CLUSTER ANALYSIS AND CLASSIFICATION OF PROCESS DATA BY USE OF PRINCIPAL CURVES

by

CORNELIA SUSANNA VAN COLLER

thesis submitted in
partial fulfilment of the
requirements for the degree of

MASTER

of

ENGINEERING SCIENCE
(Mineral Processing)

in the Department of
Chemical Engineering
at the

University of Stellenbosch

Supervisors:
Prof. C. Aldrich
Prof L. Lorenzen

-November 1999-

DECLARATION

I, the undersigned, hereby declare that the work contained in this thesis is my own original work, except where specifically acknowledged in the text. Neither the present thesis, nor any part thereof, has previously been submitted to any other university.

C.S. van Coller

November 1999

SYNOPSIS

In this thesis a new method of clustering as well as a new method of classification is proposed. Cluster analysis is a statistical method used to search for natural groups in an unstructured multivariate data set. Clusters are obtained in such a way that the observations belonging to the same group are more alike than observations across groups. For instance, long datarecords are found in mineral processing plants, where the data can be reduced to clusters according to different ore types. Most of the existing clustering methods do not give reliable results when applied to engineering data, since these methods were mainly developed in the domains of psychology and biology.

Classification analysis can be regarded as the natural continuation of cluster analysis. In order to classify objects, two types of observations are needed. The first are those observations whose group memberships are known *a priori*, which can be acquired through cluster analysis. The second kind of observations are those whose group memberships are unidentified. By means of classification these observations are allocated to one of the existing groups.

Both of the proposed techniques are based on the use of a smooth one-dimensional curve, passing through the middle of the data set. To formalise such an idea, *principal curves* were developed by Hastie and Stuetzle (1989). A principal curve summarises the data in a non-linear fashion. For clustering, the principal curve of the entire unstructured data set is extracted. This one-dimensional representation of the data set is then used to search for different clusters. For classification, a principal curve is fitted to every known group in the data set. The observations to be assigned to one of the known groups are allocated to the group closest to the new point.

Clustering with principal curves grouped engineering data better than most of the well-known clustering algorithms. Some shortcomings of this method were also established. Classification with principal curves gave similar, optimal results as compared to some existing classification methods. This classification method can be applied to data of any distribution, unlike statistical classification techniques.

OPSOMMING

In hierdie tesis word 'n nuwe metode elk vir trosanalise en klassifikasie analise voorgestel. Trosanalise is 'n statistiese tegniek waarmee natuurlike groepe in 'n ongestruktureerde meerveranderlike datastel gevind word. Groepe word op so 'n wyse verkry dat die waarnemings in dieselfde groep meer eenders is as waarnemings tussen groepe. Byvoorbeeld, in mineraalaanlegte is lang datarekords algemeen, wat deur middel van trosanalise gereduseer kan word na verskillende groepe, ooreenkomstig verskillende ertstipes. Die meerderheid bestaande groeperingsmetodes lewer nie betroubare resultate in hul toepassing op ingenieursdata nie, aangesien hierdie tegnieke meestal hul oorsprong in die sielkundige en biologiese velde het.

Klassifikasie analise kan gesien word as die natuurlike opvolging van trosanalise. Om objekte te klassifiseer, word gebruik gemaak van twee soorte waarnemings. Die eerste tipe is daardie waarnemings met *a priori* bekende groepsidentiteite, wat deur trosanalise gevind kan word. Die tweede soort is die waarnemings met onbekende groepsidentiteite. Elkeen van hierdie waarnemings kan deur middel van klassifikasie toegewys word aan een van die bestaande groepe.

Beide hierdie voorgestelde tegnieke is gebaseer op die gebruik van 'n gladde, eendimensionele kromme wat deur die middel van die datastel beweeg. Om hierdie idee te formaliseer, is *hoofkrommes* ontwikkel deur Hastie en Stuetzle (1989). 'n Hoofkromme gee 'n nie-lineêre opsomming van die data. Vir groeperingsdoeleindes word 'n hoofkromme uit die algehele ongestruktureerde datastel onttrek. Met klassifikasie word'n hoofkurwe aan elke bekende groep in die datastel gepas. Die waarneming wat aan een van die bestaande groepe toegewys moet word, word in die groep naaste aan die betrokke punt geplaas.

Groepering met behulp van hoofkrommes, het met ingenieursdata beter resultate gelewer as meeste van die bestaande tegnieke. Deur middel van praktiese voorbeelde is sekere tekortkominge van hierdie groeperingsmetode vasgestel. Klassifikasie met behulp van hoofkrommes lewer soortgelyke, optimale resultate as die van bekende vergelykende tegnieke. Die voorgestelde klassifikasie tegniek kan toegepas word op datastelle van enige verdeling, in teenstelling met die statistiese klassifikasietegnieke.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to the following persons for guidance, help and friendly assistance:

Prof. C. Aldrich
Prof. L. Lorenzen
Braam van Dyk
Juliana Steyl

To My Parents

CONTENTS

SIMULATED DATA SETS:

PROCESS DATA:

SIMULATED DATA:

PROCESS DATA:

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Cluster analysis is a grouping technique that can be applied in many different fields. This method fuses observations, which are similar in some sense, into the same group. Observations that do not have mutual characteristics are said to be dissimilar. Such observations belong to different groups or *clusters*. For instance, in order to maximise profits and to minimise the costs on chemical and metallurgical plants, the functional settings need to be adjusted regularly. For example, in the petrochemical industry blending operations necessitate frequent modification to avoid changes in crude oil feedstock (Aldrich, 1999). The physicochemical phenomena underlying the behaviour of the plant, needs to be understood thoroughly in order to make these adjustments. Unfortunately, this is not the case; therefore, engineers often have to rely on historic plant data analyses to anticipate the behaviour of the plant regarding changing conditions. Historic plant data show a tendency to cluster around major process changes, for instance the shutdown or start-up of manufacturing units, malfunctioning equipment or the introduction of new reagents, etc. Thus, cluster analysis can produce valuable information concerning process conditions.

Classification analysis is a statistical method that can be seen as the natural succession of cluster analysis. In the latter method, different clusters are identified. Classification analysis makes use of these clusters where new observations, whose group memberships are not known, are compared to the different groups. The observation is allocated to that group with which the new observation has the most features in common.

The majority of the existing clustering techniques was developed to cluster qualitative data, as found in fields such as sociology and psychology. Therefore, the application of these methods to engineering data, which are quantitative, leads to incorrect and senseless results. A clustering technique is proposed in an attempt to find a grouping method, which yields more reasonable clustering results of engineering data. This clustering method is based on the internal structure of the multidimensional data set. The internal structure of the data is obtained through a technique called *principal curves* (Hastie and Stuetzle, 1989). The similar concept is applied to classification analysis, where the internal structure of each of the identified groups is obtained. New observations are compared to every group's structure in order to assign each new point to exactly one of the known groups.

CHAPTER 2

BACKGROUND

Cluster analysis and classification analysis are two broad categories of the same subject, namely that of classification. However, they relate to different aspects. The former technique relates to situations where no *prior* knowledge concerning the structure of the multivariate data set is available. A good exploratory procedure is to search the structure of the data for 'natural' groups (Johnson and Wichern, 1988). This technique needs no assumptions regarding the number of groups or the group structure present within the data set. These groupings are determined in such a way that the units within the same group are more alike or homogeneous than objects belonging to other groups. These groups are called *clusters*. The number and features of the clusters are totally data-dependent. This form of classification problem is known as *unsupervised learning* or in statistical terms it is labelled as *cluster analysis*.

The second technique, *classification analysis,* applies in situations where the available data set consists of *k* distinct identifiable classes or groups, thus it is known to which groups the observations belong. Furthermore there are those objects with unidentified group membership, in terms of these known groups. The goal is to obtain a rule by which to assign these additional, new observations to one of the *k* prespecified groups. This second event is known as *supervised learning* or, as already mentioned, as *classification analysis* in statistical terminology.

Figure 2.1 and 2.2 illustrates the difference between cluster analysis and classification analysis, respectively.

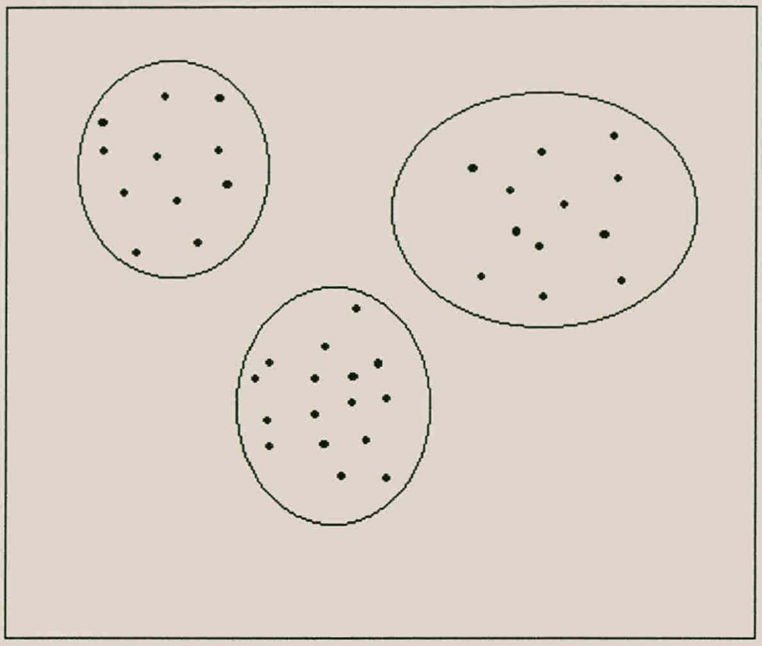

Figure 2.1: Cluster analysis seeks distinct groups present in the data.

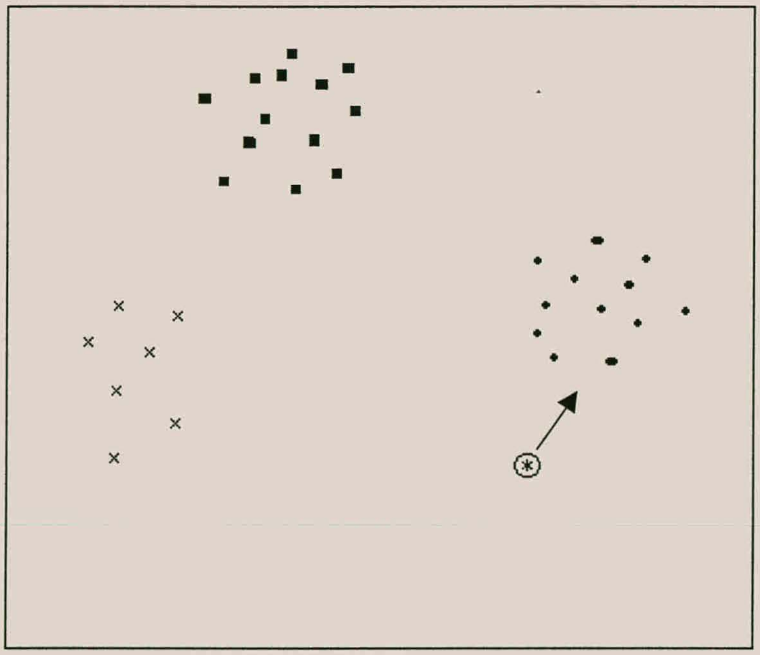

Figure 2.2: In classification analysis the distinct groups are known beforehand and 'new' points are assigned to one of the existing clusters.

2.1 Cluster Analysis

Cluster analysis is a statistical technique used to search for 'natural' groups in an unstructured set of multidimensional data. This is an important tool for exploratory data analysis and is often used in conjunction with other analyses. Objects that belong to the same group are known to be similar in some sense, whereas the opposite holds for objects that belong to different groups, which are said to be dissimilar. Clustering can either be carried out on the variables or the observations. Cluster analysis originated in the fields of biology and zoology where it was known under the name of taxonomy, however, it was not a very scientific method (Everitt, 1974). The techniques became progressively more objective, leading eventually to the development of numerical taxonomy methods, based on the ideas of Adanson (18th century).

According to the Encyclopaedia of Statistical Sciences (1981) it was in the biological and sociological disciplines where the first attempts were made to formally approach clustering, (Zubin, 1938). Only recently, in the 1950s, when more powerful computing systems became available, researchers of the mathematical and statistical disciplines started to formalise clustering methods. Owing to this, a vast number of different clustering algorithms started to see the light even to the extent that it was introduced as an independent scientific field (*Journal of Classification*, first published in 1984 and the International Federation of Classification Societies, founded in 1985).

2.1.1 Applications of Cluster Analysis

In mineral processing, the visual appearance of the froth phase is used as a controlling mechanism of industrial flotation plants (Moolman et al., 1995). Other than this, the experience of a human operator has a direct bearing on the stability of the plant. It is clear why these kinds of processes are not optimally controlled. Digitised images of the froth structure can be extracted and cluster analysis can be performed on this data to unravel the kind of structure (number of different groups) present within the data. This information can further be employed in studies such as discriminant analysis to obtain some knowledge concerning the different clusters and also to classify new observations at later stages. All this may help to interpret images with regard to the behaviour and control of the plant.

In mechanical and automation engineering the design of a cellular construction system starts with two group formation tasks, namely part-family formation and machine-cell formation (Wang, 1998). The former activity clusters parts with alike geometric features, whereas the latter assembles dissimilar machines, dedicating them to the manufacture of one or more part families. Cluster analysis is employed to find part-family or machine–cell representatives in order to obtain a set of the most extreme parts or machines. These representatives are utilised in further analyses (a linear assignment model) to obtain a highly effective group formation algorithm for machine-cell and part-family formation.

There is substantial interest in the petroleum industry in testing new blending components in such a way as to relate the blending performance of a particular octane-quality enhancing component to its concentration, as well as to the properties of the base fuel (Zemroch, 1986). The postulated model includes terms in base-fuel properties. Cluster analysis can be used to obtain a subset of design points from a list of base-fuel candidates so that these selected points have as even a spread as possible over the design space. An assumed model can be efficiently estimated from a design with such evenly spread points.

Environmental studies can also benefit from the use of cluster analysis. Neural networks together with clustering techniques, in particular a hierarchical clustering method, was employed to investigate a pollution problem in Germany. It was discovered that a considerable source of dioxin found in the river Elbe, soils from the flood plains of the river Elbe, the Hamburg harbour and soils originating from dredging materials, originated from the dioxin contaminated region of Bitterfield (Götz et al., 1998). It was indicated that metallurgy processes, as well as chemical production, lead to dioxin contamination of the Bitterfield region. Moreover, the main dioxin source accountable for the Hamburg contamination, not affected by the river Elbe, was of thermal origin and the cause was most likely a plant in Bitterfield.

In process engineering, poor running of production machinery undoubtedly has a negative influence on the company. The occurrence of serious process variations causes stable process settings to move towards unstable regions. The incidence of production faults may not easily be resolved and in situations system failures cannot be prevented, it may lead to temporary machine shutdowns. Defective products either result in product waste or the products necessitate the process of re-work. Owing to the increasing pressures on industrial demands, the development of methods aimed at improving different aspects of production strategy is imperative. Process problems that can be described by large unwanted variations in machine processes, can usually be seen in the physical evidence of the data. By means of cluster analysis, Sutanto and Warwick (1995) studied the complex internal processes of an industrial production machine in order to improve product quality and machine efficiency. They also demonstrated how to categorise machine behaviour and how to identify regions of good and poor machine behaviour. The clustering technique separated areas of the process space, describing different types of machine states. This information on machine behaviour is important where diagnosis and predictions of machine behaviour must be made. For instance, it is then possible to predict when a machine is expected to fail and thus the necessary precautions can be taken.

Cluster analysis was also applied in a situation of soil pollution, where it assisted in locating existing pollution patterns as well as the detection of main emission sources. The problem occurred at the Maxhütte Unterwellenborn, a large metallurgical plant in Thuringia (Germany). A cement mill was situated on the north-eastern part of the plant. Over a period of four decades the topsoil surrounding the plant was contaminated with heavy metals through dustlike emissions. After samples were drawn, the heavy metals were digested with aqua regia. Some of the heavy metals were analysed by means of atomic absorption spectrometry and inductively coupled plasma-atomic emission spectrometry (Einax and Soldt, 1999). The results obtained through cluster analysis showed the presence of two maim clusters; the first group contained heavily polluted points and the second group consisted of moderately and slightly polluted points. However, the data were split up into three groups, namely heavily, moderately and slightly polluted observations. Using these *a priori* classes as input into other multivariate statistical procedures, the three groups of different pollution states were confirmed. These detected pollutant patterns indicated the sources of emission.

2.1.2 Background on Cluster Analysis

The aim of the research and the type of input data will determine which clustering procedure to follow, as the numerous different combinations of algorithms and input structures result in just as many different solutions. These different clustering methods provide clusters with uncommon features, even to the same problem. Therefore, it is imperative to know exactly what kind of input you have and the results you anticipate.

The input structures for the clustering algorithms can take either the form of:

- an $n \times p$ matrix, where the rows and columns correspond to the observations and the variables, respectively. This structure is known as *two-mode* because the row and column units are different; or
- an $n \times n$ proximity matrix, where the entries can either be *similarities* or *dissimilarities* between all pairs of objects. This proximity matrix is said to be *one-mode* since the row and column units are the same.

The entries in a *dissimilarity* matrix are usually metric distances between all pairs of objects, but there are also other ways in which they can be defined. For instance, a dissimilarity matrix can also consist of subjective evaluations from more than one individual, typically found in the social sciences or market research, such as evaluating different groups' behaviour regarding sports being played on Sundays. Due to this fact we speak of dissimilarities instead of distances. Entry *(i, j)* in a dissimilarity matrix indicates to what degree objects *i* and *j* differ. The greater this value, the more these two objects differ.

The distance often used for cluster analysis is the *Euclidean* distance, defined as follows

$$d(\mathbf{x}\,,\mathbf{y}) \;=\; \sqrt{(x_1-y_1)^2+(x_2-y_2)^2+\ldots+(x_p-y_p)^2} \qquad (2.1)$$

$$= \sqrt{(\mathbf{x}-\mathbf{y})'(\mathbf{x}-\mathbf{y})}$$

where $(\mathbf{x}\,,\mathbf{y})^1$ is the transpose of the venctor $(\mathbf{x} - \mathbf{y})$

Equation (2.1) corresponds to the square on the hypotenuse in p dimensions or, in other words, the straight-line distance between the points with co-ordinates $(x_1,x_2,\ldots,x_p)$ and $(y_1,y_2,\ldots,y_p)$. In the situation where the variables do not have equal variances, or there is a correlation among them, the Euclidean distance is not useful since this distance measure does not incorporate the variances or covariances of the variables. However, standardising the variables to unit variance prior to computing the Euclidean distances, can solve the problem of unequal variances.

The statistical distance, the *Mahalanobis* distance, simultaneously resolves both cases where the Euclidean metric fails, as it standardises all variables to the same variance in the sense that a random variable with a larger variance than another, receives relatively less weight. Equally, two highly correlated variables contribute not as much than two less correlated variables, thus the Mahalanobis distance eliminates correlations. The Mahalanobis distance is:

$$d(\mathbf{x}\,,\mathbf{y}) \;=\; \sqrt{(\mathbf{x}-\mathbf{y})'\mathbf{S}^{-1}(\mathbf{x}-\mathbf{y})} \qquad (2.2)$$

where **S** is the pooled within groups covariance matrix. Owing to the use of the inverse of the matrix **S**, correlations between variables are eliminated and the variables are standardised to the same variance. However, without prior information regarding the different groups, this measure of distance cannot be computed (Johnson and Wichern, 1988). Consequently, the Euclidean distance is the most popular choice of a distance function. Additional distance metrics include the *Manhattan* (city block) distance, which is the sum of absolute differences across variables for two observations. The *Minkowski* distances is yet another distance used, with the Euclidean and Manhattan distances being special cases of the Minkowski distance, as mentioned by Kaufman and Rousseeuw (1990). Both the Euclidean distance (2.1) and the Manhattan distance satisfy the following axioms:

(D1) $d(i, j) \geq 0$

(D2) $d(i, i) = 0$

(D3) $d(i, j) = d(j, i)$

(D4) $d(i, j) \leq d(i, k) + d(k, j)$

As said previously, not all dissimilarities are computed by means of distances and in general, dissimilarities satisfy **D1** through **D3** but **D4** is not met. From these inequalities (D1 – D3) we can conclude that a dissimilarity matrix is a non-negative symmetric matrix with zeros on the diagonal.

A *similarity* matrix is in effect the opposite of a dissimilarity matrix. The similarity coefficient $s(i, j)$ signifies the likeness of objects i and j. A similarity coefficient always falls within the range between 0 and 1. As this coefficient approaches one, the more two objects are alike, and vice versa.

For a similarity matrix, the following specifications hold:

(S1) $0 \leq s(i, j) \leq 1$

(S2) $s(i, i) = 1$

(S3) $s(i, j) = s(j, i)$

Again we can conclude from these statements that the similarity matrix is a symmetric matrix with zero-entries on the diagonal, and the off-diagonal elements lie between 0 and 1. Similarities are not computed from distance functions, but are obtained through other ways such as opinions on a certain idea. Some clustering methods require as input a dissimilarity matrix and when the actual input is similarities, they can easily be transformed to dissimilarities through the following equation:

$$d(i, j) = 1 - s(i, j) \tag{2.3}$$

Clustering algorithms can primarily be grouped into two main groups, viz., *hierarchical* and *non-hierarchical/partitioning* methods, though other procedures are available but are used less frequently.

2.1.2.1 Hierarchical Clustering

As the name suggests, hierarchical clustering forms a hierarchical grouping of the objects. Hierarchical techniques can be used to group either the attributes or the items. We are concerned with two types, namely that of *agglomerative* and *divisive* clustering methods. These two hierarchical techniques work in opposite directions (Johnson and Wichern, 1988). In the former case, each object initially forms a cluster on its own and with each successive step the two closest or most similar objects are fused into one cluster. This procedure repeats itself until all the objects belong to the same cluster. Divisive clustering requires the dissimilarity matrix measured in distances as input, whereafter the procedure starts with all the objects belonging to one cluster. At the first step the cluster is split into two parts in such a way that the objects from the distinctive clusters are as dissimilar as possible. This subdividing continues in this manner until each object forms its own little cluster.

The input structure used for agglomerative clustering is either one of the proximity measures, i.e. similarity or dissimilarity measures. A number of algorithms exist in this section and they all operate according to the same fundamental pattern, where linkage methods are of the better-known techniques. *Single link* or the *nearest neighbour* method sequentially fuses those two groups having the smallest distance of all the closest two members belonging to distinct groups. The linkage method where the smallest distance of all the distances between the *furthest members (neighbours)* from different groups is the decisive factor is called *complete link. Average link* combines those two clusters having the shortest average dissimilarity among all pairs of objects in the respective groups, with pair members belonging to distinct clusters. The different linkage methods are demonstrated in figure 2.3. *Centroid cluster analysis, median cluster analysis* and *Ward's method* are some of the other agglomerative methods available.

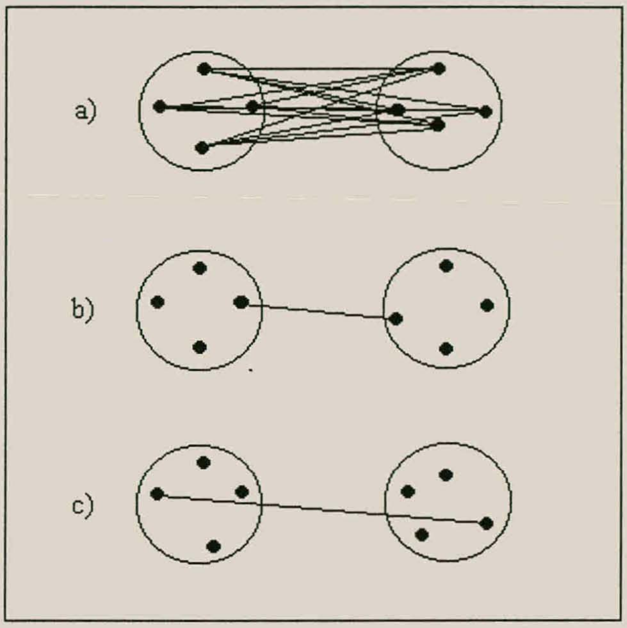

Figure 2.3: Demonstration of the different linkage methods: (a) Average link (b) Single link (c) Complete link.

The results obtained by agglomerative and divisive clustering procedures can graphically be displayed in a two-dimensional graph known as a *dendrogram.* A disadvantage of agglomerative clustering is that the reallocation of objects are not allowed, since once an object has been fused into a cluster-entity, the process cannot be reversed as this fusion is permanent. Divisive clustering does not have the problem of suffering from initial decisions, as the large clusters are established first. However, the drawback of this technique is a computational one (Kaufman and Rousseeuw, 1990). Considering all possible combinations of two distinct clusters in the first step results in an enormous amount of calculations, even for small data sets. This is the reason why this technique has not received much attention as a clustering method. The single link method tends to fuse clusters whenever the groups are not sharply separated, i.e. the clusters are touching. This tendency is referred to as *chaining*. Ward's method and the other two linkage methods have the propensity to find spherical clusters in data, even though the real clusters take on other forms. Therefore, it seems that these methods inflict certain structures on the data instead of extracting the true ones.

2.1.2.2 Non-Hierarchical Clustering

Non-hierarchical clustering techniques are also known as partitioning methods since they divide the objects into k clusters, where this integer k has to be provided by the user. This differs from hierarchical clustering where the number of groups is not set *a priori*. The main difference between hierarchical clustering and this method, apart from the fact that the raw data can be used as input in the latter case, is that this procedure allows objects to be regrouped if it becomes evident at a later stage that items had previously been incorrectly clustered.

Partitioning methods are not only used to reveal the 'natural' groups present in the data, but can also be used to inflict a certain configuration on the data. For example, dividing a city up into different polling stations for elections. Certainly, not all choices of k will yield the 'natural' clusters or the kind of clusters one wishes to acquire, therefore, it is recommended to run the algorithm several times with different values of k and choose that k which reveals the most meaningful results and interpretations.

Partitioning around medoids is one of the non-hierarchical techniques that is used on a regular basis. This algorithm searches for k representative objects, known as medoids, which are centrally situated within the clusters (cf. Kaufman and Rousseeuw, 1990). The medoids are computed in such a way that the average distance of all the objects to their 'closest' representative object is minimal. The clusters are then found by appointing the remaining objects to the group with the most similar medoid.

K-means clustering introduced by MacQueen (1967), is possibly the best known of the partitioning methods. The steps involved in this procedure are:

- Select an initial set of k clusters;
- Move each observation to the cluster with the nearest centroid/mean and recalculate both clusters' means receiving and losing the object; and
- Continue in this manner until no reassignments are made.

As with the k-medoid method, the distance referred to here is usually the Euclidean distance. These two partitioning methods (k-medoids and k-means) reveal a resemblance; where the first method attempts to minimise the average distance, k-means tries to minimise the average *squared* distance. The first step in k-means can be changed so that k initial centroids (seed points) are specified instead of clusters.

Fuzzy analysis is yet another non-hierarchical technique. This method stands apart and has nothing in common with the two previous techniques. It is said that the above mentioned methods are hard or clear-cut clustering methods since each object is allocated to one and only one cluster. An item that lies between two clusters must be assigned to one of the clusters. Fuzzy analysis, on the other hand, spreads out each object over all the different clusters and assigns a membership coefficient θ_{ij} to each object i and cluster j (Struyf et al, 1997). This coefficient θ_{ij}, specifies to which extent item i belongs to cluster j.

The membership coefficients satisfy the following:

- $\theta_{ij} \geq 0$ for all $i = 1,\ldots,n$ and all $j = 1,\ldots,k$
- $\sum_{j=1}^{k} \theta_{ij} = 1 = 100\%$ for all $i = 1,\ldots,n$

Thus, fuzzy clustering does not allocate an observation to exactly one cluster, as is the case with the other two methods. To illustrate this method, figure 2.4

shows points that can be classified with confidence as well as points of which their cluster membership is not quite clear.

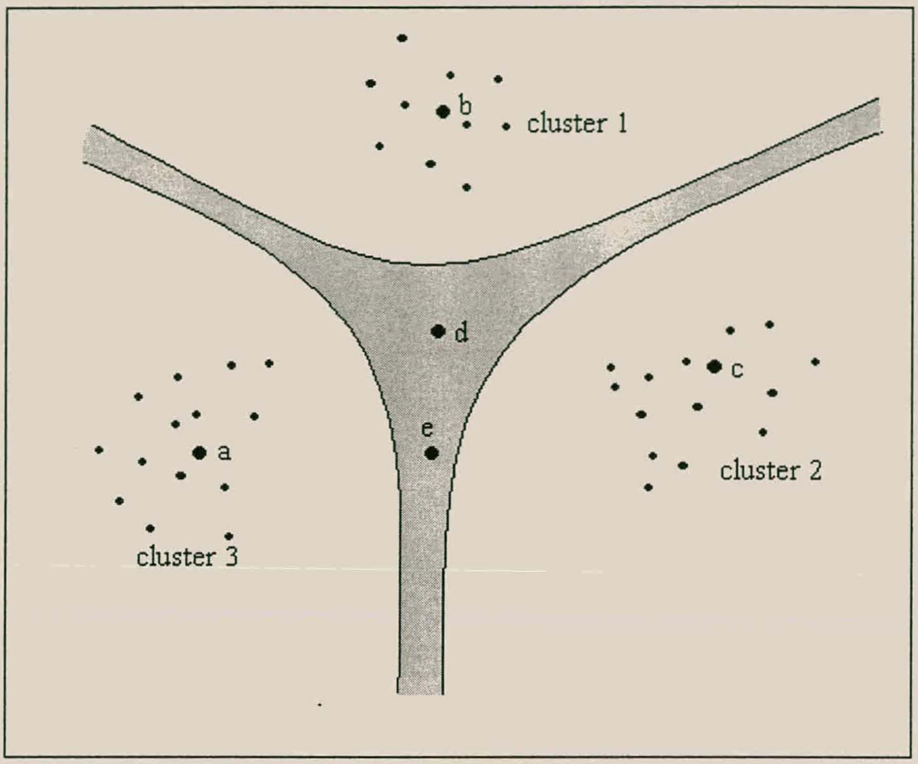

Figure 2.4: Illustration of fuzzy analysis clustering intermediate points.

First of all, there is no doubt that observations **a**, **b** and **c** belong to clusters three, one and two respectively. However, items **d** and **e** are more difficult to group since object **d** lies almost in the middle amongst all the groups and item **e** are approximately centrally located between clusters two and three. By the use of membership coefficients, fuzzy analysis solves this problem differently than other methods would. Table 2.1 lists these coefficients.

Table 2.1: Membership coefficients corresponding to figure 2.4.

	MEMBERSHIP COEFFICIENTS		
Object	**Cluster 1**	**Cluster 2**	**Cluster 3**
a	0.05	0.05	0.90
b	0.90	0.05	0.05
c	0.05	0.90	0.05
d	0.34	0.33	0.33
e	0.10	0.45	0.45

Considering the membership coefficients in the above table, we see items **a**, **b** and **c** belong mainly to clusters three, one and two, respectively. Because item **d** is situated almost in the centre between the three groups, it belongs for 34% to cluster 1 and for 33% to clusters two and three. Observation **e** is not as ambiguous as item **d**. It lies almost halfway between groups two and three, thus it belongs for 45% to both these groups and only for 10% to group 1.

The k-medoids are computed, unlike k-means where the initial clusters or seed points are chosen at random. Both these techniques assume that the clusters they seek are spherical and this is a drawback, because if the data's actual clusters are of different shapes, they will most probably not be discovered. There is a good chance that the natural groups in the data will not be discovered since the number of clusters *k*, to be distinguished, are set beforehand by the user. Fuzzy analysis returns as output an $n \times k$ matrix with the membership coefficients as the entries, thus interpreting and deciding on the final clusters is a difficult task purely because of its size.

2.1.2.3 Density Seeking Techniques

This clustering procedure concentrates on what is probably the most instinctive and logical approach to cluster analysis. By portraying the data as points in metric space, clusters will noticeably be the areas that embody dense collections of points. Clustering techniques that search the data for areas of high densities or modes fall in this category and each mode in the data is indicative of a cluster. It is mainly due to the weakness of the single link hierarchical clustering technique, namely chaining, that led to the development of many of these kind of algorithms (Everitt, 1974).

Mode analysis, developed by Wishart (1969), is the best-known method in this category. A sphere of radius *r*, which surrounds every observation, is searched for 'dense points'. Starting with a small *r*, the number of other points that fall within the observation's sphere is counted and if there is at least a specified number, say *k*, then the centre point/observation is called a *dense point* (Chatfield and Collins, 1980). The remaining points, for which less than *k* other points fall within their sphere, are labelled *non-dense points*. This radius *r* is steadily increased so that the larger the *r,* the more points become *dense* until all the points fall within the same sphere. There are four possible options regarding the introduction of each new dense point:

- The distance between the new point and all the other dense points is greater than *r*. In this situation the new point forms the nucleus of a new cluster and so the number of clusters increases by one;
- The new point falls within a distance of less than *r* from at least one of the other dense points belonging to only one cluster, and thus is added to the same cluster;
- This new point falls within a distance of less than *r* from more than one dense point, with the dense points belonging to distinct groups, which leads to the fusion of the different clusters; and
- With each introductory step of a new point, the smallest distance *d*, among dense points belonging to separate clusters is calculated and compared to some threshold value. If *d* falls beneath this threshold value, these separate

clusters are united. This threshold value is the average of the *2k* smallest distances from each individual new point.

This method proposed by Wishart (1969) is scale-dependent, furthermore the method assumes that the modes are spherical and this drawback may lead to the masking of genuine multimodality in the case where the modes are not spherical.

2.1.2.4 Mixture Models

This method is more formal than any of the techniques discussed previously since it has a probabilistic approach (Everitt and Dunn, 1991). Some researchers consider probability models for the proximity matrices, whereas others prefer to use the raw data, with a certain kind of probability density function as its model, called a *finite mixture density*. Thus, there is no need to decide upon a proximity measure prior to the clustering. A model for *k* clusters can be given by the population density

$$f = \sum_{j=1}^{k} p_j f_j \qquad (2.4)$$

This is a mixture of components f_j, in proportions p_j, which respectively represent the underlying density function of cluster *j* and the proportion of objects belonging to group *j*. A general assumption is that all f_j belong to the same parametric family. For two univariate normal groups (2.4) becomes

$$f(x) \; = \; pN(\mu_1, \sigma_1) + (1\text{-}p)N(\mu_2, \sigma_2) \qquad (2.5)$$

where *p* is the proportion of objects in group 1 and the parameters μ_1, μ_2, σ_1 and σ_2 represent the means and standard deviations of the continuous variable *x* for groups one and two, respectively. As can be seen, this method involves the estimation of parameters μ_1, μ_2, σ_1, σ_2 and p, which is usually performed by means of maximum likelihood estimation. Observations are assigned to that group for which their probability of belonging is the maximum. This is,

$$P(j|\, x) = p_j f_j(x) \,/\, f(x) \tag{2.6}$$

is maximised, where $P(j|\ x)$ is the posterior probability that observation x belongs to group j. The extension of equation (2.5) to k groups is

$$f(x) = \sum_{j=1}^{k} p_j N\left(\mu_j, \sigma_j\right) \tag{2.7}$$

In the case of the data being multivariate normal (MVN), equation (2.7) becomes

$$f(x) = \sum_{j=1}^{k} p_j MVN(\mu_j, \Sigma_j) \tag{2.8}$$

It is usually assumed that all f_j are from the multivariate normal family. However, the application of equation (2.4) is not restricted to multivariate normality but may be applied to general sample spaces. For more discussion on this topic the reader is referred to Everitt and Hand (1981).

Due to the fact that there may be more than one solution to the maximum likelihood equations, this method suffers from the problem of sub-optimal solutions.

2.1.2.5 Other Clustering Techniques

As mentioned in the beginning of this chapter, there is a wide variety of clustering algorithms and only the more popular and general techniques have been introduced. Clumping techniques, where the clusters are in fact allowed to overlap, the inverse method or Q-factor analysis, latent structure analysis and Sammon maps, are just a few of the numerous different available techniques. Everitt (1974) provides a thorough overview of all the different clustering methods.

As noted previously, variables that vary remarkably will have a definite effect on the results since the variable with the largest variance has the greatest impact on the clustering. This also holds for variables that are not commensurate. To overcome these problems the variables can be scaled to unit variance, unless some variables are more important than others, and it is possible to quantify their relative importance. Another aspect to bear in mind is correlation between variables. The solution is to perform the clustering on the principal components retained (Green, Frank and Robinson, 1967). Principal components linearly summarise the original data in such a way that the first few components account for almost all the information in the data set and the newly acquired variables (components) are not correlated. However, Press (1982) argues that information lost through the deletion of the last few components, especially in small samples, results in erroneous findings.

Another problem that can affect results tremendously, is that of outliers. For instance, in the case of partitioning methods an outlier can lead to the formation of a cluster with very dispersed group members. Also, it is known that hierarchical clustering is not robust to outliers. Therefore it is essential to solve this problem prior to any clustering.

2.2 Classification Analysis

Classification analysis is the second of two objectives or goals concerning the separation of groups. There are two objectives because there are two kinds of observations. Firstly, there are those observations with known group membership, called the *training samples*. Secondly, there are items for which no information regarding their group identity is available. The objective is to classify these objects as belonging to one of the k groups in the training samples. These observations are known as the *test samples*.

The first part of the two-fold procedure of group separation deals with the *descriptive* aspect, where the goal is mainly to obtain linear functions (*discriminant functions*). These functions combine the p attributes linearly in order to maximally elucidate the differences between the k distinct groups of the training samples. Via these discriminant functions the relative contribution of each of the p variables, regarding group separation, can be identified as well as the *discriminant space* is obtained. This representation of the projected points illustrates the construction of the k groups optimally. The descriptive aspect of group separation is called *discriminant analysis.* For a comprehensive overview of the different discriminant functions, the reader is referred to Rencher (1995).

The second stage involves the test samples and is referred to as the *predictive* aspect of discriminant analysis. The unknown points' attributes are evaluated by a linear or quadratic function called the *classification rule*, which is based on the discriminant function. This procedure leads to the assignment of each point to one of the k groups. This procedure is called *classification analysis* to clearly distinguish it from discriminant analysis.

A point to bear in mind is the possibility that the object you wish to classify may in fact belong to a total different group than those included in the training samples. As will be seen in the following sections, most of the classification rules are designed for groups coming from a normal distribution. In practice the data sets are rarely distributed normally especially in the fields of engineering. However, if the measured attributes are not restricted to only a small range of different values, the measurements can be transformed so that their distributions more closely resembles a normal one.

The performance of the classification rule can be evaluated by some available measures. These measures are based on the training samples and not the test samples. The expected cost of misclassification is one of the measures to be considered if access to such information is available, however, this is frequently not available. Results acquired through classification can conveniently be displayed in a *classification table or confusion matrix,* in order to easily count the number of correct and misclassifications. The probability of misclassification, known as the error rate, is the more frequently used evaluation measure. Some of these rates include:

- The *optimum error rate* – if all the parameters are known, this rate can be computed; and
- The *actual error rate* - the probability that the classification rule, based on the present sample, misclassifies a future observation.

Various methods, assuming normality, exist to estimate the error rates. A few estimators that can be applied in any context include:

- apparent error rate;
- leave-one-out ;
- bootstrap;
- cross-validation; and
- a method using a third sample, called the validation sample.

See appendix 1 for more information regarding these techniques.

2.2.1 Classification into Two Groups

The most popular way to assign an observation to one of two groups is by means of the *linear classification rule*, which states that observation $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{ip})$ is assigned to group g_1 if

$$\mathbf{a}'\mathbf{x} = (\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)'\,\mathbf{S}_{pl}^{-1}\mathbf{x} > \tfrac{1}{2}\,(\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)'\,\mathbf{S}_{pl}^{-1}(\bar{\mathbf{x}}_1 + \bar{\mathbf{x}}_2) \qquad (2.9)$$

and it is assigned to group g_2 if

$$\mathbf{a}'\mathbf{x} = (\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)'\mathbf{S}_{pl}^{-1}\mathbf{x} < \tfrac{1}{2}\,(\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)'\,\mathbf{S}_{pl}^{-1}(\bar{\mathbf{x}}_1 + \bar{\mathbf{x}}_2) \qquad (2.10)$$

where $\bar{\mathbf{x}}_1$ and $\bar{\mathbf{x}}_2$, n_1 and n_2 denote the vector means and sample sizes for group 1 and group 2, respectively. $\mathbf{S}_{pl}$ is the pooled sample covariance matrix

$$\mathbf{S}^2_{pl} = \frac{(n_1 - 1)\mathbf{s}_1^2 + (n_2 - 1)\mathbf{s}_2^2}{n_1 + n_2 - 2} \tag{2.11}$$

The sample covariance matrices for groups 1 and 2 are denoted by $\mathbf{s}_1^2$ and $\mathbf{s}_2^2$, respectively. Fisher (1936) proposed the linear discriminant function (LDF), $(\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)'\mathbf{S}^{-1}_{pl}\mathbf{x}$, thus in the two-group situation the linear discriminant and linear classification functions are alike. The methodology (2.9) and (2.10) were developed under the constraint that the two groups have a common covariance structure. No assumptions were made regarding the distributions of the groups. Consequently this procedure is basically nonparametric. Nevertheless, if the two populations are normally distributed with equal covariance matrices, swapping the parameter estimates with the population parameters leads to an optimal performance of the linear classification rule. In other words, it results in a minimum probability of misclassification (P) of the observations. If $P(2|1)$ is the probability of classifying an item from g_1 incorrectly as coming from g_2, and vice versa for $P(1|2)$, then in the above-mentioned situation $P = \pi(1)P(2|1) + \pi(2)P(1|2)$ is a minimum, where $\pi(i)$ is the probability that an item belongs to g_i.

Provided the proportion of observations in each group, p_1 and p_2 with $p_2 = 1 - p_1$, is known beforehand, a variation of the linear classification function can be obtained. The density functions for both groups, $f(\mathbf{x}|g_1)$ and $f(\mathbf{x}|g_2)$, are needed in order to employ the group prior probability estimates. The classification rule becomes: allocate x to group g_1 if

$$p_1 f(\mathbf{x}|g_1) > p_2 f(\mathbf{x}|g_2) \tag{2.12}$$

or otherwise assign it to group g_2. This is known as the Bayes procedure. Equation (2.12) can still be used even when the group prior probability estimates are unknown, by setting both p_1 and p_2 equal to 0.5. This implies that the prior probabilities of an item belonging to either of the groups are equal.

In the situation of the two populations being normally distributed with equal covariance matrices, as mentioned previuosly, equation (2.12) changes to

$$\mathbf{a}'\mathbf{x} = (\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)' \mathbf{S}_{pl}^{-1}\mathbf{x} > \tfrac{1}{2}(\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)'\mathbf{S}_{pl}^{-1}(\bar{\mathbf{x}}_1 + \bar{\mathbf{x}}_2) + \ln \frac{p_2}{p_1} \qquad (2.13)$$

If equation (2.13) holds, then $\mathbf{x}$ is assigned to group g_1. Because the parameter estimates are used, this procedure is only asymptotically optimal, meaning that optimality increases as the sample size enlarges. As can be seen from equation (2.13) if $p_1 = p_2$, then equation (2.13) becomes equation (2.9).

Some noteworthy remarks regarding the two-group LDF, are:

- There is a relation between the LDF and multiple regression;
- Some of the LDF's advantages include its simplicity; the standardised coefficients also reveal information concerning differences among the groups; the already mentioned graphical aid of projections of the points onto the optimal discriminant plane; the fact that the normality assumption is not required; and also the LDF's performance compared to more complicated methods are satisfactory; and
- Variable selection can be done by methods such as *stepwise variable selection* and/or computing of the Mahalanobis squared distances between the two groups for each variable. It is imperative to have the *correct* number of variables included in the discriminant function whereas too many may cause the function not to generalise and too few may not be able to capture the necessary information in the data. See Gnanadesikan (1977) for more information on all these issues.

2.2.2 Classification into Several Groups

In the case where the k groups have similar covariance structures, the population covariance matrix can be approximated with the pooled sample covariance matrix across all groups, i.e.

$$\boldsymbol{S}_{pl} = \tfrac{1}{N-k} \sum_{j=1}^{k} \left(n_j - 1\right) \mathbf{S}_j = \frac{\mathbf{E}}{N-k} \tag{2.14}$$

where n_j and $\boldsymbol{S}_j$ are respectively the size and sample covariance matrix for group j, $N = \sum_j n_j$ and $\boldsymbol{E}$ is the $p \times p$ error matrix with p the number of variables. The error matrix is denoted by

$$\begin{aligned} \boldsymbol{E} &= \sum_{j=1}^{k} \sum_{i=1}^{n} \left(\mathbf{x}_{ji} - \overline{\mathbf{x}}_{j.}\right)\left(\mathbf{x}_{ji} - \overline{\mathbf{x}}_{j.}\right)' \\ &= \sum_{ji} \mathbf{x}_{ji}\mathbf{x}'_{ji} - \sum_{j} \frac{1}{n} \mathbf{x}_{j.}\mathbf{x}'_{j.} \end{aligned} \tag{2.15}$$

The linear classification rule for the several-group case, corresponding to equation (2.9) and equation (2.10) for the two-group situation, is

$$L_j(\mathbf{x}) = \overline{\mathbf{x}}'_j \mathbf{S}_{pl}^{-1} \mathbf{x} - \tfrac{1}{2}\, \overline{\mathbf{x}}'_j \mathbf{S}_{pl}^{-1} \overline{\mathbf{x}}_j \tag{2.16}$$

which assigns item $\mathbf{x}$ to the group for which equation (2.16) is maximum. As in the two-group situation the Bayes procedure can also be implemented for several groups. Observation $\mathbf{x}$ is then allocated to that group for which $p_j f(\mathbf{x}|g_j)$ is a maximum, where $f(\mathbf{x}|g_j)$ is the density function for group j and p_j is the probability estimate of an observation belonging to it. Under the assumption of normality with equal covariance matrices for all the k groups, and the availability of the respective prior probabilities of group membership, $p_1, p_2, \ldots, p_k$, enables the use of

$$L_j'(\mathbf{x}) = \ln p_j + \bar{\mathbf{x}}_j'\mathbf{S}_{pl}^{-1}\mathbf{x} - \tfrac{1}{2}\,\bar{\mathbf{x}}_j'\mathbf{S}_{pl}^{-1}\bar{\mathbf{x}}_j \tag{2.17}$$

Again, observation **x** is assigned to that group *j* which produces the largest value in equation (2.17). In the case where all the group prior probabilities are equal, equation (2.17) becomes equation (2.16).

The linear classification rule cannot be applied when the groups' covariance matrices differ. Observations tend to be classified as coming from the group whose variables vary the most. Instead, the *quadratic classification rule* is employed, which states that **x** belongs to the group yielding the largest $Q_j(\mathbf{x})$, indicated as follows

$$Q_j(\mathbf{x}) = \ln p_j - \tfrac{1}{2}\ln|S_j| - \tfrac{1}{2}\left(\mathbf{x}-\bar{\mathbf{x}}_j\right)'\mathbf{S}_j^{-1}\left(\mathbf{x}-\bar{\mathbf{x}}_j\right) \tag{2.18}$$

Normal distributed groups render equation (2.18) the best classification rule, which in the case of equal covariance structures, reduces to the linear classification rule. When prior probabilities $p_1, p_2, \ldots, p_k$, are not available or equal, then the term $\ln p_j$ should be deleted in equation (2.18). In situations where the population means are equal, this rule overshadows the linear rule. In the case of small samples and when normality is not met, caution should be taken with the use of equation (2.18). In this situation attempts should be made to transform the data to near-normality.

2.1.1 Other Classification Methods

In cases where normality of the data sets cannot be assumed, it is a good idea to contemplate other methods such as *logistic regression classification*. In the two-group case, it is assumed that the groups are from a single sample, i.e., $n = n_1 + n_2$. A dummy variable *y*, is introduced for every item $\mathbf{x}_j$ with $j = 1, 2, \ldots, n$; where $y = 1$ for the n_1 items coming from group g_1, otherwise $y = 0$. Assuming equal prior probabilities of group membership, the linear classification function $\upsilon = \hat{\alpha} + \mathbf{x}'\hat{\beta}$ is employed, where the parameters α and β can be estimated by means of the maximum likelihood method. The likelihood function is

$$\prod_{j=1}^{n_1}\left(\frac{\exp\left(\alpha + \mathbf{x}'_j\beta\right)}{1+\exp(\alpha + x'_j\beta}\right) \cdot \prod_{i=n_1+1}^{n}\left(\frac{1}{1+\exp\left(\alpha + \mathbf{x}'_i\beta\right)}\right) \tag{2.19}$$

with the estimates chosen so as to maximise equation (2.19) and $\beta = [\beta_1, ..., \beta_p]$, where p is the number of variables. The scores obtained via the linear classification function υ are used to allocate the unknown observations. If an item's score is positive it is assigned to group g_1, whereas a negative score will lead to the observation's allocation to group g_2. Variable selection, as with the other methods, is a problematic area. It is proposed to rather use the LDF in cases where the data are distributed normally. A disadvantage of this method lies in its excessive computations.

The *nearest neighbour classification rule* was the first nonparametric classification method, which was developed by Fix and Hodges (1951). This rule is rather straightforward. The distance between the new point $\mathbf{x}_j$ and all other points are acquired by the distance function,

$$d_{ij} = (\mathbf{x}_j - \mathbf{x}_i)' \, \mathbf{S}_{pl}^{-1} \, (\mathbf{x}_j - \mathbf{x}_i) \qquad j \neq i \tag{2.20}$$

For equal group prior probabilities, $\mathbf{x}_j$ is labelled as belonging to group g_i if the majority of the h nearest points to $\mathbf{x}_j$ come from group g_i. Let h_1 indicate the number of points belonging to group g_1 and let h_2 be the number of points from group g_2, with $h_1 + h_2 = h$, then the classification rule becomes: assign $\mathbf{x}_j$ to group g_1 if

$$\frac{h_1}{n_1} > \frac{h_2}{n_2} \tag{2.21}$$

where n_1 and n_2 are the sample sizes for groups one and two, respectively. If equation (2.21) does not hold, then observation $\mathbf{x}_j$ is assigned to group g_2.

Taking group prior probabilities into consideration may lead to the further refinement of equation (2.21). Thus equation (2.21) changes to

$$\frac{h_1/n_1}{h_2/n_2} > \frac{p_2}{p_1} \tag{2.22}$$

If equation (2.22) holds, then $\mathbf{x}_j$ is allocated to g_1, otherwise to g_2. Adapting these rules to the several-group case is quite simple. That is, equation (2.21) changes as follows: assign $\mathbf{x}_j$ to that group yielding the largest proportion $\frac{h_i}{n_i}$, among the h nearest points to observation $\mathbf{x}_j$, with h_i the number of items belonging to group g_i. The value of h has to be chosen by the user and a good method would be by trying several values for h, then choosing that h yielding the best error rate, or $h \approx \sqrt{n_i}$ is another good value as proposed by Loftsgaarden and Quesenberry (1965).

Unknown or non-normal density functions can be estimated directly from the data by an approach known as the *kernel* estimator. An individual, $\mathbf{x}_j$ will be allocated to the group for which $p_i \hat{f}$ ($\mathbf{x}_j$ | g_i) is a maximum, where $i=1,2,\ldots,k$. In the case of the variable being continuous univariate, the procedure is as follows: Firstly, denote the density of a variable x by $f(x)$. By using the sample $x_1, x_2, \ldots, x_n$, the density of x is to be estimated. The proportion of observations falling into the interval ($x_0 - b$, $x_0 + b$) is an uncomplicated estimate of the density $f(x_0)$, for a random observation x_0. Denote the number of points falling into this range by $N(x_0)$, then the estimate of $P(x_0 - b < x_0 < x0 + b)$ would be $N(x_0)/n$. This is approximately equal to $2bf(x_0)$. Consequently the estimate of the density $f(x_0)$ is:

$$\hat{f}(x_0) = \frac{N(x_0)}{2bn} \tag{2.23}$$

It is possible to define $\hat{f}(x_0)$ in terms of all x_j by denoting

$$K(u) = \begin{cases} \frac{1}{2} & \text{for } |u| \leq 1 \\ 0 & \text{for } |u| > 1 \end{cases} \tag{2.24}$$

The function $K(u)$ is called the *kernel.*

This causes $N(x_0)$ to equal $2\sum_{j=1}^{n} K[(x_0 - x_j)/b]$, which changes equation (2.23) to

$$\hat{f}(x_0) = \frac{1}{bn}\sum_{j=1}^{n} K\left(\frac{x_0 - x_j}{b}\right) \tag{2.25}$$

The kernel defined in equation (2.24) is a rectangle and thus the plot of $\hat{f}(x_0)$ will be a step function, because when x_0 is too far away from x_j the kernel will be 0 and thus, there will be a drop in the graph. A smooth kernel must be used to obtain a smooth $\hat{f}(x_0)$. Various density functions can be used as the kernel. The family of the density function of the kernel does not lead to any assumptions regarding the density $f(x)$. Figure 2.5 illustrates the kernel estimate with individual kernels. Silverman (1986) provides a thorough overview on nonparametric density estimation.

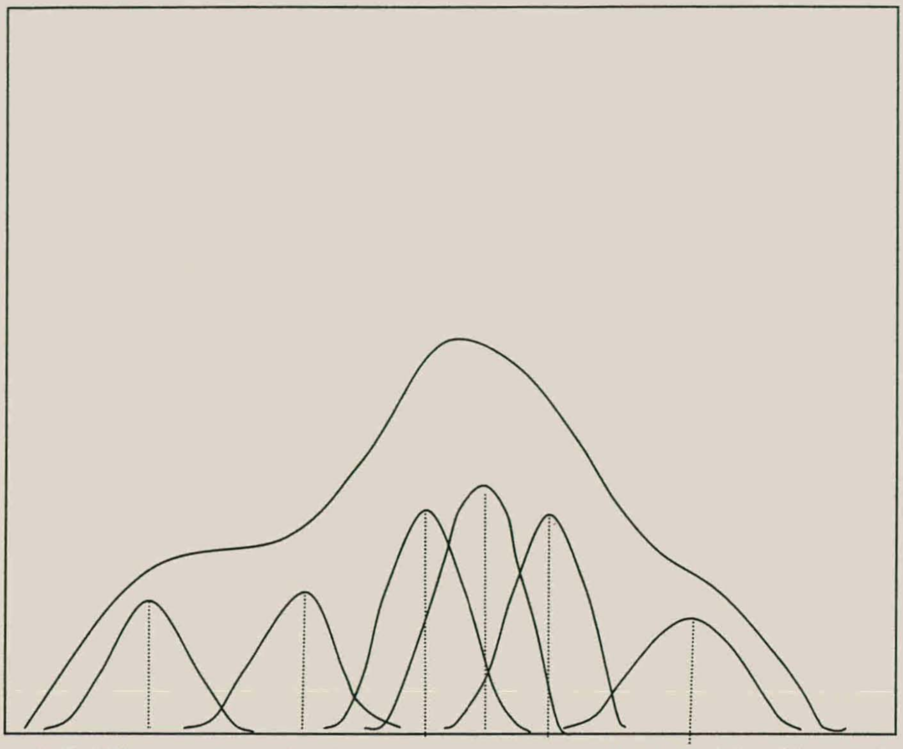

Figure 2.5: Illustration of a kernel estimator with individual kernels.

Classification trees is yet another classification method which has recently been developed. This method resembles divisive hierarchical clustering in the sense that initially there is one group, consisting of all the objects. The classification rules are determined by a procedure known as *recursive partitioning*. The group is split into two subgroups using two parts of a variable, which yield the whole variable space when combined. Each of the two subgroups are then further split according to another variable. This continues until the data are too sparse or the nodes (leaves) are pure, producing the classification rules. This procedure is computer intensive, resulting in a complicated tree. It is not on the same level as the LDF, as the descriptive feature is lost whereas LDF concentrates on the spatial partitioning of the groups. Figure 2.6 illustrates a classification tree. Each observation is evaluated by means of the rules and eventually the item is allocated to one of two groups, where the main difference among the groups are the absence or presence of a certain feature.

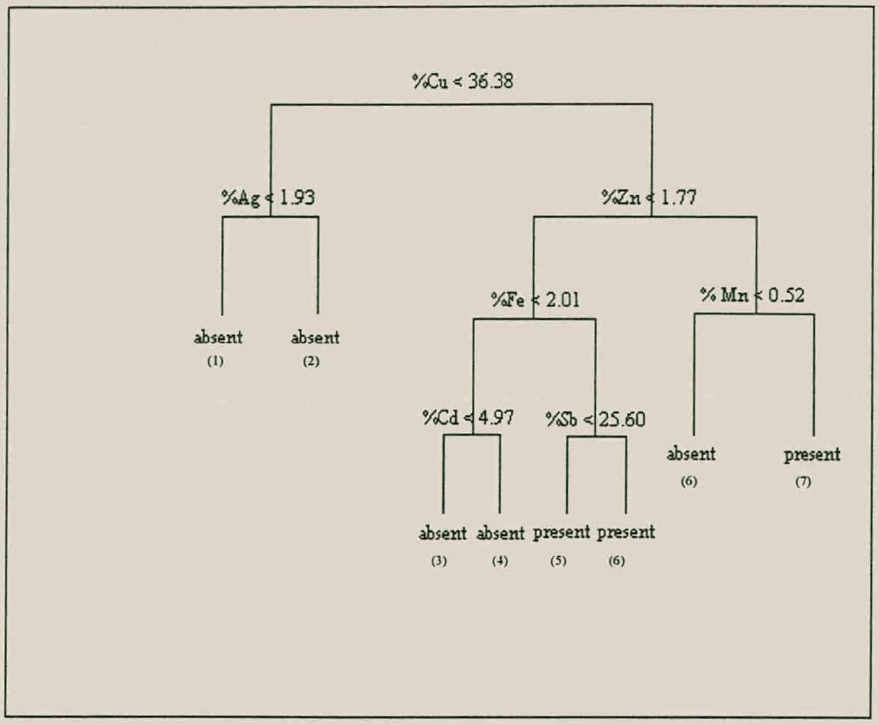

Figure 2.6: Illustration of a classification tree. The first branch can be interpreted as IF % (UL 36.38 AND % Ag < 1.93 THEN the outcome ABSENT.

Most of the clustering methods are quite appropriate for qualitative data, which yield more compact clusters, as is found in fields such as sociology and psychometrics, etc. Engineering data are of a more quantitative nature with interrelated variables that result in long stringlike clusters (Ginsberg and Whiten, 1991). Consequently the existing clustering algorithms do not yield the desired results.

Another characteristic of process data is its usually high multidimensionality with complex internal structures and thus it is difficult to handle such an enormous 'bulk' of data to obtain certain results. A solution would be to find a simpler representation of the data, i.e., mapping the data onto a lower dimension, retaining as much of the information as possible in the original data set. As expected, mapping the data to a linear function will not generate the required results for it cannot capture the essential information in the data.

A clustering method is proposed, based on a non-linear approach that closely follows the internal structure of the data. This non-linear approach is known as *principal curves,* which was developed by Hastie and Stuetzle (1989). Informally, principal curves are defined as those smooth one-dimensional curves that pass through the *middle* of a p-dimensional data set. Clustering will be performed on this one-dimensional representation of the multidimensional data, by means of a method similar in some sense to density seeking methods. Principal curves will also be incorporated in classification analysis, resulting in an alternative method of classification. The methodology of clustering and classification with principal curves is set out in chapter 3, with the experimental work regarding the topics following in chapters 4 and 5, respectively.

CHAPTER 3

METHODOLOGY OF CLUSTER ANALYSIS AND CLASSIFICATION WITH PRINCIPAL CURVES

Interpretations and conclusions regarding information in a raw data set cannot be accomplished at first sight. Therefore, statistical methods saw the light centuries ago in order to describe important features in the data. The actual problem of parameter estimation can be traced back to the Babylonian astronomers in the last three centuries B.C. (Pearson and Kendall, 1970).

Gradually the nature of data sets became more and more complicated and the need arose for evermore complex statistics to be developed. The higher the dimensionality of the data, the more troublesome the analyses become. As a result it has always been an aspiration to diminish dimensionalities of data sets, retaining the information in the data, to simplify analyses. The mean vector of a swarm of points in p-dimensions only establishes the centre of the data cloud and is seldom an adequate representation of the data.

Principal component analysis (PCA) has proven to be an excellent dimension reduction technique, in that the essential dimensionality of the data set k, is considerably less than the superficial dimension p, which is the number of variables. PCA was first introduced 100 years ago by Sylvester (1889) and is still used extensively today. The principal components are concerned with the core structure of a single sample of items measured on p variables. The components explain the variance-covariance structure of the data set by means of a few uncorrelated linear combinations of the original variables. The main goal of PCA is dimension reduction where the components can be used as input into other statistical methods, such as cluster analysis. Another use is to construct a scatterplot of the first two components to check for multivariate normality or outliers. More formally, the components linearly map the multifaceted data orthogonally onto a lower dimensional space, retaining almost all the information in the original data. Figure 3.1 illustrates the idea of principal component analysis of a two-dimensional data set.

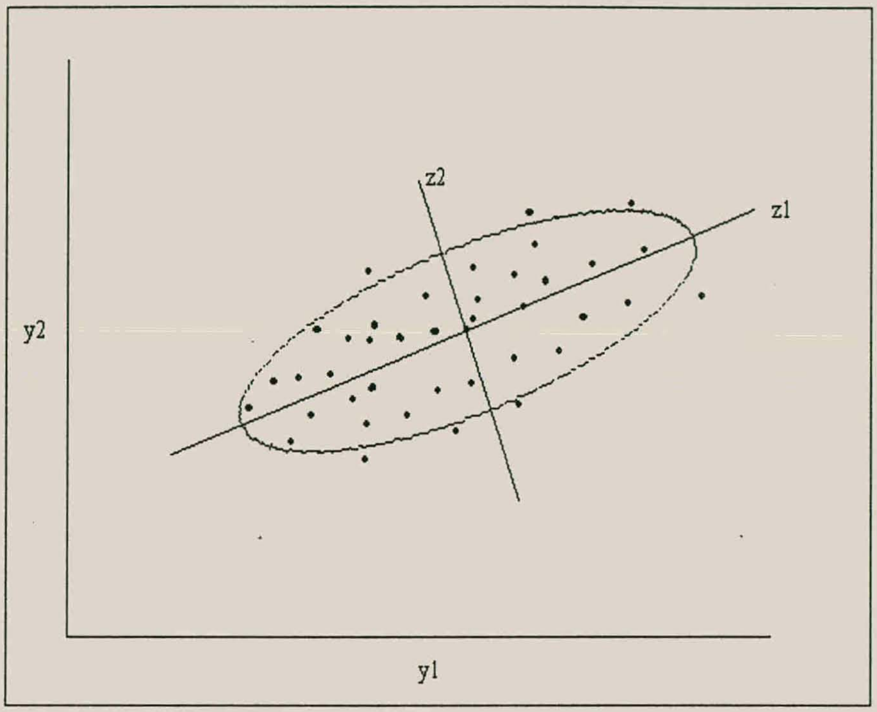

Figure 3.1: The first two principal components of the data set.

Since the variables are correlated, the points are not parallel to the axes denoted by the variables y_1 and y_2. The principal components yield the natural axes of the points, indicated by z_1 and z_2. As can be seen from the graph the points are maximally spread out along axis z_1. Thus, the first principal component z_1 accounts for the most variability in the data set of all the components. Many statistical textbooks cover the subject of principal component analysis, for example Johnson and Wichern (1988).

Engineering data are said to be data rich, but information poor. Therefore, as mentioned at the end of the previous chapter, an attempt to describe these types of data by means of a linear representation is infidelity to the data for it will result in the loss of valuable information. An illustration of such loss of information will be portrayed graphically in figure 3.4, which can be seen in section 3.2. A non-linear approach would seem more logical to summarise the data, of which many such approaches have been proposed. A method that developed the concept of a curve moving through the middle of a *p*-dimensional data set came to the light a decade ago and it is known as *principal curves*.

3.1 Principal Curves

Principal curves and principal components have two features in common. Firstly, in the event of the principal curve being a straight line, it is nothing else than a linear principal component. Secondly, both techniques concentrate on the orthogonal distances between the points and their projection onto the curve, in the sense that they both attempt to minimise the sum of the squares of these distances. That is, let Γ be a principal curve and Γ_t a smooth family of curves (denoted by the subscript) with $\Gamma_0 = \Gamma$, then

$$ {}^{d}\!/_{dt}\, d^2(\mathbf{X}, \Gamma_t)\big|_{t=0} = 0 \qquad (3.1)$$

where x is the p.dimensional data set ($X \in R^P$), d^2 is a squared distance measure (usually Euclidean) quantifying the fit of the curve to the data, Γ_t is the curve at iteration t.

Owing to the two above-mentioned communal properties, principal curves are said to be a generalisation of principal components. Figure 3.2 demonstrates the orthogonal projections of the points onto the principal curve. Hastie and Stuetzle (1989) formally defined principal curves to be those smooth one-dimensional curves that are *self-consistent* for a data set or distribution. This implies that if all the items projecting onto a point on the curve are gathered and averaged, that average should then coincide with the point on the curve. This condition should hold for all such points on the curve to meet the self-consistency property. Recall the informal description of a curve as a smooth one-dimensional curve passing through the *middle* of a data set. Accordingly, the shape of the curve is determined by the structure of the data and if the data set contains outliers, the shape of the curve will definitely be affected, again emphasising the influence such observations have on results.

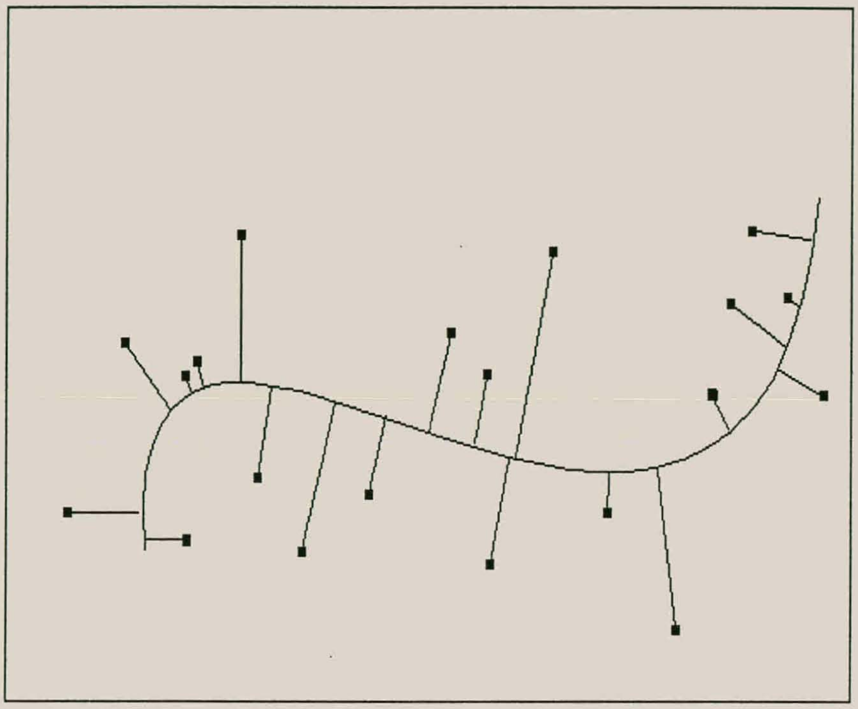

Figure 3.2: Orthogonal projections of the points onto the curve.

3.1.1 Definition of a Principal Curve

Let $\mathbf{X}_i$ indicate a random vector in $\mathbf{R}^p$ with a density of h. A one-dimensional curve in p-dimensional space is a vector $\mathbf{f}(\lambda)$ of p functions (co-ordinate functions) of a single variable λ, thus by definition if the co-ordinate functions are smooth, $\mathbf{f}$ is a smooth curve. The variable λ, parameterises the curve in terms of arc length and provides an ordering along it. In other words, λ_j is the arc length down the curve from $\mathbf{f}_1$ to $\mathbf{f}_j$, where $\lambda_1 = 0$. As a result these distances, λ_j, where $j=1, 2, \ldots, n$, are analogous to the principal component scores.

The $\mathbf{f}_j$ are the projections of the n points onto the line. When $\|\mathbf{f}'\| \equiv 1$ it is called a unit-speed parameterised curve. Let $\mathbf{f}$ be such a smooth unit-speed curve in $\mathbf{R}^p$ parameterised over $\Lambda \subseteq \mathbf{R}^1$, a closed interval, which does not intersect itself, i.e., $\lambda_1 \neq \lambda_2 \Rightarrow \mathbf{f}(\lambda_1) \neq \mathbf{f}(\lambda_2)$. Also let $\mathbf{f}$ be of finite length within any finite area inside $\mathbf{R}^p$, such as the spatial distribution of the data $\mathbf{X}$. The curve $\mathbf{f}$ is self-consistent or a principal curve of h if

$$E[\mathbf{X}_i|\ \lambda_{\mathbf{f}}(\mathbf{X}_i) = \lambda] = \mathbf{f}(\lambda) \qquad (3.2)$$

for all λ, where $\lambda_{\mathbf{f}}$ is defined as a projection index of $\mathbf{R}^p \rightarrow \mathbf{R}^1$, given by

$$\lambda_{\mathbf{f}}(\mathbf{x}) = \sup_\lambda\{\lambda\|\ \mathbf{x} - \mathbf{f}(\lambda)\|\ = \ \inf_\mu\|\ \mathbf{x} - \mathbf{f}(\mu)\|\}. \qquad (3.3)$$

The projection index $\lambda_{\mathbf{f}}(\mathbf{x})$ of $\mathbf{x}$, portrayed in equation (3.3), is the value of λ for which $\mathbf{f}(\lambda)$ is closest to $\mathbf{x}$. If several such values exist, the largest one is used.

3.1.2 The Principal Curve Algorithm

This particular algorithm is the first principal curve algorithm, which was developed by Hastie and Stuetzle (1989). The starting step uses any smooth curve, usually the first principal component, and tests this curve for self-consistency by means of projecting the data onto the curve and calculating their expected value conditional on where they project. In the situation where the conditional expectation coincides with the curve, the self-consistency feature is met and a principal curve is obtained, otherwise, the conditional expected curve is subjected to an iteration process until it (hopefully) converges.

Mathematically the process to obtain the principal curve is:

- Initialisation: Set $\mathbf{f}^{(0)}(\lambda) = \bar{\mathbf{x}} + \mathbf{a}\lambda$, with $\mathbf{a}$, the first principal component of $\mathbf{X}$.

  Set $\lambda^{(0)}(\mathbf{x}) = \lambda_{\mathbf{f}^{(0)}}(\mathbf{x})$.

- Iteration: 1. Set $\mathbf{f}^{(j)}(\cdot) = E[\mathbf{X}|\, \lambda_{\mathbf{f}^{(j-1)}}(\mathbf{X}) = \cdot]$

  2. Define $\lambda^{(j)}(\mathbf{x}) = \lambda_{\mathbf{f}^{(j)}}(\mathbf{x}),\ \ \forall\ \mathbf{x} \in h$

  3. Transform $\lambda^{(j)}$ so that $\mathbf{f}^{(j)}$ is unit speed.

  4. Compute $|d^2(\mathbf{X},\mathbf{f}^{(j)}) = d^2(\mathrm{X},\mathbf{f}^{(j+1)})| \,/\, d^2(\mathbf{X},\mathbf{f}^{(j)})$, until it falls beneath some threshold value (usually 0.001). This is the sum of squared distances of the points to their respective closest points on the current curve.

The curve is characterised by n-tuples of (λ_j,$\mathbf{f}_j$), which is assumed to be sorted in increasing order of λ to form a polygon of which its geometric form does not depend on the actual values of the λ's, but only on their order. In practice, the conditional expectation at λ_j is estimated by averaging all the items $\mathbf{x}_k$ in the sample for which λ_k falls within a neighbourhood of λ_j. This idea is graphically displayed in figure 3.3.

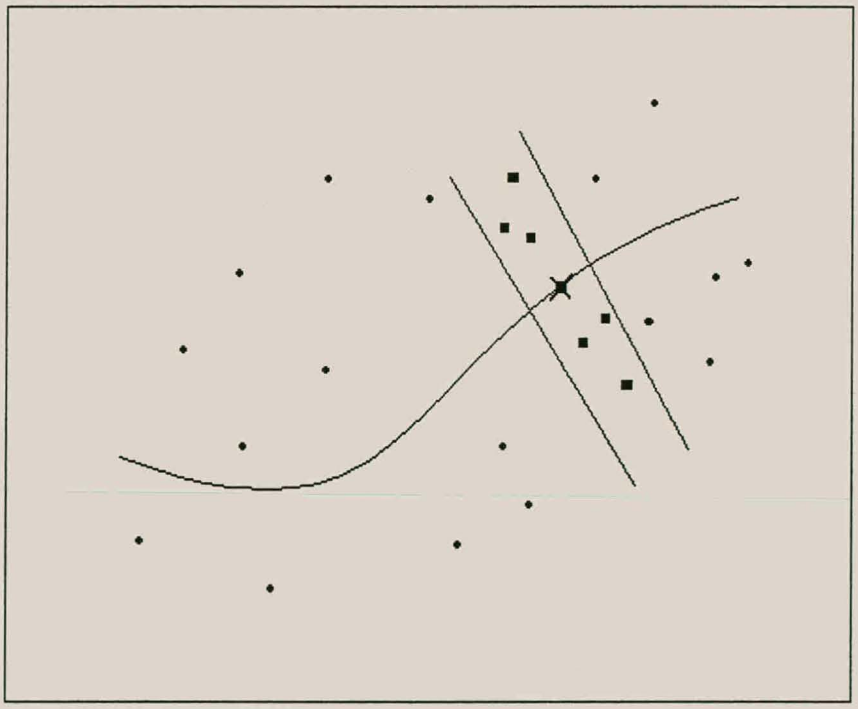

Figure 3.3: Each point on the principal curve is the average of the points that project onto it.

The fraction of points that fall inside this neighbourhood is set by means of a parameter called span. If it is too small a value, the curve will follow the data too closely, including the possible noise within the data. On the contrary, a large value seems to interpolate the points. Another method used to search for the value of span is by means of an automatic technique known as cross-validation (see appendix 1). The span that yields the smallest corresponding cross-validation-value is chosen as the parameter value. This method is used throughout the experimental work.

3.2 Cluster Analysis using Principal Curves

Employing the principal curve algorithm on the data set **X**, yields the principal curve. This one-dimensional representation of the set **X**: $n \times p$ is adopted as the 'new' data set on which all further analyses will be executed, since the principal curve embodies the core structure of **X**. As mentioned formerly, the newly acquired $1\times n$ data vector can be referred to as the *principal curve scores* as these scores are the arc lengths along the curve. Therefore, all these values are positive with the smallest being equal to zero.

The clustering technique is based on the statement that the curve is said to move through the middle of the data set, as well as on the initiative of density seeking clustering methods where clusters are defined as areas in space that enclose dense collections of points. If in fact the curve captures the core structure of the data set, and if there are natural clusters present, the representative data vector has to contain information regarding these clusters.

Firstly the principal curve scores λ_j, are sorted in increasing order. It is expected that points from a specific cluster can be associated with ordered scores on the curve that are relatively compact and their respective values will all fall within an evident range. Scores that do not fall within this scope are labelled as belonging to a different cluster. As a result, such objects are thought of as dissimilar to those objects whose respective scores fall within the above-mentioned range. When the data has clusters that are well separated there will be areas on the curve containing scores that are close together, corresponding to clusters, and there will be areas with no points. Such an empty area between two clusters is the decisive factor in this technique, namely where to distinguish between objects as belonging to different clusters.

As mentioned earlier, figure 3.4 as can be seen below, demonstrates how mapping engineering data to a one-dimensional representation can result in the loss of information. However, this data set is not process data. It contains four well-separated clusters that can easily be identified by the eye, which is not typical of process data. This data is known as the Ruspini data (1970), which he originally used to illustrate fuzzy cluster analysis.

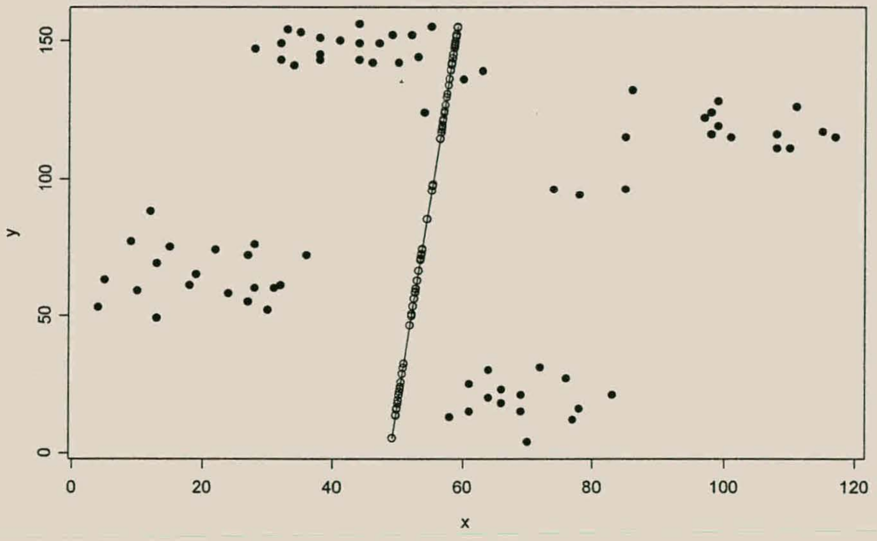

Figure 3.4: Data points with their orthogonal projections onto the first principal component.

As observed in figure 3.4, when applying the proposed clustering technique to the first principal component of the bivariate data, only three clusters seem to be distinguished. The three clusters are identified as those dense regions of the projected points, separated by less dense regions. Reducing the dimension with one by means of a linear summary, resulted in the loss of essential information in the data. This is the result of a data set, which appears easy to cluster. Process engineering data are of a much more complicated nature than this data set. Therefore, such data equally needs a superior mapping technique than the one just used, in order for minimal loss of information.

The principal curve seems to be a more logical representation of the data in one dimension than the first principal component. The principal curve is fitted to the data set used in figure 3.4, which can be seen in figure 3.5 together with the projected points portrayed as black dots on the curve.

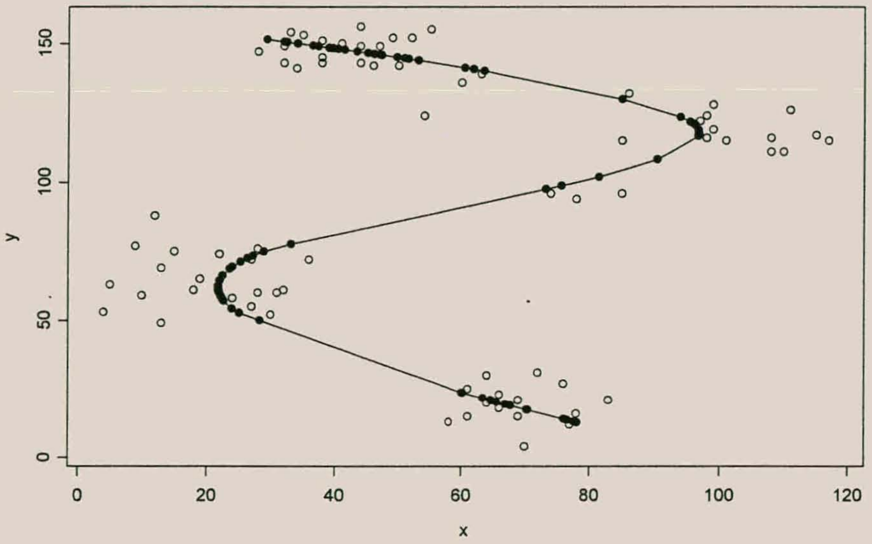

Figure 3.5: Data points with their orthogonal projections onto the principal curve.

By mapping the data onto the principal curve and considering the projected points (black dots) on the curve, four clusters can be pinpointed in figure 3.5. The empty areas clearly distinguish the four groups. Already it is possible to conclude that the principal curve captured the essential information in the data and that the curve is a good one-dimensional representation of the original data.

In order to formalise this idea, the differences between the consecutive, increasingly ordered principal curve scores are determined. The differences amongst successive scores in the same cluster ought to be quite small, as opposed to the large difference between two successive scores pertaining to separate clusters. These large differences relate to the large empty areas between clusters, as can be viewed in figure 3.6 where there are 3 large empty areas. In order to decide visually where to partition the scores into different clusters, the differences between the consecutive, ordered scores are plotted in two dimensions in the following graph.

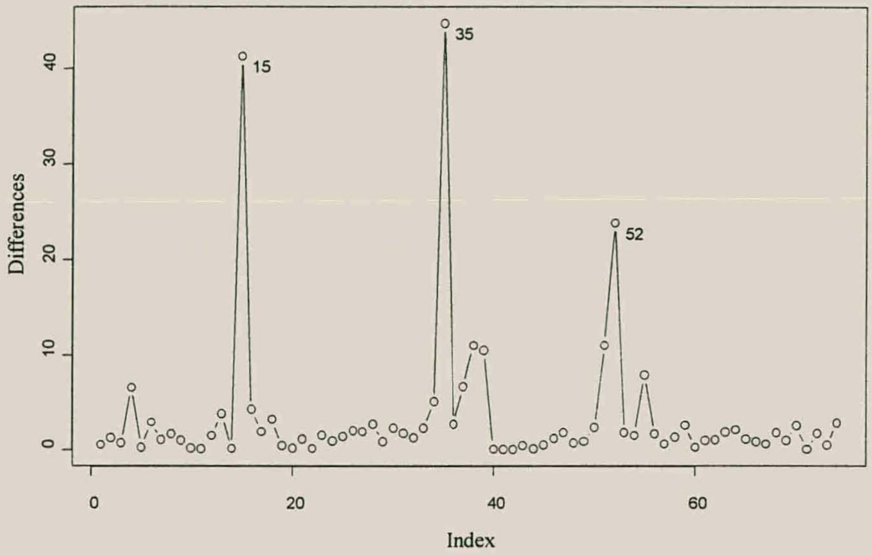

Figure 3.6: Differences between consecutive, ordered principal curve scores.

Three significant peaks are visible in figure 3.6, which indicates a large difference between two successive ordered scores, in other words it denotes an empty area. According to this information, the ordered scores are split at difference numbers 15, 35 and 52. This implies that the observations relating to the ordered scores 1 through 15, 16 to 35, 36 to 52 and 53 to the last item, respectively belong to clusters 1 to 4. The final cluster configurations based on these results can be seen in figure 3.7.

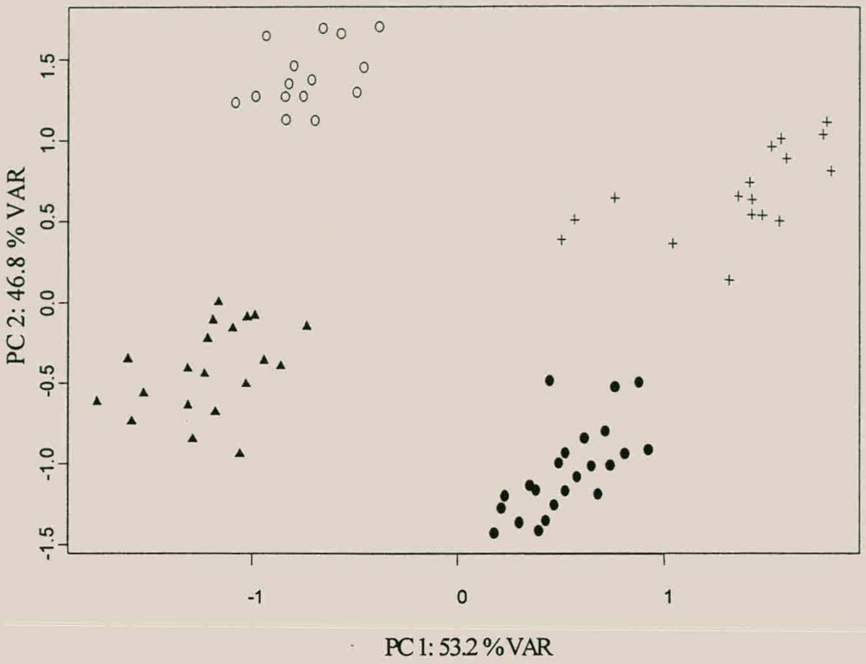

Figure 3.7: The four different clusters obtained via the splitting of the ordered principal curve scores.

Hierarchical and partitioning clustering procedures like single link and *k*-medoids, produced similar results than that obtained in figure 3.7. This implies that the proposed principal curve clustering method can in fact be regarded as a clustering technique.

In cases where the clusters are not that well defined, the principal curve technique can also be applied. The empty areas may only be smaller or even non-existing, however, dense areas will still be present. It will be more difficult to decide exactly where one cluster ends, and another begins. In this situation the researcher should execute the procedure several times, splitting the points into different cluster formations under consideration. Each time the obtained clusters should preferably be plotted in the space of the first two or three principal component axes, as this space optimally separates the data. The final cluster configurations decided upon, should be that arrangement that yields the most interpretable and reasonable results.

3.3 Classification Analysis using Principal Curves

Classification of objects can be considered as flowing directly out of the previous topic, where the different groups in the data set were identified. This is the starting point for classification analysis, where the distinct groups are known beforehand. There is now being searched for a method to maximally distinguish the k groups and learn the intrinsic structure of each of them in order to allocate new objects, of unknown origin, to one of the existing k groups.

It is known by now that a principal curve summarises the core structure of a data set in a non-linear fashion. Hence, it would seem a good idea to use the principal curve in terms of describing each group's formation and also another area to demonstrate its versatility. After acquiring each group's principal curve, the methodology is to project every new observation onto the k different principal curves. This idea has previously been proposed by Chang and Ghosh (1998), but has not been explored in detail. By means of the Euclidean distance (2.1), it is possible to calculate how close the new point is from all its projections onto the different curves. Consequently, the classification rule is to assign the new item to that group yielding the shortest distance between the new point and its projected value on the principal curve. Figure 3.8 illustrates this concept.

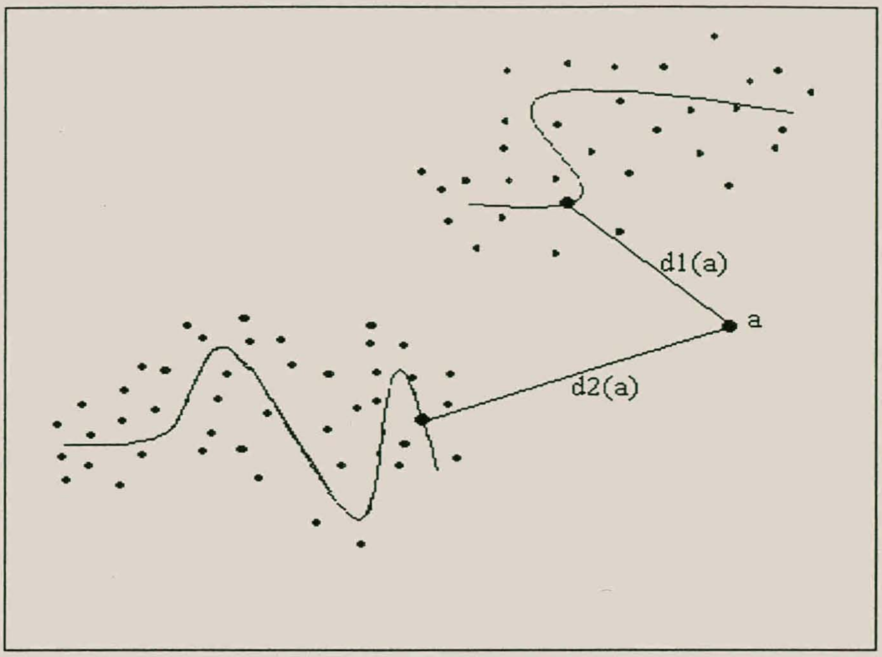

Figure 3.8: Two distinct groups with their respective fitted principal curves. The new observation, a, is to be classified to the group yielding the shortest distance between the point and its projection.

The proposed method of classification with principal curves is portrayed in figure 3.8. The new point, **a**, is projected onto the principal curve of each cluster, whereafter the Euclidean distances of the new point to all of its projections are calculated, yielding d1(**a**) and d2(**a**). The new observation is allocated to the cluster at the top, since the distance to its projection onto the principal curve of this group is the smallest.

There is one problem though, namely that it is not possible to merely project the new item onto the principal curve in order to obtain its projected value. Principal component analysis (PCA) produces a principal loading for each variable, which determines the direction of the respective principal component, enabling the projection of new points. Unfortunately, unlike PCA, the algorithm of principal curves does not supply a model for the curve, only a related principal curve score and corrected data point are determined for each observation.

These corrected data points are a matrix corresponding to the original data set **X**, giving their projections onto the curve. For each of the *k* groups, a back propagation neural network with only one hidden layer is used to acquire the relation between the original data set and the matrix containing the corrected data points. This relation is in fact nothing else than the sought after principal curve model. Using the *k* trained neural networks, the new observations are used as input into every network and in this manner projections for every new point onto every group's principal curve can be obtained.

3.3.1 Back Propagation Neural Networks

Back propagation neural networks are also known as feed-forward neural networks, which is only one of numerous kinds of neural networks available in the literature today. Neural networks are a collection of simple computational components (units) that are systematically interlinked. Basically a neural net consists of the input layer, output layer and possible hidden layers, which are situated between the input and output layers. The layers are comprised of elements or nodes. The input nodes receive only the input data and distribute them to the network. The hidden layer can have any number of nodes and there can be more than one hidden layer. The nodes in a specific layer are linked to other nodes in consecutive layers through weighted connections. The structure of a back propagation neural network with three input, two hidden and three output nodes are illustrated in figure 3.9.

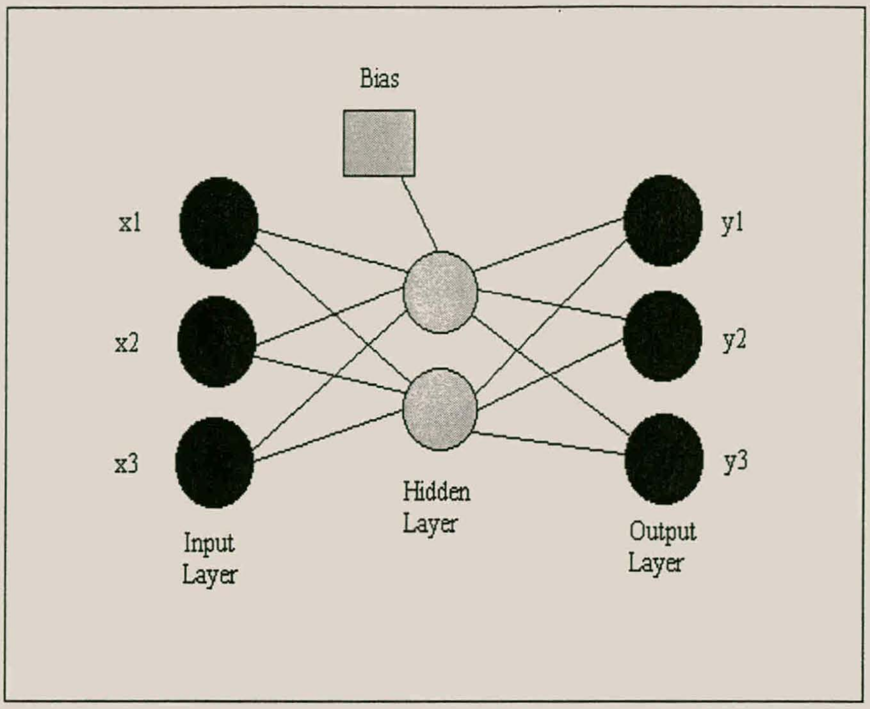

Figure 3.9: A back propagation neural net with an input-, hidden- and output layer containing three, two and three nodes, respectively.

Neural nets are applied if the functional relation between two sets of data is desired. For instance, in multiple linear regression where the expected response y is related to the values $x = (x_1, x_2, ..., x_p)$ through,

$$y = w_0 + \sum_{j=1}^{p} w_j x_j \tag{3.4}$$

the relation can be found by means of neural nets, as portrayed diagrammatically in figure 3.10.

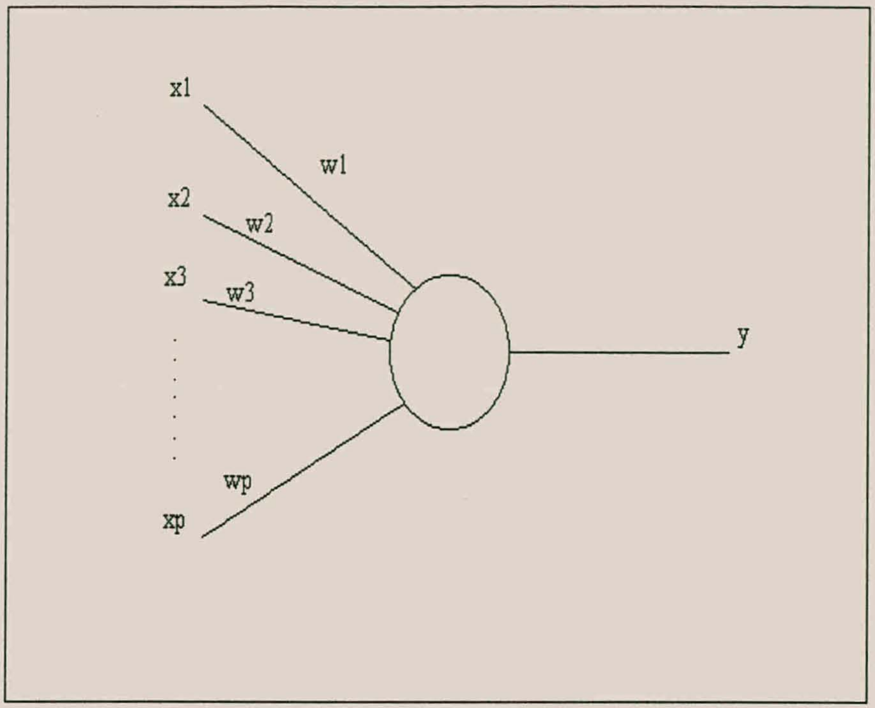

Figure 3.10: Illustration of a simple neural network.

The input layer contains as many nodes as there are variables (in this case *p*), whereas the output layer has the same number of nodes as the number of variables in the target data set (one). In this example there are no hidden layers. Equation (3.4) is actually computed in the single node of the output layer in figure 3.10.

In a general back propagation neural net the data set is fed into the net through the input layer. The sum of the weighted inputs and the bias formulate the input to the transfer/activation function F. An often used transfer function for back propagation neural nets are the log-sigmoid transfer function that generates output values between 0 and 1, as the network input goes from negative to positive infinity. Sigmoid transfer functions used for the last layer produce outputs within a small range, whereas outputs of linear transfer functions for the last layer can take on any value. The values obtained through the output layer are compared to the target values and so the errors between them are computed.

Usually back propagation neural nets have more than one hidden layer containing sigmoidal nodes, followed by an output layer with linear nodes. These multiple layers with nonlinear nodes allow the network to learn nonlinear, as well as linear relationships between input and output data sets. The first step is an initialisation step, where the weights and biases are returned for each layer. The second step involves training of the network where the weights and biases have to be determined. The back propagation learning rule is a recursive algorithm for it repeatedly attempts to match the input data with the output or target data. That is, the information is propagated back through the net to update the weights and biases, which represents the features/variables of the process. This procedure is repeated until the sum-squared error between the inputs and the outputs fall beneath some threshold value. The neural networks executed in the thesis have been performed in Matlab, of which the manual is a good starting point for new users in this field of artificial intelligence. Other references include Aldrich (1997), and Cheng and Titterington (1994) who give a review of neural networks from a statistical viewpoint.

As was mentioned earlier, classification trees are on a higher level than any of the other classification rules, discussed in chapter 2. This is due to the fact that no insight is gained into the contribution each variable has on the separation of the *k* different groups. All the variables are employed to obtain the final rules. The same applies to the proposed method of classification via principal curves. Unlike the problem of variable selection with the other techniques where it is imperative to use the correct number of preferably uncorrelated variables, all the variables can be employed with the neural networks to obtain the rules. As a consequence, this is an advantage of the proposed technique because the intercorrelations among the variables do not present a problem.

CHAPTER 4

CLUSTERING WITH PRINCIPAL CURVES

In this section the experimental work that has been done on clustering with principal curves, is presented. In the first three examples, the proposed clustering method is applied to simulated two-dimensional data sets containing clusters with some interesting formations. These data sets are not representative of any real data, but are merely to obtain an understanding of the performance of the principal curve clustering technique. After these three examples, the proposed method is applied to two process engineering data sets.

Example 4.1:

Two contiguous osculating clusters are present in this data set. Usually single link clustering methods fuse groups such as these into one cluster, because of their vulnerability to the chaining effect. Therefore, it would be interesting to see how the proposed clustering method handles this problem and whether it fails, like single link, or if it is able to circumvent this problem. The data set, together with the principal curve fitted to it, can be seen in figure 4.1.

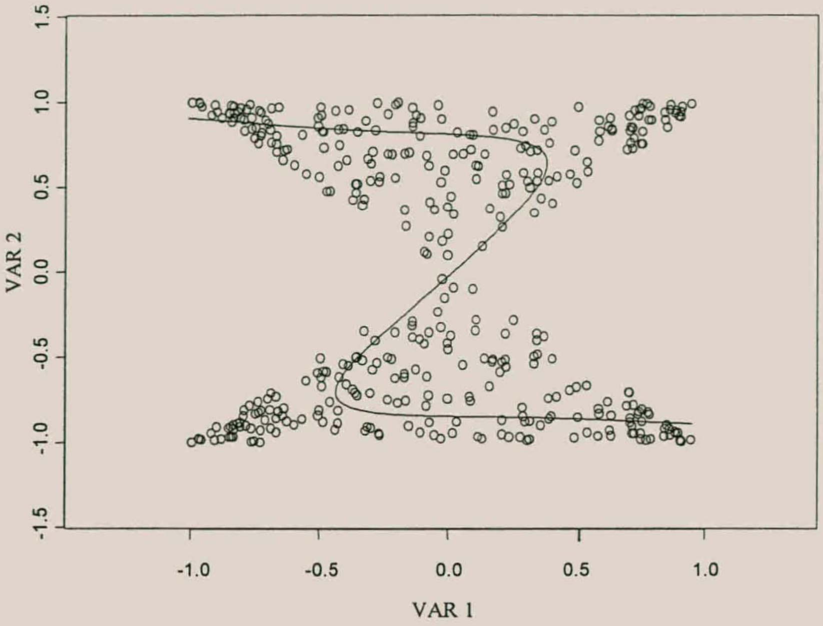

Figure 4.1: Principal curve fitted to the data from example 4.1.

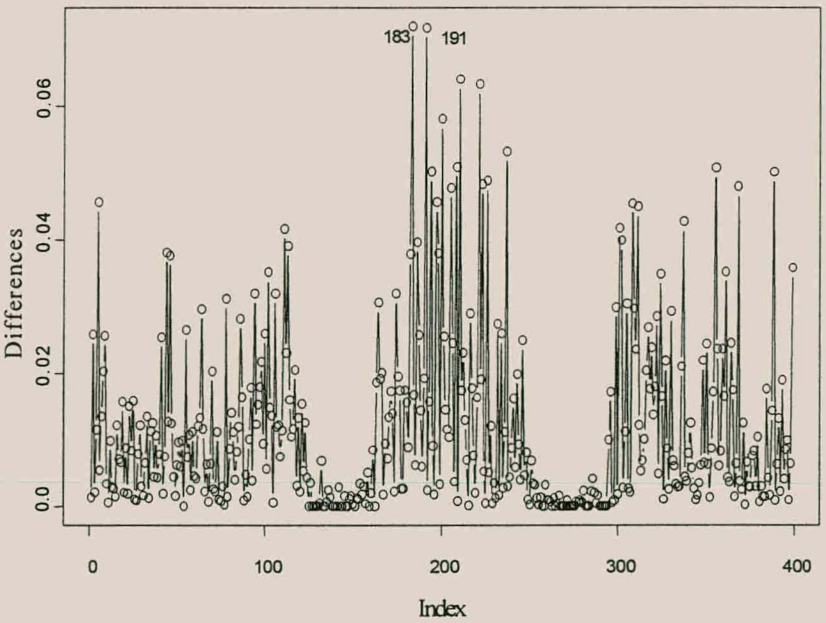

Figure 4.2: Differences between the consecutive, ordered principal curve scores (example 4.1).

As can be seen in the above figure, the plot of the differences between the successive scores will not always be as easily interpretable as was the case with the Ruspini data illustrated in chapter 3. In figure 4.2 we see that there are two areas where the differences are minimal and in the middle there are two points close together. These points indicate the largest differences and both of them were used to cluster this data set. Figure 4.3 shows the clustering when the split is based on observation number 191. The two areas of minimal difference relate to the two inflection points in the principal curve in figure 4.1, since most of the points are projected onto these areas. The three parts of obvious differences in figure 4.2 relate to the three fractions where the curve is quite straight, as relatively few points project onto these segments. These results can be ascribed to the relatively low densities of the points in each of the two triangles, and the spurious peaks in figure 4.2 tend to decline as the densities of the clusters increase.

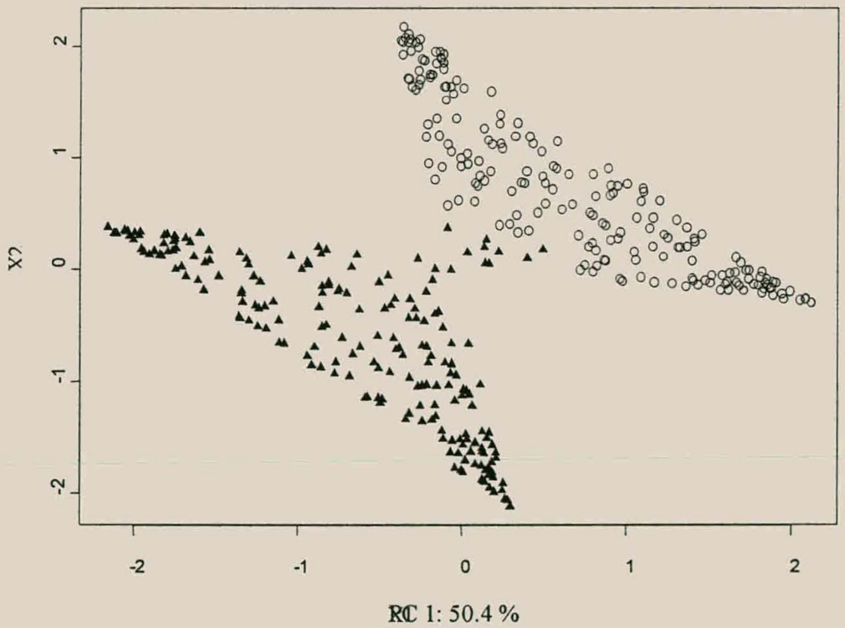

Figure 4.3: Two-dimentional clusters obtained through the principal curve-method, (example 4.1).

The few points incorrectly clustered in figure 4.3 where the two groups meet, can be attributed to the relatively low densities of the points in the triangles. Figure 4.4 displays the results obtained by single link hierarchical clustering. Complete link failed in its attempt to group this data set, probably because of the bias this method has towards spherical clusters.

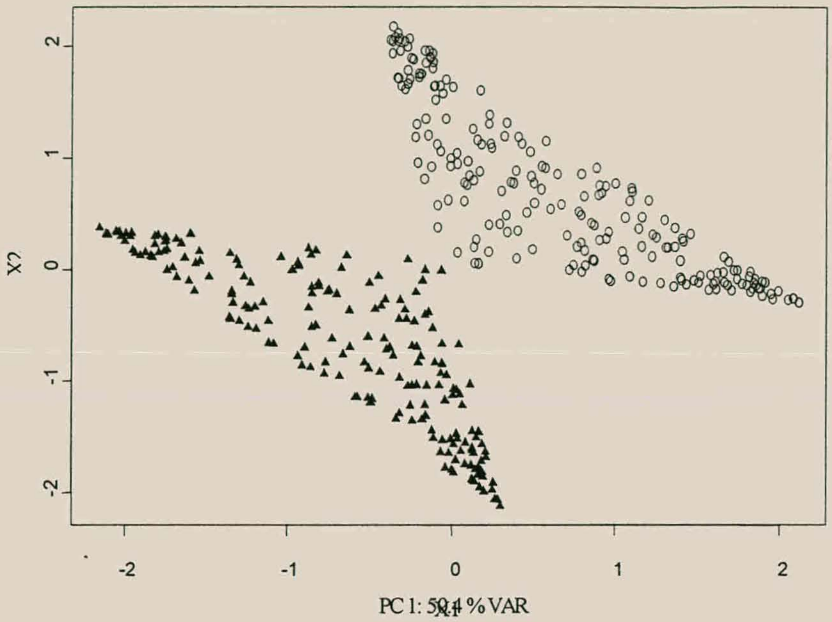

Figure 4.4: Single link clustering of osculating data (example 4.1).

Although the single link method clustered the data set very well, it failed completely when the densities of the points were increased. The reason for this is that the bridge between the clusters became denser, resulting in chaining of the clusters – a weakness of the single-link method.

Example 4.2:

The next data set has two elongated parallel clusters. It is expected that single link will cluster this data set perfectly, since there are no intermediate points between the two groups and also as single link is prone to produce long thin clusters. Figures 4.5 through 3.8, respectively yield the data set with its fitted principal curve, the graph containing the differences of the sequential, ordered curve scores, the clusters obtained via this method and finally the results from single link. Average link and complete link are not expected to cluster the data successfully, owing to their predisposition to search for spherical clusters.

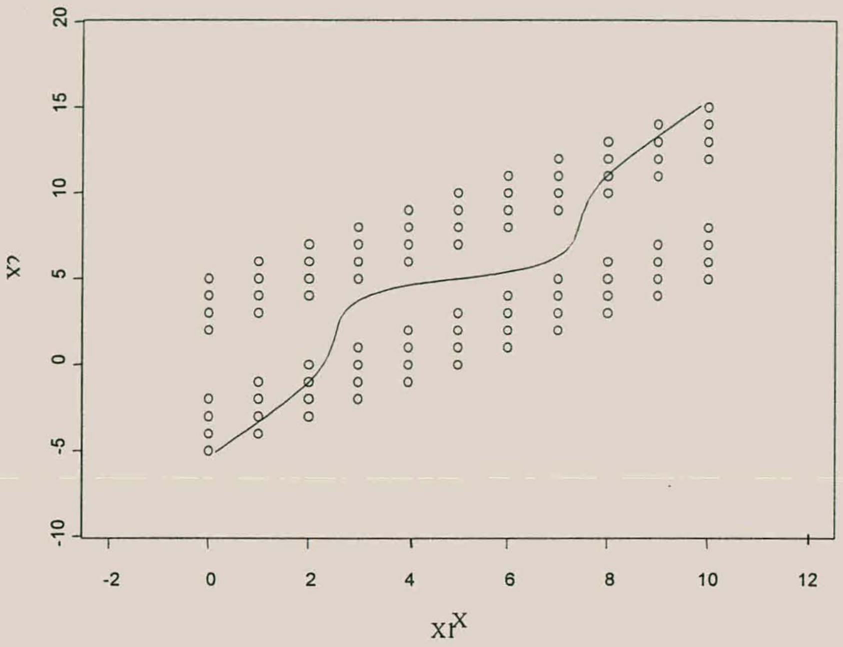

Figure 4.5: Data containing elongated clusters with the fitted principal curve (example 4.2).

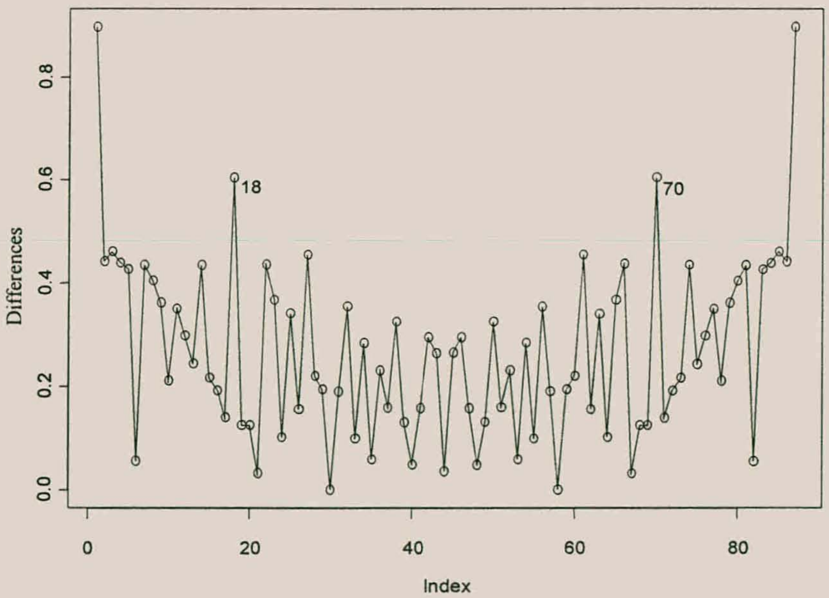

Figure 4.6: Differences of the sequential, ordered curve scores (example 4.2).

The largest differences, which yield the same value in this case, are the first and last differences. Usually the first difference is rather large, since the first ordered curve score is always zero. As a result, a fairly large difference between it and the second ordered score is obtained. Splitting the elongated data set at the observations yielding the largest differences between the ordered scores, produces three clusters as can be seen in figure 4.7. The cluster configurations acquired through this method, as well as the single link clustering results can respectively be viewed in the next two diagrams.

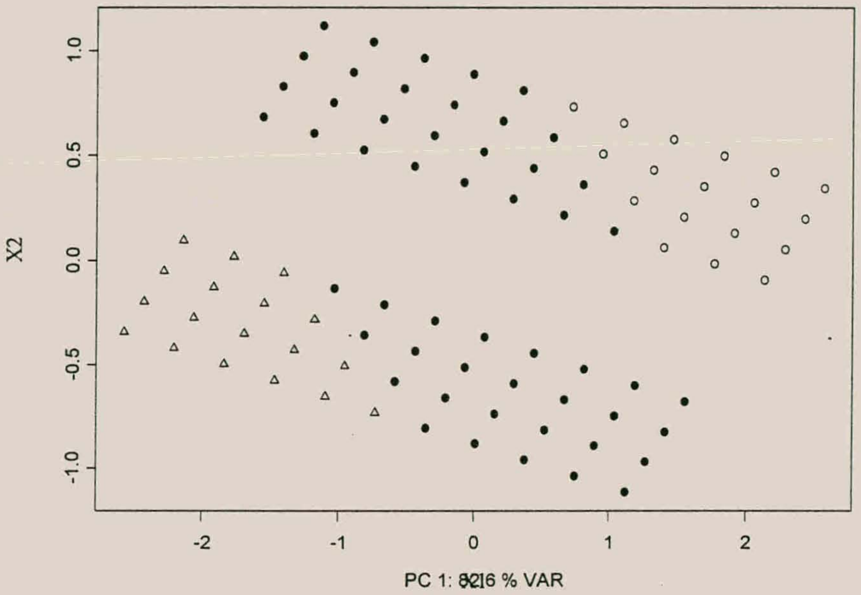

Figure 4.7: Principal curve-clusters (example 4.2).

It is evident from this graph that the principal curve method failed in its attempt to obtain the actual clusters present in the data set.

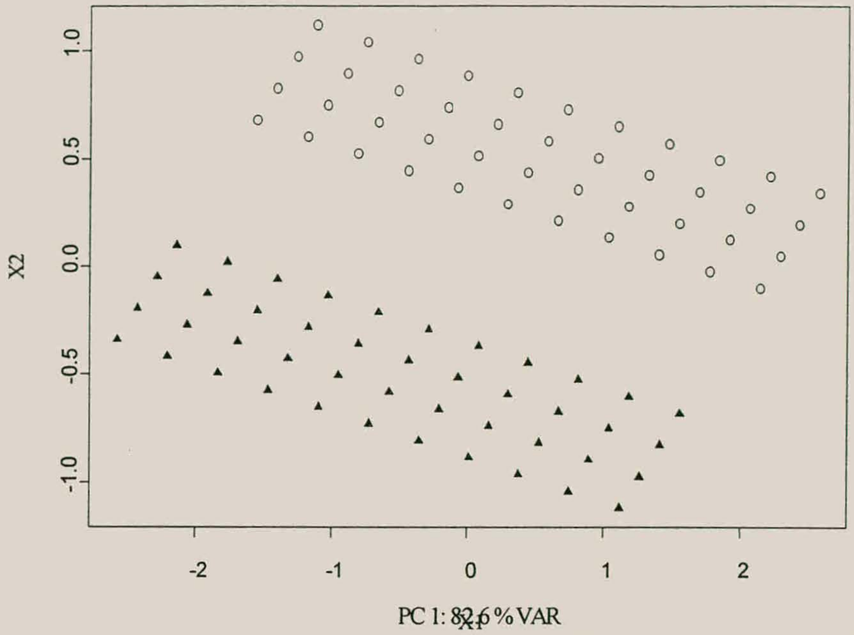

Figure 4.8: Clusters acquired via single link clustering (example 4.2).

Studying figure 4.5, it becomes evident that the scores on the principal curve cannot produce the desired results, since the two groups project symmetrically onto the curve and, therefore, the observations cannot be distinguished. Since the clusters are linearly separable, it is clear that had the points been projected onto a curve, orthogonal to the main axes of the clusters, the different groups would have been identified. However, this cannot be achieved through the principal curve algorithm. Single link, as was expected, identified the clusters successfully, whereas complete and average link both failed, giving almost similar results as with the principal curve method.

Example 4.3:

The last of these two-dimensional data sets is composed of two clusters, with the first being a spherical cluster enclosed by an annular second cluster. This data set will be difficult to separate by means of a projection technique, such as the principal curve method. Figure 4.9 illustrates the data set with the principal curve fitted to it.

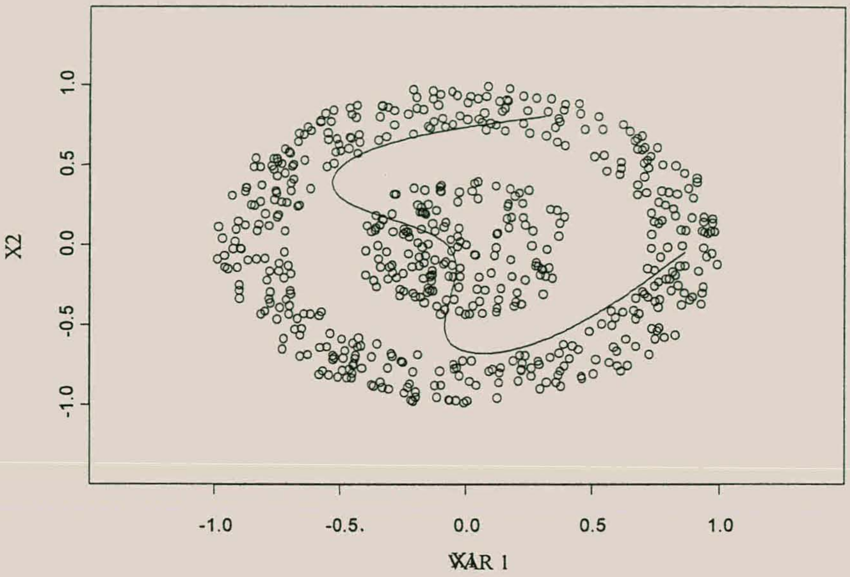

Figure 4.9: Data containing concentric circular clusters with their fitted principal curve (example 4.3).

The graph of the differences between the successive ordered scores is illustrated in figure 4.10. As can be seen, it is difficult to decide where to split the observations into different clusters. There are a large number of observations in the middle whose respective ordered curve scores do not differ at all (zero), thus, according to the methodology they fall within the same cluster. In fact these projections are all onto the same point, seeing that their differences are zero. At both ends of this area are ordered scores that successively differ to some extent as in the first example, indicating that for those areas the clusters are too sparse. The divisions were made as indicated in figure 4.10, yielding three clusters. The three obtained clusters are shown in figure 4.11.

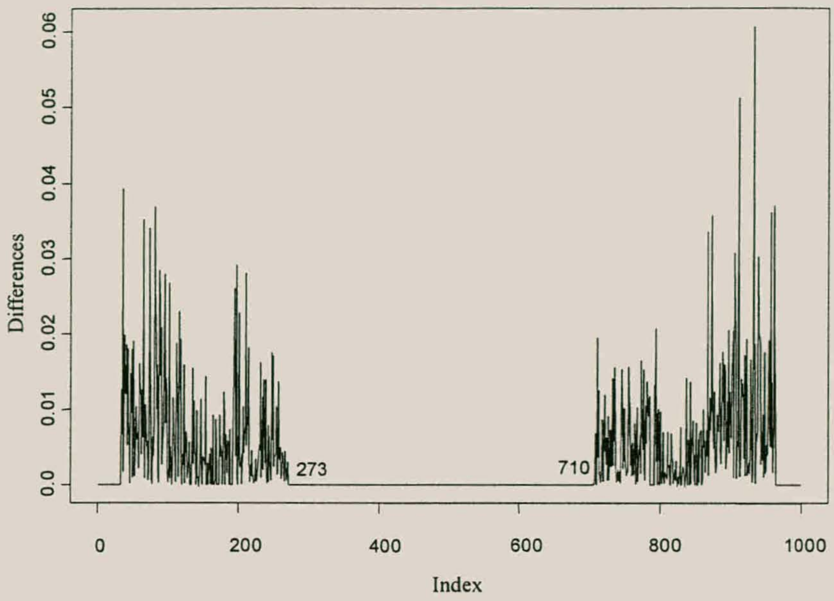

Figure 4.10: The differences between the consecutive, ordered curve scores (example 4.3).

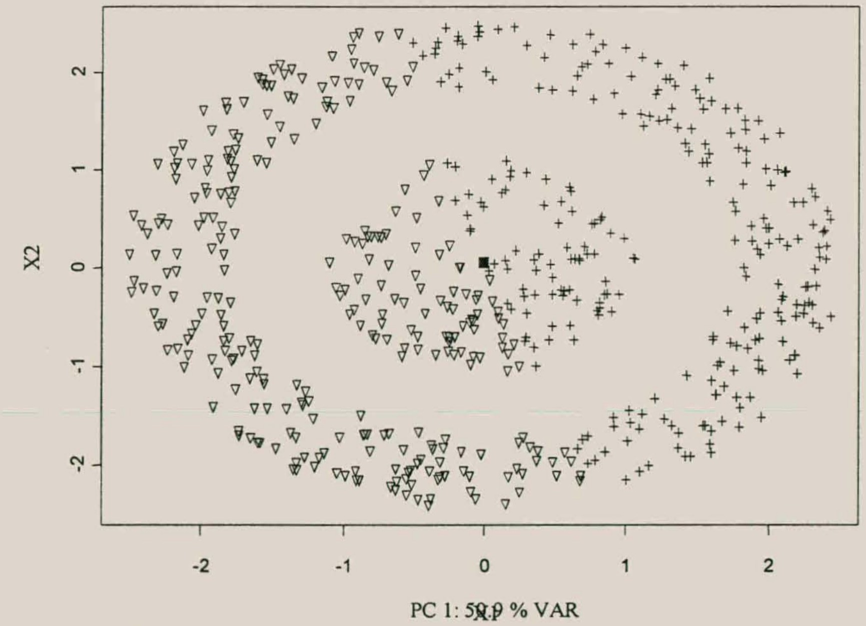

Figure 4.11: Obtained clusters via the principal curve-(example 4.3).

The block in the centre of figure 4.11 indicates the second cluster, which is the area with zero differences between the ordered scores. As mentioned previously, if all those observations project onto the same point on the curve, only then can the difference between all those ordered scores be zero, thus, yielding the block as the second cluster. Figure 4.12 portrays the results obtained by means of single link clustering.

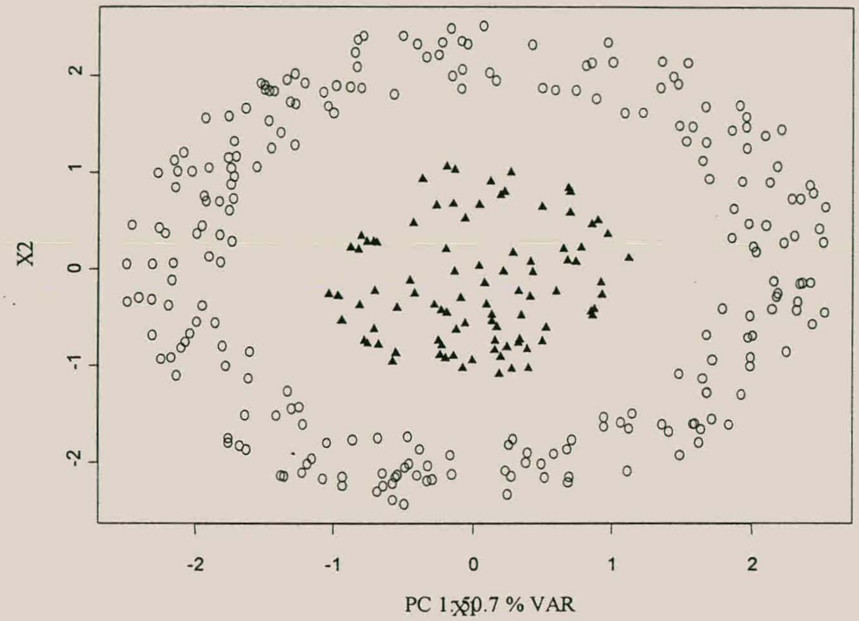

Figure 4.12: Clustering by means of single link (example 4.3).

Complete link and average link clustered this data set better than the principal curve method, but still not successfully. Single link clustered all of these data sets perfectly, whereas the principal curve method only succeeded in clustering the first data set. The initiative of this new method is to search for a technique to cluster process engineering data in which the linkage methods, in fact most of the existing techniques, fail completely. These two-dimensional data sets are interesting to use in order to compare the results of known methods with the new method, and clearly show some of the limitations of the method.

Example 4.4: Industrial Flotation Plant Data

Froth flotation is a complex process and consequently the plant operators usually evaluate the state of the flotation process on the basis of the visual appearance of the froth phase. Moolman et al. (1995) has recently automated this approach by using a computer vision system that extracts features from digitised images of the froth. Clusters found in such data can generally be associated with systematic changes in plant operation, which is often the result of some external disturbances.

The data were gathered from digitised images of the froth phase, from which five statistical variables were extracted. These five variables were measured on 297 observations. There appeared to be some outliers, which have been eliminated. See appendix 2 for more detail on the detection of these spurious objects. The first variable is inversely proportional to the bubble size. The second to fourth variables were indicative also of the bubble size as well as of the colour of the froth. The fifth variable related to the stability and mobility of the froth. The data were standardised to unit variance prior to any analyses, in order to circumvent the problem of non-commensurate features/variables with different variances. Figure 4.13 illustrates the principal curve moving through the data.

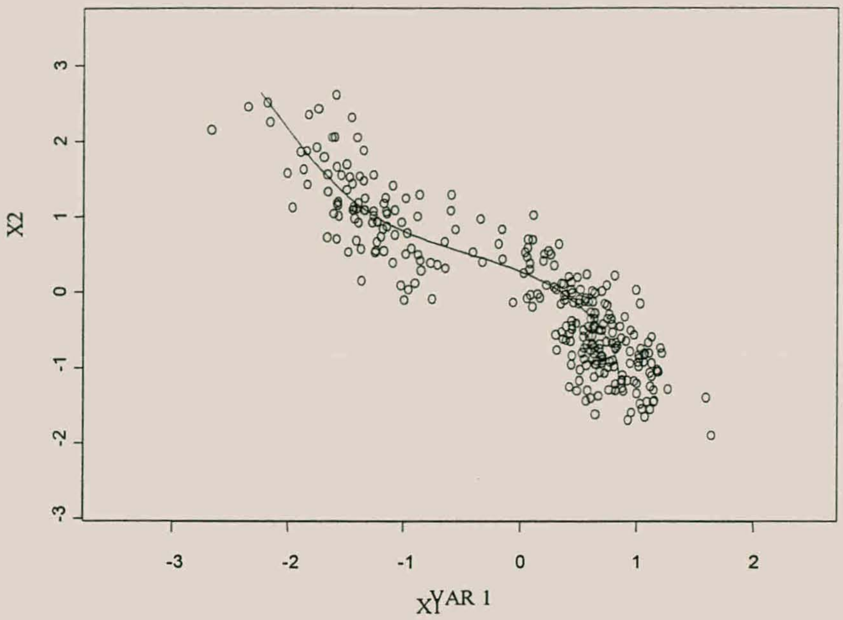

Figure 4.13: The principal curve moving through the data as plotted in the space of the first two variables (example 4.4).

Judging by the eye, it appears that there are two regions on the curve that are denser than the other areas. The differences between the consecutive, ordered principal curve scores are shown in figure 4.14.

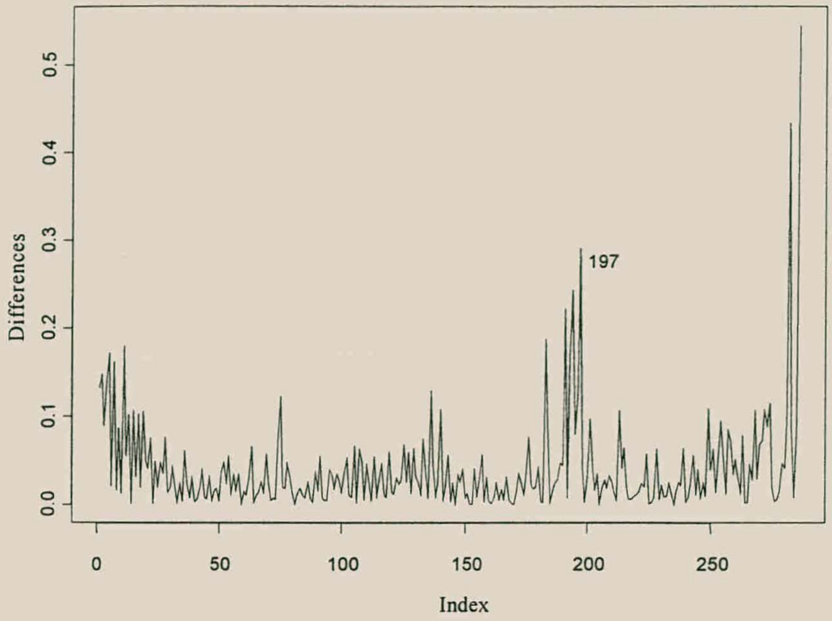

Figure 4.14: The differences between the consecutive, ordered principal curve scores (example 4.4).

One difference seems to stand out and thus the observations were split according to these results. The principal curve moving through the data together with the two clusters obtained via this method, are plotted in the space of the first two principal components, which can be viewed in figure 4.15. It was found that the first group contained heavily mineralised froths as opposed to the second cluster, which consisted of demineralised froths with small bubbles. The first two components collectively account for 93.9% of the variance in the data. Figures 4.16 and 4.17 portray the data clustered by means of the *k*-medoids method and complete link, respectively, again plotted in the space of the first two principal components. As can be seen from these figures, the corresponding methods did not cluster the data as well as the principal curve method. The *k*-medoids method grouped the data quite well, except for the few points on the left side of the top cluster. Complete link clustering failed completely. Single link was also unsuccessful in its attempt to cluster this data set.

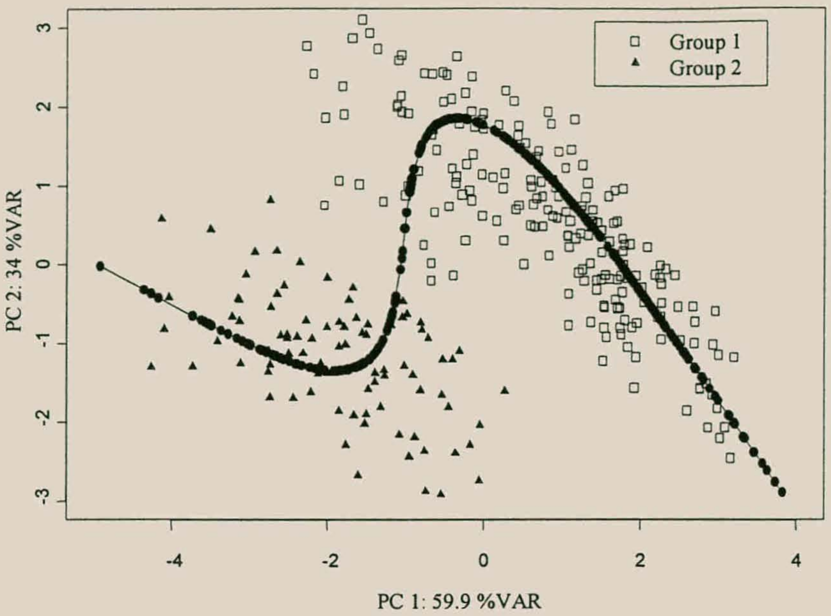

Figure 4.15: Clusters obtained via the proposed method together with the curve moving through the centre of the data, plotted in the space of the first two principal components (example 4.4).

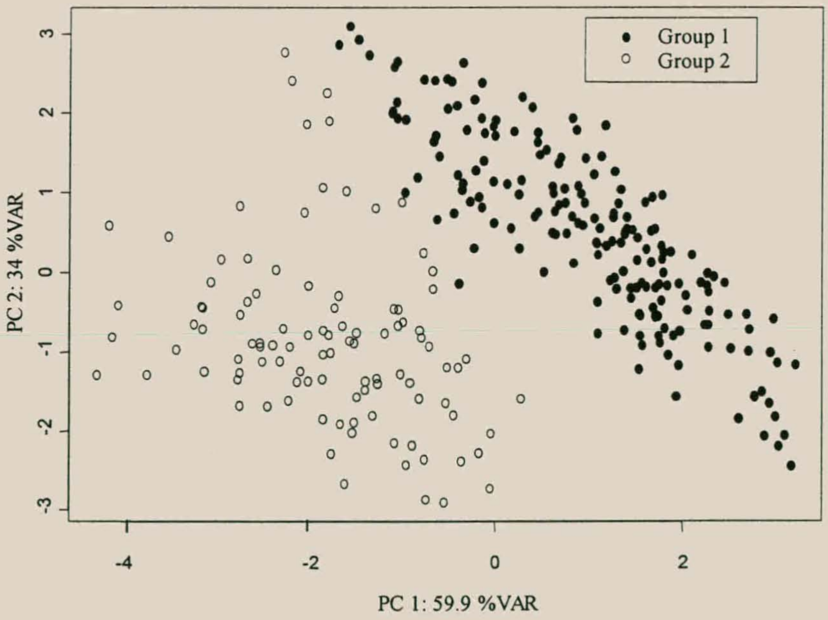

Figure 4.16: The clusters obtained through the *k*-medoids method (example 4.4).

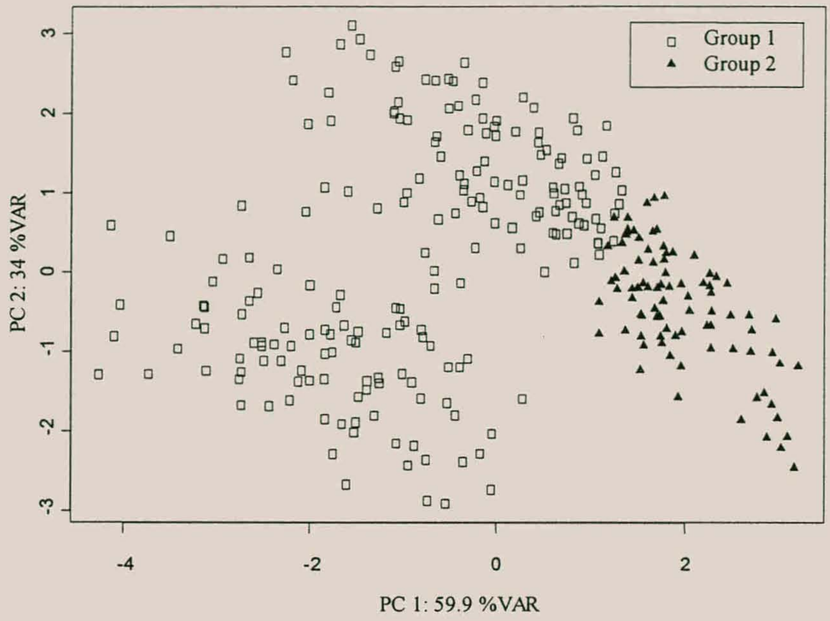

Figure 4.17: Complete link clustering of the flotation data, plotted in the space of the first two principal components (example 4.4). The bias of the method towards spherical clusters is evident.

Example 4.5: Multiphase Flow in Pipelines

This data set contains simulated observations modelling non-intrusive measurements on a pipeline. The three-phase flow of oil, water and gas in pipelines has previously been described by Bishop et al. (1997). The flow in the pipe adopts one out of three possible configurations, namely horizontally stratified, nested annular or homogeneous mixture flow. The flow was characterised by 1000 observations made on 12 features. The variables were highly correlated and in order to remove these intercorrelations, principal component analysis was applied to the raw data, which simultaneously reduced the essential dimensionality to three. These first three components accounted for 79.2% of the total variability of the raw data set. As in the previous example, the variables were scaled to unit variance and outliers also seemed to be present in the data set. Once again the reader is referred to appendix 2 for information on the detection of these points. Even though the first two components accounted for more of the variation in the data (64.2%) than the second and third components (35.2%), the latter two components separated the three groups better than the first two components. The reason for this was that the nested annular and homogeneous mixture clusters, predominantly loaded onto the first two components, which obscured the structure of the horizontally stratified flow regime. Thus, all results will be plotted in the space of the last two newly acquired variables (principal components), except of course for the graph of the differences between the successive, ordered scores. This is simply done to visualize this particular data set, and is not a prerequisite for the method in general.

Figure 4.18 illustrates the actual groups plotted in the space of the first two components, whereafter the same plot in the space of the last two components can be viewed in figure 4.19.

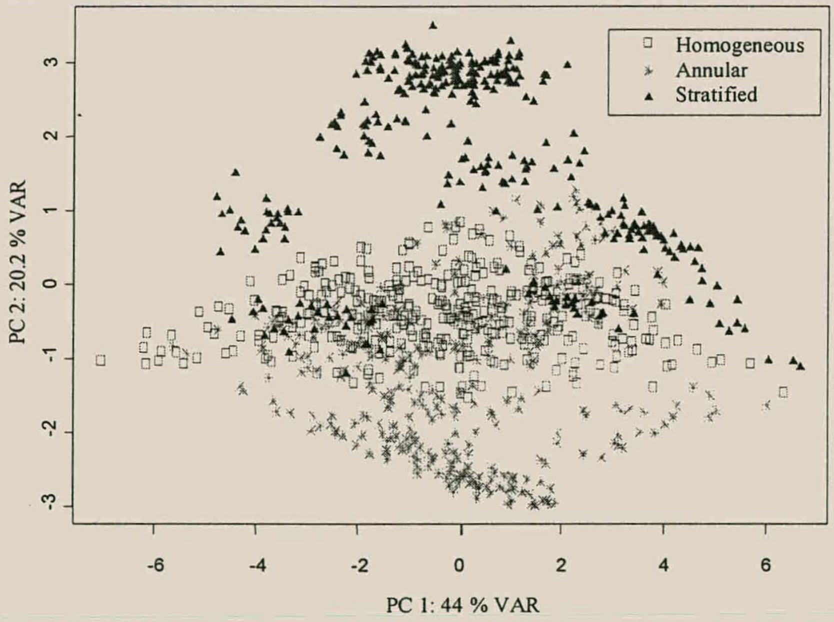

Figure 4.18: Actual clusters of the three-phase data plotted in the space of the first two (principal components).

It is evident from figure 4.18 that the first two principal components do not separate the three actual groups very well. Thus, it is hard to identify the regions of each cluster, since all three groups seem to overlap. Comparing this graph with figure 4.19, it becomes clear why it was decided to plot the results in the space of the second and third principal component. Although these two principal components collectively account for only 35.2% of the total variance in the data, the smaller clusters in the data are clearly visible.

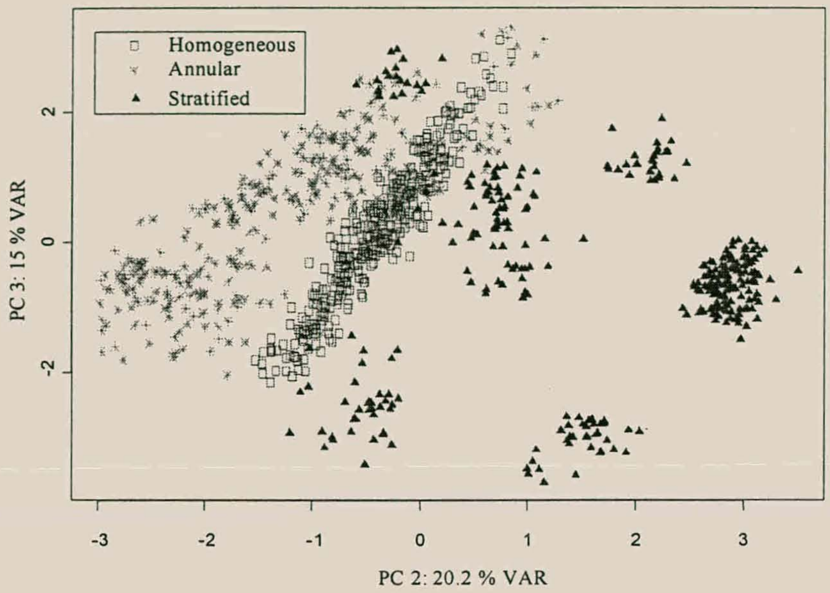

Figure 4.19: The actual clusters of the three-phase data plotted in the space of the 2nd and 3rd principal components.

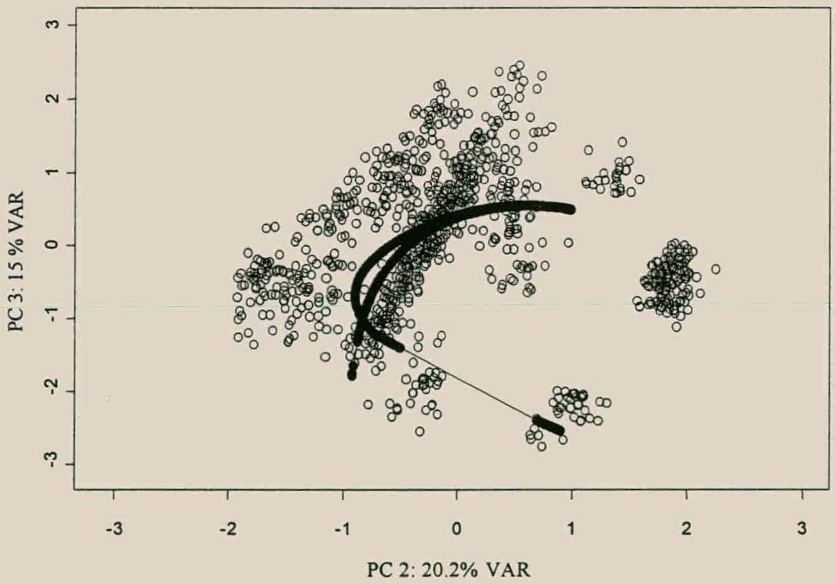

Figure 4.20: The principal curve moving through the data, as seen in the space of the 2nd and 3rd principal components.

This data set has a much more complex internal structure than the data in the previous example. It is not easy to distinguish the clusters by examining the projections of the points onto the curve. Figure 4.21 depicts the differences between the sequential, ordered curve scores.

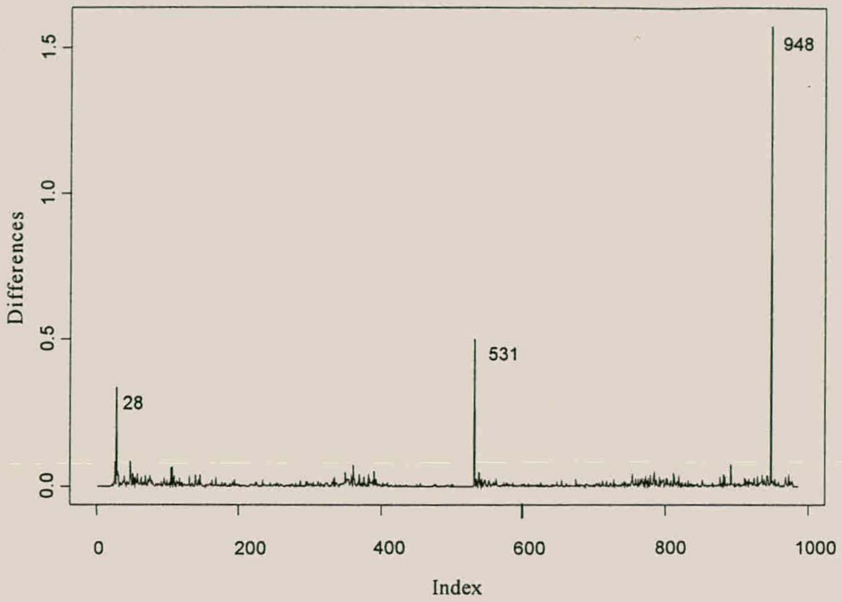

Figure 4.21: Differences of the sequential, ordered principal curve scores.

According to the results portrayed in figure 4.21, there are four clusters of which two are rather small. These four clusters can be seen in figure 4.22. Firstly, although the method found four clusters in the data set instead of three, the results are still good. The two main clusters, namely the homogeneous and annular groups, were satisfactorily identified. The horizontally stratified group consists of six smaller sub-clusters, of which two were identified, but as two different clusters. As seen in figure 4.19 the top small cluster, being one of the six sub-clusters forming the horizontally stratified group, appears to overlap the annular group. The proposed method correctly separated those overlapping points into two different groups, though it was not labelled correctly. The intention is to study these results together with plant experts, who would hopefully be able to associate some meaningful events concerning these clusters.

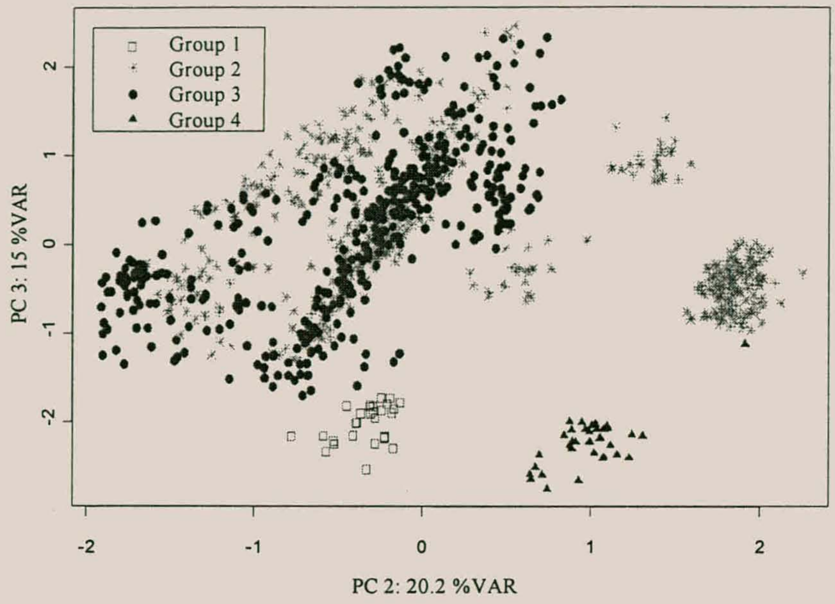

Figure 4.22: The four clusters obtained with the principal curve clustering method, plotted in the space of the last two variables.

Comparing the end results of the methodology with the results obtained from some known hierarchical clustering methods, it appears that the proposed technique performed better. Figure 4.23 shows the clusters obtained from complete link hierarchical clustering.

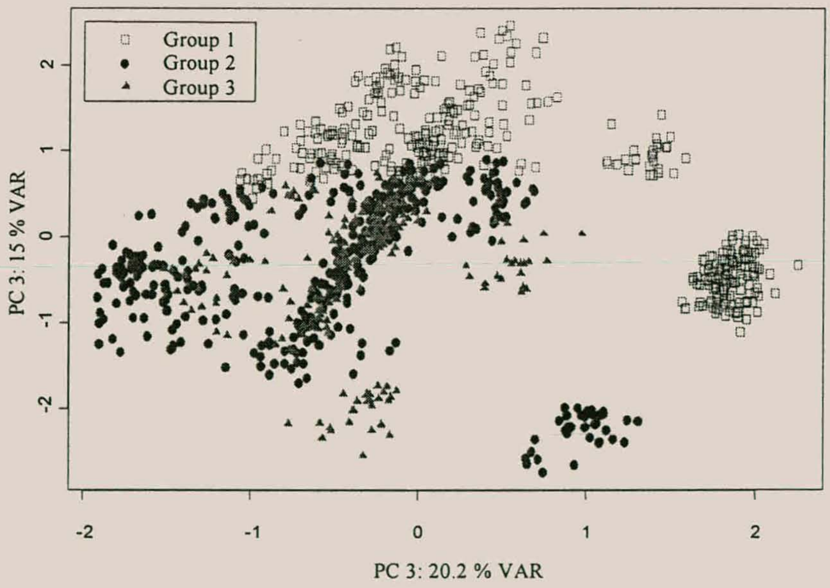

Figure 4.23: Clusters obtained through the complete link clustering method.

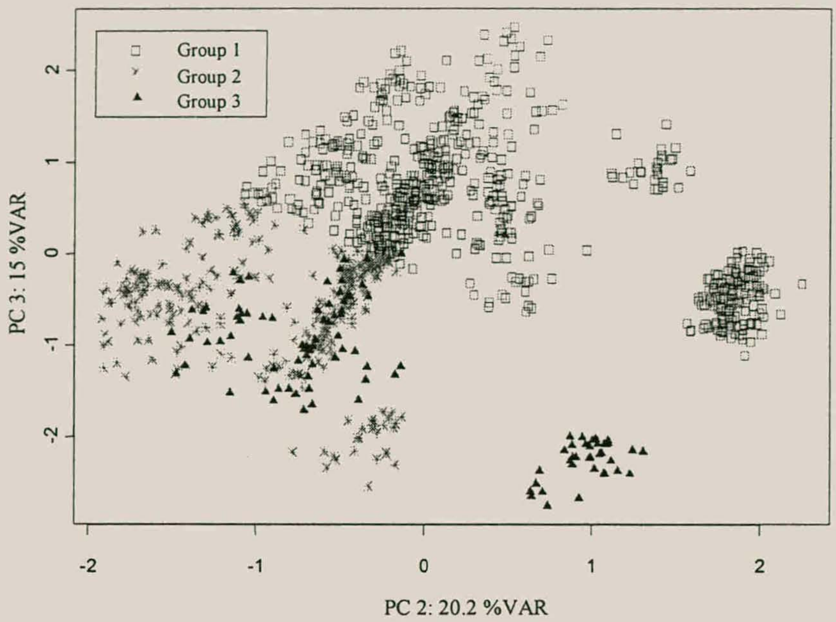

Figure 4.24: Groups acquired by means of divisive cluster analysis.

Neither divisive clustering nor any of the linkage clustering methods could cluster the data set into sensible groups. Thus, hierarchical techniques failed. K-means clustering, a partitioning method, achieved marginally better results than the hierarchical methods. A drawback of the k-means method is that the number of groups has to be specified beforehand by the user. The clusters acquired by the k-means method can be seen in figure 4.25.

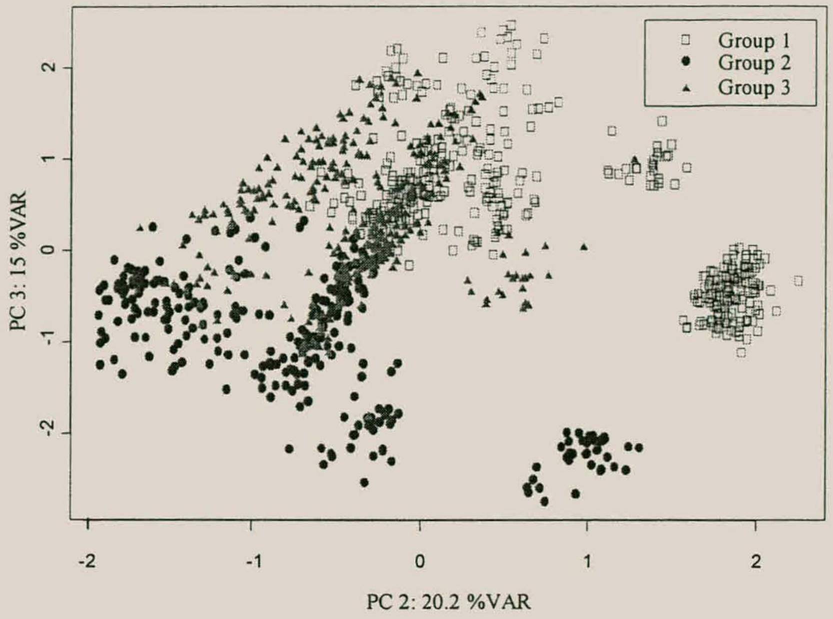

Figure 4.25: Clusters obtained by k-means clustering.

In order to compare the results of the different clustering methods, a classification table has been set up for each clustering algorithm. The rows correspond to the actual groups of the data set and the columns relate to the groups obtained via the different clustering methods. The values on the diagonal correspond to the number of correct clustered observations. The first table contains the results of the principal curve clustering method.

Table 4.1: Percentages of the correct classified observations of the multiphase-flow data as clustered by the principal curve method.

ACTUAL GROUPS	PRINCIPAL CURVE CLUSTERED OBSERVATIONS		
	Group 2	Group 3	Groups 1 & 4
Homogeneous	**34.8**	40.3	0.0
Annular	28.2	**40.5**	0.0
Stratified	37.0	19.2	**100**

Only 41.6% of the observations have been correctly grouped. This is due to the low number of observations clustered into the stratified group by the proposed method. Groups one and four, which can be viewed in figure 4.22, have been fused into one group for comparison to the actual stratified group. The next table gives the results obtained through complete link clustering.

Table 4.2: Classification table of multiphase-flow data as clustered by complete link.

ACTUAL GROUPS	COMPLETE LINK CLUSTERED OBSERVATIONS		
	Group 2	Group 1	Group 3
Homogeneous	**40.9**	19.3	53.5
Annular	36.5	**32.2**	19.3
Stratified	22.6	48.5	**27.3**

The total percentage of correctly clustered observations for complete link clustering is 34.9%. The corresponding results acquired via the divisive clustering technique are displayed in table 4.3.

Table 4.3: Classification table of multiphase-flow data as clustered through divisive hierarchical clustering.

ACTUAL GROUPS	DIVISIVE HIERARCHICAL CLUSTERED OBSERVATIONS		
	Group 1	Group 2	Group 3
Homogeneous	**31.4**	42.5	34.9
Annular	26.6	**47.0**	20.8
Stratified	42	10.5	**44.3**

Divisive clustering succeeded in clustering 37% of the data points correctly.

Table 4.4 displays the classification table of the k-means clustered observations, compared to the actual groups.

Table 4.4: Classification table of multiphase-flow data as clustered with the k-means technique.

ACTUAL GROUPS	K-MEANS CLUSTERED OBSERVATIONS		
	Group 2	Group 3	Group 1
Homogeneous	**49.4**	35.8	22.7
Annular	42.5	**38.3**	18.2
Stratified	8.1	25.9	**59.1**

K-means clustered this data set the best of all the clustering methods, with a 50.1% correct clustering rate. As a review, table 4.5 summarises the different clustering methods with the rows relating to the genuine groups and the columns corresponding to the four different methods, listing the percentages of correctly clustered observations into the three distinct groups.

Table 4.5: Summary of the correct clustered observations, expressed in percentages, of the Principal Curve (PC), Complete Link (CL), Divisive (DC) and the K-means (KM) clustering methods.

TRUE GROUP	PC	CL	DC	KM
Homogeneous	34.8	40.9	31.4	49.4
Annular	40.5	32.2	47.0	38.3
Stratified	100	27.3	44.3	59.1
Total	41.6	34.9	37.0	50.1

As can be seen from these results, the k-means method gave the best overall clustering results of this data set. The hierarchical methods did not cluster the data very well (both less than 40% accuracy). The principal curve method did not give as good results as was anticipated. It is important to bear in mind that in order to perform k-means clustering, the number of groups has to be specified *a priori* by the user.

Cluster analysis using principal curves operates differently than the existing clustering techniques. It compares to clustering methods such as density seeking techniques. As mentioned in chapter 2, these methods appear to be the logical approach to identifying groups in a multidimensional data set. The method gave the best results for the industrial flotation data. The last data set, the multiphase flow data, is in general a difficult data set to cluster. The proposed technique clustered satisfactorily, compared to the other methods, except for k-means clustering. As mentioned, this technique imposes a structure on the data set, instead of searching for natural groups, because the number of groups has to be specified prior to the clustering.

All of the experimental work in this chapter has been done in the statistical package S-PLUS version 4.5, the professional edition for Windows. The algorithm used to obtain the principal curves is available in S-PLUS. For more information on this algorithm written in S-PLUS, see appendix 3.

CHAPTER 5

CLASSIFICATION WITH PRINCIPAL CURVES

The focus of this chapter is on the natural extension of clustering, namely classification. Once the groups in a data set have been established through cluster analysis, the features or characteristics of the different groups can be utilised, in order to compare the features of new observations of unknown origin by means of the established groups. A new point is allocated or labelled as belonging to the group to which it is 'nearest' or, in other words, to the group with which it has the most features in common. The proposed classification technique employed in this chapter has been described in detail in chapter 3.

Three case studies are considered in this chapter, of which two were introduced in the previous chapter, namely, the industrial flotation plant data and the three-phase data set. The first example is a simulated data set with two overlapping normally distributed groups, which are linearly separable. Therefore, the principal curve classification method is compared with the classic two-group linear classification rule, as described in section 2.2.1.

The second and third examples are not linearly separable thus it does not make sense to compare the principal curve classification with a linear classification rule. Instead, the proposed classification method is compared with a kernel-based classifier. This classifier was implemented through a probabilistic neural network, which uses Bayesian classification methods (Specht, 1990a). These neural nets use distribution functions to estimate the likelihood of a feature vector (an observation) belonging to a particular group. Through the use of exemplars, which are input vectors whose group membership is known, these nets are trained to represent the distribution functions.

For instance, let $\mathbf{x} = (x_1,x_2\ldots,x_p)$ be an observation and assume that there are k different classes. If $f(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}) \ldots, f_k(\mathbf{x})]$ is the set of probability density functions of the different class populations and $P = [p_1, p_2, \ldots, p_k]$ is the set of prior probabilities of an observation belonging to the different classes, then the Bayes classifier compares the k values $p_1f(\mathbf{x})$, $p_2\,f(\mathbf{x})\ldots$, $p_k\,f(\mathbf{x})$. Finally, the group having the highest value, are determined.

In order to implement this decision rule, the probability density functions have to be constructed. This is accomplished through Parzen estimation (Parzen, 1962), a non-parametric method that makes no assumptions regarding the nature of the distribution functions. That is,

$$f_i(\mathbf{x}) = [B/k_i]\sum\nolimits_j \exp\left(-(x-x_{ij})'(x-x_{ij})/4\sigma^2\right) \qquad (5.1)$$

where $B = 1/\left(2\pi^{p/2}\sigma^p\right)$. The Parzen estimator is developed from k_i training data points. The exponential terms (Parzen kernels) are in fact localised multivariate Gaussian curves that are added together and smoothed (the B-term). The structure of this network is as follows: an input layer and a normalising layer, which normalises the observation vector **x**, so that $\mathbf{x}'\mathbf{x} = 1$. These two layers consist of just as many nodes as there are variables (p). The data are then distributed from the normalising layer to the pattern or exemplar layer, which represents the Parzen kernels. After the Parzen layer, a summation layer sums the kernels and finally, a competitive output layer identifies the classes. The weights, which are associated with the nodes in the output or class layer, are composed of the *a priori* probabilities, p_i. These probabilities are assumed to be equal, unless it is specified otherwise. For more information concerning probabilistic neural networks, the reader is referred to Specht (1990a, 1990b).

Example 5.1: Gaussian Data

In the first example, a two-dimensional data set containing two overlapping spherical clusters is considered. The one group is centered at (-1, -1) while the other group has a mean (1,1) and both have a standard deviation of 1. A total of 999 observations were randomly generated, containing objects of both the groups. These observations were split up into three sets, namely the training, test and validation sets. The training and test sets were used to construct the classifier discussed below. Thus, each of the three sets consisted of 333 observations.

The first step of the proposed classification method involves calculating the principal curve for each of the two groups. The training sets of both groups with their fitted principal curves are portrayed in figure 5.1. As can be seen, the option to fit the principal curve with the circle as the starting curve, instead of the usual first principal component, was employed since the two groups are spherical. In practice prior knowledge such as this may not be available, in which case the curves can also be initiated by the first principal components, as described earlier. However, a circle as the starting curve will produce better results than using the first principal component, in cases where the data seem to have a spherical distribution. Therefore, examining the distribution of the data set is imperative prior to any analyses.

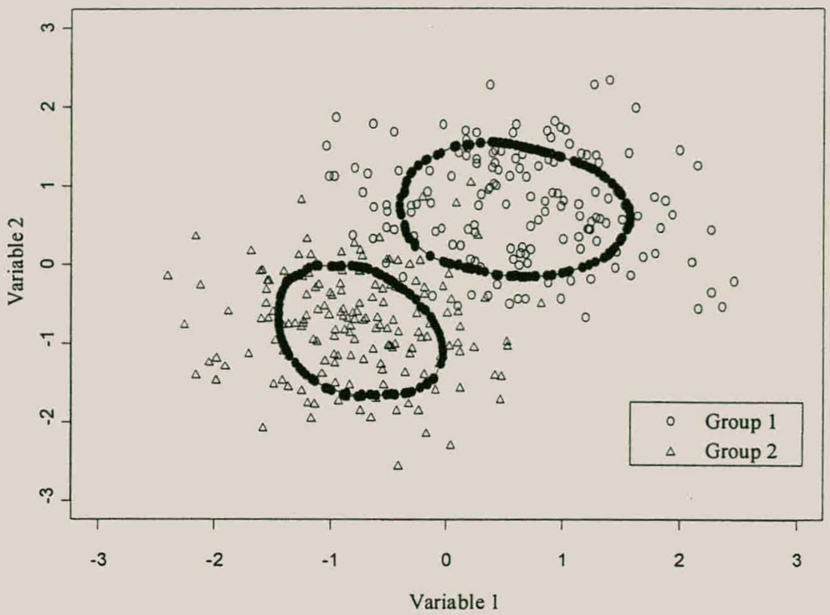

Figure 5.1: Two group Gaussian data set with each group's fitted principal curve, using the circle as the starting curve.

For each group, the relation between the observations and their projections onto the particular group's principal curve was acquired by means of a multilayer perceptron neural network. The trained networks were thus used to obtain the principal curve model for each group. The neural network of the first group contained one hidden layer with two log-sigmoidal nodes and a linear output layer. After 2000 iterations the sum-squared error (SSE) of the neural network reached 0.14. The second group was trained with a neural network including one hidden layer of two log-sigmoidal nodes. The SSE of 0.19 was achieved after 15 000 epochs. In order to test the generalisation of the trained networks, both groups' test sets were used as input to the respective networks.

The co-ordinates of each group's actual principal curve, extracted from the group's test set, had a strong positive correlation with the co-ordinates of the simulated principal curve (obtained by using the test sets as input into the trained networks). The correlation coefficients for each group can be viewed in table 5.1

Table 5.1: The correlation coefficients between each group's actual principal curve and simulated principal curve co-ordinates.

ACTUAL GROUPS	CORRELATION COEFFICIENTS	
	Variable 1	Variable 2
Group 1	0.98	0.87
Group 2	0.91	0.97

The next step was using the validation set, which represented the 'new observations' of unknown origin, as input for both the groups' neural nets. This was to project the new observations onto every group's principal curve. Finally, the Euclidean distances between each point and their projections onto every group's curve were obtained. The group nearest to the new observation, is the group to which the point was allocated. A classification table is shown in table 5.2, which contains the allocations of every observation in the validation set, to one of the two groups. The rows of the table correspond to the actual group and the columns contain the allocated points, as classified by means of the principal curve classification method. Thus, the entries on the diagonal, running from the top left to the bottom right, represent the percentage of correct classifications.

Table 5.2: Classification table of the Gaussian data set, with the percentage of correct classified observations on the diagonal, as obtained through principal curve classification.

ACTUAL GROUPS	PRINCIPAL CURVE CLASSIFIED GROUPS	
	Group 1	Group 2
Group 1	**81.2**	2.3
Group 2	18.8	**97.7**

The overall percentage of correct classifications was 87.7%. As can be seen in figure 5.1, the two clusters are alike, considering their shape and distribution, and they overlap considerably. Considering all this, the method performed well.

Classic two-group linear classification was applied to this data set after testing both groups statistically for a similar covariance structure. The training and test sets were fused into one set to obtain the classification rule. The validation set was classified according to this obtained rule. The results are listed in table 5.3.

Table 5.3: Classification table of the Gaussian data set, with the percentage of correct classified observations on the diagonal, as obtained through linear discriminant analysis.

ACTUAL GROUPS	PRINCIPAL CURVE CLASSIFIED GROUPS	
	Group 1	Group 2
Group 1	**92.1**	8.9
Group 2	7.9	**91.1**

The overall percentage of correct classification was 91.6%. This method of classification performed slightly better than the principal curve method, however, the difference was statistically insignificant. The groups are linearly separable and the linear classification method, utilising this feature, gave similar results as the principal curve method.

Example 5.2: Multiphase Flow in Pipelines.

There are three distinct known groups in this data set, which include the horizontally stratified (Strat), nested annular (Ann) and the homogeneous mixture flow (Hom) regimes. The 12 original variables were used to derive the principal curves for each of the three groups. The 12 variables of the data set were standardised to unit variance prior to deriving the principal curves.

The original data set consisted of 3000 observations of which 30 were labelled as outliers and removed accordingly. The resulting data set was split into three sets, namely the training set, the test set and the validation set. Each of these sets contained 990 observations and 12 variables. The relation between the data points and their projections onto the principal curve for the first group, namely the annular group, were obtained by a neural network consisting of one hidden layer with three tan-sigmoidal nodes and a linear output layer. After 15 000 iterations this neural net was trained with a sum-squared error (SSE) of 0.64.

The neural net of the homogeneous group, contained also only one hidden layer but with three log-sigmoidal nodes and a linear output layer. After 15 000 epochs the net trained to a SSE of 0.42. The net representing the last group, namely the horizontally stratified group, consisted of one hidden layer with three tan-sigmoidal nodes as well as a linear output layer. The SSE after 15 000 iterations was 0.23, the best of all three groups.

The results obtained from a neural network presented with the test set are shown in table 5.4. This table lists the correlation coefficients between the co-ordinates of each group's actual principal curve and the co-ordinates of the simulated principal curve.

Table 5.4: The correlation coefficients between the three groups' actual principal curve and simulated principal curve co-ordinates.

ACTUAL GROUP	CORRELATION COEFFICIENTS					
	V1	V2	V3	V4	V5	V6
Hom	0.97	0.94	0.97	0.94	0.97	0.94
Ann	0.96	0.96	0.95	0.94	0.96	0.93
Strat	0.99	0.98	0.99	0.98	0.99	0.98
	V7	V8	V9	V10	V11	V12
Hom	0.97	0.94	0.97	0.94	0.97	0.93
Ann	0.95	0.94	0.95	0.94	0.95	0.94
Strat	0.99	0.99	0.99	0.98	0.84	0.96

Table 5.4 indicates that there were very strong positive correlations between the co-ordinates of the simulated curves and actual curves' co-ordinates, for all three groups. This denotes that the neural networks generalised satisfactorily.

Finally, classifying the observations in the validation set into one of these three groups, on the basis of the shortest Euclidean distance between a point and its projected values onto both the curves, yields the results summarised in table 5.5.

Table 5.5: Classification table of the multiphase flow data set, containing percentages of correct classified observations on the diagonal, as obtained through principal curve classification.

ACTUAL GROUPS	PRINCIPAL CURVE CLASSIFIED GROUPS		
	Homogeneous	Annular	Stratified
Homogeneous	**93.7**	0.3	0
Annular	4.6	**99.7**	0
Stratified	1.7	0	**100**

The overall percentage of correct classification of the observations in the validation set, performed by the proposed method, was 97.7%.

The probabilistic neural network was trained using 900 observations, after which the remaining 100 observations were classified as belonging to one of the three groups. The exemplar layer consisted of 100 nodes and the output layer of the actual three groups. The classification table obtained by using the probabilistic neural network to classify the 100 'new' observations can be seen in table 5.6.

Table 5.6: Classification table of the multiphase flow data set, containing percentages of correct classified observations on the diagonal, as obtained by the probabilistic neural network.

ACTUAL GROUPS	PRINCIPAL CURVE CLASSIFIED GROUPS		
	Homogeneous	**Annular**	**Stratified**
Homogeneous	**100**	5.88	0.0
Annular	0.0	**94.12**	0.0
Stratified	0.0	0.0	**100**

On average, the probabilistic neural net thus classified 98.04% of the 'new' observations correctly. This is approximately similar to the results obtained by the principal curve method.

Example 5.3: Industrial Flotation Plant Data

The two groups, as established in chapter 4, are the groups known *a priori* in this example. That is, group 1 consisted of heavily mineralised froths whereas group 2 was composed of demineralised froths with small bubbles. As in the previous case studies, the data set was split up into the training, test and validation sets. Each set consisted of five features and 95 observations. Each of these sets was standardised to unit variance in order to eliminate the possible domination of variables with larger variances. The principal curve, for each of the two groups present in the training sample, was extracted. Figure 5.2 shows the two groups, as well as their respective curves, plotted in the space of the first two variables.

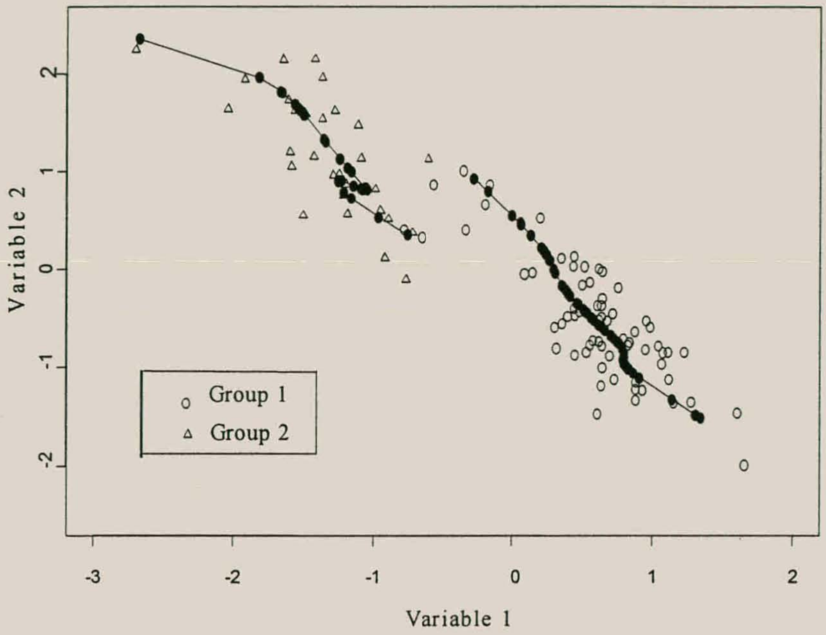

Figure 5.2: Two known groups in the data set with their respective fitted principal curves.

The neural networks for both groups were obtained by using each cluster's data points, and their projections onto the group's corresponding curve, as the inputs and targets respectively. Group 1 was trained with a neural net containing one hidden layer with two tan-sigmoidal nodes. After 2832 epochs the network's SSE reached 0.02. The second group was trained on a network with a single hidden layer containing two tan-sigmoidal nodes. Both groups' networks had a linear output layer. After 10 000 iterations the SSE was 0.017. Table 5.7 indicates how well the networks generalised.

Table 5.7: The correlation coefficients between each group's actual principal curve and simulated principal curve co-ordinates.

ACTUAL GROUPS	CORRELATION COEFFICIENTS				
	V1	V2	V3	V4	V5
Group 1	0.88	0.98	0.97	0.99	0.98
Group 2	0.99	0.98	0.83	0.89	0.93

Table 5.8 gives the classification table, which was obtained after the validation set was fed into both the neural networks. The two networks represented the two groups. Each point in the validation set was classified as belonging to that group, for which the point's projection onto the particular group's principal curve delivered the shortest Euclidean distance.

Table 5.8: Classification table of the industrial flotation plant data set, containing percentages of correct classifications on the diagonal, as obtained through principal curve classification.

ACTUAL GROUPS	PRINCIPAL CURVE CLASSIFIED GROUPS	
	Group 1	Group 2
Group 1	**100**	3.1
Group 2		**96.9**

The observations in the validation data set were 98.9% correctly classified by the proposed classification method. Similar results were obtained by classifying the observations in the validation set by means of a probabilistic neural network.

Good results were obtained via the principal curve classification method. It compares to the known linear classification method, as well as to classification with a probabilistic neural network. An advantage of the principal curve classification method is that a data set of any distribution can be classified. Classification with some known statistical methods requires prior analyses, such as testing for similar covariance structures. Different statistical classification rules are used depending on the outcome of the test for similar covariance structures.

The principal curve classification method can be used in conjunction with the principal curve clustering method. The clusters identified through the latter technique can be used as the groups known *a priori* in classification with principal curves.

The analysis of the data in this chapter, as in chapter 3, was done in S-PLUS, except for the neural networks. The Neural Network Toolbox of MATLAB version 4.2c for Windows was used for this purpose.

CHAPTER 6

CONCLUSIONS

Methods based on the use of principal curves for clustering and classification of particularly process engineering data are proposed in this study. Both these techniques are based on a one-dimensional representation of the data. This representation is obtained through a technique known as *principal curves,* developed by Hastie and Stuetzle (1989). Principal curves are said to move through the *middle* of the data cloud.

Cluster analysis by means of principal curves, employs the increasingly ordered principal curve scores of the observations. When the values of the successive ordered scores do not differ significantly, the scores will appear to be compact or close together. The observations corresponding to these dense points are grouped to form the clusters. Results obtained from this clustering method were compared to results of some of the well-known clustering techniques such as k-means clustering, the hierarchical linkage methods, as well as divisive hierarchical techniques.

This clustering technique is in some sense similar to density seeking techniques. However, the proposed method considers the distribution of the differences between the successive scores, unlike the density seeking methods, which concentrates on the density distribution of the scores.

Drawbacks of the proposed method include the interpretation of the distribution of the ordered principal curve scores, furthermore, small data sets present a problem since the densities of points within clusters may be comparable to the densities of points between clusters. In cases where the clusters in a data set tend to be aligned with the general direction of the principal curve, or when data sets contain complicated cluster configurations such as enclosed groups, the method failed in its attempt to identify the clusters.

It is known that most of the established clustering algorithms cannot cluster engineering data sets, since they were developed to cluster qualitative data. These techniques also have a computational drawback, since they employ the entire data set in the actual computation. Engineering data are of a quantitative nature and are usually highly dimensional. Needless to say, the well-known algorithms, especially hierarchical techniques, struggled to cluster such data and the results obtained through these methods are not reliable. The dimensionality of a data set is not an obstacle for the principal curve method, as the dimension is reduced. Consequently, the principal curve method yielded mainly superior results compared to the other methods, and results comparable to the partitioning method k-means. The proposed method has the advantage over the k-means method in so far as the number of groups need not be specified prior to the analyses.

Owing to the fact that the principal curve method reduces the dimensionality of the data to one, much less computer memory is required and the actual computation is quite faster than that of the standard methods. As a result, much larger data sets can be handled than with the other techniques.

The second aspect of this study is classification by use of principal curves. The relation between each group's data points and their projections onto the group's principal curve is determined by means of a back propagation neural network. The actual classification of the observations of unknown origin involves the projection of every 'new' data point onto each known group's principal curve. The group for which the new point is closest, in Euclidean space, to its projection onto the corresponding curve, is labelled as the class to which the new observation belongs.

Neither the proposed classification method nor the other techniques to which the principal curve method was compared, yielded superior results. A very high correctly classified percentage rate of the unidentified observations was acquired through all the different classification methods.

Unlike some of the standard statistical classification analyses, such as linear discriminant analysis, the proposed classification procedure does not need any assumptions regarding the data set. Regardless of the distribution of the data set or the number of classes present, the method is applied without any difficulty. The usual statistical classification procedures are applicable to two groups and need to be extended in the case where the data set has more than two groups. The principal curve method does not need to be adjusted in certain situations, for the same procedure is followed each time. An important issue in statistical classification is the equality of the different groups' covariance structures, which is unnecessary to test prior to the execution of the principal curve classification procedure.

Clusters that tend to overlap may lead to the incorrect classification of the observations. In such cases better results may be acquired through the linear separation of such groups. Analogous to classification trees, the principal curve classification method does not produce any information regarding the contribution each variable has in the separation of the groups. In some cases not all the variables are needed in the separation of the groups and may, therefore, be discarded. Consequently, since the principal curve method does not posses this aspect, it may have an economical impact, financially and time-wise, since the collection and measurement of these superfluous variables are not required.

Appendix 1: Estimates of Error Rates

This annotation describes various ways of the estimation of error rates, as mentioned in chapter 1, which does not assume normality but may be applied in any situation.

The *apparent error rate* uses a method known as *resubstitution.* This method entails that the training samples, used to acquire the classification rule, be re-used for classification in order to estimate the error rate. Owing to this, the estimator is over optimistic for small samples whereas for larger samples the amount of bias is rather small.

Leave-one-out is a method quite similar to the *jackknife* method proposed by Quenouille (1956) as a general technique to reduce bias in an estimator. For a one-step jackknife, this technique is as follows. Suppose $x_1,\ldots,x_n$ is a random sample and let the estimator of a parameter θ, be denoted by $T_n = T_n(x_1,\ldots,x_n)$. The n statistics $T_n^{(i)}, i = 1,\ldots,n$ are computed in order to "jackknife" T_n, where $T_n^{(i)}$ is calculated just as T_n except for x_i being omitted from the sample. The jackknife estimator of θ is denoted by

$$\mathrm{JK}(T_n) = nT_n - \frac{n-1}{n}\sum_{i=1}^{n} T_n^{(i)} \tag{A1.1}$$

The jackknife estimator $\mathrm{JK}(T_n)$ will in fact have a smaller bias than T_n. See Miller (1974) for a review on the properties of the jackknife. In the situation of classification, the application of this method is that each observation is deleted in turn, with the classification rule being computed from the remaining observations. The observation left out is then classified with the obtained rule. The number of misclassified observations is counted, which results in an almost unbiased estimate, however it has a large variance and mean square error.

The *bootstrap* method is another technique, such as cluster analysis, where the application became feasible because of the availability of ever stronger growing computing systems. An illustration of how this method works is as follows: Let $\mathbf{x} = (x_1,\ldots,x_n)$ denote n independent observations and let $s(\mathbf{x})$ be a statistic of interest computed from the n observations. Suppose the statistic of interest is the mean. It is known that the *estimated standard error* of the mean $\bar{x} = \sum_{i=1}^{n} x_i/n$ is given by the equation

$$\sqrt{\frac{s^2}{n}} \qquad \text{(A1.2)}$$

where $s^2 = \sum (x_i - \bar{x})^2/(n-1)$. The bootstrap estimate of the standard error is acquired by first obtaining the *bootstrap samples*, $\mathbf{x}^{*b} = (x_1^*,\ldots,x_n^*)$, with b = 1,…, B. A bootstrap sample is obtained by sampling randomly n times from the original data set $\mathbf{x} = (x_1,\ldots,x_n)$, with replacement. For each bootstrap sample a corresponding *bootstrap replication* of the statistic *s,* namely $s(\mathbf{x}^{*b})$, is calculated. The bootstrap estimate of the standard error is the standard deviation of the bootstrap replications, that is

$$\hat{s}_{\text{boot}} = \left\{\sum_{b=1}^{B}\left[s\left(\mathbf{x}^{*b}\right) - s(\cdot)\right]^2/(B-1)\right\}^{\frac{1}{2}} \qquad \text{(A1.3)}$$

where $s(\cdot) = \sum_{b=1}^{B} s\left(\mathbf{x}^{*b}\right)/B$. According to Efron (1993), assuming that $s(\mathbf{x})$ is the mean $\bar{x}$, as n approaches infinity (A1.3) approaches $\left\{\sum_{i=1}^{n}(x_i - \bar{x})^2/n^2\right\}^{\frac{1}{2}}$.

The calculation of the bootstrap process for estimating the standard error of the statistic $s(\mathbf{x})$ is depicted in figure A1.1. In comparison to the other error rate estimators, this method consists of the best features of the two preceding estimators, namely it has a small variance and is almost unbiased. A huge drawback is that this technique is rather time-consuming in that the number of classification rules must equal the number of replicates. The bootstrap is also unable to handle large biases.

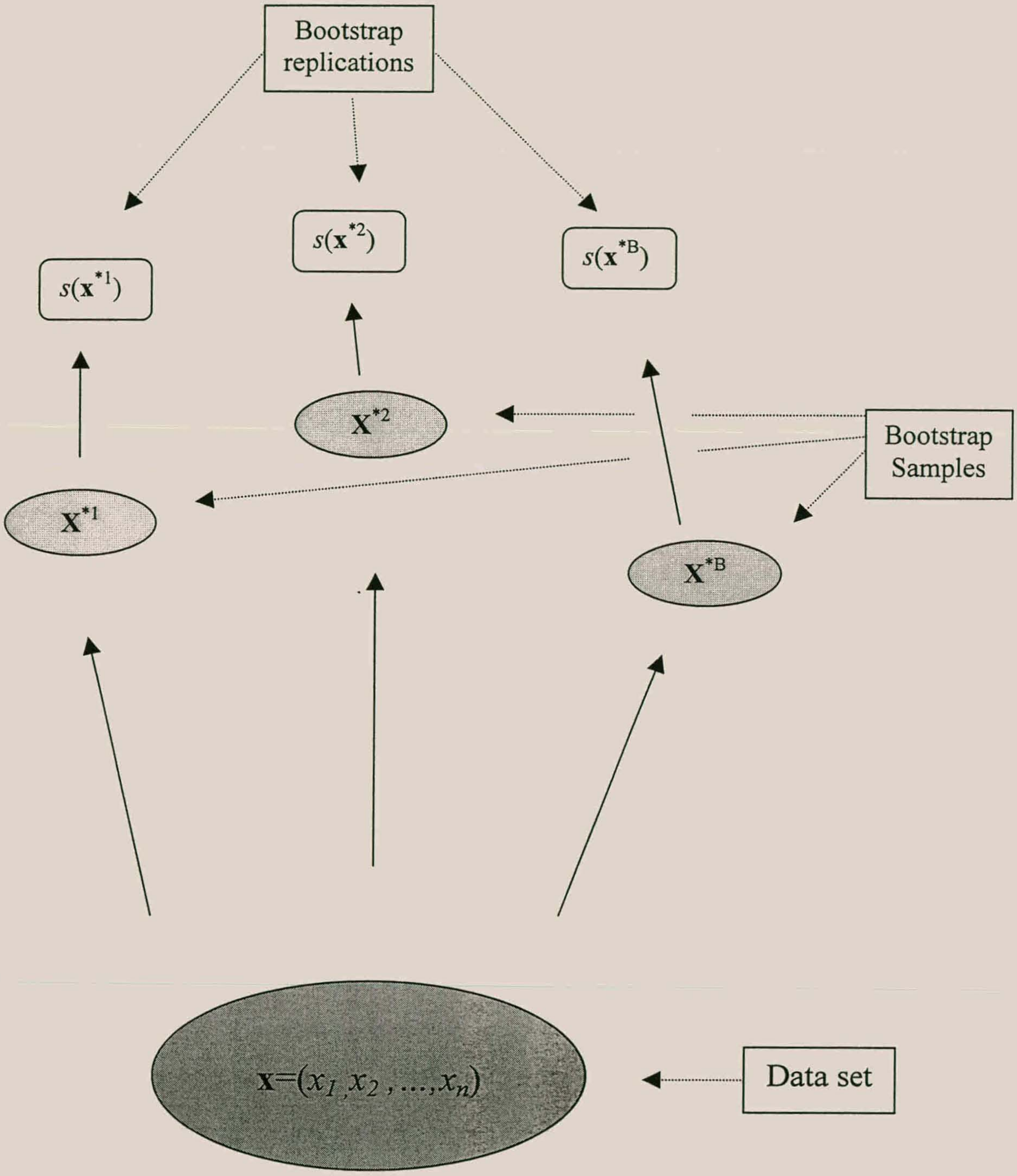

Figure A1.1: Schematic illustration of the bootstrap process.

Cross-validation is another method used to estimate the error rate. When the training sample is rather large it is split into two equal parts, where the one part is used to compute the rule, the second part is classified with the rule obtained from the first part. Otherwise, for smaller sets "*k*-fold" cross-validation is used, which entails the following: The sample is split into k parts. One part is omitted from the calculation of the classification rule from the remaining $k - 1$ parts, the *k-th* part is classified and the error rate is determined. This procedure is repeated k times, omitting each part in turn, and finally the k error rates are averaged. Usually k is set equal to n, then consequently the cross-validation error rate estimation is similar to the leave-one-out method (jackknife). The cross-validation method appears to be similar to the jackknife, because of the observations being omitted in turn, however no apparent parameter is being jackknifed and consequently there are no deeper relation between the two methods (Efron, 1982).

In the principal curve algorithm, cross-validation was also used to determine the spans for each of the p co-ordinate functions. For each $i = 1,\ldots,n$, the point x_i is predicted by means of a smoother applied to the sample, excluding the ith observation. Let $\hat{x}_{(i)}$ be this predicted value. The cross-validated residual sum of squares is defined by

$$\text{CVRSS} = \sum_{i=1}^{n} \left(x_i - \hat{x}_{(i)}\right)^2 \tag{A1.4}$$

CVRSS/n is approximately an unbiased estimator of the expected squared prediction error. When the span used in the smoother is too large, the curve will not capture the essential features in the data, resulting in a large CVRSS for the bias component will dominate. Conversely, when too small a span is picked the curve will follow the data very closely, including the noise in the data, again yielding a large CVRSS because the variance component increases. The span chosen is that value which yields the smallest CVRSS. Having a large enough training sample results in an unbiased estimate, however the mean square error tends to be rather large as in the case of the jackknife.

Another method, which differs from the rest, is one where the sample is split into two parts, the one being the training sample and the other is named the *validation sample*. The classification rule is computed from the former sample and then evaluated with the validation sample. This results in an unbiased error rate. If the sample size is small this procedure is not recommended. Keep in mind, that this rule is not the classification rule that will be used in practice. Preferably, the whole data set is used to construct the classification rule, in order to minimise the error rate's variance. This method occupies only half of the data set and the obtained error rate can vary significantly compared to the whole set used to construct the rule.

Appendix 2: Outliers

In chapter 4, page 60 it was mentioned that prior to any analysis, the data were screened for outliers. When suspect observations were detected they were eliminated. Some researchers do not believe in rejecting dubious observations, but rather to down-weight them (Venables and Ripley, 1997). Still, observations that are completely wrong can be eliminated. Only such observations were deleted since data that are "too clean" underestimate the variance. It is widely known in which way these spurious observations affect results, thus, it is always important to screen the data for outliers and not to assume that the data set does not contain any of these observations.

More attention has been paid to the detection of univariate outliers than to their multivariate counterparts. There are various ways to seek for multivariate outliers, of which no method is optimal. Even though a two or three-dimensional scatter diagram of all possible combinations of the *p* variables can be considered as the most elementary method, this diagram may reveal observations on the edge of the swarm of points which are distinctly separate from the other observations (Barnett and Lewis, 1994).

This method may not expose all possible outliers, therefore, the change of the co-ordinate basis, or in other words the rotation of the axes, may assist in disclosing dubious observations not observed previously. Another use of principal components is illustrated in this regard. By plotting the data in the space of the first and last few principal components respectively, different kinds of outliers can be pointed out. See figure A2.1 for an example of outlying observations in the space of the first two principal components. According to Gnanadesikan and Kettering (1972), outliers that inflate variances or covariances can be highlighted by the first few principal components, whereas the last few components are responsive towards outliers adding spurious dimensions to the data.

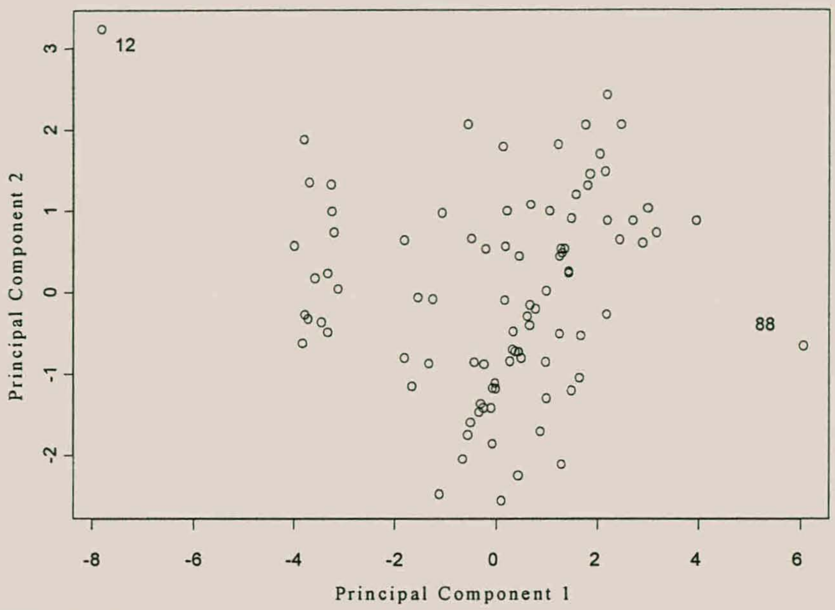

Figure A2.1: Projections of data onto the first two principal components to search for outliers.

It is evident in figure A2.1 that observations 12 and 88 are separated from the swarm of points, thus they can be distinguished as extremes. The individual principal component scores z_i, can be used in univariate outlier tests or be plotted against plotting positions, such as the normal probability plot. These plotting positions are only useful when assumptions can be made regarding the distribution of the original data set, otherwise they are useless. However, if p is fairly large it leads to the z_i being samples from approximately normal distributions, because of the linear transformations of the data with principal component analysis. In this case, observations that lie off the linear relationship of the normal probability plot are indicative of outliers.

Another method to search for outliers is to compute the Mahalanobis distances between all the observations, as in equation (1.2). As proposed by Johnson and Wichern (1998), producing a Chi-square plot (with p degrees of freedom) of the ordered distances will also reveal outliers. The observations furthest from the origin and also deviating from the straight line can be labelled as outliers. An example of this can be seen in figure A2.2. Outliers can also be indicated by comparing the distances with the critical value $\chi_{p,0.05}$. Those distances that exceed this critical value are also suspicious observations.

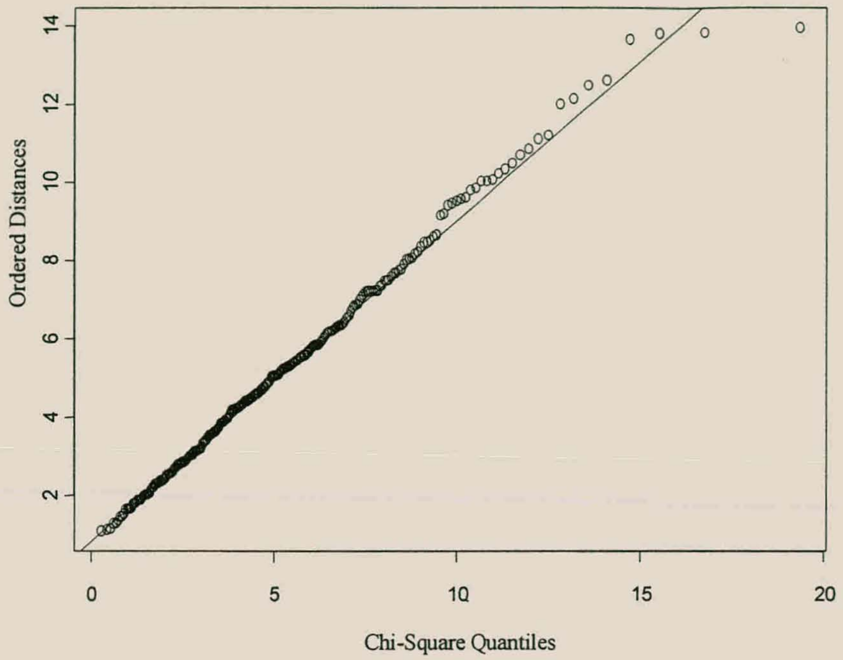

Figure A2.2: Chi-square plot of ordered Mahalanobis distances.

The top observation in figure A2.2 deviates from the fitted line and lies further apart than the other observations. The Mahalanobis distance can be compared to a more conservative critical value, namely the interquartile range (IQR), which is computed by subtracting the first quartile from the third quartile. This is a measure of dispersion of the data set and the decision is to label an observation whose corresponding Mahalanobis distance d_i > 1.5·IQR. These two methods employing the Mahalanobis distances are some of the methods, which transform multidimensional data to one-dimensional data, whereafter univariate methods can be applied to search for outliers.

As stated earlier, there is no superior method in the search for outliers, it is a trade-off between all of these methods. Observations that tend to show up as outliers in most of these techniques can be dealt with accordingly.

Appendix 3: Software

The algorithm for principal curves is not included in this version of S-PLUS, however, it is available on the World Wide Web at the addresses:

http://lib.stat.cmu.edu/S/
http://www.hensa.ac.uk/ftp/mirrors/statlib/

The principal curve function call is as follows:

principal.curve(x, start, thresh, plot.true, maxit, stretch, smoother, trace, ...)

The required arguments are:

x a matrix of points in arbitrary dimension

start either a previously fit principal curve, or else a matrix of points that in row order define a starting curve. If missing, then the first principal component is used. If the smoother is "periodic.lowess", then a circle is used as the start.

thresh convergence threshold on shortest distances to the curve; default is 0.001.

plot.true If TRUE the iterations are plotted.

maxit maximum number of iterations; default is 10.

stretch a factor by which the curve can be extrapolated when points are projected. Default is 2 (times the last segment length). The default is 0 for smoother = "periodic.lowess"

smoother choice of smoother. The default is "smooth.spline" and other choices are "lowess" and "periodic.lowess". The latter allows one to fit closed curves. Beware, you may want to use `iter = 0' with lowess().

trace If TRUE, the iteration information is printed

... additional arguments to the smoothers

The value obtained through the algorithm includes:

An object of class "principal.curve" is returned, that describes a smooth curve passing through the middle of the data x in an orthogonal sense. This curve is a nonparametric generalisation of a linear principal component. If a closed curve is fit (using `smoother = "periodic.lowess"') then the starting curve defaults to a circle, and each fit is followed by a bias correction suggested by J. Banfield. It has components:

s a matrix corresponding to x, giving their projections onto the curve.

tag an index, such that s[tag,] is smooth.

lambda for each point, its arc-length from the beginning of the curve. The curve is parameterised approximately by arc-length, and hence is unit-speed.

dist the sum-of-squared distances from the points to their projections.

The clustering algorithms used in this thesis are all included in the S-PLUS package.

1. The k-means clustering function call is as follows:

kmeans(x, centers, iter.max=10)

The required arguments are:

x matrix of multivariate data. Each row corresponds to an observation, and each column corresponds to a variable. Missing values are not accepted.

centers matrix of initial guesses for the cluster centers, or integer giving the number of clusters. If centers is an integer, hclust and cutree will be used to get initial values. If centers is a matrix, each row represents a cluster center, and thus centers must have the same number of columns as x. The number of rows in centers, (there must be at least two), is the number of clusters that will be formed. Missing values are not accepted.

The argument that is not obligatory is:

iter.max maximum number of iterations.

The value returned by the algorithm is an object of class kmeans with the following components:

cluster vector of integers, ranging from 1 to nrow(centers), with length the same as the number of rows of x. The ith value indicates the cluster in which the ith data point belongs.

center matrix like the input centers containing the locations of the final cluster centers. Each row is a cluster center location.

withinss vector of length nrow(centers). The ith value gives the within cluster sum of squares for the ith cluster.

size vector of length nrow(centers). The ith value gives the number of data points in cluster i.

2. The hierarchical clustering function call is:

hclust(dist, method = "compact", sim =)

The only required arguments are exactly one of dist or sim. The rest of the arguments are optional, which include:

dist a distance structure or distance matrix. Normally this will be the result of the function dist, but it can be any data of the form returned by dist, or a full, symmetric matrix. Missing values are not allowed.

method character string giving the clustering method. The three methods currently implemented are "average" (average link), "connected" (single linkage) and "compact" (complete linkage). The first three characters of the method are sufficient.

sim structure giving similarities rather than distances. This can either be a symmetric matrix or a vector with a "Size" attribute. Missing values are not allowed.

The returned value of the function includes:

Firstly, a "tree" representing the clustering, i.e., a list consisting of the following components:

merge an (n-1) by 2 matrix, if there were n objects in the original data. Row i of merge describes the merging of clusters at step i of the clustering. If an element j in the row is negative, then object -j was merged at this stage. If j is positive, then the merge was with the cluster formed at the (earlier) stage j of the algorithm.

height a vector of the clustering "height"; i.e., the distance between merged clusters at the consecutive stages.

order a vector giving a permutation of the original objects suitable for plotting, in the sense that a cluster plot using this ordering will not contain crossings of the branches.

REFERENCES

Aldrich, C., 1999, "Cluster analysis of mineral process data with autoassociative neural networks.", *Chemical Engineering Communications (in press)*

Aldrich, C., 1997, Neural Networks for the Process Industries, University of Stellenbosch, Stellenbosch

Barnett, V. and Lewis, T., 1994, Outliers in Statistical Data, 3rd ed., Wiley, Chichester, West Sussex

Bishop, C.M., Svensén, M. and Williams, C.K.I., 1996, "GTM: A principled alternative to the self-organising map.", *Lecture Notes in Conputer Science*, **1112**, pp.165 -170

Chang, K.Y. and Ghosh, J., 1998, "Principal curve classifier – a nonlinear approach to pattern classification.", *Proceedings of the IEEE World Congress on Computational Intelligence (WCCI '98)*, Anchorage, Alaska, May, 4-9

Chatfield, C. and Collins, A.J., 1980, Introduction to Multivariate Analysis, Chapman and Hall, London

Cheng, B. and Titterington, D.M., 1994, "Neural networks: A review from a statistical perspective.", *Statistical Science*, **9**(1), pp. 2-54

Efron, B.,1982, The Jackknife, the Bootstrap and Other Resampling Plans, *SIAM*, Philadelphia

Efron, B. and Tibshirani, R.J.,1993, An Introduction to the Bootstrap, Chapman and Hall, New York

Einax, J.W. and Soldt, U., 1999, "Geostatistical and multivariate statistical methods for the assessment of polluted soils – merits and limitations.", *Chemometrics and Intelligent Laboratory Systems*, **46**, pp. 79-91

Everitt, B.S. and Dunn, G., 1991, Applied Multivariate Data Analysis, Edward Arnold, London

Everitt, B.S. and Hand, D.J., 1981, Finite Mixture Distributions, Chapman and Hall, London

Everitt, B., 1974, Cluster Analysis, Heinemann Educational Books, London

Fix, E. and Hodges, J.L., 1951, "Discriminant analysis, nonparametric discrimination: consistency properties.", Report No.4., Project No. 21-49-004, Brooks Air Force Base, USAF School of Aviation Medicine

Ginsberg, D.W. and Whiten, W.J., 1991, "Cluster analysis for mineral processing applications.", *Transactions of the Institution of Mining and Metallurgy (Section C - Mineral Processing and Extractive Metallurgy)*, **100**, pp. C139-C146

Gnanadesikan, R. and Kettering, J.R., 1977, "Methods for statistical data analysis of multivariate observations." New York, Wiley, *Biometrics*, **28**, pp. 81-124

Gnanadesikan, R., 1989, see panel on discriminant analysis, classification and clustering

Götz, R., Steiner, B., Sievers, S., Friesel, P., Roch, K., Schwörer, R. and Haag, F., 1998, "Dioxin, dioxin-like PCBS and organotin compounds in the river Elbe and the Hamburg harbour: identification of sources.", *Water Science and Technology*, **37**(6-7), pp. 207-215

Green, P.E., Frank, R.E. and Robinson, P.J., 1967, "Cluster analysis in test market selection.", *Management Science*, **13**(8), pp. 387-400

Hastie, T. and Stuetzle, W., 1989, " Principal Curves.", *Journal of the American Statistical Association*, **84**(406), pp. 502-516

Johnson, R.A. and Wichern, D.W., 1998, Applied Multivariate Statistical Analysis, 4th ed., Prentice-Hall, Englewood Cliffs, New Jersey

Johnson, R.A. and Wichern, D.W., 1988, Applied Multivariate Statistical Analysis, 2nd ed., Prentice-Hall, Englewood Cliffs, New Jersey

Kaufman, L. and Rousseeuw, P.J., 1991, Finding Groups in Data: An Introduction to Cluster Analysis, Wiley, New York

Kotz, S., Johnson, N.L. and Read, C.B., 1982, *Encyclopaedia of Statistical Science*, 2nd ed., Nine Volumes, Wiley, New York

Loftsgaarden, D.O. and Quesenberry, C.P., 1965, "A nonparametric estimate of a multivariate density function.", *The Annals of Mathematical Statistics*, **36**, pp. 1049-1051

Macqueen, J. B., 1967, "Some methods for classification and analysis of multivariate observations.", *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability,* **1**, Berkeley, California: University of California Press, pp. 281-297

Miller, R.G., 1974, "The Jacknife – a review.", *Biometrika*, **61**, pp. 1-15

Moolman, D.W., Aldrich, C., Van Deventer, J.S.J. and Stange, W.W., 1995, "The classification of froth structures in a copper flotation plant by means of a neural net.", *International Journal of Mineral Processing*, **43**, pp. 193-208

Panel on Discriminant Analysis, Classification and Clustering, 1989, "Discriminant Analysis and Clustering.", *Statistical Science*, **4**(1), pp. 34-69

Parzen, E., 1962, "On estimation of a probability density function and mode.", Annals of Mathemetical Statistics, **33**, 1065-1076.

Pearson, E.S. and Kendall, M.G., 1970, Studies in the History of Statistics and Probability, Griffin, London

Press,S.J., 1982, Applied Multivariate Analysis using Bayesian and Frequentist Methods of Inference, 2nd ed., Robert E. Krieger Publishing Company, Malabar, Florida

Quenouille, M.H., 1956, "Notes on bias in estimation.", *Biometrika*, **43**, pp. 353-360

Rencher, A.C., 1995, Methods of Multivariate Analysis, Wiley, New York

Ruspini, E.H., 1970, "Numerical methods for fuzzy clustering.", *Information Science: An International Journal*, **2**, pp.379-350

Silverman, B.W., 1986, Density Estimates for Statistics and Data Analysis, Chapman and Hall, London

Specht, D.F., 1990a, "Probabilistic neural networks.", *Neural Networks*, **3**, pp. 109-118

Specht, D.F., 1990b, "Probabilistic neural networks and the polynomial Adaline as complementary techniques for classification.", *IEEE Transactions on Neural Networks*, **1**, pp. 111-121

Struyf, A., Hubert, M. and Rousseeuw, P.J., 1997, "Integrating robust clustering techniques in S-PLUS.", *Computational Statistics & Data Analysis*, **26**, pp. 17-37

Sustanto, E.L. and Warwick, K., 1995, "Multivariable cluster analysis for high-speed industrial machinery.", *IEE Proceedings: Science, Measurement and Technology*, **142**(5), pp. 417-423

Sylvester, J.J., 1889, "On the reduction of a biline quantic of the n^{th} order to form a sum of n products by a doubles orthogonal substitution.", *Messenger of Mathematics*, **19**, pp. 42-46

Venables, W.N. and Ripley B.D., 1997, Modern Applied Statistics with S-PLUS, 2nd ed., Springer, New York, NY

Wang, J., 1998, "A linear assignment algorithm for formation of machine cells and part families in cellular manufacturing.", *Computers and Industrial Engineering*, **35**(1-2), pp. 81-84

Wishart, D., 1969, Mode Analysis: In Numerical Taxonomy, edited by A.J. Cole, Academic Press, New York

Zemroch, P.J., 1986, "Cluster analysis as an experimental design generator, with application to gasoline blending experiments.", *Technometrics*, **28**(1), pp. 39-49

Zubin, J., 1938, "Socio-biological types and methods for their isolation.", *Psychiatry*, **1**, pp. 237-247