NPL Report CMSC 37/04

Report to the National Measurement System Policy Unit, Department of Trade and Industry

From the Software Support for Metrology Programme

Classification techniques and their application to metrology

By
M. Byng and S. Langdell
NAG Ltd., Oxford

February 2004

Classification techniques and their application to metrology

M. Byng and S. Langdell
NAG Ltd., Oxford

February 2004

ABSTRACT

In this report we consider techniques for analyzing and classifying data. The report is divided into two sections: methodology and case studies. The methodology section provides a basic overview of some common statistical techniques in the fields of data reduction, cluster analysis and classification methods. The second section provides two case studies showing how these techniques can be applied to practical problems in the field of metrology.

The first case study reports on the calculation and use of a visual representation, called a dendrogram, for groupings of spectroscopy measurements of different samples of the same substance. The analysis identified those samples which had been distressed in some manner. The utility of the dendrogram in comparison to standard techniques for visualizing measurement groupings is demonstrated. We conclude that the methods used in this case study not only help the analyst better understand the data but could, with additional training data, form the basis for an automated classification procedure. This procedure could aid in grouping spectra of the same substances and identifying possibly contaminated samples.

The second case study reports on a probabilistic method for identifying outlying values in measurement data where the outlier is due to the presence of an external factor, in this case the presence of dust particles on the surface being measured. The techniques used in this second case study are highly flexible and allow the incorporation of prior information in a rigorous manner.

ISSN 1471–0005

Authorised by Dr Dave Rayner,
Head of the Centre for Mathematics and Scientific Computing

National Physical Laboratory,
Queens Road, Teddington, Middlesex, United Kingdom TW11 0LW

Contents

1 Methodology

The statistical methods described in this document can be roughly split into three categories: dimension reducing, cluster analysis and discriminant analysis. There are many statistical techniques that can be used to perform these three tasks. This document covers the commonly used ones.

Dimension reducing covers those techniques used to select from a large number of observed variables a smaller subset of variables while still retaining a high proportion of the information in the original data. One of the most commonly used techniques for this purpose, principal components analysis, is described in Section 1.1. Once the number of variables in a data set has been reduced to a more manageable number the reduced data can then be used for other analyses, see, for example, Section 2.1 were a principal components analysis is followed by a cluster analysis. Other methods that can be used to reduce the number of dimensions include multi-dimensional scaling [15], variable subset selection and generalized linear models [18].

The aim of a cluster analysis is to find groups of similar observations. For example, with a database of customers' buying habits, a retailer may wish to group together customers with similar purchasing patterns. Given these groups, further analysis can be carried out. A cluster analysis starts with a measure of how similar or different, two observations are. Observations that are most similar are then clustered together. Within this document we present two forms of cluster analysis, k-means (Section 1.2) and hierarchical clustering (Section 1.3). In Section 1.4 we describe a graphical method of presenting the results of a hierarchical cluster analysis, called a dendrogram. The case study described in Section 2.1 gives an example of a cluster analysis using some of these techniques. The groups defined during a cluster analysis can be used to provide some insight into a data set of interest or they can be used as the input to other analysis techniques, for example, a discriminant analysis.

The aim of a classification is to assign a new observation to one of a number of categories or groupings. Unlike cluster analysis, where the groupings or clusters are constructed as part of the analysis, classification requires a set of predefined categories. Therefore all classification methods require a training data set. This is a set of data for which each observation has been assigned to a category. The same variables and scalings used to produce a training data set should be used to collect any subsequent data used to validate a model. The categories used in the training data set may have been observed. For example, in a customer survey they may consist of the type of product bought. Alternatively, the categories could have been created in a previous cluster analysis. Once the discriminant model has been trained it can be

used to categorize new observations. Within this document we describe a number of discrimination methods including k-nearest neighbour (Section 1.5), discrimination via binomial regression (Section 1.6) and radial basis functions (Section 1.7).

Finally, we describe hidden Markov models. Hidden Markov models are a general purpose model that can be used for, amongst other things, cluster analysis and outlier identification. Unlike the previous methods, hidden Markov models are usually implemented within a Bayesian framework. The case study described in Section 2.2 uses a hidden Markov model to identify measurements that have been potentially contaminated by dust particles when the shape of the surface of a mirror is being assessed.

With the exception of the radial basis function and hidden Markov models, all of the methods described in this document are readily available from a wide range of statistical packages such as R [14] or Genstat [26], and have been implemented in a number of statistical libraries (for example, the NAG Fortran [20] and DMC^2 [21] libraries). NAG DMC^2 also contains routines for fitting radial basis functions.

1.1 Principal Components Analysis

Principal components analysis (PCA) is a tool for reducing the number of variables that need to be considered in an analysis. PCA derives a set of orthogonal, i.e., uncorrelated, variables that contain a known proportion of the information (measured in terms of variance or sums of squares) in the original data. The new variables, called principal components, are calculated as linear transformations of the original data.

Let X be an $n \times p$ data matrix of n observations on p variables, $X = \{x_{ij}, i = 1, \ldots, n, j = 1, \ldots, p\}$. Let S be the $p \times p$ variance-covariance matrix of X. A vector a_1 of length p can be found such that

$$a_1^T S a_1 \text{ is maximized subject to } a_1^T a_1 = 1.$$

The first principal component, z_1, is then given by $z_1 = a_1^T x$ which gives the linear combination of the variables that explains the maximum variation. In other words, for any vector a of unit length we can calculate the variance of the n-vector Xa. The vector a_1 is a vector for which this variance is maximal. The kth principal component, z_k ($k \leq p$), is found in a similar manner, such that

$$\begin{aligned} a_k^T S a_k \text{ is maximized subject to } & a_k^T a_k = 1 \\ & \text{and } a_k^T a_i = 0, \text{ for all } i < k \end{aligned} \tag{1}$$

and $z_k = a_k^T x$. This gives the linear combination of variables that is orthogonal to the first $k-1$ principal components and which explains the maximum amount of the remaining variation. The vectors a_i, for $i = 1, 2, \ldots, p$ are the eigenvectors of the matrix S. Denoting the corresponding eigenvalues λ_i, for $i = 1, 2, \ldots, p$, the percentage of the total variance in a data set explained by the first k principal components is given by

$$\frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{p} \lambda_i} \times 100\% \tag{2}$$

When calculating the principal components, the correlation matrix, sums of squares and cross-products matrix or a standardized sums of squares and cross-products matrix may be substituted for the variance-covariance matrix S in (1). If the standardized sums of squares and cross-products matrix is used, S is replaced by $\sigma^{-1/2} S \sigma^{-1/2}$ for a diagonal matrix σ with positive elements.

Having performed a principal components analysis, one important question that needs to be addressed is the size of k, i.e., the number of principal components to use. If k is too small, important information is being ignored; if k is too large, the model may over–fit the data and provide a poor predictive tool. There are many possible ways of choosing k, some of which are detailed below:

1. The value of k can be chosen as the minimum number of principal components needed to explain a given percentage of variation in a data set as defined by (2).

2. The value of k can be chosen based on the following test for the equality of the remaining $p - k$ eigenvalues. If S is the variance–covariance matrix in (1), then a test statistic defined as:

$$(n - (2p + 5)/6) \left\{ - \sum_{i=k+1}^{p} \log(\lambda_i) + (p - k) \log \left(\sum_{i=k+1}^{p} \lambda_i (p - k) \right) \right\}$$

has, asymptotically, a χ^2 distribution with $\frac{1}{2}(p - k - 1)(p - l + 2)$ degrees of freedom, and forms the basis of a test for the equality of the remaining $p-k$ eigenvalues. Equality of the remaining eigenvalues indicates that if any more principal components are to be considered then they all should be considered.

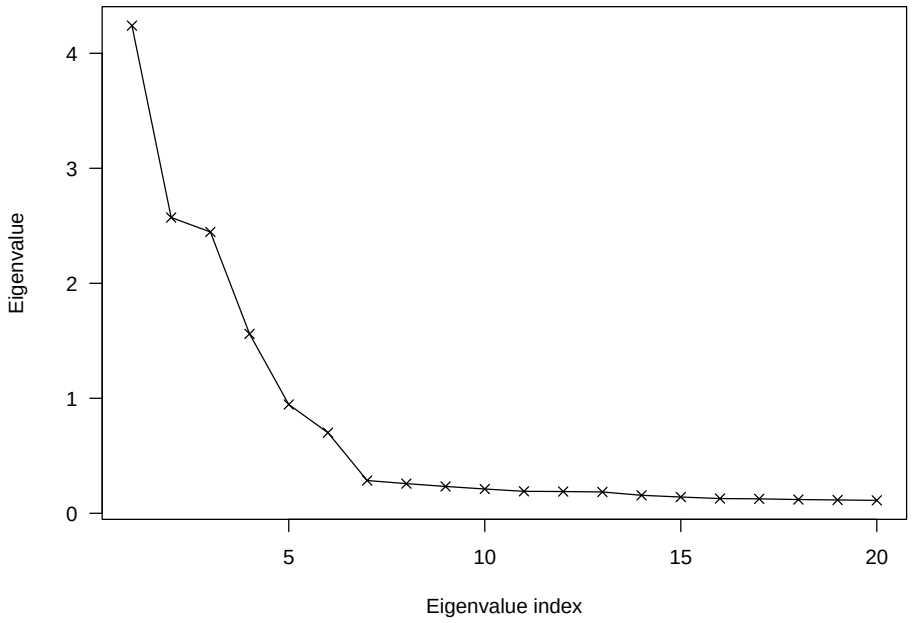


Figure 1: An example of a scree plot based on simulated data. Eigenvalues are plotted in decreasing order of magnitude. The clear bend at an index of 7 indicates suggests the use of the first seven principal components.

3. A visual method for choosing a suitable k is by plotting the eigenvalue, λ_i against its index i. This type of plot, sometimes called a "scree plot", results in a figure with a distinct "elbow bend". Principal components associated with eigenvalues to the right of this bend can be deemed not critical for accurately representing a data set. Figure 1 shows an example of a scree plot with 20 eigenvalues, suggesting a value of k equal to seven.

4. In some instances it is possible to "reconstruct" a data set using k principal components. This reconstructed data can then be compared with the original data, and if they differ significantly then a higher value for k may be considered.

Principal components analysis can provide a simple way to not only reduce the dimensionality of a problem, but also identify possible data groupings. Pairs of principal components can be plotted against each other and groups of observations identified. For example, Figure 7 in Section 2.1 shows a

plot of the first two principal components from two principal components analyses.

Classification on the basis of plots of pairs of principal components can have limited effectiveness. Firstly, unless the two principal components chosen explain a large percentage of the variation, the plots represent a limited subset of the information represented by the data. Secondly, a visual inspection of the plots is likely to be subjective with two different people arriving at two classifications. See Section 1.4 for an alternative way of visualizing groups formed by a cluster analysis.

1.2 k-Means Clustering

The aim of k-means clustering is to split a data set into k groups, each made up of similar observations. Before starting the analysis, the analyst must choose a value for k, i.e., how many groups or clusters there are in a data set. Each observation in a data set is initially allocated to one of these k groups. Given the number of clusters and an initial allocation, the k-means algorithm searches iteratively for the partition with locally optimal within-cluster sum of squares by moving observations between clusters until the grouping with the smallest total within-cluster distance is found.

Let $X = \{x_{ij} : i = 1, \ldots, n, j = 1, \ldots, p\}$ be an $n \times p$ data matrix of n observations on p variables. Let S_l denote the set of observations in the lth cluster, $(l = 1, \ldots, k)$, then the within-cluster distance is defined by:

$$\sum_{l=1}^{k} \sum_{i \in S_l} \sum_{j=1}^{p} w_i (x_{ij} - \hat{x}_{lj})^2$$

where:

$$\hat{x}_{lj} = \sum_{i \in S_l} \frac{1}{w_i} \sum_{i \in S_l} w_i x_{ij}$$

and w_i is the weight on the ith observation. In many cases $w_i = 1$, for $i = 1, 2, \ldots, n$.

The initial allocation of observations to clusters, can be performed in a number of different ways. One approach is to select k individuals at random and use these as the group centres. The remaining $n - k$ observations are then allocated to the group with the nearest group centre. Other approaches to finding initial clusters include using prior information or key individuals as group centres. The results obtained from a k-means cluster analysis may vary depending on the initial allocations.

Different numbers of clusters (i.e., different values for k) can highlight different patterns within a data set. Therefore, unless there is prior knowledge

about the number of groups, an analysis should be repeated with different values of k to see which value is best suited.

Section 1.3 introduces hierarchical cluster analysis, a clustering technique where neither k nor the initial allocation of observations to clusters needs to be specified.

1.3 Hierarchical Cluster Analysis

As with k-means cluster analysis the aim of hierarchical cluster analysis is to split a data set into a number of different groups (or clusters), each containing similar observations. Agglomerative hierarchical clustering starts with n clusters, one for each observation, and then combines these clusters step-by-step. At each of the $n-1$ steps two clusters are merged. After the final step there is only a single group which contains all observations. One advantage of this hierarchical approach over k-means clustering is that the order and distance at which the groups are joined together gives some indication of how many clusters are present in a data set and how well defined they are. There is no need to specify the number of clusters when performing a hierarchical cluster analysis, and no requirement to provide an initial starting point. Although the hierarchical clustering methods described in this report are agglomerative, hierarchical cluster analysis can also be divisive, i.e., start with a single cluster containing all of the observations and generate an additional cluster at each step.

At each stage of a hierarchical cluster analysis the two clusters which are the most similar are merged to form a new cluster. There are six common methods for calculating how far apart two clusters are: (a) single link (also called nearest neighbour), (b) complete link (also called furthest neighbour), (c) group average, (d) centroid, (e) median and (f) minimum variance. Let i, j and k denote three clusters with n_i, n_j and n_k observations in each. Let d_{ij} denote a measure of dissimilarity between clusters i and j. The metric d_{ij} is often referred to as the distance between cluster i and j. Suppose that clusters j and k are merged together to form a new cluster, jk, then, for the six common methods, the distance between this new cluster and cluster i, denoted $d_{i.jk}$, is defined by one of:

(a) Single link:

$$d_{i.jk} = \min(d_{ij}, d_{ik})$$

(b) Complete link:

$$d_{i.jk} = \max(d_{ij}, d_{ik})$$

(c) Group average:

$$d_{i.jk} = \frac{n_j d_{ij} + n_k d_{ik}}{n_j + n_k}$$

(d) Centroid:

$$d_{i.jk} = \frac{n_j(n_j + n_k)d_{ij} + n_k(n_j + n_k)d_{ik} - n_j n_k d_{jk}}{(n_j + n_k)^2}$$

(e) Median:

$$d_{i.jk} = \frac{2d_{ij} + 2d_{ik} - d_{jk}}{4}$$

(f) Minimum variance:

$$d_{i.jk} = \frac{(n_i + n_j)d_{ij} + (n_i + n_k)d_{ik} - n_i d_{jk}}{(n_i + n_j + n_k)}$$

To illustrate some of the differences between these different distance calculations the results of six hierarchical cluster analyses, performed on simulated data with ten observations, are given in Figure 2. The results are presented visually in the form of a dendrogram. A dendrogram is a visual technique for describing the results from a hierarchical clustering analysis (see Section 1.4).

The standard approach to carrying out hierarchical clustering is to first compute the distance matrix for all individuals and then update this matrix as the analysis proceeds. This approach is suitable for at most a few thousand individuals. However, a more efficient computational method exists for two of the six common methods described above, namely the group average and minimum variance methods. Both of these methods have the property that group members can be replaced by the group centroid or mean when calculating the between-group distances, so only the group means need to be stored at any one time. This approach is efficient computationally as it possible to compute many merges in parallel and does not require the storage of a n by n distance matrix.

Before applying a hierarchical clustering algorithm to a set of data a measure of dissimilarity must be constructed. This measure must quantify how similar (or dissimilar) two observations are. There are six commonly used distance measures: (a) Euclidean, (b) absolute, (c) threshold, (d) scaled rank and (e) simple matching. The absolute distance is also referred to as the city block metric or the Manhattan distance. Of these five common metrics, the first three apply to continuous variables and the last two (the

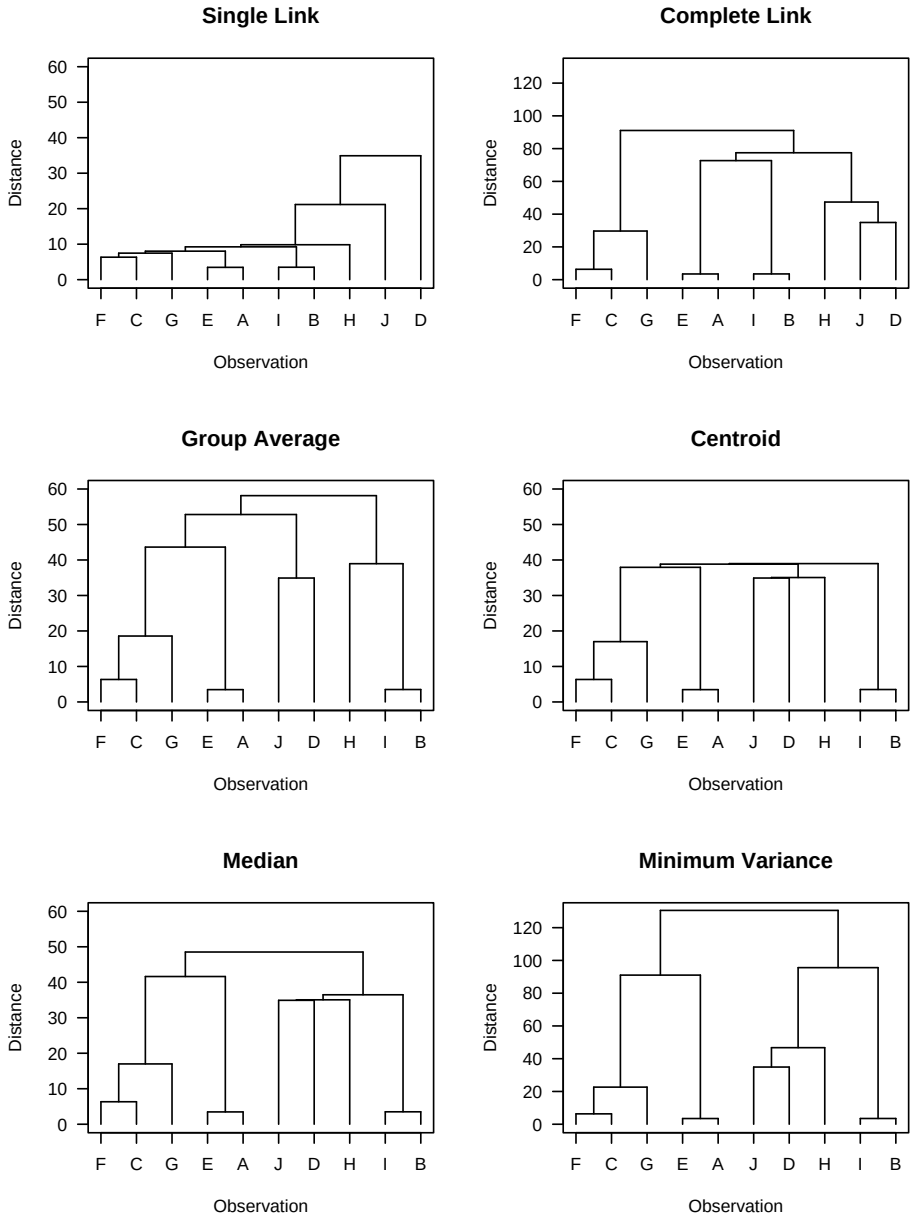


Figure 2: Results on performing hierarchical clustering on simulated data. The analysis was repeated six times, using a different method for calculating the distance between clusters each time. The data was simulated to have 10 observations, labelled A to J.

scaled rank and simple matching distance functions) to discrete variables. Let $X = \{x_{ij}, i = 1, \ldots, n, j = 1, \ldots, p\}$ be an $n \times p$ data matrix of n observations on p variables. Then the distance between variable k in observation i and variable k in observation j, d_{ij}^k, can be defined as:

(a) Euclidean distance:

$$d_{ij}^k = (S_k(x_{ik} - x_{jk}))^2$$

(b) Absolute distance:

$$d_{ij}^k = |S_k(x_{ik} - x_{jk})|$$

(c) Threshold distance:

$$d_{ij}^k = \begin{cases} 1 & \text{if } (x_{ik} - x_{jk}) > S_k \\ 0 & \text{otherwise} \end{cases}$$

(d) Scaled rank distance:

$$d_{ij}^k = \frac{|x_{ik} - x_{jk}|}{(c_k - 1)}$$

(e) Simple matching:

$$d_{ij}^k = \begin{cases} 1 & \text{if } x_{ik} = x_{jk} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where S_k is a scaling factor and, in the case of the scaled rank distance and simple matching, c_k is the number of categories for variable k, and x_{ik} must take values $1, \ldots, c_k$. A number of scaling factors can be used, including the standard deviation or range of variables or scalings based on a priori information.

Given a distance function for each of the p variables, the distance between observation i and observation j, d_{ij}, is calculated as:

$$d_{ij} = \sqrt{\sum_{k \in P_a} d_{ij}^k} + \sum_{k \in P} d_{ij}^k$$

where P_a denotes those parameters for which a Euclidean distance is used and P denotes all other parameters.

In addition to the five general distance functions described above and those given in other sources, see Everitt [8] or Krzanowski [16], domain-specific distance functions can be used.

An example of the application of hierarchical clustering is given in Section 2.1.

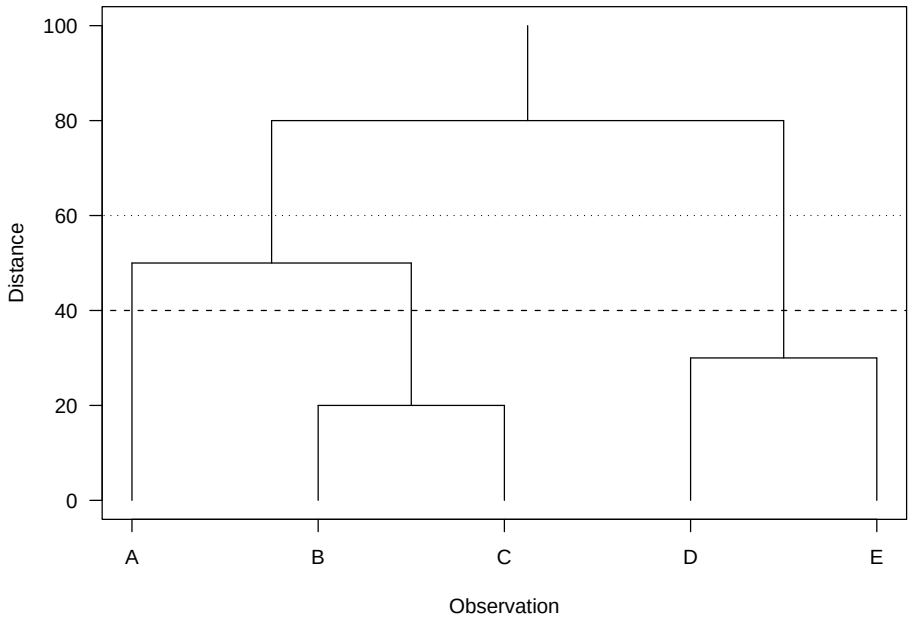


Figure 3: An example of a dendrogram based on simulated data.

1.4 Dendrogram

A dendrogram is a way to visualize the results from a hierarchical cluster analysis (see Section 1.3). The dendrogram displays which clusters merge and at what distance. The distance displayed on the dendrogram is defined by the intra-cluster distance measure adopted for the hierarchical cluster analysis.

Figure 3 gives an example of a dendrogram, with five observations (labelled A to E) and using an intra-cluster distance measure that has been scaled to give values between zero (clusters are identical) to 100. This plot indicates that at a distance of zero, each of the observations are in their own cluster. Observations B and C form a cluster at a distance of 20, observation A joins this cluster at a distance of 50. Observations D and E form their own cluster at a distance of 30, and these two clusters {A, B, C} and {D, E} join together at a distance of 80. Using a dendrogram it is easy to observe which observations are clustered together, at any given distance. For example, the dashed line indicates that at a distance of 40, we would use three clusters, {A}, {B, C} and {D, E}, where as at a distance of 60 (dotted line) we would

use two clusters {A, B, C} and {D, E}. Other examples of dendrograms and how to interpret them can be found in Section 2.1; see also Everitt [8] and Krzanowski [16].

1.5 k-Nearest Neighbours

Let $T = \{x_{ij} : i = 1, \ldots, n, j = 1, \ldots, p\}$ denote an $n \times p$ training data set. Associated to T is a vector $y = \{y_i : i = 1, \ldots, n\}$ of labels indicating to which of c categories each observation in T belongs. Given a new observation, $v = \{v_j : i = 1, \ldots, p\}$, the aim of a k-nearest neighbour analysis is to assign v to one of c possible categories.

The k-nearest neighbour approach searches the set of training observations T to find the k-nearest observations to v. For computational efficiency, nearest neighbours are found by using a binary tree search, see, for example, Bentley [2]. The proximity of v to each member of T is defined by a distance metric calculated over the p variables. Letting x_{ij} denote the jth variable of the ith training set two common measures of distance between v and the ith member of T are:

1. The ℓ_1-norm or Manhattan distance:

$$\sum_{j=1}^{p} |v_j - x_{ij}|.$$

2. The ℓ_2-norm or Euclidean distance:

$$\left[\sum_{j=1}^{p} (v_j - x_{ij})^2 \right]^{1/2}.$$

Let S be a set containing the k nearest neighbours in T to v, and h_l be the number of members of S belonging to the lth category. The posterior probability θ_l of v belonging to the lth category is given by:

$$\theta_l = \frac{p_l h_l}{\sum_{m=1}^{c} p_m h_m}, \quad l = 1, \ldots, c$$

where p_l is the prior probability of the lth category, i.e., the probability of assigning an observation to the lth category prior to viewing any data. Unless there is strong evidence to the contrary, it is common to use a uniform prior for p by setting $p_l = 1/c$, for all l.

Letting q denote the index of the maximum value of $\theta_l : l = 1, \ldots, c$, then observation v is classified as belonging to category q if $\theta_q > \rho$, where ρ is

a pre-specified tolerance. If $\theta_q \leq \rho$ then observation v remains unclassified. High values of ρ can lead to observations remaining unclassified, whereas low values for ρ can lead to observations being assigned to categories they have little in common with.

In addition to specifying the tolerance, ρ, a value for k must also be decided upon. Different values of k can, potentially, lead to different classifications. The larger the value of k the more likely an observation will be assigned to groups with a large number of distant neighbours compared to a group with a small number of close neighbours. Small values of k may lead to over–fitting of models, a value of $k = 1$ would give a perfect fit to the training data set, but is likely to lead to poor predictions for additional observations. More details on k-nearest neighbour can be found in Duda and Hart [6] and Hastie et al. [11], for example.

1.6 Binomial Regression

Binomial regression can be used to classify data into one of two groups. It is an example of a generalized linear model, where the error structure of a data set is assumed to be binomial. (Logistic regression is a special case of binomial regression.) In order to perform a discriminant analysis with binomial regression, a model must first be constructed using a training data set. This data set consists of n groups of observations that have been classified into one of two categories, labelled c_1 (successes) and c_2 (failures). Let $T = \{x_{ij} : i = 1, \ldots, n, j = 1, \ldots, p\}$ denote an $n \times p$ training data set. Let t_i be the number of observations belonging to the ith group in T. The y_i values are assumed to be realizations of a random variable Y drawn from the binomial probability distribution function:

$$P(Y = y_i) = \binom{t_i}{y_i} \pi_i^{y_i} (1 - \pi_i)^{(t_i - y_i)}, y_i = 0, 1, \ldots, t_i$$

where $\pi_i^{y_i}(1-\pi_i)^{(t_i-y_i)}$ is the probability of obtaining y_i successes and $t_i - y_i$ failures.

A binomial regression model consists of the following elements:

1. A linear model:
$$\eta_i = \sum_j \beta_j x_{ij}$$

2. A link function $\eta_i = g(\mu_i)$, linking the linear predictor, η_i, and the mean of the distribution, μ_i. Although it is possible to use a number of link functions the three common ones are:

(a) logistic link: $\eta_i = \log\left(\frac{\mu_i}{n - \mu_i}\right)$

(b) probit link: $\eta_i = \Phi^{-1}\left(\frac{\mu_i}{n}\right)$, where $\Phi(\cdot)$ is the Normal cumulative distribution function

(c) complementary log-log link: $\eta_i = \log\left(-\log\left(1 - \frac{\mu_i}{n}\right)\right)$.

3. A measure of fit, the deviance:

$$\sum_{i=1}^{n} \text{dev}(y_i, \hat{\mu}_i) = 2\sum_{i=1}^{n}\left[y_i \log\left(\frac{y_i}{\hat{\mu}_i}\right) + (t_i - y_i)\log\left(\frac{t_i - y_i}{t_i - \hat{\mu}_i}\right)\right]$$

When this model is fitted to the training data set estimates $\hat{\beta}$ of the parameters β are found.

Given a new observation $v = \{v_j : j = 1, \ldots, p\}$, with a unknown classification we can estimate the probability that v belongs to category c_1 by the following:

1. Obtain an estimate of the linear predictor for v, $\eta_v = \sum_j \hat{\beta}_j v_j$.
2. Obtain an estimate of $\hat{\pi}_v$ of π_v from $g^{-1}(\eta_v)$.

For example, in the case of logistic regression (i.e., binomial regression using a logit link [18]),

$$\hat{\pi}_v = \frac{\exp(\eta_v)}{\exp(1 - \eta_v)}$$

If $\hat{\pi}_v > 0.5$, v can be assigned to category c_1; if $\hat{\pi}_v < 0.5$, v can be assigned to category c_2; otherwise $\hat{\pi}_v = 0.5$ and we have no information about which category to assign the new observation.

The grouped data formulation given here for a binomial regression model is the most general one. It includes individual data as the special case of n groups of size 1, i.e., $t_i = 1$ for all i; and the case when all observations belong to a single group, i.e., $t_1 = n$.

Further information on the fitting of generalized linear models, including binomial regression, can be found in McCullagh and Nelder [18].

1.7 Radial Basis Functions

The basic premise behind the use of radial basis functions (RBF) is similar to that behind other discrimination methods in that given a set of test data

T consisting of n vectors of p independent variables, $\{x_i : i = 1, \ldots, n\}$ and a dependent variable y_i, $i = 1, \ldots, n$, then the following assumption is made:

$$y_i = f(x_i) + \epsilon, \quad f : \mathbb{R}^p \to \mathbb{R}$$

for some unknown function $f(\cdot)$ and noise ϵ drawn at random from a Normal distribution with zero mean and unit variance. Unlike in linear discrimination where the function $f(\cdot)$ is assumed to be a linear combination of the x_i, a radial basis function model approximates $f(\cdot)$ by a linear combination of t radial basis functions. The value of a RBF is defined by a centre location c, an observation x_i and a function $\phi(\cdot)$. Approximating $f(\cdot)$ by a linear combination of RBFs gives an approximate value $\hat{y}_i$ for the ith dependent value given by:

$$\hat{y}_i = \sum_{k=1}^{t} w_k h_{i,k} + b, \quad w_k, b \in \mathbb{R} \tag{4}$$

where b is a scalar bias term and:

$$h_{i,k} = \phi(z_{i,k}), \quad \begin{cases} z_{i,k} \in \mathbb{R} \\ \phi : \mathbb{R} \to \mathbb{R} \end{cases}$$

is the value of an RBF $\phi(\cdot)$ for a radial distance $z_{i,k}$ between x_i and a centre location c_k. The values of the weights, w_k, are estimated during the fitting process by, for example, minimizing the penalized (regularized) sum of squares error function:

$$\sum_{i=1}^{n} (y_i - \hat{y}_i)^2 + \lambda \sum_{k=1}^{t} w_k^2$$

where λ is the pre-specified regularization parameter.

The centre locations $\{c_k;\ k = 1, 2, \ldots, t\}$ can be any suitable p-dimensional vectors. Common methods for selecting these centres include:

(a) A random subset of t of the n independent data vectors $\{x_i;\ i = 1, 2, \ldots, n\}$.

(b) The results of a cluster analysis of the n independent data vectors for t clusters.

The above strategies are particularly well suited to classification tasks when there are often natural choices for the centres, i.e., as the centroids of data belonging to each category.

The radial distance $z_{i,k}$ between observation i and centre c_k may be calculated by using any distance function, common choices are:

1. Euclidean distance:

$$z_{i,k} = \left[(x_i - c_k)S^{-1}(x_i - c_k)^T\right]^{1/2}$$

in which S is a standardization metric S, for example:

   (a) a diagonal matrix where the jth diagonal element is the standard deviation of the j independent variable, for $j = 1, 2, \ldots, p$;

   (b) the variance-covariance matrix of data values of p independent variables, i.e., giving the Mahalanobis distance.

2. Absolute (Manhattan) distance:

$$z_{i,k} = \sum_{j=1}^{p} |(x_{i,j} - c_{k,j})/s_j|$$

where the s_j are scalings, for example, the standard deviation of the jth independent variable.

The radial basis function $\phi(\cdot)$ can also take many forms, common choices include:

1. Linear: $\phi(z) = z$.
2. Cubic: $\phi(z) = z^3$.
3. Thin plate spline: $\phi(z) = z^2 \ln(z)$.
4. Gaussian: $\phi(z) = \exp^{-z^2}$.
5. Multiquadric: $\phi(z, \alpha) = (z^2 + \alpha^2)^{1/2}$.
6. Inverse multiquadric: $\phi(z, \alpha) = (z^2 + \alpha^2)^{-1/2}$.
7. Cauchy: $\phi(z, \alpha) = (z^2 + \alpha^2)^{-1}$.

Figure 4 shows the form of the last six of these functions.

Once a radial basis function model has been fitted to the test data set T, then, given a new observation $v = \{v_j : i = 1, \ldots, p\}$, with unknown dependent variable, y, an estimate of the value of y can be obtained from (4), with $z_{i,k}$ set to the distance between variable v_i and the centre c_k.

When RBF models are used for classification tasks care must be taken to encode the categories of the dependent variable, y. In general, this encoding is of the form 1–of–c for c categories. In the case of two categories this requires a single dependent variable with categories labelled 0 and 1. For $c > 2$, c

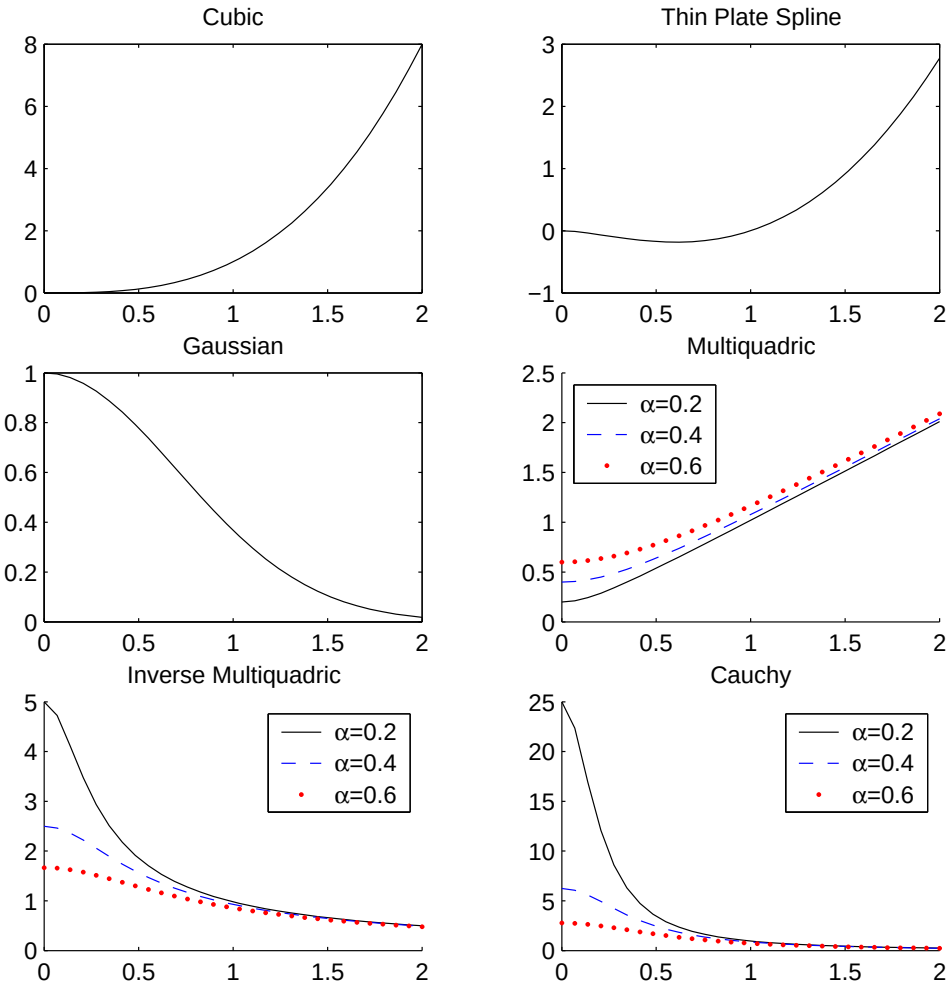


Figure 4: The forms of six radial basis functions; cubic, thin plate spline, Gaussian, multiquadric, inverse multiquadric and Cauchy. In each plot, the x–axis represents radial distances and the y–axis gives the values of the RBF. The effect of varying the free parameter, α, is demonstrated in the plots of the multiquadrics and the Cauchy function.

binary dependent variables are required and a value 1 on the kth dependent variable signifies that a data record belongs to the kth category. Because of the nature of least squares optimization, failure to encode response variables in this manner is equivalent to introducing a bias into the solution.

More information on radial basis functions can be found in Light [17], Miccheli [19] and Powell [22].

The RBF model (4) may also be implemented as a support vector machine (SVM), e.g., see Christianini [5], in which case centres are chosen automatically during an optimization process that seeks to maximize the separation between two categories in the space of the RBF functions. Hence the SVM RBF model looks for differences between two categories whereas the RBF classification model computes a similarity measure between data and centroids of known groups; see the NPL report [3] for further detail on SVMs for classification.

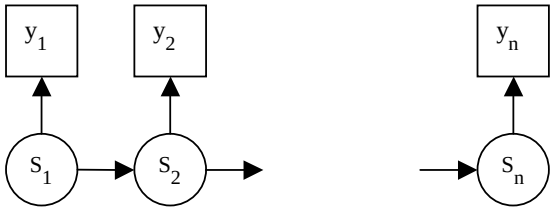
1.8 Hidden Markov Models

Let X be a set of observed independent variables, and let $Y = \{y_i : i = 1, \ldots, n\}$ be the set of observed values we are interested in modelling. If we are unable to calculate $P(Y|X)$, but are able to calculate $P(Y|X, S)$ where S is an unobserved variable, and S can be expressed in terms of a Markov chain, then we can use a hidden Markov model (HMM) to model Y. A simple example of a HMM relates to throwing two dice, one taking values 1 to 6, the other taking values 1, 3 and 5. The observed data would be the results from the die tosses (i.e., a value between 1 and 6) and the unobserved state relates to the particular die used for each toss. Hidden Markov models are widely used in the fields of speech recognition (where the observed data may be phonetic sounds, and the unobserved values the start of words, phrases or sentences) and, more recently, bioinformatics.

The basic concept underlying a HMM is that of a Markov chain. A kth order, discrete state, Markov chain is a stochastic process, $\{S_1, S_2, \ldots S_n\}$ with the property that

$$P(S_t|S_1, S_2, \ldots, S_{t-1}) = P(S_t|S_{t-k}, S_{t-k+1}, \ldots, S_{t-1}),$$

i.e., the probability of S_t given knowledge of all the previous states is the same as that given only knowledge of the previous k states. For a first order Markov chain, i.e., $k = 1$, only the previous state has a bearing on the current state. For a first order Markov chain, given the current value the past and present are independent. The probabilities $P(S_t|S_{t-1})$ are known as the transition probabilities. A hidden Markov model can be represented as:



where the $Y = \{y_i : i = 1, \ldots, n\}$ represent the n observed data, the $S = \{S_i : i = 1, \ldots, n\}$ represent the unobserved (or hidden) states of a Markov chain and the arrows indicate relationships, so the probability of y_i depends on S_i, but given S_i is independent of y_{i-1}. A HMM is therefore suitable for situations where, if we had additional information about some unobserved or unobservable variable, we would know the form of our observed data. One such example is given in the case study detailed in Section 2.2.

In order to fit a HMM, we require a model for the observed data Y, given the unobserved values, S, that is $P(Y|S,\Theta)$, where Θ are a set of model parameters. We also need the transition probabilities for the Markov chain defined by S, that is $P(S_i|S_1,\ldots,S_{i-1},\rho)$ where ρ is a set of transition parameters. In the case of a first order Markov chain this simplifies to $P(S_i|S_{i-1},\rho)$.

Once a HMM model has been formulated there are several options on how to proceed. The two most common methods are using a full Bayesian analysis or using a dynamic programming algorithm.

A full Bayesian analysis requires calculations involving the posterior probability, $P(\Theta,\rho,S|Y)$. This gives the joint probability of all possible states of the Markov chain and all possible values for the model and transition probabilities, given the observed data. Using Bayes theorem [1] we have that

$$\begin{aligned} P(\Theta,\rho,S|Y) &= \frac{P(Y|\Theta,\rho,S)P(\Theta,\rho,S)}{P(Y)} \\ &= \frac{P(Y|\Theta,S)P(S|\rho)P(\Theta)P(\rho)}{P(Y)} \end{aligned}$$

The values $P(Y|\Theta,S)$, $P(S|\rho)$ and $P(\Theta)$ are known, as they are specified when defining the model. The remaining term $P(Y)$ is specified by the requirement that the posterior distribution has unit volume and can be expressed as:

$$P(Y) = \int_\Theta \int_\rho \sum_S P(Y|\Theta,S)P(S|\rho)P(\Theta)P(\rho)$$

where $\int_\Theta$ and $\int_\rho$ denote the integral over all possible values of Θ and ρ respectively, and $\sum_S$ denotes the sum over all possible configurations of the Markov chain S. For simple problems with a small number of possible values for Θ, ρ and S, the constant of integration can be calculated directly. For more complex models alternative approaches exist, for example, Markov chain Monte Carlo (MCMC) techniques like Gibbs sampling [4] or the Hastings-Metropolis algorithm [12]. The description of these techniques are beyond the scope of this report, but all of them are computationally intensive and, potentially, suffer from issues of convergence. A detailed description of MCMC techniques, along with practical examples and solutions to common types of problems, can be found in Gilks et al. [10].

Given a fully specified HMM, where all the parameters, Θ and ρ are known, an alternative to performing a full Bayesian analysis is to use a dynamic programming algorithm called the Viterbi algorithm [25], which proceeds as follows:

1. Set $t = 1$.
2. For all i, calculate:

$$\delta_t(i) = \max_j \{\delta_{t-1}(j) P(S_t = i | S_{t-1} = j) P(y_t | S_t = i)\} \qquad (5)$$

3. Set $\phi_t(i) = k$, where k is the value of j in (5) which resulted in the maximum value.
4. Increment t, if $t \leq n$ repeat steps (2) and (3).

where $P(S_1 = i | S_0 = j)$ is defined to be $P(S_1 = i)$ and $\delta_0(j)$ is defined to be one, for all j. One common way to obtain $P(S_1 = i)$ is to assume the Markov chain underlying the HMM is stationary, that is $P(S_t = i) = P(S_{t+k} = i)$ for all t, k and i, and to solve the following series of equations:

$$\begin{aligned}
\sum_i P(S_1 = i) &= 1 \\
\sum_i P(S_2 = 1 | S_1 = i) P(S_1 = i) &= P(S_1 = 1) \\
\sum_i P(S_2 = 2 | S_1 = i) P(S_1 = i) &= P(S_1 = 2) \\
&\vdots \\
\sum_i P(S_2 = m | S_1 = i) P(S_1 = i) &= P(S_1 = m)
\end{aligned}$$

where m is the number of possible states.

Letting p_n equal the value of i for which $\delta_n(i)$ is maximized, and defining $p_t = \phi_{t+1}(p_{t+1})$, for $t = n-1, \ldots, 1$, the series $\{p_1, \ldots, p_n\}$ defines the most likely state for the underlying hidden Markov chain.

The Viterbi algorithm has several advantages over a full Bayesian analysis. It is computationally efficient and is guaranteed to converge to the most likely solution in a single iteration. On the other hand, whereas a full Bayesian analysis gives the probability of all possible solutions to a problem, the major drawback with using the Viterbi algorithm is that only the most likely solution is obtained. In addition the Viterbi algorithm gives no indication on how likely this solution is compared to any other possible solutions. It is possible to extend the Viterbi algorithm to obtain the k most likely solutions at the cost of some computational efficiency. The number of solutions required, k, must be specified a priori.

In order to implement the Viterbi algorithm, the HMM needs to be fully specified, i.e., we need to know that values of all the model parameters Θ

and all the transition probabilities ρ. This is not always possible, in some cases we need to estimate some or all of these parameters from the observed data. The Baum-Welch algorithm [23], also called the Forward-Backward algorithm, allows us to do this. The Baum-Welch algorithm proceeds as follows:

1. Obtain an initial state for the HMM, $S^{(0)}$ and set i to 0. The initial state can either be set randomly or through the use of prior information.
2. Given $S^{(i)}$, estimate values for Θ and ρ, label the estimates $\Theta^{(i)}$ and $\rho^{(i)}$ respectively. The estimates for the transition probabilities, ρ are often obtained by counting, i.e., an estimate of $P(S_j = 0|S_{j-1} = 1)$ is the number of times a one is followed by a zero, divided by n, the number of steps in the Markov chain. The estimation of the model parameters, Θ, depend on the form of the model fitted.
3. Increment i by 1.
4. Use the Viterbi algorithm to obtain the most likely set of states, $S^{(i)}$, given $\Theta^{(i-1)}$ and $\rho^{(i-1)}$.
5. Calculate $L^{(i)}$ the log likelihood, $\log(P(Y|\Theta^{(i-1)}, S^{(i)}))$.
6. If $i = 1$ or $L^{(i)} - L^{(i-1} > \epsilon$ repeat steps (2) to (5).

A good introduction to hidden Markov models, the Viterbi algorithm (and its extensions), the Baum-Welch algorithm and their implementation is given in Durbin et al. [7]. The case study presented in Section 2.2 uses a hidden Markov model, implemented via Baum-Welch algorithm, to identify measurements that are likely to have been contaminated by dust particles.

2 Case Studies

The first case study describes the analysis of two data sets comprising of spectra measured using circular dichroism spectroscopy. The second case study identifies outliers due to contamination by dust particles in data recorded by a co-ordinate measuring machine (CMM) experiment.

2.1 Circular Dichroism Spectroscopy

Spectra were measured at far– and near–ultraviolet wavelengths for samples of the same substance, possibly either diluted, concentrated or heated. Data from each wavelength region was analysed using statistical classification tools to verify or improve on the classification by inspection of spectra.

Each analysis was carried out in two stages. Firstly, a principal components analysis was used to reduce the dimensionality of the highly correlated spectra data, whilst retaining much of the original information. Secondly, an agglomerative hierarchical clustering of the transformed data was used to define groupings of spectra.

The results of the analysis show that the combination of principal components analysis and clustering is a powerful tool in the analysis of circular dichroism data. Such tools would enable classifications of spectra in the far–ultraviolet region to be standardized, thereby providing consistency. In the near–ultraviolet region further investigations are required to the describe the influence of experimental factors in the classifications.

2.1.1 Experimental Detail

The data consisted of readings from circular dichroism (CD) spectroscopy experiments using a spectro-polarimeter. Circular dichroism spectroscopy is a technique widely used in the study of proteins and other biomolecules (Rodger [24]). The spectro-polarimeter measures the difference in absorption of left– and right–circularly polarised light at a series of wavelengths. Measurements were made by using a Jasco J-720 spectro-polarimeter on dilute solutions of a protein known as CRM197 (Giannini [9]), a commonly used carrier protein in vaccine production. CRM197 has been the subject of many biophysical studies, including those using circular dichroism spectroscopy (Ho [13]). The protein was studied at a concentration of approximately 1 mg per ml, diluted into a sodium phosphate buffer. CD baseline measurements were made using sodium phosphate buffer and were subtracted from all spectra before further analysis.

2.1.2 Data

Data were available in both the far and near–ultraviolet frequency domain. The far–ultraviolet data contained 36 circular dichroism spectra measured at 151 equally spaced wavelengths in the interval $[185\text{nm}, 260\text{nm}]$. The near–ultraviolet data contained 26 circular dichroism spectra measured at 141 equally spaced wavelengths in the interval $[250\text{nm}, 320\text{nm}]$.

Of the 36 far–ultraviolet spectra: 17 were from a repeatability assay, i.e., batches of spectra recorded consecutively from the same sample (the sample cell was emptied and re-filled between batches); one was from a concentrated sample; two were from diluted samples; and the remaining 16 were heated to a temperature of either 20°C, 25°C, 30°C, 35°C, 40°C or 45°C. In order to improve the ratio of signal to noise in the data, each spectrum in the data is an average of the measurements of four scans across the reported wavelength range.

Of the 26 near–ultraviolet spectra: 21 were from a repeatability assay; 3 were incubated overnight at 37°C; and 2 were recorded at different dilutions. Each spectrum in the data is an average of a number (16, with the exception of 12 for datum 030227b1) of scans across the reported wavelength range.

Although some information on the spectra was known a priori, all labels were removed from the spectra prior to analysis and only replaced when the final groupings had been determined. Therefore the analysis made no assumptions about the number or contents of any potential groupings.

Plots of the far– and near–ultraviolet spectra can be seen in Figures 5 and 6 respectively. As can be seen by comparing these two figures, the general profile of the far–ultraviolet spectra is relatively simple compared to that for the near–ultraviolet spectra.

2.1.3 Analytical Method

Due to the large number of frequencies at which the spectra were measured compared to the number of spectra (151 frequencies and 36 spectra in the far–ultraviolet range; 141 frequencies for 26 spectra in the near–ultraviolet range), the first step in each analysis was to reduce the dimensionality of the problem. A principal components analysis was applied to each data set, with the aim of reducing the number of variables from over 140 to less than 20 and yet retaining most of the information about the shape of the spectra. Each PCA was carried out using the variance-covariance matrix associated with the spectra. A plot of the first two principal components was produced.

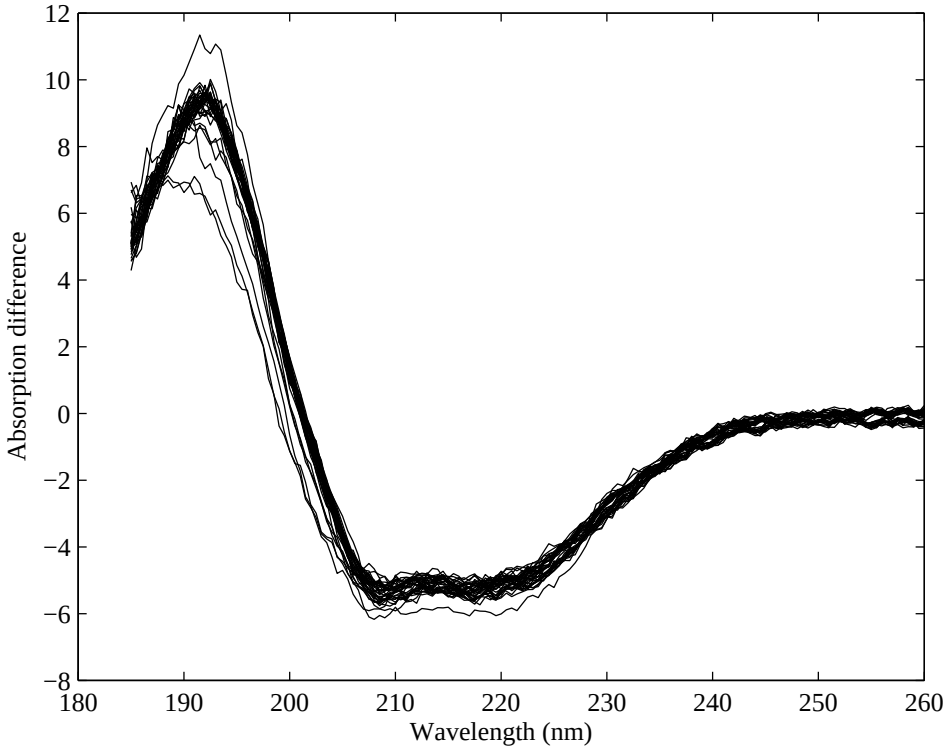


Figure 5: A plot of the 36 spectra available in the far-ultraviolet region.

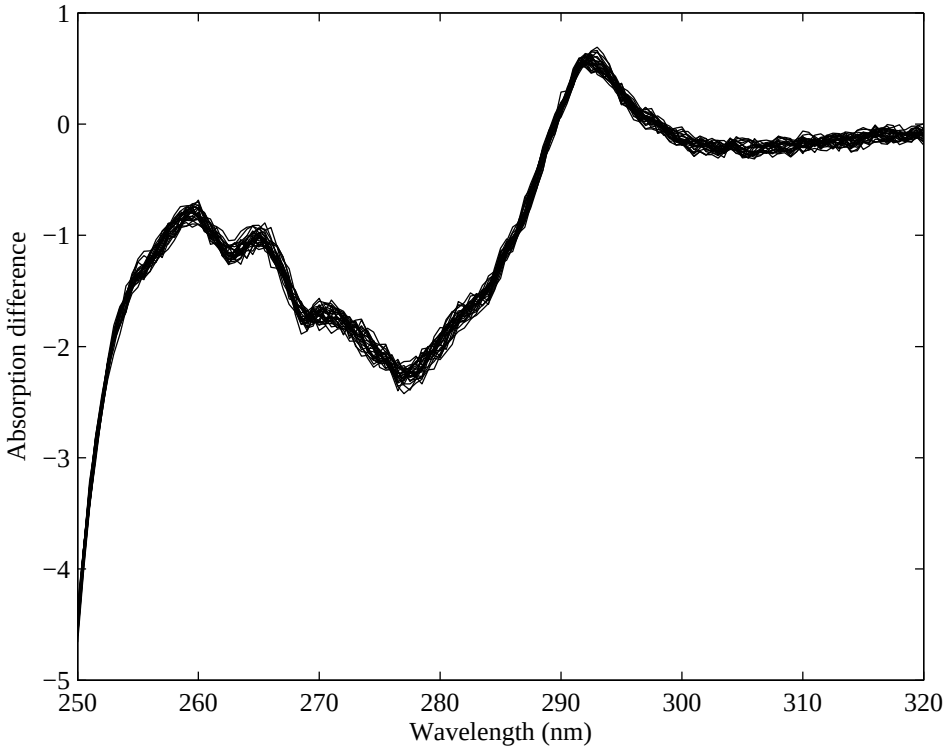


Figure 6: A plot of the 26 spectra available in the near-ultraviolet region. This plot does not show as much separation as that seen in the far–ultraviolet spectra given in Figure 5.

The number of principal components, k, used in the subsequent analysis was determined through the use of scree plots and by examining the amount of variation in the spectra explained by the first k principal components. As a final test of accuracy of the dimensionality reduction obtained by the PCA the original spectra were reconstructed from the first k principal components. After reconstruction, residuals were computed for each spectrum by subtracting the reconstructed spectrum from the original spectrum. The root mean sum of squares (RMS) of these residuals was then used as an indication of the accuracy of the reconstruction. Overlay plots of the reconstructed and original spectra were produced.

The second stage of the analysis consisted of using the k principal components identified in stage one in a hierarchical cluster analysis. The primary cluster analysis was performed using an Euclidean distance metric, with a standardization based on the standard deviation. The group average method was used to combine clusters. In order to ascertain the sensitivity of the results to a particular distance metric or clustering method, a number of other distance metrics and clustering methods were also tried. The results from only the primary cluster analysis for each data set are given in Section 2.1.4, and these are presented in the form of dendrograms.

Further details on principal components analysis, hierarchical cluster analysis and dendrograms can be found in Sections 1.1, 1.4 and 1.3, respectively, of this report.

2.1.4 Results

Far–Ultraviolet Region

The far–ultraviolet spectra, as shown in Figure 5, show a relatively simple profile. By visual examination of the troughs and peaks the spectra appear to fall into at least four groups.

Figure 7(a) shows a plot of the first two principal components obtained from the PCA outlined in Section 2.1.3. This simple graphical method for identifying clusters of observations clearly shows at least six distinct groups. Figure 7(b) shows the same plot, this time with the points colour coded, with the colours corresponding to the clusters defined in Table 1. It is not surprising that, in this case, the first two principal components allow the data to be classified so well, as they account for 86% of the variation in the spectra.

Figure 8 gives a scree plot for the 37 highest eigenvalues from the PCA. The characteristic "elbow bend" of a scree plot can clearly be seen, indicating that the first four principal components may be sufficient to summarise the

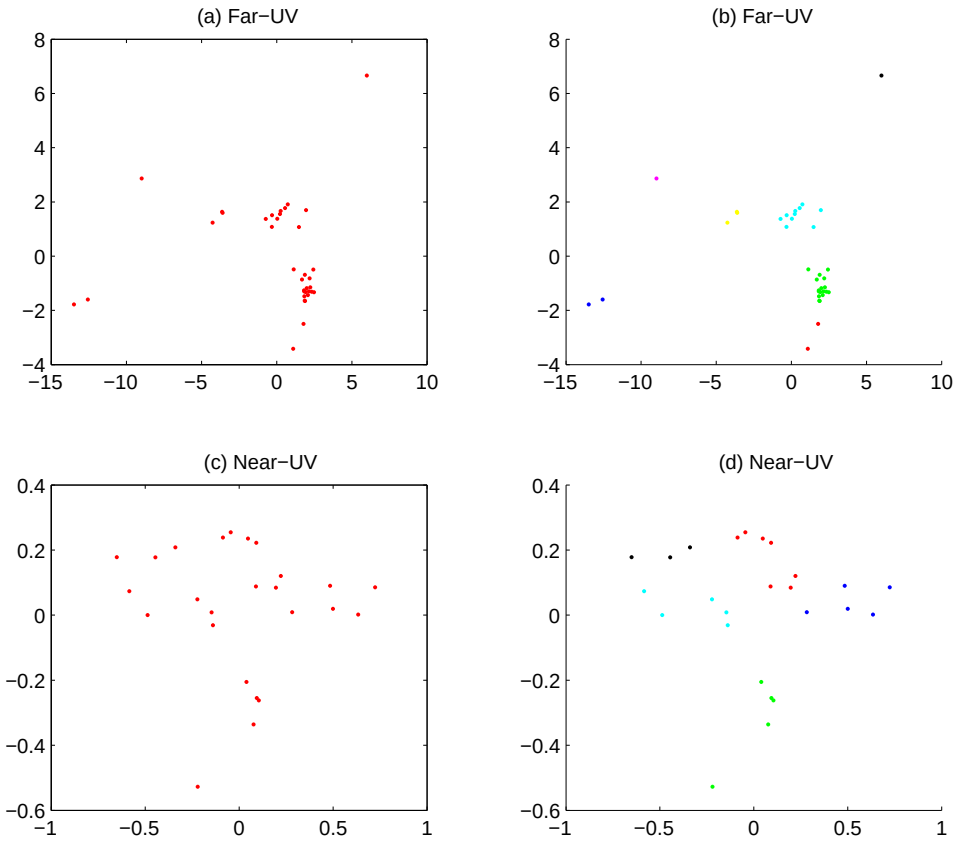


Figure 7: Plots of the first two principal components for the far– and near–ultraviolet spectra. The colour coding in the plots (b) and (d) correspond to the groupings described in Tables 1 and 2 respectively.

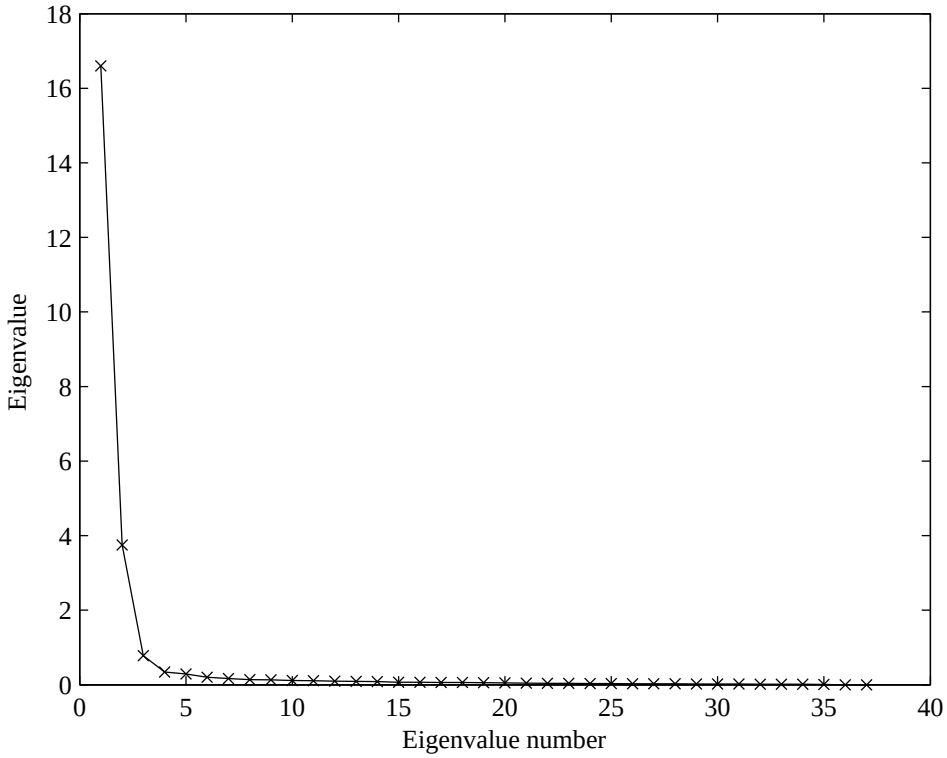


Figure 8: Scree plot of the 37 highest eigenvalues from a principal components analysis of the far–ultraviolet spectra.

data. This is confirmed by the fact that the first four principal components account for 92% of the total variation in the data. As a final check, the 36 spectra were reconstructed using only the first four principal components and the root mean square between the reconstructed spectra and the original spectra calculated. The RMS gives an indication of the accuracy of reconstruction of each spectra, with a value of zero indicating a perfect match. Figure 9 shows a plot of the original spectrum and its reconstruction for the observation with the highest RMS (0.15), and hence the worst reconstruction. The best reconstruction had an RMS of 0.089, with the mean RMS being 0.118 (standard deviation of 0.06). It can be seen from Figure 9, that even in the worse case, a reconstruction based on the first four principal components is a fairly close match to the original spectrum. Figure 10 shows the dendrogram obtained from performing a hierarchical clustering analysis on the first four principal components of the far–ultraviolet spectra. The dendrogram shows well-defined groups, i.e., few clusters are merged after a relatively low distance. Choosing a cut-off of 2.2 results in the seven groups described in Table 1.

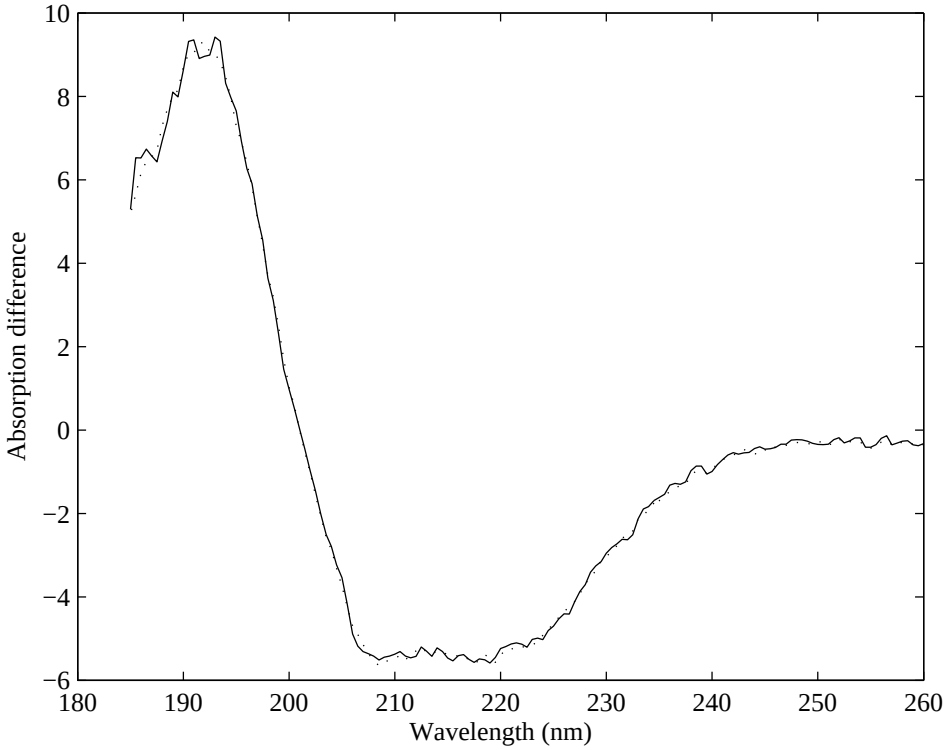


Figure 9: A plot of the least accurate spectrum reconstruction as measured by RMS. The dotted line is the reconstruction of the solid line spectrum.

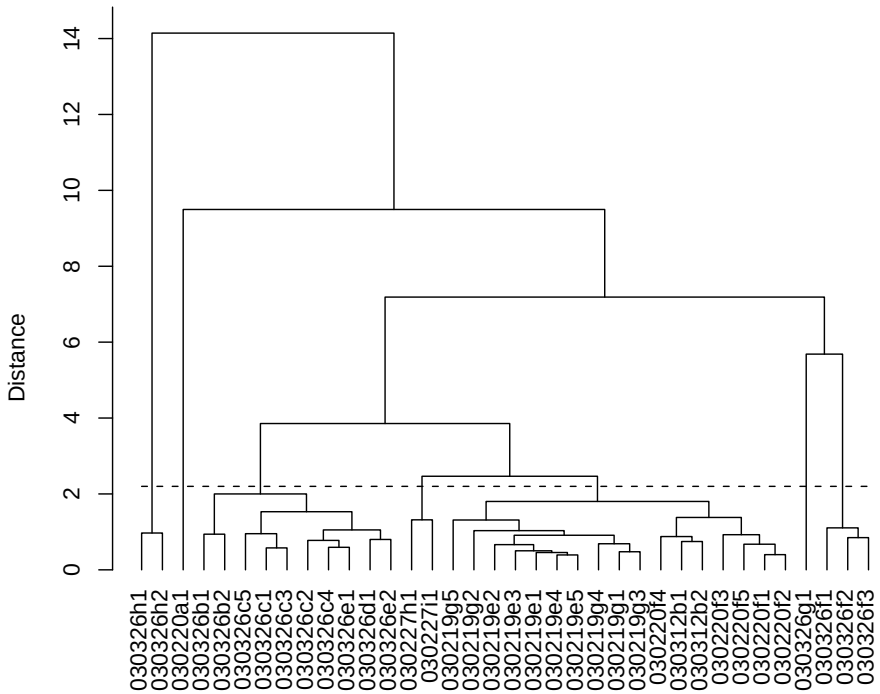


Figure 10: Clustering of far–ultraviolet circular dichroism data. The dotted line indicates the cut-off used to define the clusters reported in Table 1.

No.	Cluster members	Comments
1	030326h1, 030326h2	55°C
2	030220a1	Concentrated sample
3	030326b1, 030326b2, 030326c3, 030326c1, 030326c5, 030326d1, 030326e2, 030326c2, 030326c4, 030326e1	20°C, 25°C, 30°C
4	030227h1, 030227i1	Diluted samples
5	030220f4, 030312b2, 030312b1, 030220f1, 030220f2, 030220f3, 030220f5, 030219g5, 030219e2, 030219e4, 030219e1, 030219e3, 030219e5, 030219g2, 030219g1, 030219g3, 030219g4	Repeatability assay
6	030326g1	45°C
7	030326f3, 030326f1, 030326f2	40°C

Table 1: Cluster details for the seven groupings used to describe the far–ultraviolet data. These groupings have been constructed from Figure 10 using a distance cut-off of 2.2.

We can see in column three of Table 1 that clusters identified by the clustering analysis readily identify the different types of samples. Accordingly, it is not surprising that the spectra from the samples heated to 20°C, 25°C and 30°C appear in the same group.

The cut-off used in a cluster analysis to define the groups is problem specific and there are no hard and fast rules for its choice. As can be seen from Figure 10 choosing a slightly higher cut-off value of 2.4 would have led to the merging of clusters 4 and 5. However, it can be seen clearly that the six groups defined at this slightly higher cut-off are quite distinct, with the next merge occurring at a value close to four. Choosing a slightly lower cut-off, say, 1.8, would cause group 3 to be split into two groups, one containing spectrum 030326b1 and spectrum 030326b2 and the other group containing all other spectra. Smaller cut-off values would dramatically increase the number of groups being looked at, as the majority of the clustering occurs at these low levels. When interpreting the results from a cluster analysis it is usually informative to look at the clusters that would be formed if a different cut-off value had been chosen.

Use of different clustering methods and different distance metrics gave similar results to those shown in Figure 5, giving some confidence in the consistency of the results.

Near–Ultraviolet Region

The near–ultraviolet spectra, as shown in Figure 6, show a more complex profile than that seen in the far–ultraviolet spectra and, as such, there are no easily discernible groups.

Figure 7(c) shows a plot of the first two principal components obtained from the applying PCA outlined in Section 2.1.3 to the near–ultraviolet spectra. Although the five clusters are readily identifiable when colour coded, as in Figure 7(d), unlike in the far–ultraviolet case they are not as easy to identify in Figure 7(c). This is because the first two principal components in the near–ultraviolet analysis only account for 42% of the variation in the spectra.

Figure 11 gives a scree plot for the 25 highest eigenvalues from PCA. The scree plot indicates that the first six principal components may be sufficient to summarize the data. However, the first six principal components account for only 67% of the total variation in the data. In order to account for at least 90% of the variation, the first 17 principal components would be needed (which explain 93% of the variation). Given the relatively complex nature of spectra in the near–ultraviolet region, it is not surprising that a decision on the suitable number of principal components differs for the scree plot and the computation of explained variance. The 26 spectra were reconstructed using both the first six and the first 17 principal components. Little difference was observed between these reconstructions, and therefore the analysis was performed using only the first six principal components. Figure 12 plots the original spectrum and the worst reconstruction, based on six principal components. The RMS of this reconstruction was 0.465, compared with an RMS of 0.129 for the best reconstruction and an average RMS of 0.366 (standard deviation of 0.221).

Figure 13 shows the dendrogram obtained from performing a hierarchical clustering analysis on the principal components of the near–ultraviolet spectra. Due to the similarity of the spectra, Figure 13 shows clusters being formed and merged at values of distance relatively high compared to the distance when all groups merge (1.5). However, there are five easily identifiable clusters defined by a cut–off distance of 0.85. Table 2 describes these five groups.

We can see in column three of Table 2 that clusters identified by the clustering analysis do not readily identify the different conditions of the sample, i.e., the clustering distributes the 21 repeatability assays over all five groups. Further information on the spectra labels provided the following detail. Members of cluster number 1 were taken from readings on the same day as were those in cluster 3. However, members of clusters 2 and 4 were

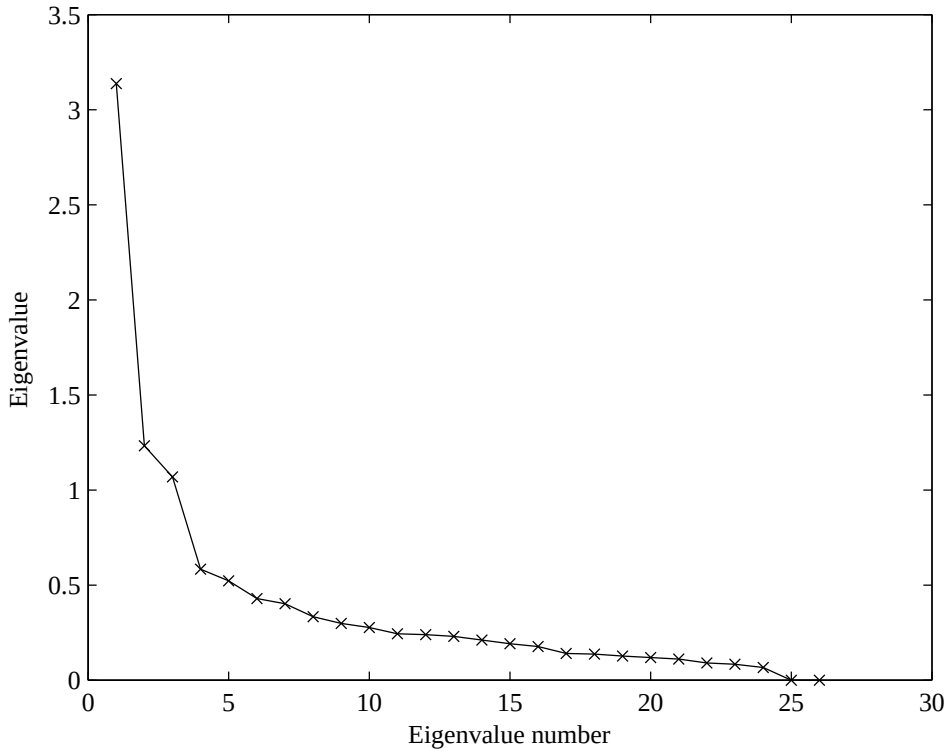


Figure 11: Scree plot of the 25 highest eigenvalues from a principal components analysis of the near–ultraviolet spectra.

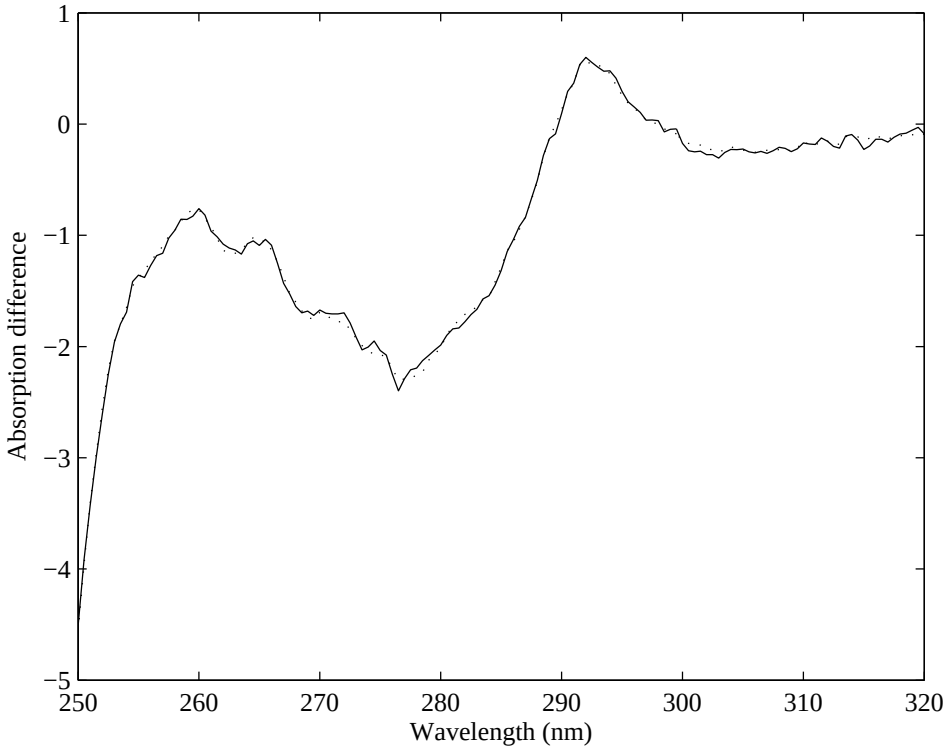


Figure 12: A plot of the least accurate spectrum reconstruction, based on six principal components, as measured by RMS. The dotted line is the reconstruction of the solid line spectrum.

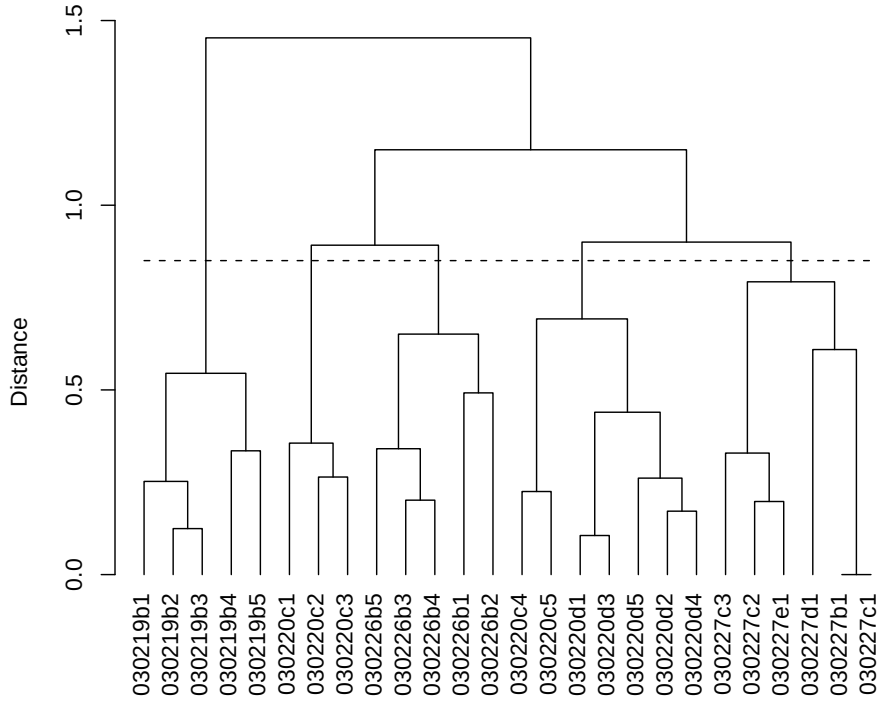


Figure 13: Clustering of near–ultraviolet circular dichroism data. The dotted line indicates the cut-off used to define the clusters reported in Table 2.

No.	Cluster members	Comments
1	030219b4, 030219b5, 030219b1, 030219b2, 030219b3	Repeatability assay
2	030220c2, 030220c1, 030220c3	Repeatability assay
3	030226b4, 030226b5, 030226b1, 030226b2, 030226b3	Repeatability assay
4	030220c4, 030220c5, 030220d4, 030220d5, 030220d2, 030220d1, 030220d3	Repeatability assay
5	1. 030227b1 2. 030227c1, 030227c2, 030227c3 3. 030227d1, 030227e1	1. Repeatability assay 2. Incubated at 37^oC overnight 3. Diluted by 1 per cent. and 2 per cent., respectively

Table 2: Cluster details for the five groupings defined in Figure 13 at a distance of 0.85.

taken on the same day suggesting that another factor is present. This factor may be the batch of data to which spectra belong. Such batches are defined by fills of the sample cell (a quartz glass container of fixed optical path-length into which the sample is placed). Between each batch of recordings the sample cell was emptied, dried and re-filled with a sample of the same substance.

Use of different clustering methods and different distance metrics gave similar results to those shown in Figure 6, once again giving some confidence in the consistency of the results.

2.1.5 Conclusions and Further Work

Principal components analysis coupled with hierarchical clustering provides a simple yet effective means to visualise the differences between spectra obtained by circular dichroism. Even when a simple classification method, like the visual inspection of a principal components plot, cannot identify the clusters present, the use of a hierarchical cluster analysis allows this identification to be automated.

The groupings found using hierarchical clustering for the far–ultraviolet region concur with those determined by the experimenter using overlay plots. Hence the potential to automate classification of samples from circular dichroism spectroscopy experiments is realized.

The groupings found using hierarchical clustering for the near–ultraviolet region have highlighted some experimental variances in the data as well as the effect of challenging the sample. This multi–level model or nested classification requires more investigation with new data that is collected in a structured, designed manner. In particular, enough data is required to adequately investigate a multi–level model of the kind: challenged/unchallenged samples within days within sample cell fills.

Given more data a measure of the uncertainty of groupings could be computed by using the ratio of within cluster sum of squares to between clusters sum of squares.

The suspected presence of nested groups in the data should also be investigated in the near–ultraviolet region. That is, we need to be sure that we are grouping together spectra based on challenged or unchallenged samples rather than the day and fill of the measurements.

Further work in this area would be to discriminate between substances of known classes. This task could be approached in one of two ways. Firstly, substances could be classified as one–against–all, i.e., examples of acceptable

spectra (for a given substance) against all others. Secondly, we could discriminate between n-classes of substances. This latter option would require more spectra data than currently is available. These discriminations could be based on the groupings of a cluster analysis similar to the one described in this case study.

2.2 Contamination from Dust Particles

In order to ensure that a surface has been manufactured to the required specification, measurements of the distance from a reference point to the surface are taken and then compared to theoretical values defined by the specification. These measurements are made at a number of locations across the surface. If the differences between the observed and expected values are large then the surface is deemed to be out of specification, and is rejected. Due to the low tolerances involved, the presence of dust particles can affect the results obtained and lead to false rejection of the surface.

This second case study describes the analysis of surface measurements made on a convex mirror. The purpose of the analysis was to identify those measurements contaminated by dust particles. The analysis was carried out using a two-state hidden Markov model (HMM), with the states corresponding to the presence or absence of a dust particle. The Forward-Backward algorithm (Section 1.8) was then used to ascertain the most likely set of states, given the observed data.

The most likely configuration of the hidden Markov model indicated dust particles at nine of the 401 observations.

2.2.1 Data

The data consisted of 401 measurements taken on the surface of a convex mirror. Each observation comprised of an (x, y) co-ordinate, giving the location at which the measurement was taken, and a residual. The residuals were calculated as the difference between the observed measurement and the expected measurement, defined by the specification. An unknown number of measurements had been contaminated by dust particles.

Although the data was collected from a two-dimensional plane, the measuring device used moved linearly, that is, it is possible to label the residuals, $r = \{r_i : i = 1, \ldots, 401\}$, by the order in which they are measured, so that r_1 corresponds to the first measurement taken, r_2 the second and so forth.

2.2.2 Analytical Method

There are two underlying mechanisms through which a dust particle can contaminate a particular measurement. The dust particle can be present on the surface of the mirror at that location or the dust particle could have been present at a previously measured location, and become attached to the measuring device. If a dust particle present at measurement point i sticks

to the measurement device it can move to point $i+1$, but cannot move to point $i+k$, without first passing through points $i+1$ to $i+k-1$. This is an example of the first order Markov property (see Section 1.8) as if S_i corresponds to the presence, $S_i = 1$ or absence, $S_i = 0$, of a dust particle at measurement i for $i = 1, \ldots, n$, then

$$P(S_i|S_1, \ldots, S_{i-1}) = P(S_i|S_{i-1})$$

As the presence or absence of a dust particle at measurement i is unknown a natural way to model this contamination mechanism this would be via a hidden Markov model.

In the absence of dust particles, it would be natural to assume that the residuals were Normally distributed, with mean zero and variance σ_0^2. It also seems reasonable to assume that the diameter of a dust particle, d, is Normally distributed with mean λ_1, and variance σ_d^2. Therefore:

$$P(r_i|S_i = j) \sim \mathrm{N}(\lambda_0, \sigma_1^2)$$

where $\lambda_0 = 0$ and $\sigma_1^2 = \sigma_d^2 + \sigma_0^2$. Modeling the residuals in this way we have made the assumption that the fluctuations in the mirror are small enough that several fluctuations could happen between two measurement points in close proximity to each other. This means that, in the absence of dust particles, the residuals are independent of each other.

In order to formulate transition probabilities for our HMM we make the assumption that dust particles are present on the surface of the mirror at random and denote the probability that a dust particle is present at any position as ρ_r. Letting ρ_s denote the probability that a dust particle sticks to the measurement device, then the four transition probabilities for the HMM become:

$$\begin{aligned}
P(S_i = 1|S_{i-1} = 0) &= \rho_r \\
P(S_i = 1|S_{i-1} = 1) &= 1 - (1-\rho_s)(1-\rho_r) \\
&= \rho_r + \rho_s - \rho_s\rho_r \\
P(S_i = 0|S_{i-1} = 0) &= 1 - P(S_i = 1|S_{i-1} = 0) \\
P(S_i = 0|S_{i-1} = 1) &= 1 - P(S_i = 1|S_{i-1} = 1)
\end{aligned}$$

The initial state of the hidden Markov chain, S, was randomly set, using a value of 0.05 for the probability of setting state S_i to 1 (dust particle present). This would lead (on average) to an initial state indicating the presence of 20.45 (0.05×409) dust particles. The value of λ_0 was forced to be equal to zero.

The Forward - Backward algorithm used to fit the HMM is an iterative maximization algorithm, and therefore the results may be dependent on the

Index	x co-ordinate	y co-ordinate	residual
187	59.7913	84.3371	3.4148
209	66.9727	4.9069	3.3071
210	66.9720	-0.3869	2.4736
211	66.9716	-5.6816	2.1037
212	74.1534	-5.6814	1.5217
213	74.1534	-0.3858	1.8059
214	74.1534	4.9096	1.9909
215	74.1534	10.2022	2.3955
304	102.8793	36.6796	4.4099

Table 3: The nine observations at which dust particles were predicted.

starting values used. That is, different results may occur, depending on the initial state attributed to the HMM. The analysis was repeated one hundred times with randomly chosen initializations. The initializations were chosen conditional on $P(S_i = 1) < P(S_i = 0)$, for all i, that is, the probability of a dust particle being present was less than the probability of it being absent.

2.2.3 Results

Figure 14 shows the data, r_i, against index, where the index indicates the order in which the measurements were taken. The second plot, Figure 15, shows the position of each measurement on a plane. The axes correspond to the (x, y) co-ordinates supplied with the data, and the numbers presented on the plot are the indices of the observations.

The hidden Markov model described in Section 2.2.2 indicated that dust particles were likely to have been present at nine of the 401 observations. These nine observations are detailed in Table 3. These nine observations correspond to the values shown in red in Figure 14 and Figure 15. As indicated by Figure 15 it appears likely that six of the nine observations identified as having been contaminated by dust particles (210 to 215) were due to a dust particle present at observation 209 sticking to the measuring device.

The average size of a dust particle, λ_1, was estimated to be 2.603, with a standard deviation of 0.964. The average size of the residual, λ_0, was set to zero in the analysis; its standard deviation was estimated to be 0.443. The probability of a dust particle being present at a particular location on the surface of the mirror, ρ_r was estimated to be 0.008 and that of a dust particle sticking to the measuring device, ρ_s, was estimated to be 0.667.

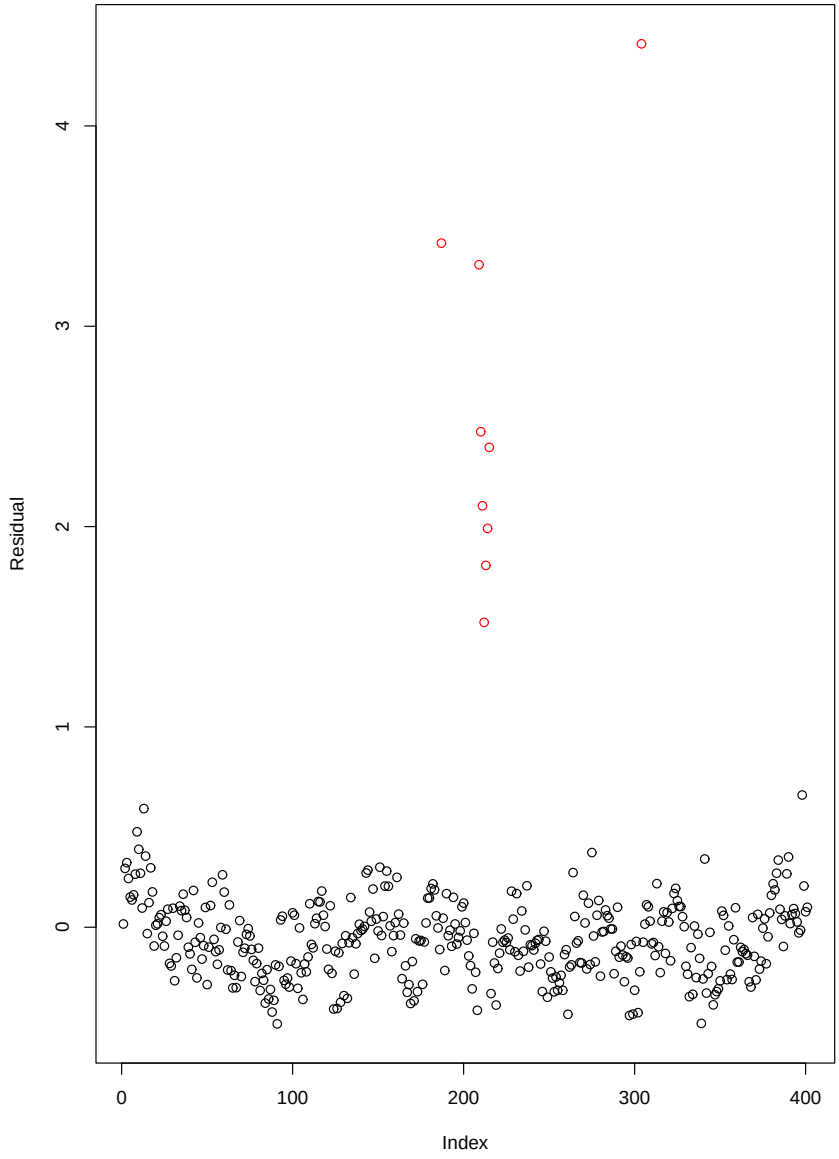


Figure 14: Data (r_i) vs index (i). Points marked in red correspond to the presence of a predicted dust particles.

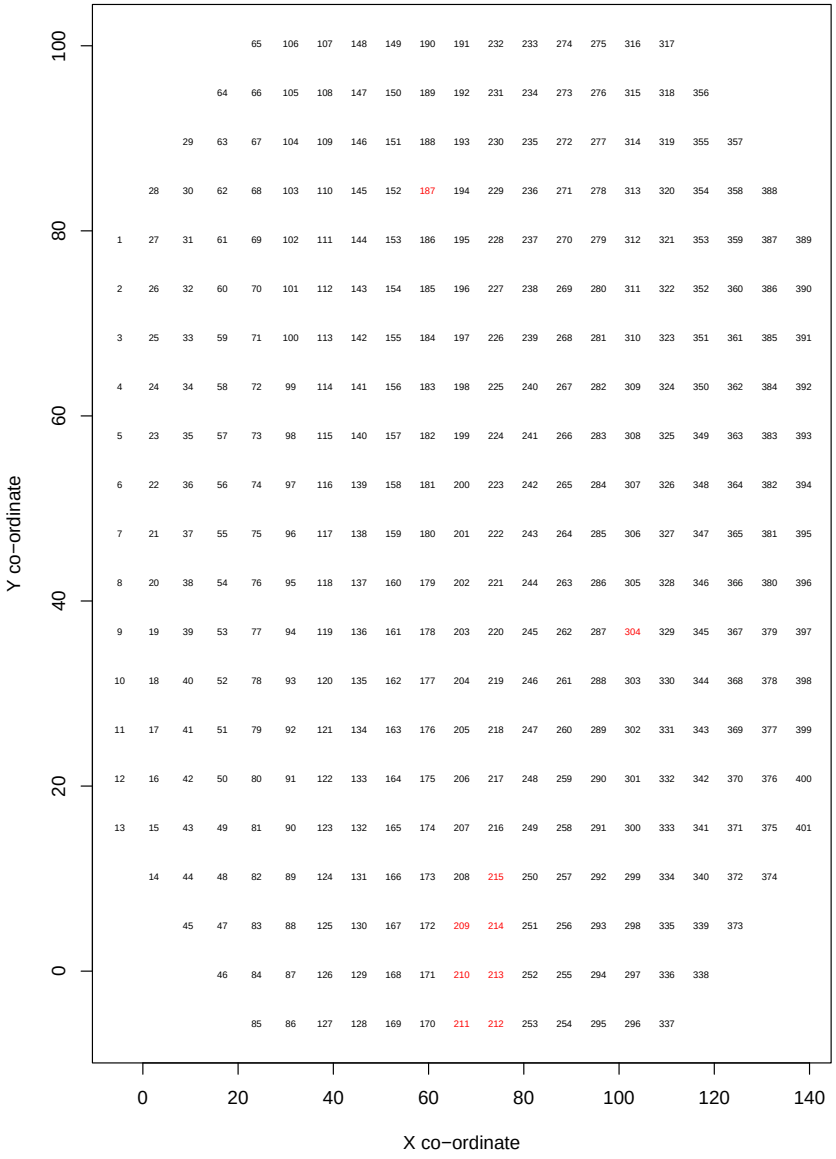


Figure 15: Scatter plot indicating the positions of each measurement. The figures represent the index numbers of the observations. The indices in red, (187, 209 to 215 and 304), correspond to the presence of a predicted dust particle.

As stated in Section 2.2.2 the algorithm used to fit the HMM is an iterative maximization algorithm, and therefore the results may be dependent on the starting values used. With 100 repeat analyses, two sets of results were observed. In 85% of the cases the results presented in Table 3 were observed. In the remaining 15 analyses all observations with the exception of the nine listed in Table 3 and observation 398, ($x = 138.7879$, $y = 31.3874$ and residual $= 0.66$), were estimated as having $S_i = 1$, i.e., for all observations except 398, the 0 and 1 flags used were reversed when compared to the original analysis. Because λ_0 was forced to be zero, this led to $\lambda_1 < 0$. This situation could have been avoided by incorporating additional prior information into our model, we know that the mean diameter of dust particle is a positive value and so could therefore have constrained $\lambda_1 \geq 0$. It should be noted that observation 398 had the next largest residual value after the nine observations listed in Table 3.

As stated in section 1.8 it is not possible to assign a probability (or degree of belief) to the classification obtained from this analysis. In order to calculate such a probability a full Bayesian analysis would need to be performed, with a probability being calculated for each of the possible 2^{401} possible classifications.

2.2.4 Conclusions and Further Work

As shown in Figure 14 the analysis indicates that the nine largest residuals are most likely to be due to the presence of dust particles, and this result is invariant to the starting conditions used to initialize the model.

A hidden Markov model is a natural choice for the analysis of problems of this type, where the form of the data is known (i.e., the residuals), conditional on an unobserved state (the presence or absence of dust particles). Although the data used for this case study was simple, with, as shown in Figure 14, the high residuals being easily identified by eye, the methodology can be applied to more complicated data. Although it was deemed unnecessary in this simple example, the methodology used can easily be extended to include additional information into the model, for example, prior knowledge about the size of dust particles, information about how often dust particles are likely to appear and how often they stick to the measuring device.

Acknowledgements. This work has been supported by the National Measurement System Directorate of the UK Department of Trade and Industry as part of its NMS Software Support for Metrology programme. The authors are grateful to Dr David Schiffmann, Centre for Optical and Analytical Metrology, NPL, for his help with the case study on circular dichroism spectroscopy and to Prof Alistair Forbes, Centre for Mathematics and Scientific

Computing, NPL, manager of the Fusion of Data project.

References

[1] T Bayes. An essay towards solving a problem in the doctrine of chances. *Philosophical Transactions of the Royal Society*, pages 330–418, 1763.

[2] J L Bentley. Multi-dimensional binary search trees used for associative searching. *Communications of the ACM*, 18:509–517, 1975.

[3] R Boudjemaa, A B Forbes, P M Harris, and S Langdell. Multivariate empirical models and their use in metrology. Technical Report CMSC 32/03, National Physical Laboratory, Teddington, December 2003.

[4] G Casella and E L George. Explaining the Gibbs sampler. *American Statistician*, 46:167–174, 1992.

[5] N Christianini and J Shaw-Taylor. *Introduction to support vector machines*. CUP, 2000.

[6] R O Duda and P E Hart. *Pattern Classification and Scene Analysis*. Wiley, 1972.

[7] R Durbin, S R Eddy, A Krogh, and G Mitchison. *Biological Sequence Analysis*. Cambridge University Press, 1998.

[8] B S Everitt. *Cluster Analysis*. Heinemann, 1974.

[9] G Giannini, R Rappuli, and G Ratti. The amino-acid sequence of two non-toxic mutants of diphtheria toxin: CRM45 and CRM197. *Nucleic Acids Research*, 12:4063–4069, 1984.

[10] W R Gilks, S Richardson, and D J Spiegelhalter. *Markov Chain Monte Carlo in Practice*. Chapman and Hall, 1996.

[11] T Hastie, R Tibshirani, and J Friedman. *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. Springer, 2003.

[12] W K Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57:97–109, 1970.

[13] M M Ho, X Lemercinier, B Bolgiano, D. Crane, and M J Corbel. Solution stability studies of the subunit components of meningococcal C oligosaccharide-CRM_{197} conjugate vaccines. *Biotechnology and Applied Biochemistry*, 33:91–98, 2001.

[14] Ross Ihaka and Robert Gentleman. R: A language for data analysis and graphics. *Journal of Computational and Graphical Statistics*, 5(3):299–314, 1996.

[15] Richard A Johnson and Dean W Wichen. *Applied Multivariate Statistical Analysis (fourth edition)*. Prentice and Hall, 1999.

[16] W J Krzanowski. *Principles of Multivariate Analysis*. Oxford University Press, 1990.

[17] W A Light. Some aspects of radial basis function approximation. *Approximation Theory, Spline Functions and Applications*, 356:163–190, 1992.

[18] P McCullagh and J A Nelder. *Generalized Linear Models (Second Edition)*. Chapman and Hall, 1989.

[19] C A Miccheli. Interpolation of scattered data: Distance matrices and conditionally positive data. *Constructive Approximation*, 2:11–22, 1986.

[20] NAG. *Fortran Library: Mark 20*. Numerical Algorithms Group Ltd, Oxford, 2001.
http://www.nag.co.uk/numeric/FL/FLdocumentation.asp.

[21] NAG. *DMC*^{2}. Numerical Algorithms Group Ltd, Oxford, 2003. http://www.nag.co.uk/numeric/DR/DRuserguide.asp.

[22] M J D Powell. Radial basis functions for multivariable interpolation: A review. *In: Algorithms for Approximation*, 1985. Editors: Cox, M G & Mason, J C Clarendon Press, Oxford.

[23] L R Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77:257–285, 1989.

[24] A Rodger and M A Ismail. Introduction to circular dichroism. *In: Spectrophotometry and spectrofluorimetry: a practical approach*, 2000. Editors: Gore, M G, OUP.

[25] A J Viterbi. Error bounds for convolutional codes and an asymptotically optial decoding algorithm. *IEEE Trans. Information Theory*, IT–13:260–269, 1967.

[26] VSN. *GenStat for Windows, 6th Edition*. VSN Internationl Ltd, Hemel Hempstead, 2003. http://www.vsn-intl.com.