, COLOUR and CSOUND.

3 - a subset composed of features ONSET, OCCUR, SEV, DURAT, TYPE, SITE, RAD and PROG.

4 - a subset whose members are the remaining two features: PPAIN and PILL.

The rationale for this subdivision is the following. First, a division was made with respect to the nature of the features. In this way, the discriminating power of continuous and discrete features can be assessed separately with algorithms suited to the variables in question.

A second division concerns the discrete features only. Specifically, three subsets of discrete features were established. The first subset groups all the features that 'describe' the patient but are not related to the pain itself. The second subset groups all the features that describe the pain. Finally, the third subset includes two features which are related to the patient history.

For the continuous features, feature selection was achieved through the application of stepwise linear discriminant analysis as described by Cooley and Lohnes (COOL71). This starts with a 'best' feature and sequentially adds those features which have the largest additional separating power as expressed by the F-ratio's. The procedure, as implemented in ISPAHAN (GELS80), allows the control of F-values thresholds for inclusion and removal from the selected set of features.

The algorithms applied for the selection of discrete features were those applied with the head injured patients data set, with the exception of the algorithm that employed selection criterion C. One reason justifies this exclusion. The implementation of the algorithm requires that objects are sampled from the universe and this was, with the present data set, not generally the case.

A final remark concerns the selection of the subsets of discrete features. A maximum of five features was always requested (this limit is immaterial as far as the last subset is concerned).

The various partial selections yielded sets of variables that could include both discrete and continuous features. For the final selection, only algorithms suitable for discrete data were applied. Specifically, the same four algorithms used in the partial selection of discrete features were again used here. If continuous features were involved, these were first discretized. The discretization was done in such a way as to obtain a uniform histogram.

Whenever a continuous feature was consistently ranked best in the final selection, the selection was repeated using only the discrete features. These reruns were done in view of the use of the location model. As explained in chapter 5, this model is suited for mixed feature types. It decomposes the joint probability of continuous and discrete features, into the product of the marginal probability of the discrete features and the conditional probability of the continuous features given the discrete features. The former is then modeled using the basic model. Therefore, and for the marginal probability, a 'best' features combination and structure were selected appropriately.

6.3.5. Classifiers and error evaluation.

In order to evaluate the discriminating power of the feature combinations and, for some cases, the structures selected, several discriminators and error estimators were applied. The discriminators were:

1) Bayes classifiers assuming the basic model just as was done for the head injury data set.

2) Classifiers based on the logistic model. This model can work with discrete, continuous or mixed feature types without requiring any adaptation as the basic model does. Nevertheless, it was considered for use only with mixed data or whenever there was an ordered discrete feature. Each continuous feature accounts for N_c - 1 parameters in the model, N_c being the number of classes (see chapter 5).

3) Classifiers based on the location model. This model can only be applied to mixed data types (see chapter 5).

The error estimators used were those which were also applied to the head injured patients data set. Estimators other than the hold-out technique were used only if objects belonging to a group to be discriminated (a group is a collection of subjects belonging to one or more of the classes defined above: NON, MIN, MIP, AP), could be considered as sampled from the universe of objects associated with the group. The way these techniques were implemented requires that this condition be satisfied. Otherwise, the results obtained would not be consistent. For logistic and location classifiers, only the hold-out method was applied, in view of the long computation times required by the other methods. As already explained (see chapter 5), the logistic and location models require an iterative procedure (logistic model) involving a matrix inversion, in order to estimate their parameters.

6.3.6. NON versus CARD (MIN and MIP and AP).

The corresponding decision tree is shown in fig. 6.9. Partial selection among the continuous features, resulted in the selection of feature AGE only. For the final selection, feature AGE was discretized with seven bins.The final results are presented in table 6.5.

All four algorithms selected feature AGE as the best feature. Also to be noted is the consistent selection and high rank of feature SITE. Concentrating only on the first 4 features, the structures selected vary between the full independence and dependence between at most 2 features. Again, the results are widely varying even if only 2 or 3 features are considered.

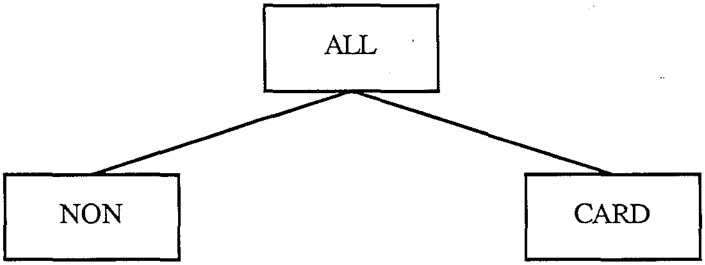


Fig. 6.9: A decision tree to discriminate those cases where the source of the pain is in the heart.

The conditions of the stopping criterion were never fulfilled in spite of the fact that structures were selected requiring a number of parameters as large as 56. The failure of the stopping criterion is a problem that has manifested itself more or less constantly. At the end of this chapter, an attempt will be made to explain this behaviour and a revised version will be proposed.

Since feature AGE was always ranked first and in order to apply the location model, the final feature selection was repeated considering only discrete features. Feature SITE was twice ranked first and twice ranked second (in these cases, feature PILL was the best). Also to be noted is the failure of the stopping criterion, admitting in one case a structure requiring 79 parameters.

Error estimates are presented in fig. 6.10 as functions of the number of features used according to the ranking, and, in the case of the basic and location model, also according to the structure selected. With no features the estimates shown were obtained by using only prior probabilities. The chart corresponding to the location model contains the error estimates as functions of the number of discrete features only, feature AGE being always considered. For this model, zero marginal probabilities did not allow all features selected to be tested, since maximum likelihood estimates could not be obtained.

Only one error estimator was applied, specifically the hold-out estimator, since the group formed by classes MIN, MIP and AP, could not be considered as being sampled from the associated universe of objects.

Algorithm	Ranking	Structure
Search strategy B1 Selection criterium A MLK estimates	1 - Age 2 - Ppain 3 - Cold 4 - Mood 5 - Site 6 - Sev 7 - Csound 8 - Type 9 - Disp	1 - 2 3 4 5 - 6 7 8 - 9
Search strategy B1 Selection criterium A Bayesian estimates	1 - Age 2 - Type 3 - Site 4 - Sev 5 - Pill 6 - Mood 7 - Onset 8 - Cold 9 - Ppain	1 - 7 2 - 6 3 - 8 - 9 4 5
Search strategy B1 Selection criterium B MLK estimates	1 - Age 2 - Rad 3 - Site 4 - Sev 5 - Pill 6 - Type 7 - Csound 8 - Onset 9 - Mood 10 - Ppain	1 - 2 3 - 4 5 - 7 - 8 6 9 - 10
Search strategy B1 Selection criterium B Bayesian estimates	1 - Age 2 - Pill 3 - Site 4 - Cold 5 - Ppain 6 - Rad 7 - Csound 8 - Sev 9 - Type 10 - Onset	1 2 3 - 5 - 6 4 7 8 9 - 10

Table 6.5: This table presents the results of the feature selection algorithms when applied to node ALL of the tree in fig. 6.9 (see also the caption in table 6.2).

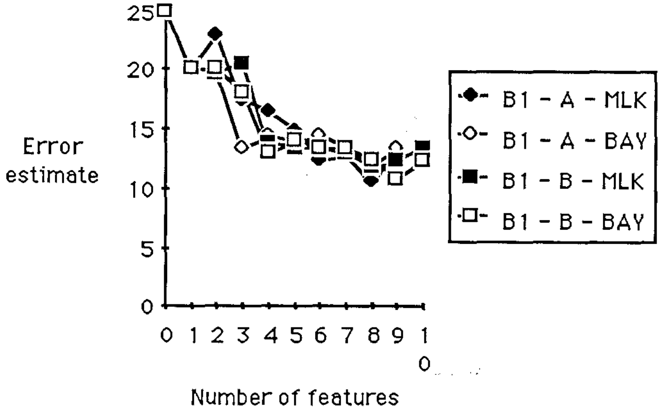


Fig. 6.10 (see following page)

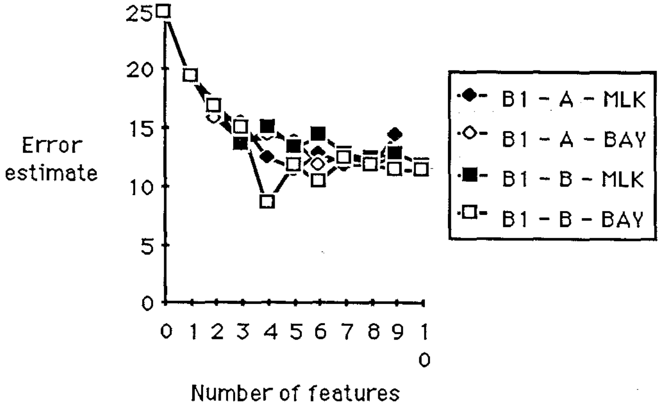


Fig. 6.10 (see following page)

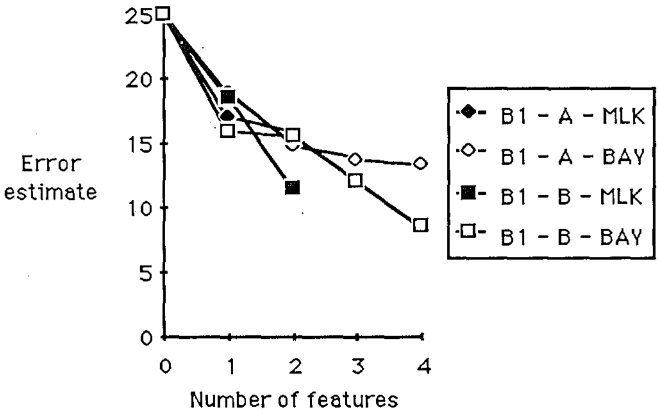


Fig. 6.10: Error estimates (hold-out estimator) of the decision tree in fig. 6.9. The charts show the errors of classifiers based on the basic model (top, previous page), the logistic model (bottom, previous page) and the location model. In every chart, each error function is associated with the feature ranking produced by a feature selection algorithm.

An analysis of fig. 6.10 reveals the following points:

1 - With a small number of features the classifiers based on models able to handle mixed data types performed better than the basic model.

2 - The progression of the error curves is similar when comparing the various feature rankings.

3 - The error curves stabilize after 3 or 4 features.

4 - An error rate less than 10 % can be achieved with 3 or 4 features.

Concerning the first point, we associate the behaviour indicated there, with the shortcomings of the discretization required by the basic model and the discriminative power of feature AGE. The discretization was done in such a way as to minimize the loss of information. This does not necessarily imply that the loss of discriminant information is also minimized.

The confusion matrices (see fig. 6.11 for a typical confusion matrix) show that errors occur mainly with the classification of objects from class NON. In view of the use of Bayes classifiers and the large difference between the prior probabilities (the prior probability of class NON was assumed to be equal to .25), this behaviour is to be expected.

	NON	CARD
NON	34	16
MIN	3	18
MIP	0	29
AP	2	48

Fig. 6.11: A typical confusion matrix resulting from the test of a classifier at node ALL in fig. 6.9. See also the caption in fig. 6.7.

6.3.7. MIN versus MIP versus AP.

The second stage in the classification of patients with acute chest pain concerns the discrimination between the various classes associated with pain originating within the heart. Three classes (MIN, MIP and AP) are to be considered.

Three decision strategies were analyzed: a single level strategy, a two levels strategy where at the first level discrimination is attempted between class MIP and the group formed by classes MIN and AP (see fig. 6.12), and finally, another two levels classifier where at the first level discrimination is attempted between classes AP and the group formed by classes MIN and MIP (see fig. 6.13). Fig. 6.12 shows a decision strategy which first tries a discrimination based on whether or not further problems developed, whereas the decision strategy shown in fig. 6.13 first tries a discrimination based on the problem that caused the pain.

As usual, systematic feature selection (4 algorithms) followed by testing (i.e. error estimation) of classifiers was done.

Starting with the single level approach, partial feature selection among the continuous features, selected feature AGE. For the final selection feature AGE was

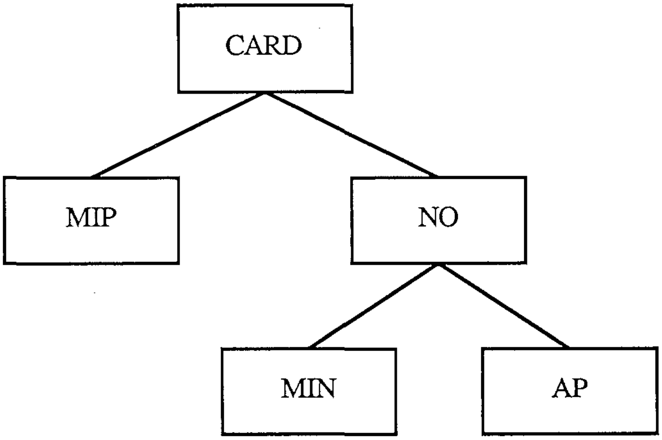


Fig. 6.12: A two levels decision tree for discrimination between classes MIN, MIP and AP.

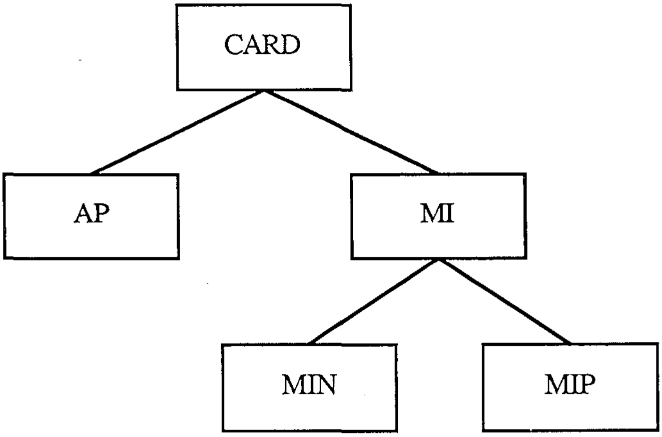


Fig. 6.13: Another two levels decision tree for discrimination between classes MIN, MIP and AP.

Algorithm	Ranking	Structure
Search strategy B1 Selection criterium A MLK estimates	1 - Sweat 2 - Onset 3 - Pill 4 - Disp 5 - Ppain	1 - 5 2 3 - 4
Search strategy B1 Selection criterium A Bayesian estimates	1 - Sweat 2 - Age 3 - Ppain 4 - Occur 5 - Pill 6 - Sev 7 - Durat 8 - Site	1 - 2 - 3 4 - 6 - 7 5 8
Search strategy B1 Selection criterium B MLK estimates	1 - Disp 2 - Durat 3 - Pill 4 - Ppain 5 - Site 6 - Csound 7 - Sweat 8 - Vomit	1 - 2 - 3 - 4 5 - 6 - 7 8
Search strategy B1 Selection criterium B Bayesian estimates	1 - Disp 2 - Sweat 3 - Csound 4 - Ppain 5 - Site 6 - Pill 7 - Onset 8 - Durat	1 2 - 3 - 4 5 6 - 7 8

Table 6.6: This table presents the results of the feature selection algorithms for the single level strategy to discriminate between classes MIN, MIP and AP (see also the caption in table 6.2).

discretized using 4 bins in view of the number of objects available. The rankings and structures obtained are shown in table 6.6.

The results are widely varying, suggesting difficulties in the discrimination, both due to a large overlap between the various classes and also to the small number of objects available for training. Features DISP and SWEAT were twice ranked first. In just one case was feature AGE selected. Again, the stopping criteria failed to act in time allowing the selection of totally inadequate structures, in view of the large number of parameters required, and the selection of too many features (see the error charts related to this decision tree).

For error evaluation, only classifiers based on the basic model were used, in spite of the fact that feature AGE and feature DURAT (an ordered discrete feature) were selected. It was not considered worthwhile to use the logistic and the location models because for both features only one algorithm ranked them in a meaningful position.

Three error estimation techniques were used: the hold-out technique, the leave-one-out approach and averaged rotation. As usual, averaged rotation was applied using 20 partitions of the data. Each partition resulted in a test set with a number of objects approximately equal to 1/3 of the total number of objects. The results obtained are shown in fig. 6.14.

An analysis of the results indicates a relative failure to discriminate between the three classes. The error curve related to the selection using algorithm B1-B-MLK is always the worst. This behaviour is explained by the large number of parameters required by the structures selected, i.e. the failure of the selection algorithm. As expected, the hold-out estimator yielded the worst results but it may be recalled that this estimator yields pessimistic estimates. The other two estimators yielded similar results with estimated variances for the averaged rotation ranging between 10.5 % and 20.2 %. These values are larger than those obtained with the head injury data set. Since the acute chest pain data set has far less objects than the head injured data set, the differences found were already expected. With 3 features, an error rate of 40 % is achieved. A typical confusion matrix is shown in fig. 6.15.

The decision strategy shown in fig. 6.12 is now considered. Feature selection algorithms were applied to node CARD yielding the results shown in table 6.7. Consistent results may be found in this table. Specifically, features DISP, SWEAT and CSOUND, were selected as the three best features by three selection algorithms. Also, the same three algorithms selected a structure which assumes non-independence between features SWEAT and CSOUND. The consistency of these results suggests that the combination of these features and structure are indeed significative. The effectiveness of algorithm B1-A-BAY (the only one breaking this consistency) was compromised already at the second step because a feature and a structure were selected that required a large number of parameters. Also to be noted is the general failure of the stopping criteria.

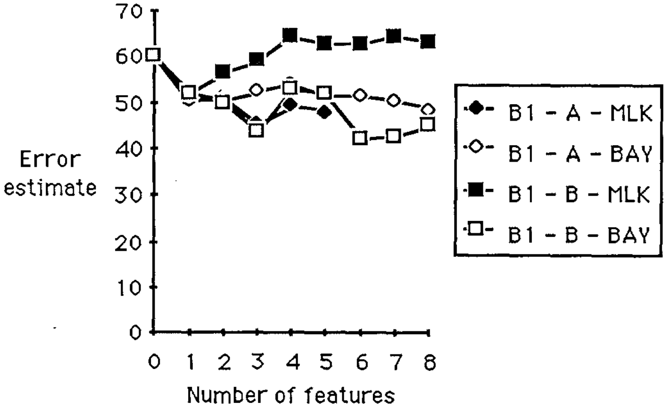


Fig. 6.14 (see following page)

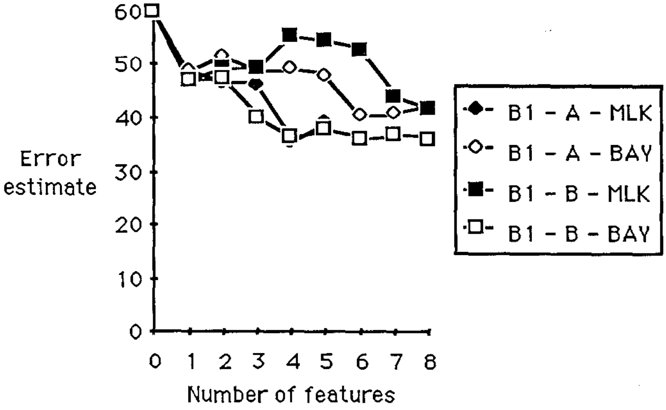


Fig. 6.14 (see following page)

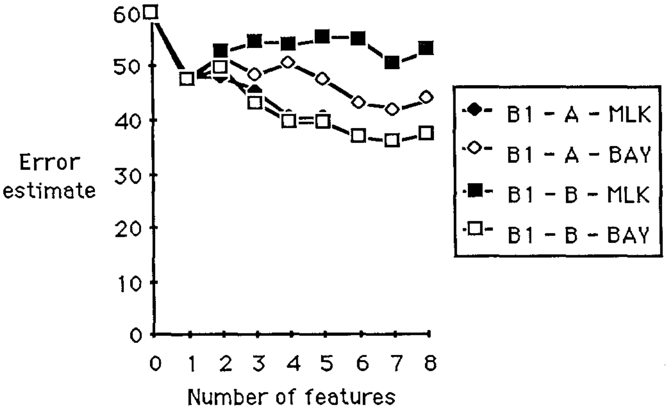


Fig. 6.14: Error estimates of the single level decision strategy for discrimination between classes MIN, MIP and AP. The charts show the errors of classifiers based on the basic model. The top chart (previous page) relates to the hold-estimator, the bottom chart (previous page) to the leave-one-out estimator using all data available, and the present chart to the averaged cross validation estimator again using all the data available.

	MIN	MIP	AP
MIN	12	2	7
MIP	5	20	4
AP	12	18	20

Fig. 6.15: A typical confusion matrix resulting associated with the single level decision strategy. See also the caption in fig. 6.7.

Algorithm	Ranking	Structure
Search strategy B1 Selection criterium A MLK estimates	1 - Disp 2 - Sweat 3 - Csound 4 - Durat 5 - Pill 6 - Onset 7 - Nausea 8 - Sev	1 - 2 - 3 4 - 5 - 6 7 - 8
Search strategy B1 Selection criterium A Bayesian estimates	1 - Disp 2 - Mood 3 - Site 4 - Ppain	1 - 2 3 4
Search strategy B1 Selection criterium B MLK estimates	1 - Disp 2 - Sweat 3 - Csound 4 - Site 5 - Durat 6 - Sputum 7 - Pill 8 - Sev	1 2 - 3 - 4 5 - 7 6 8
Search strategy B1 Selection criterium B Bayesian estimates	1 - Sweat 2 - Csound 3 - Disp 4 - Site 5 - Durat 6 - Pill 7 - Sputum 8 - Type	1 - 2 - 6 3 4 5 7 - 8

Table 6.7: This table presents the results of the feature selection algorithms when applied to node CARD of the tree in fig. 6.12 (see also the caption in table 6.2).

Bayes classifiers assuming the basic model were used in order to obtain error estimates. Only the hold-out technique was used because objects from the group formed by classes MIN and AP cannot be considered as being sampled from the universe associated with these two classes. The results are shown in fig. 6.16.

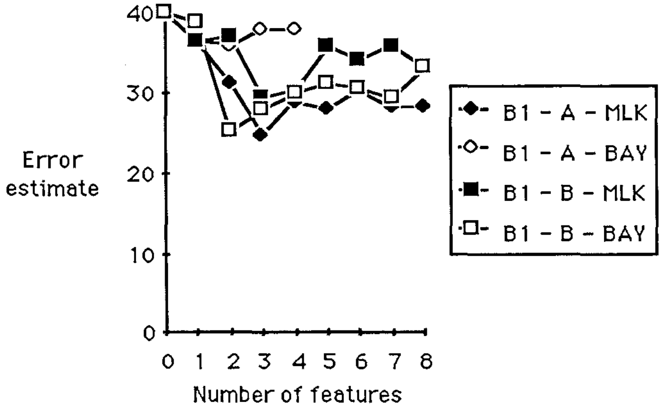


Fig. 6.16: Error estimates at node CARD of the decision tree in fig. 6.12. The chart shows the hold-out estimate of the error of a classifier based on the basic model.

The error curve associated with the selection algorithm B1-A-BAY is the worst. It has already been indicated above that this algorithm gave results deviating from the other three. This illustrates that one of the advantages of using several selection algorithms, is the ability to find consistent results without being subject to spurious results arising from specific behaviours associated with a particular training set which will be reflected in the performance of a classifier when tested on an independent test set. In this case, the error evaluation confirmed that the selection made by algorithm B1-A-BAY was indeed unusual and poor.

Either the combination of features DISP, SWEAT and CSOUND or the combination of features SWEAT and CSOUND show the best results, with an estimated error rate of approximately 25%. A related confusion matrix is shown in fig. 6.17.

Next the decision level at node NO was analyzed. The results of feature selection are shown in table 6.8. Again, consistent results may be found in this table. Specifically, features PPAIN, PILL and ONSET, were selected as the three best

	MIP	NO
MIN	0	21
MIP	15	14
AP	9	41

Fig. 6.17: A typical confusion matrix associated with node CARD in fig. 6.12. See also the caption in fig. 6.7.

features by three selection algorithms. Another kind of 'consistency' (similarity with other results), is the failure of the stopping criteria.

Error estimates for classifiers based on the basic model are shown in fig. 6.18. The three error estimators that have been used throughout this chapter were also applied here. The error curves associated with the selection algorithm B1-B-MLK, is always an upper bound for the other error curves. This is explained by the large number of parameters required by the structures selected and the errors associated with the estimation of the parameters of the classifiers. The comment made above concerning the use of several selection techniques, also applies here since algorithm B1-B-MLK was not consistent with the other three.

With only 2 features (PPAIN and PILL) an error rate of 25 % is achieved. Combining this value with the error rate at node CARD and the prior probability there, an overall error estimate is obtained which is approximately equal to 40 %. Therefore there is no improvement when this decision strategy is applied as compared to the single level strategy. Estimated variances for the averaged rotation ranged between 13.8 % and 21.2 %. A related confusion matrix is shown in fig. 6.19.

As indicated above, another decision strategy was explored. As usual, systematic feature selection followed by error estimation of Bayes classifiers based on the basic model (no continuous features were selected) was done. At node CARD, the results were fairly poor. The error rates obtained were either only slightly better than the error rate of a classifier based on prior probabilities, or worse. At node MI, some discrimination was achieved. Feature DISP was always ranked first and

Algorithm	Ranking	Structure
Search strategy B1 Selection criterium A MLK estimates	1 - Ppain 2 - Pill 3 - Onset 4 - Disp 5 - Nausea 6 - Sev 7 - Occur 8 - Vomit	1 - 4 2 3 5 6 - 8 7
Search strategy B1 Selection criterium A Bayesian estimates	1 - Ppain 2 - Pill 3 - Onset 4 - Sputum 5 - Sev 6 - Disp 7 - Nausea 8 - Vomit	1 - 4 - 5 2 3 - 7 6 - 8
Search strategy B1 Selection criterium B MLK estimates	1 - Ppain 2 - Disp 3 - Onset 4 - Vomit 5 - Site 6 - Pill 7 - Durat 8 - Sweat	1 2 - 3 - 4 5 - 8 6 - 7
Search strategy B1 Selection criterium B Bayesian estimates	1 - Ppain 2 - Pill 3 - Onset 4 - Vomit 5 - Disp 6 - Cold 7 - Site 8 - JVP	1 - 6 2 - 3 - 4 - 8 5 7

Table 6.8: This table presents the results of the feature selection algorithms when applied to node NO of the tree in fig. 6.12 (see also the caption in table 6.2).

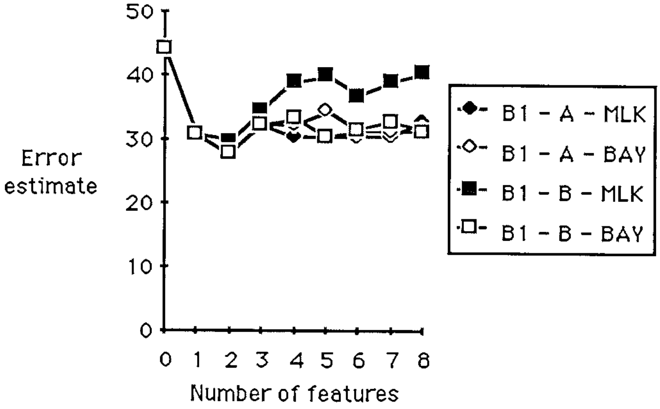


Fig. 6.18 (see following page)

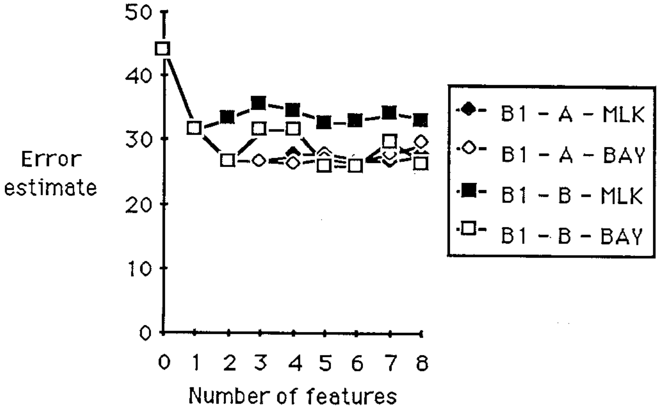


Fig. 6.18 (see following page)

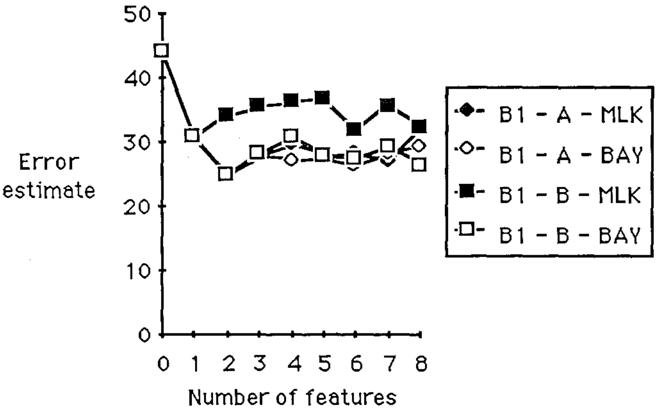


Fig. 6.18: Error estimates at node NO of the decision tree in fig. 6.12. The charts show the errors of classifiers based on the basic model. The top chart (previous page) relates to the hold-estimator, the middle chart (previous page) to the leave-one-out estimator using all data available, and the current chart to the averaged cross validation approach again using all data available.

	MIN	AP
MIN	12	9
AP	12	38

Fig. 6.19: A typical confusion matrix associated with node NO in fig. 6.12. See also the caption in fig. 6.7.

either feature SITE or feature SPUTUM were ranked second. Nevertheless, and in view of the results at node CARD, no further improvements could be obtained over the other two strategies.

Summing up, it may be concluded that a reasonable error rate could not be obtained when discrimination was attempted between classes MIN, MIP and AP. It remains an open question, whether or not a large overlap really exists between these classes since the number of objects available for training was small. This is also evidenced by the large estimated variances obtained with the averaged rotation approach.

6.3.8. Mappings.

Using correspondence analysis, factorial plots were obtained. The set of features used was assembled by combining the best subsets selected (as assessed by the error estimates) at the various levels of decision shown in figs. 6.9, 6.12, and 6.13 (the case of MI versus AP was excluded in view of the extremely poor results obtained). The following features were therefore chosen: AGE (seven bins were used for discretization), PILL, SITE, COLD, PPAIN, ONSET, DISP, SWEAT and CSOUND.

The factorial axes were obtained using the class-approach (see chapter 5) and the objects present in the training set. The first and second axes account for 72.6 % and 20.2 % of the total variance, respectively. Fig. 6.20 shows the projections of the four classes onto the plane spanned by the first two factorial axes, fig. 6.21 the projections of objects in the training set, fig. 6.22 the projections of objects in the test set and, finally, fig. 6.23 the projections of features AGE, SWEAT, DISP and PPAIN (these were the features ranked first at the various decision levels).

The first axis appears to be associated with the discrimination between class NON and all the others, since objects from class NON tend to project on one side whereas the other three classes tend to project on the other side. Considering the projection on the second axis, it may be seen that objects from classes AP and MIN tend to project on the lower part, whereas objects from class MIP tend to project on the upper part. Therefore, the second axis may be associated with the typification of those objects with or without further complications.

The projections of the objects indicate three degrees of overlap: a large overlap between classes MIN and AP, a smaller but still large overlap between class MIP and the other two, and finally, a small overlap between class NON and the other three. This is consistent with the numerical discrimination results found. The projections of objects in the training and test set are similar.

The projection of feature AGE shows a progression along the first axis followed by a progression along the second axis. The first part corresponds to younger patients and the bin associated with the older patients projects next to the projections of objects from class MIP. The projections of feature SWEAT indicates that sweating is an indication of pain caused by a heart condition with possible further complications. As for feature DISP, the occurrence of dispnoea both in the past and at present, is also an indicator of possible further complications. Finally, the projection of feature PPAIN indicates that the previous occurrence of chest pain is again associated with cardiac pain.

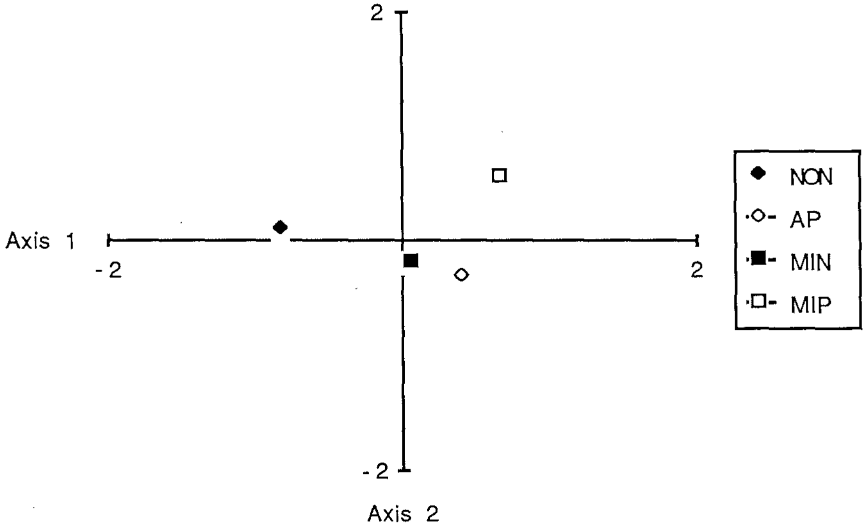


Fig. 6.20: Mapping of the four classes onto the first two factorial axes. The factorial axes were obtained using the class approach.

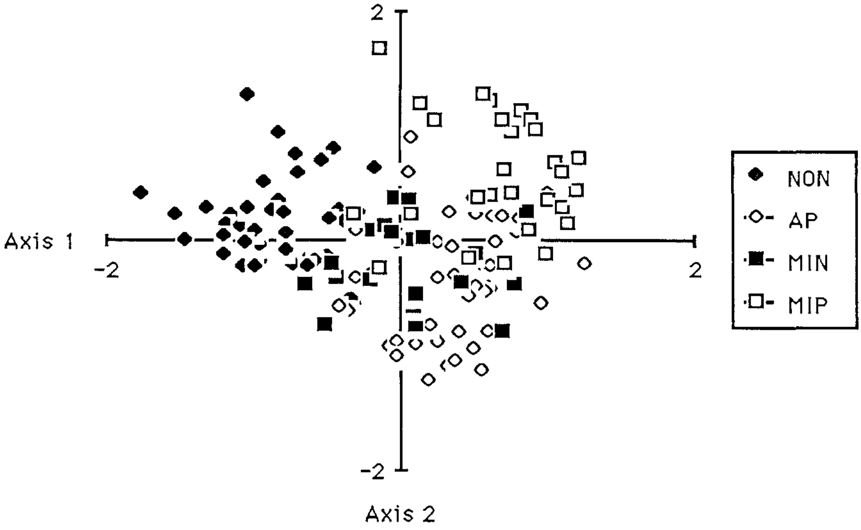


Fig. 6.21: Mapping of all objects in the training set onto the first two factorial axes.

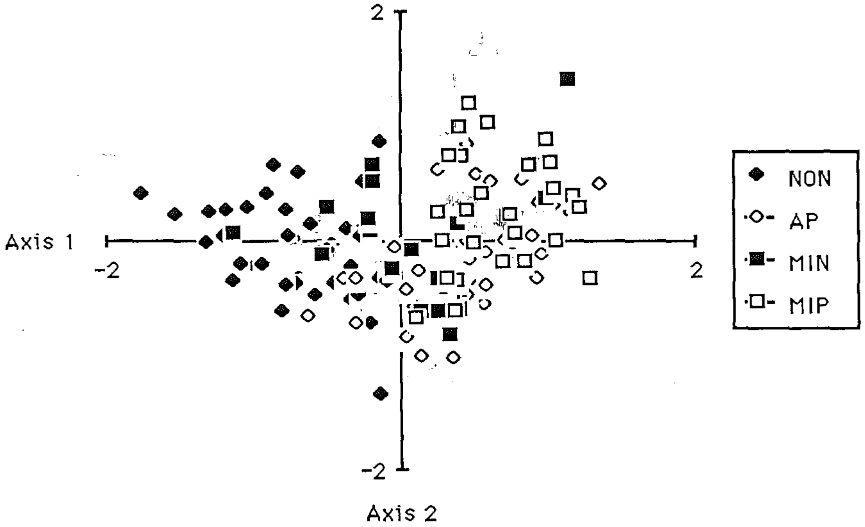


Fig. 6.22: Mapping of all objects in the test set onto the first two factorial axes.

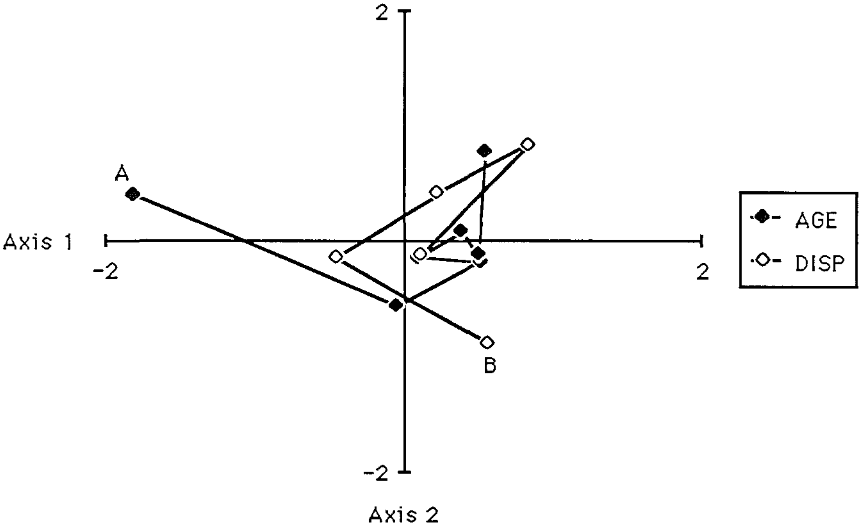


Fig. 6.23: Mapping of features AGE, DISP, PPAIN and SWEAT onto the first two factorial axes. Each feature is represented by a set of points (each one associated with a category), joined by straight lines. See also following page.

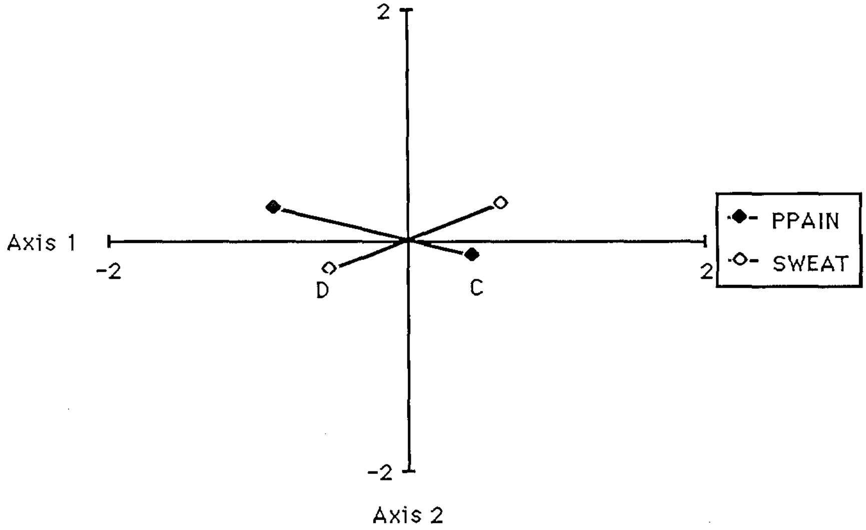


Fig. 6.23: Continuation.

6.3.9. Discussion.

A study has been made concerning patients with acute chest pain. It consisted of systematic feature selection followed by classifier evaluation and mappings. Also several decision strategies were tried.

The separation between class NON and all the others, might be considered reasonable although a question remains about its usefulness in a practical case. As for the attempted discrimination between the other three classes, the results were poor.

The study in de Dombal (DOMB83) does not specify which classifier is being used. Also, it appears that no systematic feature selection was done. The results shown cannot be compared with the results obtained here because there is no clear indication of the way the errors were estimated.

In view of the small number of objects available for training purposes, nothing can be said about the ultimate overlap between the classes. It is recommended that further studies be undertaken, possibly using some of the results obtained here as a starting point.

One of the difficulties encountered in the analysis of this data, was the organization of the data itself. Some of the features had a large number of categories requiring that some merging was done. This was done heuristically. Although an algorithm could be defined that merged the features according to some criterion, it is in our opinion better to more strictly define the categories and not allow any deviation from a pre-defined set.

The various feature selection algorithms yielded different rankings and structures, although consistent results could be found if only a small number of features is considered. The performance of the feature selection algorithms, as assessed by the error estimates, was not always similar. In some cases, structures were selected that required a large number of parameters. When the associated classifiers were tested, large errors resulted. This explains the most significant differences in performance. Furthermore, consistency of results was associated with comparatively better performances. Since the stopping criteria also failed to act in time with the head injury data set, this behaviour will further be discussed in a separate section.

The various error estimators yielded results that were worse for the hold-out technique. This was expected. Both the leave-one-out and the averaged rotation approaches, gave similar results. If an estimate of the variance of the error estimate is not required, then the leave-one-out method is more attractive since it can be implemented in such a way that it is faster to apply.

6.4. The problem of the stopping criteria.

Throughout this chapter, failures have been noted of the stopping criteria used by the selection algorithms with selection criteria directly related to the error rate. As explained elsewhere (see chapters 3 and 5), the stopping criterion associated with selection criterion A, acts when the selection criterion does not decrease after two more features are added, whereas the stopping criterion associated with selection criterion B, acts if the selection criterion does not decrease. In both cases, the failure to act in time, resulted either in too many features selected and/or in the selection of structures that required too many estimates. Especially the latter is particularly damaging.

The task required from a stopping criterion, is a complex one. At each step in the selection, several estimates are computed (each one corresponding to a given structure and feature under consideration) and a selection is made according to their relative values. If the choice of the best among a set of possible cases is already a complex task, particularly sensitive to the variance of the estimator used, the task required from a stopping criterion is even more complex.

In principle, more restrictive versions of the stopping criteria could be implemented, e.g. forcing the selection to stop if the estimated selection criterion (error or error related estimates) does not decrease by more than a given threshold, greater than zero. However, this may have negative consequences on the performance of the selection algorithm as a whole. It may happen, that at a certain step in the selection, there is no decrease in the value of the selection criterion, whereas in the following step the decrease may be significant. Examples may easily be constructed which show that this may occur even with true (not estimated) values of the error as a selection criterion.

In our opinion, a more efficient and effective stopping criterion results if the relation between the number of estimates required and the number of objects available is considered. Structures not obeying such predefined relations should be discarded. A possible way of doing this is to use the results in chapter 2, ignoring the assumptions that lead to them. A maximum admissible design error may be defined and the curves in fig. 2.1 may be used to check the relation between the number of objects and the number of parameters (full multinomials require a number of parameters equal to the number of different elements in the space minus 1: therefore, the curves in fig. 2.1 can also be seen as curves relating the expected design error to the number of objects and also to the number of parameters) required by a given structure. The assumptions underlying the results in chapter 2 and ignored if these results would be incorporated in a general stopping rule, concern the admissible structures (the results in chapter 2 only considered full multinomial structures), the estimation technique (the results in chapter 2 only consider the case of maximum likelihood) and sampling from the universe of objects (the acute chest pain data was a data set obtained by sampling per class). Furthermore, since the results in chapter 2 only apply to two-class discrimination problems, a simple heuristic would also have to be devised in order to account for more than two classes. A conservative approach is to check only the combination of the two classes that have the smaller number of objects.

6.5. Concluding remarks.

The results of the analysis of the two medical data sets may not justify practical applications, in view of the error rates obtained. It is stressed that a rigorous Bayes approach was followed and risk factors were not used.

In view of the large number of objects available in the head injury data set and of the results presented in TITT81, it can be concluded that there is a large overlap between the classes. Therefore, better results require either the introduction of new features or the definition of different classes or both.

On the other hand, improvements with the acute chest pain data set are not to be excluded because the data set analyzed had a small number of objects inhibiting the exploration and use of more features and more complex structures. The estimated variances obtained with the averaged rotation error estimator support this conclusion. Nevertheless a rearrangement of the data is required.

In both cases, the data sets were assembled over a large span of time. It is therefore frustrating to obtain results which are not sufficiently good. It appears to us that periodic analyses similar to the ones just presented, applied during the formation of the data set and yielding preliminary conclusions and suggestions of changes to implement (e.g. redefinition of classes, introduction of new features, etc.), is a good approach that will increase the chances of success. Also, a close interaction between the medical specialist and the data analyst is of utmost importance. This was not the case here.

Finally, a fairly large number of algorithms and techniques were explored. Their application was facilitated and enhanced by their integration into the interactive system ISPAHAN (see GELS80).

CHAPTER 7

FINAL REMARKS

This thesis is devoted to various aspects related to the application of pattern recognition techniques for the analysis of medical data bases. In the following, all chapters but chapter 5 are briefly reviewed with emphasis on the main points discussed and the identification of new problems.

In chapter 2 new error bounds, applicable in a given situation are established. Their main advantage is that they take into account sampling fluctuations. Furthermore, one of the bounds is a function of variables that do not require an estimate. It would be very important if other non-trivial bounds, also functions of variables that do not have to be estimated and taking into account sample fluctuations, could be found for other feature spaces and classifiers. These efforts should be directed at the classification error itself and not at errors in the estimation of a particular distribution. This last point stems from the fact that the condition of good estimates of probability density functions is sufficient but not necessary for good classifiers.

Chapter 3 is mainly devoted to feature selection. The most important concept introduced there is the concept of structure selection embedded in the process of feature selection. Whenever a new feature is selected, it brings with it the requirement that one or more additional parameters (depending on the model assumed) will have to be estimated. It may happen that not all these parameters are important for discrimination. With feature selection incorporating structure selection, a more refined analysis of the discriminating power of a feature is made. In chapter 3, the need for further study of some of the selection criteria available, was also noted. In particular, those that are directly or indirectly related to the error rate are of special interest. Specifically, new studies will have to take into account the fact that these criteria are used in an optimization procedure. What is presently known about the properties (e.g. bias, variance) of some criteria, does not apply when these criteria are used in an optimization procedure.

Chapter 4 is dedicated to missing values. Two points should be stressed here. The first point is that missing values are a nuisance and should be avoided as much as possible. The second point is the application of pattern recognition techniques in order to estimate missing values.

Chapter 6 presents the application of pattern recognition techniques to two medical data bases. The availability and integration of specific algorithms in an interactive system for pattern recognition (ISPAHAN) proved its value. Also the use of different algorithms in the same problem, enabled the identification of spurious results. The need for periodical analysis of medical data bases in order to yield preliminary conclusions was suggested. Also, the importance of a close interaction between the medical specialist and the data analyst, was stressed.

APPENDIX A

It is shown here that

$$\sum_{j=1}^{m} q(j) \qquad (\sum_{i=0}^{n} \; \sum_{t=0}^{\min(i,n-i)}$$

$$n! / (i!\; t!\; (n-t-i)!) \; (.5\; (k(j)-q(j)))^{i} \; (.5\; (k(j)+q(j)))^{t} \; (1-k(j))^{n-i-t} \quad -$$

$$.5 \sum_{i=0}^{f(n)} n! / (i!\; i!\; (n-2i)!) \; (.5\; (k(j)-q(j)))^{i} \; (.5\; (k(j)+q(j)))^{i} \; (1-k(j))^{n-2i} \qquad)$$

$$\text{(A.1)}$$

is an increasing function of k(j). Expression A.1 is obtained by making a change of variables in 2.12 according to 2.14. Let us first consider the derivative with respect to k(j) of the terms under the double summation signal. It is equal to

$$q(j) \qquad (\sum_{i=1}^{n} \; \sum_{t=0}^{\min(i,n-i)}$$

$$.5\; n! / ((i-1)!\; t!\; (n-t-i)!) \; (.5\; (k(j)-q(j)))^{i-1} \; (.5\; (k(j)+q(j)))^{t} \; (1-k(j))^{n-i-t} \quad +$$

$$\sum_{i=1}^{n-1} \; \sum_{t=1}^{\min(i,n-i)}$$

$$.5\; n! / (i!\; (t-1)!\; (n-t-i)!) \; (.5\; (k(j)-q(j)))^{i} \; (.5\; (k(j)+q(j)))^{t-1} \; (1-k(j))^{n-i-t} \quad -$$

$$\sum_{i=0}^{n-1} \; \sum_{t=0}^{\min(i,n-i-1)}$$

$$n! / (i!\; t!\; (n-t-i-1)!) \; (.5\; (k(j)-q(j)))^{i} \; (.5\; (k(j)+q(j)))^{t} \; (1-k(j))^{n-i-t-1} \qquad) \quad =$$

$$q(j) \quad (\sum_{i=0}^{n-1} \sum_{t=0}^{\min(i+1,n-i-1)}$$

$$.5\ n! / (i!\ t!\ (n-t-i-1)!)\ (.5\ (k(j)-q(j)))^{i}\ (.5\ (k(j)+q(j)))^{t}\ (1-k(j))^{n-i-t-1} \quad -$$

$$\sum_{i=0}^{n-1} \sum_{t=0}^{\min(i,n-i-1)}$$

$$.5\ n! / (i!\ t!\ (n-t-i-1)!)\ (.5\ (k(j)-q(j)))^{i}\ (.5\ (k(j)+q(j)))^{t}\ (1-k(j))^{n-i-t-1} \quad +$$

$$\sum_{i=1}^{n-1} \sum_{t=0}^{\min(i,n-i-1)}$$

$$.5\ n! / (i!\ t!\ (n-t-i-1)!)\ (.5\ (k(j)-q(j)))^{i}\ (.5\ (k(j)+q(j)))^{t}\ (1-k(j))^{n-i-t-1} \quad -$$

$$\sum_{i=0}^{n-1} \sum_{t=0}^{\min(i,n-i-1)}$$

$$.5\ n! / (i!\ t!\ (n-t-i-1)!)\ (.5\ (k(j)-q(j)))^{i}\ (.5\ (k(j)+q(j)))^{t}\ (1-k(j))^{n-i-t-1} \quad) \quad =$$

$$q(j) \quad (\sum_{i=0}^{f(n)-1} .5 \quad *$$

$$n! / (i!\ (i+1)!\ (n-2i-2)!)\ (.5\ (k(j)-q(j)))^{i}\ (.5\ (k(j)+q(j)))^{i+1}\ (1-k(j))^{n-2i-2} \ -$$

$$\sum_{i=0}^{\substack{f(n)-1 \ (n \text{ even}) \\ f(n) \ (n \text{ odd})}} .5 \quad *$$

$$n! / (i!\ i!\ (n-2i-1)!)\ (.5\ (k(j)-q(j)))^{i}\ (.5\ (k(j)+q(j)))^{i}\ (1-k(j))^{n-2i-1} \quad)$$

Considering now the remaining terms, their derivative is

$$.5 \quad q(j) \quad (\quad - \sum_{i=1}^{f(n)}$$

$$.5\ n! / (i!\ (i-1)!\ (n-2i)!)\ (.5\ (k(j)-q(j)))^{i-1}\ (.5\ (k(j)+q(j)))^{i}\ (1-k(j))^{n-2i} \quad -$$

$$\sum_{i=1}^{f(n)}$$

$$.5\ n! / (i!\ (i-1)!\ (n-2i)!)\ (.5\ (k(j)-q(j)))^{i}\ (.5\ (k(j)+q(j)))^{i-1}\ (1-k(j))^{n-2i} \quad +$$

$$\sum_{i=0}^{\substack{f(n)-1\ \ (n\ \text{even})\\ f(n)\ \ (n\ \text{odd})}}$$

$$n! / (i!\ i!\ (n-2i-1)!)\ (.5\ (k(j)-q(j)))^{i}\ (.5\ (k(j)+q(j)))^{i}\ (1-k(j))^{n-2i-1} \quad) \quad =$$

$$.5 \quad q(j) \quad (\quad - \sum_{i=0}^{f(n)-1}$$

$$.5\ n! / (i!\ (i+1)!\ (n-2i-2)!)\ (.5\ (k(j)-q(j)))^{i}\ (.5\ (k(j)+q(j)))^{i+1}\ (1-k(j))^{n-2i-2} \ -$$

$$\sum_{i=0}^{f(n)-1}$$

$$.5\ n! / (i!\ (i+1)!\ (n-2i-2)!)\ (.5\ (k(j)-q(j)))^{i+1}\ (.5\ (k(j)+q(j)))^{i}\ (1-k(j))^{n-2i-2} \quad +$$

$$\sum_{i=0}^{\substack{f(n)-1\ \ (n\ \text{even})\\ f(n)\ \ (n\ \text{odd})}}$$

$$n! / (i!\ i!\ (n-2i-1)!)\ (.5\ (k(j)-q(j)))^{i}\ (.5\ (k(j)+q(j)))^{i}\ (1-k(j))^{n-2i-1} \quad)$$

Summing all terms, we are left with

$$.25 \quad q(j) \quad (\quad \sum_{i=1}^{f(n)-1}$$

$$n! / (i!\ (i+1)!\ (n-2i-2)!)\ (.5\ (k(j)-q(j)))^{i}\ (.5\ (k(j)+q(j)))^{i+1}\ (1-k(j))^{n-2i-2} \ -$$

$$\sum_{i=0}^{f(n)-1}$$

$$n! \,/\, (i!\ (i+1)!\ (n-2i-2)!)\ (.5\ (k(j)-q(j)))^{i+1}\ (.5\ (k(j)+q(j)))^{i}\ (1-k(j))^{n-2i-2}\ \ ) =$$

$$.25 \quad q(j)^{2} \quad \sum_{i=0}^{f(n)-1}$$

$$n! \,/\, (i!\ (i+1)!\ (n-2i-2)!)\ (.5\ (k(j)-q(j)))^{i}\ (.5\ (k(j)+q(j)))^{i}\ (1-k(j))^{n-2i-2}$$

This last expression is greater or equal to zero. Therefore A.1 is a non decreasing function of k(j). This completes the proof.

APPENDIX B

It is considered here the maximization with respect to q(1) of

$$q(1) \quad \left(\sum_{i=f(n)+1}^{n} n! \,/\, (i!\,(n-i)!)\; (.5\,(1-q(1)))^{i}\; (.5\,(1+q(1)))^{n-i} \quad + \right.$$

$$\left. .5\; n! \,/\, (f(n)!\; f(n)!)\; (.5\,(1-q(1)))^{f(n)}\; (.5\,(1+q(1)))^{f(n)} \quad \right)$$

for n even, and

$$q(1) \quad \left(\sum_{i=f(n)+1}^{n} n! \,/\, (i!\,(n-i)!)\; (.5\,(1-q(1)))^{i}\; (.5\,(1+q(1)))^{n-i} \right. \qquad \text{(B.1)}$$

for n odd.

Making a change of variable of the form $q(1) = 1 - 2f$ and rewriting B.1 we have

$$(1 - 2f) \quad \left(\sum_{i=f(n)+1}^{n} n! \,/\, (i!\,(n-i)!)\; f^{i}\; (1-f)^{n-i} \quad + \right.$$

$$\left. .5\; n! \,/\, (f(n)!\; f(n)!)\; f^{f(n)}\; (1-f)^{f(n)} \quad \right)$$

for n even, and

$$(1 - 2f) \quad \left(\sum_{i=f(n)+1}^{n} n! \,/\, (i!\,(n-i)!)\; f^{i}\; (1-f)^{n-i} \right.$$

for n odd.

Since $0 \leq q \leq 1$, than $0 \leq f \leq .5$. We have therefore in both cases binomial distributions. These distributions can be approximated by a gaussian distribution with parameters

$$n\,f,\; \sqrt{}\,(n\,f\,(1-f))$$

Since $0 \leq f \leq .5$, the skewness of the binomial is positive. Therefore, the use of the gaussian distribution, leads to larger values. We can now write

$$(1 - 2f) \quad \phi \quad (x \leq - (n/2 - nf) / \sqrt{} (n\, f\, (1-f)) \quad =$$

$$2 \sqrt{}(f(1-f))\ (.5-f) \sqrt{}n / (\sqrt{}n \sqrt{}(f\ (1-f)))\ \phi\ (x \leq - (.5-f) \sqrt{}n / \sqrt{}(f\ (1-f)) \tag{B.2}$$

where ϕ (x) is the standard normal distribution. The maximum of $(u\ \phi\ (x \leq -u))$ with respect to u is equal to .17. Therefore

$$B.2 \leq (2 \sqrt{}\ (f\ (1-f))\ .17) / \sqrt{}n \leq .17 / \sqrt{}n$$

This completes the proof.

APPENDIX C

These expressions complete the definition of expression 3.12 in chapter 3.

$$F_1({n^A}_1,{n^B}_1,n^A,n^B,m,\alpha) = \Big(\sum_{k=0}^{{n^B}_1} \alpha^{{n^B}_1-k} (1-\alpha)^{n^B-{n^B}_1+m-1+k} g({n^B}_1,{n^B}_1-k+1) \,/$$

$$g(n^B-{n^B}_1+m-1+k,n^B-{n^B}_1+m-1) \Big) \;*$$

$$\Big(({n^A}_1+1)! \,/\, g(n^A+m,n^A-{n^A}_1+m-1) \;-$$

$$\Big(\sum_{k=0}^{{n^A}_1+1} \alpha^{{n^A}_1+1-k} (1-\alpha)^{n^A-{n^A}_1+m-1+k} g({n^A}_1+1,{n^A}_1-k+2) \,/$$

$$g(n^A-{n^A}_1+m-1+k,n^A-{n^A}_1+m-1) \Big) \Big)$$

if element 1 is classified as class A,

$$F_1({n^A}_1,{n^B}_1,n^A,n^B,m,\alpha) = \Big(\sum_{k=0}^{{n^B}_1+1} \alpha^{{n^B}_1-k+1} (1-\alpha)^{n^B-{n^B}_1+m-1+k} g({n^B}_1+1,{n^B}_1-k+2) \,/$$

$$g(n^B-{n^B}_1+m-1+k,n^B-{n^B}_1+m-1) \Big) \;*$$

$$\Big({n^A}_1! \,/\, g(n^A+m-1,n^A-{n^A}_1+m-1) \;-$$

$$\Big(\sum_{k=0}^{{n^A}_1} \alpha^{{n^A}_1-k} (1-\alpha)^{n^A-{n^A}_1+m-1+k} g({n^A}_1,{n^A}_1-k+1) \,/$$

$$g(n^A-{n^A}_1+m-1+k,n^A-{n^A}_1+m-1) \Big) \Big)$$

if element 1 is classified as class B,

$$F_2(n^A{}_1,n^A{}_i,n^A{}_1,n^B{}_i,n^A,n^B,m,\alpha) \quad =$$

$$(\sum_{k=0}^{n^B{}_1} \alpha^{n^B{}_1-k} (1-\alpha)^{n^B-n^B{}_1+m-1+k} \; n^B{}_i! \; g(n^B{}_1,n^B{}_1-k+1) \, /$$

$$g(n^B-n^B{}_1+m-1+k,n^B-n^B{}_1-n^B{}_i+m-2) \,) \; *$$

$$(\, (n^A{}_i+1)! \; n^A{}_1! \, / \, g(n^A+m,n^A-n^A{}_1-n^A{}_i+m-2) \; -$$

$$(\, (n^A{}_i+1)! \sum_{k=0}^{n^A{}_1} \alpha^{n^A{}_1-k} (1-\alpha)^{n^A-n^A{}_1+m-1+k} \; g(n^A{}_1,n^A{}_1-k+1) \, /$$

$$g(n^A-n^A{}_1+m+k,n^A-n^A{}_1-n^A{}_i+m-2) \,) \,)$$

if element i is classified as class A,

$$F_2(n^A{}_1,n^A{}_i,n^B{}_1,n^B{}_i,n^A,n^B,m,\alpha) \quad =$$

$$(\sum_{k=0}^{n^B{}_1} \alpha^{n^B{}_1-k} (1-\alpha)^{n^B-n^B{}_1+m+k} \; (n^B{}_i+1)! \; g(n^B{}_1,n^B{}_1-k+1) \, /$$

$$g(n^B-n^B{}_1+m+k,n^B-n^B{}_1-n^B{}_i+m-2) \,) \; *$$

$$(\, n^A{}_1! \; n^A{}_i! \, / \, g(n^A+m-1,n^A-n^A{}_1-n^A{}_i+m-2) \; -$$

$$(\sum_{k=0}^{n^A{}_1} \alpha^{n^A{}_1-k} (1-\alpha)^{n_A-n^A{}_1+m-1+k} \; n^A{}_i! \; g(n^A{}_1,n^A{}_1-k+1) \, /$$

$$g(n^A-n^A{}_1+m-1+k,n^A-n^A{}_1-n^A{}_i+m-2)\ )\ )$$

if element i is classified as class B. Function g(a,b) above is either equal to the product of all integers in the interval [b,a] ($b \leq a$) or equal to 1 if $a < b$.

APPENDIX D

These expressions complete the definition of expression 3.15 in chapter 3.

$$G_1(n^A_1,n^B_1,n^A,n^B,m,\alpha) = (n^A_1+1) / (n^A+m) \quad * \quad (1 / (n^B+m-1)! - 1 / n^B_1! \quad * \quad \sum_{k=0}^{n^B_1} \alpha^{n^B_1-k} (1-\alpha)^{n^B-n^B_1+m-1+k} \quad * \quad g(n^B_1,n^B_1-k+1) / (n^B-n^B_1+m-1+k)! \;)$$

if element 1 is classified as class A,

$$G_1(n^A_1,n^B_1,n^A,n^B,m,\alpha) = (n^B_1+1) / (n^B+m)! - 1 / n^B_1! \quad * \quad (\sum_{k=0}^{n^B_1+1} \alpha^{n^B_1+1-k} (1-\alpha)^{n^B-n^B_1+m-1+k} \quad * \quad g(n^B_1+1,n^B_1-k+2) / (n^B-n^B_1+m-1+k)! \;)$$

if element 1 is classified as class B,

$$G_2(n^A_i,n^B_1,n^B_i,n^A,n^B,m,\alpha) = (n^A_i+1) / (n^A+m) \quad * \quad (1 / (n^B+m-1)! - 1 / n^B_1! \quad * \quad (\sum_{k=0}^{n^B_1} \alpha^{n^B_1-k} (1-\alpha)^{n^B-n^B_1+m-1+k} \quad * \quad g(n^B_1,n^B_1-k+1) / (n^B-n^B_1+m-1+k)! \;) \;)$$

if element i is classified as class A,

$$G_2(n^A{}_i,n^B{}_1,n^B{}_i,n^A,n^B,m,\alpha) = (n^B{}_i+1) \; * \; \left(1/(n^B+m)! - 1 / n^B{}_1! \; * \left(\sum_{k=0}^{n^B{}_1} \alpha^{n^B{}_1-k} \, (1-\alpha)^{n^B-n^B{}_1+m+k} \; * \; g(n^B{}_1,n^B{}_1-k+1) \, / \, (n^B-n^B{}_1+m+k)! \right) \right)$$

if element i is classified as class B. For the definition of function g(a,b), see appendix C.

APPENDIX E

This appendix contains a brief description of the features present in the acute chest pain data base. The analysis of this data base is presented in chapter 6.

Feature ONSET Binary feature with categories 'sudden' and 'gradual' which typifies the onset of the pain.

Feature OCCUR Binary feature which indicates whether the pain is continuous or intermittent.

Feature SEV Binary feature which indicates whether the pain is moderate or severe.

Feature DURAT Discrete feature that indicates the duration of the pain, grouped in a number of intervals (≤ 1 hour, ≤ 2 h, ≤ 4 h, ≤ 12 h, ≤ 24 h, ≤ 1 week, more than 1 week). It is therefore an ordered discrete feature.

Feature TYPE Discrete feature that indicates the type of the pain. In the data set made available to us, 34 different categories can be found, which are combinations of one or more of the following categories: tight, sharp, crushing, gripping, burning, ache, dull, stabbing, nagging and squeezing.

Feature SITE Discrete feature that indicates the site of the pain. It has 12 different categories, again combinations of one or more of the following categories: central, chest, across, left side, right side and epigastrium.

Feature RAD Discrete feature that indicates whether or not the pain is radiating to other sites. If yes, it also indicates where to the pain radiates. It has 41 categories, one associated with no radiation, and the remaining 40 indicating the location of the radiation. These 40 categories are again combinations of one of the following categories: left arm, right arm, both arms, back, chest, right shoulder, left shoulder, both shoulders, neck, jaw, throat, hands and/or fingers and epigastrium.

Feature PROG Discrete feature that indicates the progress made. It has 4 categories: better, same, worse and varied.

Feature DISP Discrete feature related to the occurrence of dispnoea. It has 5 categories which are combinations of one or more of the following categories: no, now and past.

Feature SPUTUM Binary feature indicating if the patient has or does not have sputum.

Feature NAUSEA Binary feature indicating whether the patient suffers from nausea.

Feature VOMIT Binary feature indicating whether vomiting has occurred.

Feature SWEAT Binary feature indicating whether the patient is sweating.

Feature COLD Binary feature indicating whether the patient feels cold/clammy.

Feature COUGH Discrete feature that indicates whether the patient has or had cough. It has 5 categories which are combinations of one or more of the following categories: no, now and past.

Feature MOOD Discrete feature that indicates the patient's mood. It has 10 categories, which are combinations of one or more of the following categories: normal, anxious, distressed, lethargic and in shock.

Feature COLOUR Discrete feature that indicates the patient's aspect. It has five categories: normal, pale, flushed, cyanosed and pale and cyanosed.

Feature RMOV Binary feature that indicates whether the patient's respiratory movements are normal or abnormal.

Feature JVP Binary feature that indicates whether the jugular venous pressure is normal or increased.

Feature HSOUND Binary feature that indicates whether the patient heart sounds are normal or abnormal.

Feature CSOUND Discrete feature that indicates the patient chest sounds. It has 7 categories which are combinations of one or more of the following categories: normal, rhonci, crepitations and decreased.

Feature PPAIN Binary feature that indicates whether or not the patient previously had chest pain.

Feature PILL Discrete feature that indicates whether the patient had any previous relevant illnesses (e.g. myocardial infarction, angina, bronchitis, hypertension, diabetes, etc.) and if so, what were those illnesses. It has a total of 23 categories.

Feature NUMB Binary feature that indicates the presence or absence of numbness.

Feature APPET Binary feature that indicates whether the patient's appetite is normal or has decreased.

Feature TENDER Binary feature that indicates whether or not the pain responds to pressure.

Feature BOWELS Discrete feature that indicates whether the bowels function normally, constipated, or whether the patient has diarrhoea.

Feature TEMP Discrete feature that indicates whether the patient's temperature is normal, below normal or above normal.

Feature PERC Discrete feature that indicates whether the results of percussion are normal, dull or hyper resonant.

Feature AGE Continuous feature that shows the patient's age in years.

Feature PULSE Continuous feature that shows the patient's pulse rate in beats per minute.

Feature BPSYS Continuous feature that shows the patient's systolic blood pressure.

Feature BPDIA Continuous feature that shows the patient's diastolic blood pressure.

Feature RRATE Continuous feature that shows the patient's respiratory rate.

APPENDIX F

A brief account is given here of the software developments that were implemented into the interactive system for pattern recognition ISPAHAN (see GELS80). These developments, made the package able to handle mixed data types and missing values.

This appendix is organized in a number of sections, each one containing a description of facilities which are grouped in an ISPAHAN main menu.

F.1. Discrete populations. Creation of new populations.

A module is available that allows the user to make known to the system the type of features under study. Three types of features are allowed: continuous features, ordered discrete features and non-ordered discrete features. The information entered is kept by the system and used whenever required.

A module is available that allows the generation of random populations. Multivariate gaussian populations and discrete populations may be generated. The discrete populations are generated according to a loglinear model (see chapter 5) selected by the user.

F.2. Missing values.

A set of modules is available that give information, per feature or per object, about the missing values present in the group of objects under consideration.

A module allows the removal of objects with missing values in the group under consideration, according to various criteria defined by the user. These objects are excluded from further analysis.

A module allows the random insertion of missing values in the group of objects under consideration.

A set modules allows the estimation and replacement of missing values. The algorithms implemented are the four algorithms described in chapter 4 (two suitable for continuous features and two for discrete features).

REFERENCES

AITC76 - Multivariate binary discrimination by the kernel method, Aitchison J., Aitken C.G.G., Biometrika, vol. 63, pp. 413-420, 1976.

ALBE81 - Probit and logistic discriminant functions, Albert A., Anderson J.A., Communications in Statistics - Theoretical Methods, vol. A10, pp. 641-657, 1981.

ALBE81 - Stepwise probit discrimination with specific application to short-term prognosis in acute myocardial infarction, Albert A., Chapelle J.P., Smeets J.P., Compt. Biomed. Research, vol. 14, pp. 391-398, 1981.

ANDE72 - Separate sample logistic discrimination, Anderson J.A., Biometrika, vol. 59, pp. 19-35, 1972.

ANDE74 - Diagnosis by logistic discriminant function: further practical problems and results, Anderson J.A., Applied Statistics, vol. 23, pp. 397-404, 1974.

ANDE79 - Multivariate logistic compounds, Anderson J.A., Biometrika, vol. 66, pp. 17-26, 1979.

ANDE82 - Logistic discrimination, Anderson J.A., Handbook of Statistics, vol. 2, eds. P.R. Krishnaiah, L.N. Kanal, North-Holland, Amsterdam, pp. 169-191, 1982.

BAHA61 - A representation of the joint distribution of responses to n dichotomous items, Bahadur R.R., Studies in item analysis and prediction, ed. H. Solomon, pp. 158-168, 1961.

BEAL75 - Missing values in multivariate analysis, Beale E.M.L., Little R.J.A., Journal of the Royal Statistical Society B, vol. 37, pp. 129-145, 1975.

BEN80 - On the sensitivity of the probability of error rule for feature selection, Ben-Bassat M., IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. PAMI-2, pp. 57-60, 1980.

BENZ80 - L'Analise des Donnees II - Analyse des Correspondences, Benzecri J.P., Dunod, Paris, 1980.

BEZD77 - Prototype classification and feature selection with fuzzy sets, Bezdek J.C., Castelaz P.F., IEEE Trans. on System Man and Cybernetics, vol SMC-7, pp. 87-92, 1977.

BIRC63 - Maximum likelihood in three-way contingency tables, Birch M.W., Journal of the Royal Statistical Society B, vol 25, pp. 220-233, 1963.

BISH75 - Discrete multivariate analysis - theory and practice, Bishop Y.M.M., Fienberg S.E., Holland P.W., MIT Press, Cambridge Massachusetts, 1975.

CHAN71 - Independence of measurements and the mean recognition accuracy, Chandrasekaran B., IEEE Trans. on Information Theory, vol. IT-17, pp. 452-456, 1971.

CHAN74 - Quantization complexity and independent measurements, Chandrasekaran B., Jain A.K., IEEE Trans. on Computers, pp. 102-106, 1974.

CHAN75 - Independence, measurement complexity and classification performance, Chandrasekaran B., Jain A.K., IEEE Trans. on Systems Man and Cybernetics, vol. SMC-5, pp. 240-244, 1975.

CHOW68 - Approximating discrete probability distributions with dependence trees, Chow C.K., Liu C.N., IEEE Trans. on Information Theory, vol. IT-14, pp. 462-467, 1968.

COOL71 - Multivariate data analysis, Cooley W.W., Lohnes P.R., Wiley, New York, 1971.

DAY67 - A general maximum likelihood discriminant, Day N.E., Kerridge D.F., Biometrics, vol. 23, pp. 313- 323, 1967.

DILL78 - On the performance of some multinomial classification rules, Dillon W.R., Goldstein M., Journal of the American Statistical Association, vol. 78, 1978.

DIXO79 - Pattern recognition with partly missing data, Dixon J.K., IEEE Trans. on Systems Man and Cybernetics, vol. SMC-9, pp. 617-621, 1979.

DOMB83 - Evaluation of decision making by humans and computers in acute abdominal and acute chest pain, de Dombal F.T., Lecture Notes in Medical Informatics, vol. 22, Springer-Verlag, pp. 42-54, 1983.

DUIN77 - A sample size dependent error bound, Duin R.P.W., Proc. of the 3rd Int. Joint Conf. on Pattern Recognition, Colorado, 1977.

DUIN78 - On the accuracy of statistical pattern recognizers, Duin R.P.W., Ph D thesis, Delft, 1978.

DUIN78 - On the evaluation of independent binary features, Duin R.P.W., v. Haersma Buma C.E., Roosma L., IEEE Trans. on Information Theory, vol IT-24, pp. 248-249, 1978.

EVER77 - The analysis of contingency tables, Everitt B.S., Chapman and Hall, London, 1977.

FINB70 - An iterative procedure for estimation in contingency tables, Finberg S.E., Annals of Mathematical Statistics, vol. 41, pp. 907-917, 1970.

GELS80 - ISPAHAN: An interactive system for pattern analysis: structure and capabilities, Gelsema E.S., Pattern Recognition in Practice, eds. E.S. Gelsema, L.N.Kanal, North Holland, Amsterdam, 1980.

GELS82 - The formalism of correspondence analysis as a means to describe object samples, Gelsema E.S., Queiros C.E., Timmers T., Proc. of the 6th International Conference on Pattern Recognition, Munich, 1982.

GELS84 - The use of correspondence analysis in the assessment of morphological changes during carcinogenesis, Gelsema E.S., Hunink M., Queiros C.E., Timmers T., Cytometry, vol. 5, pp. 463-468, 1984.

GIFI81 - Non-linear multivariate analysis, Gifi A., Dept. of Data Theory, University of Leiden, 1981.

GILB68 - On discrimination using qualitative variables, Gilbert E.S., Journal of the American Statistical Association, pp. 1399-1412, 1968.

GLES72 - Towards automated medical decisions, Gleser M.A., Collen M.F., Comp. and Biomed. Res., vol. 5, pp. 180-189, 1972.

GLIC73 - Sample based multinomial classification, Glick N., Biometrics, vol. 29, pp. 241-256, 1973.

GLIC78 - Additive estimators for probabilities of correct classification, Glick N., Pattern Recognition, vol. 10, pp. 211-222, 1978.

GOLD75 - Selection of variates for the two-group multinomial classification problem, Goldstein M., Rabinowitz M., Journal of the American Statistical Association, vol. 70, pp. 776-781, 1975.

GOLD78 - Discrete discriminant analysis, Goldstein M., Dillon W.R., Wiley, New York, 1978.

GOOD70 - The multivariate analysis of qualitative data: interactions among multiple classifications, Goodman L.A., Journal of the American Statistical Society, vol. 65, pp. 226-256, 1970.

HABB78 - Variable kernel density estimation in discriminant analysis, Habbema J.D.F., Hermans J., Remme J., COMPSTAT, pp. 178-185, 1978.

HABB81 - A computer program for selection of variables in diagnostic and prognostic problems, Habbema J.D.F., Gelpke G.J., Computer Programs in Biomedicine, vol. 13, pp. 251-270, 1981.

HAND74 - A preliminary note on pattern classification using incomplete vectors, Hand D.J., Batchelor B.G., Proc. 2nd International Joint Conference on Pattern Recognition, pp. 15-17, 1974.

HAND76 - The classification of incomplete vectors, Hand D.J., Batchelor B.G., Proc. 3rd International Joint Conference on Pattern Recognition, pp. 263-266, 1976.

HILD78 - The measurement of performance in probabilistic diagnosis III; methods based on continuous functions of the diagnostic probabilities, Hilden J., Habbema J.D.F., Bjerregaard B., Methods of Information in Medecine, vol. 17, pp. 238-246, 1978.

HILL67 - Discrimination and allocation with discrete data, Hills M., Applied Statistics, vol. 16, pp. 237-250, 1967.

HILL73 - Reciprocal averaging: an eigenvector method of ordination, Hill M.O., Journal of Ecology, vol. 61, pp. 237-251, 1973.

HILL74 - Correspondence analysis: a neglected multivariate method, Hill M.O., Applied Statistics, vol. 23, pp. 340-354, 1974.

HUGH68 - On the mean accuracy of statistical pattern recognizers, Hughes F.G., IEEE Trans. on Information Theory, vol. IT-14, pp. 55-63, 1968.

JAIN82 - Dimensionality and sample size considerations in pattern recognition practice, Jain A.K., Chandrasekaran B., Handbook of Statistics, vol. 2, eds. Krishnaiah and Kanal, North-Holland, pp. 835-855, 1982.

KEND66 - The advanced theory of statistics, Kendall M.C., Stuart A., Griffin, London, 1966.

KITT78 - Classification of incomplete pattern vectors using modified discriminant functions, Kittler J., IEEE Trans. on Computers, vol C-27, pp. 367-375, 1978.

KRZA75 - Discrimination and classification using both binary and continuous variables, Krzanowski W.J., Journal of the American Statistical Association, vol. 70, pp. 782-790, 1975.

KRZA79 - Some linear transformations for mixtures of binary and continuous variables, with particular reference to linear discriminant analysis, Krzanowski W.J., Biometrika, vol. 66, pp. 33-39, 1979.

KRZA80 - Mixture of continuous and categorical variables in discriminant analysis, Krzanowski W.J., Biometrics, vol. 36, pp. 493-499, 1980.

KULL59 - Information theory and statistics, Kullback S., Wiley, New York, 1980.

LACH67 - An almost unbiased method of obtaining confidence intervals for the probability of misclassification in discriminant analysis, Lachenbruch P.A., Biometrics, pp. 639-645, 1967.

LACH73 - On a stepwise procedure for two population Bayes decision rules using discrete variables, Lachin J.M., Biometrics, vol. 29, pp. 551-564, 1973.

LEBA77 - Techniques de la description statistique, Lebart L., Morineau A., Tabard N., Dunod, Paris, 1977.

LEON74 - A minimum cost feature selection algorithm for binary valued features, Leonard M.S., Kilpatrick K.E., IEEE Trans. on Systems Man and Cybernetics, vol. SMC-4, pp. 536-542, 1974.

LESA86 - Logistic discrimination analysis with applications in electrocardiography, Lesaffre E., Ph D thesis, Leuven (Belgium), 1986.

LI84 - The selection of significant binary features, Li X., Dubes R.C., Proc. of the 7th International Conference on Pattern Recognition, pp. 260-263, 1984.

LORE77 - Use of correspondence analysis in processing hemodynamic data from acute myocardial infarction, Lorente P., Delabre M.,Comput. Biomed. Research, vol. 10, pp. 213-235, 1977.

MART72 - Probability models, estimation, and classification for multivariate dichotomous populations, Martin D.C., Bradley R.A., Biometrics, pp. 203-221, 1972.

MILL74 - The jackknife - a review, Miller R.G., Biometrika, vol. 61, pp. 1-15, 1974.

MOOR73 - Evaluation of five discrimination procedures for binary variables, Moore D.H., Journal of the American Statistical Association, vol. 68, pp. 399-404, 1973.

NAKA77 - Multiple correspondence analysis as an aid to medical data processing, Nakache J.P., Lebart L., Lorente P., Proceedings Medinfo 77, eds. Shires, Wolf, North Holland, Amsterdam 1977.

NAKA78 - Stepwise barycentric discrimination on qualitative data: application to a medical example, Nakache J.P., Morice V., Gremy F., MEDIS78, Int. Symp. on Medical Information Systems, Osaka, pp. 69-73, 1978.

NAKA80 - Methodes de discrimination sur variables de nature quelconque, theorie et pratique, Nakache J.P., Ph D thesis, Universite Pierre et Marie Curie, 1980.

NAKA80 - Discrimination sur variables binaires basee sur l'estimation des distributions de probabilite par group, Nakache J.P., Morice V., Golmard J.L., Data Analysis and Informatics, eds. Diday E., North Holland, 1980.

NISH80 - Analysis of categorical data: dual scaling and its applications, Nishisato S., University of Toronto Press, 1980.

OTT76 - Some classification procedures for multivariate binary data using orthogonal functions, Ott J., Kronmal R.A., Journal of the American Statistical Association, vol. 71, pp. 391-399, 1976.

QUEI83 - Correspondence analysis in the context of pattern recognition, Queiros C.E., Gelsema E.S., Timmers T., Pattern Recognition Letters, vol. 1, pp. 229-236, 1983.

QUEI84 - On feature selection, Queiros C.E., Gelsema E.S., Proc. 7th Int. Conf. on Pattern Recognition, vol. 1, pp. 128-130, 1984.

RAIF61 - Statistical decision theory approach to item selection for dichotomous test and criterion variables, Raiffa H., Studies in item analysis and prediction, ed. H. Solomon, pp. 187-220, 1961.

SCHM83 - Comparative performance of four discriminant analysis methods for mixtures of continuous and discrete variables, Schmitz P.I.M., Habbema J.D.F., Hermans J., Raatgever J.W., Communications in Statistics, B12, pp. 727-751, 1983.

SCHM85 - A simulation study of the performance of five discriminant analysis methods for mixtures of continuous and binary variables, Schmitz P.I.M., Habbema J.D.F., Hermans J., Journal of Statistical Computation and Simulation, vol. 23, pp. 69-95, 1985.

SEBE62 - Decision making processes in pattern recognition, Sebestyen G.S., McMillan New-York, 1962.

SIEG76 - Nonparametric statistics for the behavioral sciences, Siegel S., McGraw-Hill Kogakusha Tokyo, 1976.

STOF74 - A classifier design technique for discrete variable pattern recognition problem, Stoffel J.C., IEEE Trans. on Computers, vol. C-23, pp. 428-441, 1974.

TALL75 - A general classification model with specific application to response to adrenalectomy in women with breast cancer, Tallis G.M., Leppard P., Sarfaty G., Comp. Biomed. Research, vol. 8, pp. 1-7, 1975.

TITT81 - Comparison of discrimination techniques applied to a complex data set of head injured patients, Titterington D.M., Murray G.D., Murray L.S., Spiegelhalter D.J., Skene A.M., Habbema J.D.F., Gelpke G.J., Journal of the Royal Statistical Society A, vol. 144, pp. 145-175, 1981.

TOUS74 - Bibliography on estimation of misclassification, Toussaint G.T., IEEE Trans. on Information Theory, vol. 20, pp. 472-479, 1974.

TOUS74 - Recent progress in statistical methods applied to pattern recognition, Toussaint G.T., Proc. 2nd International Conference on Pattern Recognition, pp. 479-488, 1974.

WONG79 - DECA: a discrete-valued data clustering algorithm, Wong A.K.C., Wang C.C., IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-1, pp. 342-349, 1979.

YOUN81 - Quantitative analysis of qualitative data, Young F.W., Psychometrika, vol. 46, pp. 357-388, 1981.

YOUN81 - Statistical pattern classification with binary variables, Young Y.Y., Philip S.L., Rondon R.J., IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-3, pp. 155-163, 1981.

SUMMARY

This thesis was devoted to aspects related to the analysis of medical data bases in the context of pattern recognition. It contains both theoretical aspects and practical applications and its scope includes questions and problems that arise when applying pattern recognition methods and techniques to this type of data.

Two concepts form a central theme:

1 - The occurrence of discrete features, either in isolation (discrete feature spaces) or in combination with continuous features (mixed features spaces).

2 - The occurrence of missing values.

It was attempted to treat these two complications in pattern recognition in a systematic way. In doing so, it soon became clear that it was desirable to develop the treatment of both areas further than was available in the literature. This gave rise to the following subjects:

1 - Theoretical work leading to a new error bound. It is valid for a certain type of classifier. Since the bound is an expected value with respect to all possible training sets of a given size, it takes into account sample fluctuations.

2 - Development of new methods for feature selection. Specifically, these new methods were built around the concept of structure selection as opposed to the concept of feature selection. A simulation study, presenting results in a well controlled environment, was also performed.

3 - Analysis of the problems associated with the occurrence of missing values. New approaches to estimate missing values were presented and a simulation study was performed in order to better understand the effects of incomplete data in the design and use of classifiers. Both discrete and continuous data were considered.

4 - Analysis of two different medical data bases, where some of the techniques described or proposed, were applied. The two medical data sets used were a data set containing records of patients with acute chest pain, and a data set containing records of severely head injured patients.

Briefly, it can be said that, given a major general problem (in our case, the application of pattern recognition techniques to medical data bases), specific problems were identified and solutions were devised and/or proposed.

PATROONHERKENNING MET DISCRETE EN GEMENGDE GEGEVENSBESTANDEN: THEORIE EN PRAKTIJK.

SAMENVATTING.

Dit proefschrift behandelt enkele aspecten van de analyse van medische gegevensbestanden met behulp van patroonherkenningstechnieken. Het beschrijft zowel theoretisch onderzoek als enkele practische toepassingen. De vraagstelling van het onderzoek was het identificeren en oplossen van problemen, die zich bij dit soort gegevensbestanden voordoen.

Twee complicerende factoren bij de analyse van medische gegevensbestanden werden onderzocht:

1. Het optreden van discrete kenmerken, hetzij in isolatie (in zuivere discrete kenmerk-ruimtes) dan wel in combinatie met continu variërende kenmerken (gemengde kenmerk-ruimtes).

2. Het optreden van incomplete kenmerk-vectoren.

Bij de systematische behandeling van deze twee complicerende factoren werd al snel duidelijk, dat het nodig was om de theoretische grondslagen, zoals die uit de literatuur bekend zijn verder te ontwikkelen. Dit gaf aanleiding tot de volgende onderwerpen, die in dit proefschrift aan de orde komen:

1. Theorievorming met als resultaat de afleiding van een nieuwe bovengrens voor de foutenkans van een bepaald type klassificator. Deze bovengrens is een verwachtingswaarde over alle mogelijke leerbestanden en verdisconteert daarom de fluctuaties in de leerverzameling.

2. De ontwikkeling van nieuwe methodes van kenmerk-selectie, gebaseerd op het principe van de selectie van de structuur van de kenmerk-ruimte. Resultaten van een simulatie studie aan de hand van gegenereerde bestanden worden in het proefschrift beschreven.

3. Analyse van de problemen verbonden aan het vóórkomen van incomplete kenmerk-vectoren. Nieuwe manieren van de schatting van ontbrekende waarden werden ontwikkeld en geëvalueerd d.m.v. een simulatie-studie. Dit leidde tot een beter inzicht in het effect van incomplete kenmerk-vectoren op het ontwerp en het gebruik van klassificatoren. Zowel discrete als continue kenmerken werden hierbij in beschouwing genomen.

4. De analyse van twee medische gegevensbestanden, waarbij sommige van de ontwikkelde technieken werden toegepast. Eén bestand betreft patiënten met acute pijn in de borst, het andere betreft patiënten met ernstig hoofdletsel.

Samenvattend kan gezegd worden, dat bij de analyse van dit soort bestanden een aantal problemen werden geidentificeerd, waarvoor oplossingen werden ontwikkeld, die vervolgens aan de praktijk werden getoetst.

ACKNOWLEDGEMENT

The research presented in this thesis was done at the Department of Medical Informatics at the Free University in Amsterdam.

I would like to express my gratitude both to Prof. E. S. Gelsema and Prof. J. H. van Bemmel for their trust, encouragement, advice and stimulating criticisms.

Furthermore, I am also indebted to all my other colleagues in the Department of Medical Informatics for their companionship and for creating a stimulating working environment. Without trying to be exhaustive, I would like to name Gert Landeweerd, Teun Timmers, Arie Hasman, Jan Talmon, Jan Kors, Johan v. d. Lei, Jan Trim, Ed de Moel and Arnold Smeulders.

The financial support of S.T.W. (Stichting voor de Technische Wetenschappen), which allowed for my stay at Medical Informatics, is here also acknowledged.

Finally, I would like to thank the support of I.N.I.C. (Instituto Nacional de Investigação Cientifica) for the printing of this thesis.