Statistical Techniques in Metabolic Profiling

Maria De Iorio, Timothy M. D. Ebbels, David A. Stephens[*]

1 Introduction

Metabolic profiling, also known as metabonomics (Nicholson et al., 1999) or metabolomics (Raamsdonk et al., 2001), is a rapidly developing field in biomedical science that combines the application of spectroscopic techniques with multivariate statistical analysis in studies of the molecular composition of biofluids, cells and tissues. The experimental focus is on profiling *metabolites* the thousands of low molecular weight molecules which are the building blocks of the cell. These compounds interact with macromolecules such as proteins and nucleic acids in the fundamental metabolic processes that keep organisms alive. Thus, the metabolic state of an organism, as delineated by its metabolic profile, can reveal information on the genetic, physiological or functional status the system. In addition, in contrast to DNA or RNA, metabolic profiles also reflect the entirety of internal and external influences on a organism or tissue, including environment, behaviours, disease, drugs and interactions with parasitic or symbiotic organisms (Nicholson et al., 2002; Nicholson and Wilson, 2003). This leads to important applications in the molecular diagnosis and prognosis of disease, drug metabolism and toxicity, functional genomics and the investigation of fundamental biological processes. From the statistical point of view, however, metabolic profiles are complex and information rich, presenting unique challenges to data analysis and modelling.

The term metabolome refers to the total metabolite complement of a cell, tissue or organism. Whilst there are currently no methods capable of delivering an exhaustive measurement of the metabolome, there are several technologies that can approach this ideal. A gold standard technique would be one which is a) non-selective, in that it is not biased toward particular chemical classes of compounds, b) quantitative and accurate, thus producing data which can be meaningfully modelled by statistical procedures, c) sensitive to a wide range of metabolite concentrations and d) able to resolve the complex mixture into separate signals from each metabolite. These requirements are satisfied, though not perfectly, by the techniques of nuclear magnetic resonance (NMR) spectroscopy and mass spectrometry (MS).

[*] Order of authors is alphabetical.

1.1 Spectroscopic Techniques

The two spectroscopic assay technologies, NMR and MS, are highly complementary and therefore often used in parallel. NMR has the advantage that it is highly non-selective and quantitative; it also requires little sample preparation, and is non-destructive thus permitting multiple measurements on each sample. MS approaches are much more sensitive and generally higher resolution than NMR, yet suffer from inaccuracies introduced by differential ionisation efficiencies. These can be partially alleviated by preceding the MS detection with a physical separation step using liquid or gas chromatography, leading to LC- or GC-MS (Wilson et al., 2005). Figure 1 shows examples of NMR and LC-MS spectroscopic metabolic profiles of urine.

Figure 1 about here

The resulting spectra (or profiles) consist of several thousands of individual measurements at different resonances or masses. The spectra are complex and require dedicated analysis procedures. The inherent high-dimensionality immediately precludes many conventional statistical analysis approaches. In common with other spectroscopic data, the spectral variables are usually highly collinear, a characteristic that arises for both chemical analytic and biological reasons. For example, one metabolite may generate more than one spectral signal (e.g. isotopic patterns in MS) leading to several highly correlated peaks. Alternatively, signals from different metabolites may show high correlations because they are involved in the same biological process. While the above properties are typical of data from many so-called "-omic" techniques such as those from transcriptomics and proteomics, metabolic profiling poses its own challenges. The most important challenge is the problem of unidentified signals; while thousands of signals may be detected, typically only a very small fraction will be identified with a known chemical structure - so such prior chemical information is extremely helpful for correct interpretation and identification. Another difficulty is the sensitivity and resolving power of the analytical technique, as well as the complexity of the analysed mixture. A related problem is that the total number of detected analytes is in general unknown and will vary between different biological conditions. Thus, in metabolic profiling, there is a great emphasis on methods of exploratory data analysis and visualisation, which allow the analyst to probe their data for the presence of poorly detected or resolved signals and help in identifying unknown metabolites.

1.2 Data Preprocessing

Metabolic profiles generally undergo some preprocessing prior to more advanced statistical modelling, to account for spectral artefacts, and render profiles from different samples more comparable. The preprocessing is necessary to achieve comparability between replicate samples, to achieve some form of measurement noise reduction, and to more accurately distinguish the different chemical constituents by means of peak deconvolution and separation.

In this chapter, we do not describe these issues in detail, but instead focus statistical feature extraction and discrimination procedures. However, a brief discussion is given below.

One of the most serious problems affecting NMR profiles is that peak positions can be affected by the pH or ionic strength of the sample. This phenomenon seriously affects a minority of metabolites, and means that the spectral intensity at a given NMR chemical shift (or wavelength) may be due to both changes in molecular concentrations as well as shifts in resonance positions, making the interpretation of statistical models difficult. While there exist techniques that account for or correct such shifts automatically (Holmes et al., 1992; Stoyanova et al., 2004), these are most suited to correction of small spectral regions, and have not yet been successfully applied to full-width NMR profiles. A similar difficulty occurs in liquid chromatography (LC) where obtaining highly reproducible separation of metabolites is difficult and peak alignment algorithms are often used to match LC profiles from different samples. In mass spectrometry, various processes can lead to the signal from each metabolite being broken up into multiple peaks. For example, the natural occurrence of multiple isotopes of various nuclei leads to a characteristic isotopic pattern for each primary ion, while the formation of adducts (e.g. the addition of metal ions to the primary ion) ensures each metabolite is characterised by a pattern of different peaks. In MS approaches, differential ionisation efficiency also complicates interpretation of the data. Finally, regardless of the technique employed, the profiles will be affected by gross changes in sample concentration which can obscure important but subtle treatment related changes. To alleviate this, profiles are often normalised to unit total intensity, so that subsequent values for any given metabolite relate to its concentration relative to the rest of the mixture, rather than an absolute measure. In most cases, this does not detract from, and can even improve, the biological interpretation, though when there are large changes between profiles, interpretation of profiles normalised in this way is less clear.

Several metabolic profile analysis software packages are available. One popular and effective is the AMDIS software (Davies, 1998; Stein, 1999), which extracts individual component spectra from LC or gas chromatography (GC) MS data files and then uses them to identify compounds by matching the spectra to a reference library of chemical compounds housed at the National Institute for Standards and Technology (NIST). The analysis proceeds in four steps (noise analysis, component ”perception” (extraction), spectral deconvolution and compound identification) in an automatic fashion using a variety of multivariate modelling techniques and heuristic testing procedures. The entire approach has proved to be very successful for analyzing data from these platforms.

The aims of data analysis in metabolic profiling will depend on the scientific objectives of the study. However, the objectives of analysis typically fall into one or more of the following categories. Firstly, we wish to visualize the relationships between groups of both samples and spectral variables. For example, this could include clustering individuals, or detecting significant correlations between variables. In addition, we wish to determine whether there is a significant difference between groups related to the effect of interest. The latter is a classic small n, large p inference problem (West, 2002), that is, each individual datum (metabolic profile) consists of a large vector of inter-related (dependent) observations, yet the number of samples in the study is relatively small. Finally, and perhaps most importantly, we are interested in finding out which metabolites are responsible for these changes. This chapter introduces some of the current techniques used in the statistical analysis of metabolic profiles. We give the basic principles and algorithms behind each method and illustrate their use with example data sets. We will denote with $\boldsymbol{X}$ the $n \times p$ matrix of n metabolic profiles, where p is the dimension of the spectrum (usually p is in the order of thousands). Therefore, each row of $\boldsymbol{X}$ matrix represents a metabolic profile. We will define the sample covariance matrix $\boldsymbol{S} = \boldsymbol{X}'\boldsymbol{X}/n$, assuming that each column of $\boldsymbol{X}$ is standardised to have zero mean, and take $\boldsymbol{y}$ to be a n-dimensional vector of responses.

1.3 Example data

Several of the methods described in this chapter are illustrated with the help of data from a study profiling the metabolic consequences of Hydrazine toxicity (Lindon et al., 2005) (figures 2, 3, 4, 8, 10). Thirty Sprague-Dawley rats were randomly and equally assigned to three treatment groups (control, low dose and high dose) and hydrazine was administered orally at doses of 0, 30 and 90 mg/kg respectively. Urine samples were collected at 10 time points over 8 days, including 2 pre-treatment samples. All procedures were carried out in accordance with relevant national legislation and were subject to appropriate local review. ^{1}H NMR spectra were measured at 600 MHz and 300K using a robotic flow-injection system (Bruker Biospin, Karlsruhe, Germany). After phasing and baseline correction, each NMR spectrum was segmented into M=205 variables by integrating the signal in regions of equal width (0.04 ppm) in the chemical shift ranges δ 0.20-4.50 and δ 5.98-10.0, excluding spectral artefacts in the region δ 4.50-5.98. All segmented spectra were then normalized to a constant integrated intensity of 100 units to take account of large variations in overall urine concentration. The PCA and PLS analyses shown were performed using SIMCA-P+ version 11 (Umetrics AB, Umea, Sweden). Other analyses were computed using in house software written in the MATLAB programming environment (version R2006a, The MathWorks, Natick, MA).

2 Principal Components Analysis and Regression

2.1 Principal Components Analysis (PCA)

Principal component analysis (PCA) is a well known method of dimension reduction (Massy, 1965; Jolliffe, 1986) which seeks linear combinations of the columns of $\boldsymbol{X}$ with maximal variance, or equivalently high information. It is routinely applied in chemometrics with the goal of providing the most compact representation of the data. The original p variables $\boldsymbol{X} = [\boldsymbol{x}_1 \dots \boldsymbol{x}_p]$ are transformed in a new predictor set $\boldsymbol{T} = [\boldsymbol{t}_1 \dots \boldsymbol{t}_k]$, with $k \leq min(n-1, p)$. The new variables $\boldsymbol{t}_j$, called *scores*, are a weighted average of the original $\boldsymbol{X}$ variables. The *principal components* are the eigen-vectors, $\boldsymbol{u}_j$ from the eigen-decomposition of $\boldsymbol{X}'\boldsymbol{X}$ (and of the sample covariance matrix $\boldsymbol{S}$, up to a constant). PCA sequentially maximizes the variance of a linear combination of the original predictor variables

$$\boldsymbol{u}_j = \arg \max_{\|\boldsymbol{u}\|=1} \operatorname{Var}(\boldsymbol{X}\boldsymbol{u})$$

subject to the constraint that $\boldsymbol{u}_i'\boldsymbol{S}\boldsymbol{u}_j = 0$ for all $1 \leq i < j$. This ensures that $\boldsymbol{t}_j = \boldsymbol{X}\boldsymbol{u}_j$ is uncorrelated with all the previous linear combinations $\boldsymbol{t}_i = \boldsymbol{X}\boldsymbol{u}_i$. The principal components are ordered in terms of the amount of variation of the original data they account for. The first principal component direction has the property that $\boldsymbol{t}_1 = \boldsymbol{X}\boldsymbol{u}_1$ has the largest sample variance amongst all normalised linear combinations of the columns of $\boldsymbol{X}$. Each subsequent component gives combinations with as large as possible variance which are uncorrelated with those which have been taken earlier.

There are various standard approaches to find the principal components, for example taking the Singular Value decomposition of $\boldsymbol{X}$. In chemometrics it is common to estimate the principal components using the NIPALS (Nonlinear Iterative Partial Least Squares) algorithm (Wold, 1966). This is because the number of required components is usually much less than the total possible number ($k \ll p$).

NIPALS algorithm for PCA

```
0. Standardize each $\boldsymbol{x}_j$ to have mean zero.
1. Initialize $j = 1$, $\boldsymbol{X}_j = \boldsymbol{X}$ and $\boldsymbol{t}_j$ to any of the column of $\boldsymbol{X}$.
2. Project $\boldsymbol{X}_j$ on $\boldsymbol{t}_j$ to find the corresponding loadings $\boldsymbol{u}_j$: $\boldsymbol{u}_j = \boldsymbol{X}_j'\boldsymbol{t}_j/\|\boldsymbol{t}_j\|$.
3. Normalize $\boldsymbol{u}_j$ to have unit length: $\boldsymbol{u}_j = \boldsymbol{u}_j/\|\boldsymbol{u}_j\|$.
4. Project $\boldsymbol{X}_j$ on $\boldsymbol{u}_j$ to find the corresponding score vector $\boldsymbol{t}_j$: $\boldsymbol{t}_j = \boldsymbol{X}_j\boldsymbol{u}_j/\|\boldsymbol{u}_j\|$
5. Check convergence: compare $\boldsymbol{t}_j$ used in step 2 and $\boldsymbol{t}_j$ calculated in step 4
   (check if the difference is larger than a pre-defined threshold, e.g. $10^{-6}$).
   If they are the same, the iteration has converged and continue to step 6.
   Otherwise return to step 2.
6. Remove the estimated PCA principal component from $\boldsymbol{X}_j$: $\boldsymbol{X}_{j+1} = \boldsymbol{X}_j - \boldsymbol{t}_j\boldsymbol{u}_j'$.
7. Let $j = j + 1$ and repeat steps 1 to 7 until $j = k$.
```

If $k = min(n-1, p)$, then we have found all the principal components. We can now form the scores matrix $\boldsymbol{T}$ and the loadings matrix $\boldsymbol{U}$ with columns $\boldsymbol{t}_j$ and $\boldsymbol{u}_j$, respectively. These matrices are such that

$$\boldsymbol{X} = \boldsymbol{T}\boldsymbol{U}'$$

where $\boldsymbol{U}$ is an orthogonal matrix and the $\boldsymbol{t}_j$ are orthogonal. Note that $\lambda_j = \|\boldsymbol{t}_j\|^2$ and $\boldsymbol{u}_j$ are eigenvalues and eigenvectors of $\boldsymbol{X}'\boldsymbol{X}$, respectively. The NIPALS algorithm can be modified to account for missing data (Christoffersson, 1970).

A key issue is the number of principal components are necessary to describe a dataset in a parsimonious way but without loss of important information. After k runs of the NIPALS algorithm the $\boldsymbol{X}$ matrix is decomposed as

$$\boldsymbol{X} = \boldsymbol{t}_1\boldsymbol{u}_1' + \cdots + \boldsymbol{t}_k\boldsymbol{u}_k' + \boldsymbol{X}_{k+1}$$

We want to choose k such that $\boldsymbol{X}_{k+1}$ represents only noise (if $k = \min(n-1, p), \boldsymbol{X}_{k+1} = \boldsymbol{0}$) and all the features of $\boldsymbol{X}$ are captured by the $\boldsymbol{t}_j \boldsymbol{u}_j'$, $j \leq k$.

The maximum number K of principal components is determined by the number of non-zero eigenvalues, which coincides with the rank of $\boldsymbol{S}$ and $K \leq \min(n-1, p)$. However, usually k is chosen through cross-validation (Stone, 1974; Wold, 1978). The proportion of the total variance explained by a PCA model with k components, is quantified by the R^2 parameter defined as

$$R_k^2 = 1 - \sum_{i=1}^{n} \|\widehat{\underline{x}}_i - \underline{x}_i\|^2 / \mathrm{SS}$$

where $\underline{x}_i$ denotes a row of $\boldsymbol{X}$, $\widehat{\underline{x}}_i$ is the estimated $\underline{x}_i$from the PCA model and SS is the total sum of squares of $\boldsymbol{X}$, i.e. $\mathrm{SS} = \sum_{i=1}^{n} \|\underline{x}_i\|^2$.

R^2 varies in the range zero to one, taking its maximum value when $k = min(n-1, p)$. The cross-validation (CV) procedure quantifies how robust the model is to perturbation of the data and thus avoids over-fitting. In each round of CV, a proportion (usually around 10%) of the data is held out $\boldsymbol{X}_o$, the model computed with the remaining data $\boldsymbol{X}_{-o}$, and the predicted values of the held out data points computed using $\hat{\boldsymbol{X}}_o = \sum_{i=1}^{k} \boldsymbol{t}_i \boldsymbol{u}_i$. This is repeated until all data points have been held once. The parameter Q^2 is the CV equivalent of R^2 using the predicted values of the data:

$$Q^2 = 1 - \mathrm{PRESS}/\mathrm{SS}$$

where $\mathrm{PRESS} = \sum_{i \in O} \|\widehat{\underline{x}}_i - \underline{x}_i\|^2$, where O is the set of individuals left out . Q^2 is bounded above by the value of R^2 and a high ratio of Q^2/R^2 indicates a robust model which is little affected by perturbations. As more components are computed, both R^2 and Q^2 generally rise as more of the variance is explained. However, a point will be reached where the structure represented by the $(m+1)$th component is mostly noise. At this point, the predicted data will deviate from the true data and Q^2 will start to decrease, indicating that a $k = m$ is the correct number of components to use in the model.

Figures 2 and 3 illustrate the use of PCA on the Hydrazine dataset. Figure 2 shows the scores plot of the first two principal components. The plot shows the characteristic L shaped trajectory of Hydrozine toxicity; high dose profiles initially resemble controls but over the time course move to a well separated region of the plot. Figure 3 shows the PCA loadings of the first two principal components. Spectral bins which have a high variance in the $\boldsymbol{X}$ space are indicated by points lying far from the origin and can be used to interpret the distribution of data on the scores plot.

Figures 2 and 3 about here

2.2 Principal Components Regression (PCR)

In most applications we are interested in predicting $\boldsymbol{y}$ from the sets of inputs $\boldsymbol{x}_j$, e.g. we are interested in studying the relation between blood pressure and metabolic profile of an individual. From PCA we derive a representation of the data matrix $\boldsymbol{X}$ as score matrix $\boldsymbol{T}$. *Principal Component Regression* (PCR) uses the score vectors $\boldsymbol{t}_j$ as explanatory variables and regresses $\boldsymbol{y}$ on $\boldsymbol{t}_1, \ldots, \boldsymbol{t}_k$ for some $k \leq \min(n-1, p)$:

$$\boldsymbol{y} = \alpha + \sum_{i=1}^{k} \beta_i \boldsymbol{t}_i + \boldsymbol{\epsilon}$$

where α denotes the intercept, β_i are the regression coefficients and $\boldsymbol{\epsilon}$ is a Normal error term. As the $\boldsymbol{t}_j$ are orthogonal, PCR reduces to a sum of univariate regression (Hastie et al., 2001). Usually only the first principal components are used and the $p-k$ smallest eigenvalue components are discarded. PCR has major advantages over standard multivariate regression when $\boldsymbol{X}$ is singular and when $n < p$ as a dimension reduction tool. In the latter case, the eigenvalues $\lambda_n = \cdots = \lambda_p = 0$ and so at most $n-1$ components can be included.

3 Partial Least Squares and related methods

3.1 Partial Least Squares (PLS)

Partial Least Squares Regression (Wold, 1975) is extensively used in chemometrics as an alternative to Ordinary Least Square in ill-conditioned problems (e.g. $n < p$). It is most often used when the explanatory variables are highly collinear and when they outnumber the observations.

PCA (and therefore PCR) finds, in some way uncritically, those latent variables (scores) that describe as much as possible of the variation in $\boldsymbol{X}$. In PCR, the latent variables are calculated without consideration of the response and it is possible that useful predictive information for $\mathbf{y}$ is discarded as noise. If there is a lot of variation in $\boldsymbol{X}$ not correlated with the response then the latent variables found by PCR might not be adequate to describe $\mathbf{y}$. PLS tries to solve this problem by constructing latent variables that are *relevant* for describing $\mathbf{y}$. The goal of PLS is to construct linear combinations of the original variables that have symultaneously high variance and high correlation with the response (Frank and Friedman, 1993; Hastie et al., 2001):

$$\mathbf{u}_j = \arg\max_{\|\mathbf{u}\|=1} \mathrm{Corr}^2(\mathbf{y}, \boldsymbol{X}\mathbf{u}) \mathrm{Var}(\boldsymbol{X}\mathbf{u}) \tag{1}$$

subject to the constraint that $\boldsymbol{u}_i' \boldsymbol{S} \boldsymbol{u}_j = 0$ for all $1 \leq i < j$. Therefore the goal of PLS is to find optimal weights $\mathbf{u}_j$ to form a small number of latent variables that best predict the response $\mathbf{y}$.

PLS was introduced as a modification of the NIPALS algorithm for PCA (Wold, 1966) but it is most often presented as a latent factor regression method and produces a sequence of models $\mathbf{y}_j$. The algorithm to compute the first k latent variables is described below.

PLS algorithm

```
0. Standardize each $\boldsymbol{x}_j$ to have mean zero and variance 1.
   Standardize $\mathbf{y}$ to have zero mean.
1. Initialize $j = 1, \boldsymbol{X}_j = \boldsymbol{X}$ and $\mathbf{y}_j = \mathbf{y}$.
2. Compute the univariate regression coefficient of $\mathbf{y}$ on each of the $\boldsymbol{x}$: $\boldsymbol{w}_j = \boldsymbol{X}_j'\mathbf{y}_j$
3. Normalize $\boldsymbol{w}_j$ to have unit length: $\boldsymbol{w}_{\boldsymbol{j}} = \boldsymbol{w}_{\boldsymbol{j}}/\|\boldsymbol{w}_j\|$
4. Project $\boldsymbol{X}_j$ on $\boldsymbol{w}_j$ to find the corresponding score vector $\boldsymbol{t}_j$: $\boldsymbol{t}_j = \boldsymbol{X}'\boldsymbol{w}_j$
5. Regress $\mathbf{y}$ on $\boldsymbol{t}_j$ to get the the ordinary least square regression coefficient $\widehat{b}_j$: $\widehat{b}_j = \boldsymbol{t}_j'\mathbf{y}/\|\boldsymbol{t}_j\|$.
6. Project $\boldsymbol{X}_j$ on $\boldsymbol{t}_j$ to find the corresponding loadings $\boldsymbol{u}_j$: $\boldsymbol{u}_j = \boldsymbol{X}_j'\boldsymbol{t}_j/\|\boldsymbol{t}_j\|$.
7. Orthogonalize $\boldsymbol{X}_j$ with respect to $\boldsymbol{t}_j$: $\boldsymbol{X}_{j+1} = \boldsymbol{X}_j - \boldsymbol{t}_j\boldsymbol{u}_j'$.
8. Let $\mathbf{y}_{j+1} = \mathbf{y}_j - \widehat{b}_j\boldsymbol{t}_j$.
9. Let $j = j + 1$ and repeat steps 2 to 9 until $j = k$.
```

Similarly to the NIPALS algorithm for PCA, the number of components k is usually determined by cross-validation procedures. The emphasis in PLS is not only on regression but also in uncovering latent structure in $\boldsymbol{X}$ and $\mathbf{y}$. This latent structure is made up of pairs of latent vectors $\boldsymbol{t}_j$ and $\boldsymbol{u}_j$. The latent vectors are determined through the process of estimating the weights $\boldsymbol{w}_j$ for the linear combination of the $\boldsymbol{X}$ variables. The weights $\boldsymbol{w}_j$ correspond to directions in the space of $\boldsymbol{X}$ with highest covariance with $\mathbf{y}$ and are such that large variation in $\boldsymbol{X}$ are accompanied by large variation in $\mathbf{y}$. Steps 2 and 3 of the algorithm show that the $\boldsymbol{w}_j$ are obtained by normalizing the covariance matrix between $\boldsymbol{X}$ and $\mathbf{y}$. In step 4 we construct the latent variable $\boldsymbol{t}_j$ (also called partial least square direction) by reweighting the $\boldsymbol{x}_i$ by the strength of their univariate effect on $\mathbf{y}$ (the weights are given by the covariance between $\boldsymbol{x}_i$ and $\mathbf{y}$). In step 5 the response $\mathbf{y}$ is regressed on $\boldsymbol{t}_j$ to obtain the univariate OLS estimate and then in step 6 we obtain the latent variables $\boldsymbol{u}_j$ by regressing the columns of $\boldsymbol{X}$ on $\boldsymbol{t}_j$. PLS produces a bilinear representation of the data (Martens and Naes, 1989):

$$\begin{aligned}\boldsymbol{X} &= \boldsymbol{t}_1\boldsymbol{u}_1' + \cdots + \boldsymbol{t}_k\boldsymbol{u}_k' + \boldsymbol{X}_{k+1}\\ \mathbf{y} &= \widehat{b}_1\boldsymbol{t}_1 + \cdots + \widehat{b}_k\boldsymbol{t}_k + \mathbf{y}_{k+1}\end{aligned}$$

where the $\boldsymbol{t}_j$ are orthogonal, $\boldsymbol{X}_{k+1}$ and $\mathbf{y}_{k+1}$ are the residuals. The $\boldsymbol{u}_j$ and the $\widehat{b}_j$ are estimated by regression and PLS fits a sequence of bilinear models by least squares, thus the name of partial least squares. In step 7 and 8 the residuals $\boldsymbol{X}_{j+1}$ and $\mathbf{y}_{j+1}$ are calculated and in each iteration only the subspace of $\boldsymbol{X}$ that is orthogonal to the earlier linear combinations developed in the $\boldsymbol{X}-$space is used and the $\mathbf{y}-$space is projected on the space orthogonal to the previous $\boldsymbol{X}$ component. The basic PLS algorithm can be generalised to handle multiple responses.

The maximum number K of components is $\leq \min(n-1, p)$. The number of components k should be chosen so that $\boldsymbol{X}_{k+1}$ contains no information on $\mathbf{y}_{k+1}$ (i.e. they are uncorrelated) and similarly to PCA, k is usually chosen through cross-validation. The first few PLS components are retained as they account for most of the covariance between $\boldsymbol{X}$ and $\mathbf{y}$ and therefore lead to a more parsimonious representation of the data. We refer to Stone and Brooks (1990), Frank and Friedman (1993), Breiman and Friedman (1997), Burnham et al. (1999) and Butler and Denham (2000) for a review of the statistical properties of PLS.

3.2 PLS and Discrimination

In many applications it is common that the response variable $\boldsymbol{y}$ is categorical, representing, say, class membership, e.g. control/dosed, affected/unaffected individuals etc. PLS Discriminant Analyis (PLS-DA) consists of a classical PLS regression where the response variable is a categorical one (replaced by the set of dummy variables describing the categories). The goal of PLS-DA is to sharpen the separation between groups of observations, by hopefully rotating PCA components such that a maximum separation among classes is obtained, and to understand which variables carry the class separating information. Figure 4 shows the application of PLS-DA to a subset of the Hydrazine data. To illustrate class discrimination, we have restricted the analysis to samples from control and low dose animals obtained at 8 to 72 hours post dose. The separation between the groups is evident on the first latent variable and metabolites responsible for such separation can be found by examining the corresponding loadings.

Figure 4 about here

3.3 Orthogonal Projections to Latent Structure

The Orthogonal Projections to Latent Structure (O-PLS) is a modification of the original PLS algorithm and was proposed with the goal of removing variation from $\boldsymbol{X}$ that is not correlated with $\mathbf{y}$ (Trygg and Wold, 2002). Spectra often contain systematic variation that is unrelated to the response $\mathbf{y}$. This systematic variation may be due to experimental effects (such as a temperature drift in the spectrometer) or systematic biological variation (e.g. differences in diets between human subjects which are not easily controlled) and may often constitute the major part of the variation of the sample spectra. It is important to deal with the variation in $\boldsymbol{X}$ uncorrelated with $\mathbf{y}$ as it affects the construction and interpretation of statistical models, may lead to unreliable predictions for new samples and also affect the robustness of the model over time. In the analytical chemistry literature differentiation and signal correction are commonly used to remove systematic variation from the sample (see, for example, Savitsky and Golay (1964), Geladi et al. (1985)).

O-PLS provides a method to remove the variation in $\boldsymbol{X}$ orthogonal to $\mathbf{y}$, therefore improving the interpretation of PLS models and reducing model complexity. It analyzes the disturbing variation in each regular PLS component. Also O-PLS produces a bilinear representation of the data:

$$\begin{aligned}\boldsymbol{X} &= \boldsymbol{t}\boldsymbol{u}' + \boldsymbol{t}_{o1}\boldsymbol{u}'_{o1} + \cdots + \boldsymbol{t}_{ok}\boldsymbol{u}'_{ok} + \boldsymbol{X}_{k+1} \\ \mathbf{y} &= \widehat{b}\boldsymbol{t} + \mathbf{y}_{k+1}\end{aligned}$$

As in PLS $\boldsymbol{t}$ represents the score vector for $\boldsymbol{X}$ and $\mathbf{y}$, $\boldsymbol{u}$ is the vector of orthonormal loadings, $\widehat{b}$ is a scalar and $\boldsymbol{X}_{k+1}$ and $\mathbf{y}_{k+1}$ are the respective residual matrices for $\boldsymbol{X}$ and $\mathbf{y}$. $\boldsymbol{t}_{oj}$ are the scores orthogonal to $\mathbf{y}$ and $\boldsymbol{u}_{oj}$ are the corresponding loadings. The O-PLS method requires only a small modification of the PLS algorithm and therefore can be embedded as an integrated part of the regular PLS modelling. It provides similar predictions to PLS (Cloarec et al., 2005), but leads to more parsimonious models. The number of correlated O-PLS components is reduced to one in the case of a single response, thus making interpretation of the model easier. Moreover, the structured noise is modelled separately from the variation common to $\boldsymbol{X}$ and $\mathbf{y}$ and this gives the opportunity to analyse the uncorrelated variation and possibly explain its presence. An extension of the O-PLS algorithm, known as O2-PLS has been proposed by Trygg and Wold (2003). O2-PLS improves on the original algorithm when dealing with multiple responses by allowing, for example, the calculation of orthogonal latent vectors in the $\boldsymbol{Y}$ space.

Example We illustrate the use of O-PLS method with data from a study on paracetamol (acetaminophen) toxicity (Coen et al., 2003). 26 rats, divided into dose ($n = 17$) and control ($n = 9$) groups were exposed to 150 mg/kg and 0 mg/kg paracetamol respectively. Liver tissue was extracted and ^{1}H NMR spectra acquired to ascertain any metabolic difference due to paracetamol toxicity. Figure 5 shows the scores plot from an initial PCA model and shows poor separation between the groups, along with a clear drift over time. This drift was found to be the result of temperature variations in the spectrometer over the course of the experiment, introducing variation in the profiles which was unrelated to the toxicity effect. O-PLS-DA was used to remove the confounding effect of the temperature drift. This is shown in Figure 6 where variation orthogonal to the class difference has been removed from the first component and a clear separation between the groups is now observed on the scores plot. As mentioned above, a benefit of the O-PLS approach is that one may examine the orthogonal part of the data to ascertain the reasons for the confounding variation. Figure 7 plots the scores on the first orthogonal component and a clear trend with run order is seen, indicating that the time-ordered drift has been removed. The orthogonal loadings can also be examined (data not shown) giving an indication of the metabolites which are most affected by the time drift.

Figures 5, 6 and 7 about here

4 Clustering Procedures

Cluster analysis identifies subgroups or clusters in multivariate data, in such a way that objects belonging to the same cluster resemble each other, whereas objects in different clusters are dissimilar. In two or three dimensions, clusters can often be visualized, but in higher dimensions, we need some kind of analytical assistance.

Data sets for clustering of N observations can have either of the following structures:

- an $N \times p$ **data** matrix, where rows contain the different observations, and columns contain the different variables.
- an $N \times N$ **dissimilarity** matrix, whose $(i,j)^{th}$ element is d_{ij}, the **distance** or **dissimilarity** between observations i and j that has the properties
  - $d_{ii} = 0$
  - $d_{ij} \geq 0$
  - $d_{ji} = d_{ij}$
- The most typical distance measures for between two continuously measured data points i and j with measurement vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ is the *Euclidean* distance

$$d_{ij} = \sqrt{\sum_{k=1}^{p} (x_{ik} - x_{jk})^2} = \sqrt{(\mathbf{x}_i - \mathbf{x}_j)^T (\mathbf{x}_i - \mathbf{x}_j)}$$

Clustering algorithms fall into two categories:

1. **Partitioning Algorithms:** A partitioning algorithm describes a method that divides the data set into k clusters, where the integer k needs to specified. Typically, bf the algorithm is run for a range of k -values. For each k, the algorithm carries out the clustering and also yields a quality index which allows **selection of** the ìbestî value of k afterwards.
2. **Hierarchical Algorithms:**. A hierarchical algorithm yields an entire hierarchy of clusterings for the given data set. *Agglomerative methods* start with the situation where each object in the data set forms its own cluster, and then successively merges clusters until only one large cluster (the entire data set) remains. *Divisive methods* start by considering the whole data set as one cluster, and then splits up clusters until each object is separated.

4.1 Partitioning Methods

Partitioning methods are based on specifying an initial number of groups, and iteratively reallocating observations between groups until some equilibrium is attained. Several different algorithms are available

1. The **k-Means** algorithm: In the k-means algorithm the observations are classified as belonging to one of k groups. Group membership is determined by calculating the centroid for each group (the multidimensional version of the mean) and assigning each observation to the group with the closest centroid. The k-means algorithm iterates between calculating the centroids based on the current group memberships, and reassigning observations to groups based on the new centroids. Centroids are calculated

using least-squares, and observations are assigned to the closest centroid based on least-squares. This assignment is performed in an iterative fashion, either from a starting allocation or configuration, or from a set of starting centroids. The k-means algorithm is essentially model free **(though it assumes the clusters are compact and spherically symmetric), and** is heavily dependent on initial group assignments, the use of the least-squares penalty.

2. **Partitioning around medoids (PAM):** The PAM method uses medoids rather than centroids, that is, medians rather than means in each dimension. This approach increases robustness relative to the least squares approach given above.

4.2 Hierarchical Clustering

Hierarchical clustering procedures can be carried out in two ways

- **Heuristic Criteria** The basic hierarchical agglomeration algorithm starts with each object in a group of its own. At each iteration it merges two groups to form a new group; the merger chosen is the one that leads to the smallest increase in the sum of **within-group sums of squares**. The number of iterations is equal to the number of objects minus one, and at the end all the objects are together in a single group. This is known as *Ward's method*, the *sum of squares method*, or the *trace method.* The hierarchical agglomeration algorithm can be used with criteria other than the sum of squares criterion, such as the *average*, *single or complete linkage* methods described below.

- **Model-Based Criteria** Model-based clustering is based on the assumption that the data are generated by a mixture of underlying probability distributions. Specifically, it is assumed that the population of interest consists of k different subpopulations (usually assumed multivariate normally distributed), and that the density of an observation from the the subpopulation is **specified by** some unknown vector of parameters **which are to be determined**.

In conventional hierarchical clustering, the method of agglomeration or combining clusters is determined by the distance between the clusters themselves, and there are several available choices. For merging two clusters C_i and C_j, with N_1 and N_2 elements respectively, the following criteria can be used; (a) *average linkage* clustering, where the two clusters that have the smallest *average distance between the points in one cluster and the points in the other* are merged, (b) *connected (single linkage, nearest-neighbour)* clustering, where the two clusters that have the smallest *distance between* **any** *point in the first cluster and* **any** *point in the second cluster* are merged, (c) *compact (complete linkage, furthest-neighbour)* clustering, where the two clusters that have the largest *distance between* **any** *point in the first cluster and point in the second cluster* are merged. Efficient algorithms to achieve hierarchical clustering exist, and are readily available in standard statistical packages such as `R`.

4.3 Model-Based Hierarchical Clustering

Another approach to hierarchical clustering is **model-based clustering**, which is based on the assumption that the data are generated by a mixture of K underlying probability distributions. Given data matrix $\mathbf{X} = (\mathbf{x}_1, ..., \mathbf{x}_N)^T$, let

$$\gamma = (\gamma_1, ..., \gamma_N)$$

denote the cluster labels, where $\gamma_i = k$ if the i^{th} data point comes from the k^{th} subpopulation. In the classification procedure, a maximum likelihood procedure is used to choose the parameters in the model.

Commonly, the assumption is made that the data in the different subpopulations follow multivariate normal distributions, with mean μ_k and covariance matrix Σ_k for cluster k If

$$\Sigma_k = \sigma^2 I_p \qquad I_p = diag\,(1, ..., 1)\,, \text{ a } p \times p \text{ matrix.}$$

then maximizing the likelihood is the same as minimizing the sum of within-group sums of squares.

More general covariance models have been implemented in the statistical package `R` in the library `mclust`. We give brief details; other forms of Σ_k yield clustering methods that are appropriate in different situations. The key to specifying this is the eigen decomposition of Σ_k, given by eigenvalues $\lambda_1, ..., \lambda_p$ and eigenvectors $\mathbf{v}_1, ..., \mathbf{v}_p$, as in Principal Components Analysis. The eigenvectors of Σ_k, specify the orientation of the k^{th} cluster, the largest eigenvalue λ_1 specifies its variance or size, and the ratios of the other eigenvalues to the largest one specify its shape. Further, if $\Sigma_k = \sigma_k^2 I_p$, the criterion corresponds to hyperspherical clusters of different sizes; this is known as the *spherical* criterion. Another criterion results from constraining only the shape to be the same across clusters. This is achieved by fixing the eigenvalue ratios

$$\alpha_j = \frac{\lambda_j}{\lambda_1} \qquad j = 2, 3, ..., p$$

across clusters; different choices for the specification yield ellipsoidal, linear or spherical clusters.

4.4 Choosing The Number Of Clusters

A hierarchical clustering procedure gives the sequence by which the clusters are merged (in agglomerative clustering) or split (in divisive clustering) according the model or distance measure used, but does not give an indication for the number of clusters that are present in the data (under the model specification). This is obviously an important consideration. One advantage of the model-based approach to clustering is that it allows the use of statistical model assessment procedures to assist in the choice of the number of clusters. A common method is to use approximate Bayes factors to compare models with different numbers of clusters. This method gives a systematic means of selecting the parameterization of the model, the clustering method, and also the number of clusters. The **Bayes factor** is the posterior odds for one model against the other assuming neither is favored *a priori*. A common approximation to the Bayes factor is the the Bayes Information Criterion, which for model M is given by

$$BIC_M = -2 \log L_M + const \approx -2 \log L_M(\widehat{\theta}) + d_M \log N$$

where L_M is the Bayesian marginal likelihood, $L_M(\widehat{\theta})$ is the maximized log likelihood of the data for the model M, N is the number of data points, and d_M isthe number of parameters estimated in the model. The number of clusters is not considered a parameter for the purposes of computing the BIC. The first term is a measure of fidelity to the data, the second is a penalty for the number of parameters in the model. The more negative the value of the BIC, the stronger the evidence for the model.

4.5 Displaying And Interpreting Clustering Results

The principal display plot for a clustering analysis is the *dendrogram* ,which plots all of the individual data linked at successive levels by means of a binary "tree". Such a plot is displayed in Figure 8. The distance up the tree, or height, represents overall similarity between the samples, can be useful in determining the number of clusters that are present in the data.

Figure 8 displays the hierarchical clustering results for a subset of the Hydrazine dataset (control and low dose 8 to 72 hours). The control samples are shown in black while the dosed samples are shown in red. There are two main branches corresponding to control-like samples and dosed samples. The two groups of subjects are almost perfectly delineated by the clustering procedure if a similarity cut is made at similarity level 0.6 units.

Figure 8 about here

Hierarchical clustering has often been used to investigate metabolic profiling data. Some studies have investigated relationships between the metabolic profiles themselves (Beckonert et al., 2003) while others have applied the method to elucidate dependencies between metabolite concentrations (by clustering the transposed data matrix $\boldsymbol{X}'$ (Dumas et al., 2002)). In both cases, dendrograms generated intuitive visualisations of the hierarchy, allowing effects such as outliers, misclassifications and chemical structure correlations to be observed. Clustering approaches do not, of themselves, offer diagnostic information on the reasons for classification of any given object. This can be important if one wishes to know, for example, the metabolites which are critical to determining cluster membership. Mean cluster profiles, and inter-cluster differences can be inspected, though information can still be lacking on the extent of cluster overlap and the overall relationship between clusters. Finally, note that validation of the results of a clustering exercise is recommended. This can take the form of data peturbation and reclustering, to ensure that the removal of a minority of data points is not pivotal to the clustering inferences or the number of clusters selected.

5 Neural Networks, Kernel Methods and Related Approaches

An Artificial Neural Network (ANN) is a multi-layered statistical model that represents the variation in a potentially highly complex response or output variable to a collection of input variables via varying numbers of unobserved or latent variables linked through simple mathematical functions and a series of probabilistic dependence assumptions. ANNs are a member of a broad class of supervised learning algorithms that attempt to approximate ground truth by mathematical models deduced from observation of cases essentially via regression arguments. These models have been perceived as mechanistic approximations to actual biological neural networks, although this interpretation is neither necessary nor uniformly helpful. We summarize the nature of such models below; see, for example, (Ripley, 1996, Chapter 5) for a comprehensive description of statistical aspects.

5.1 Mathematical Formulation

The simplest mathematical formulation of an ANN involves three levels of interlinked variables;

- The *outputs*, $\boldsymbol{Y}$ (a $q \times 1$ vector), interpreted as a single/collection of continuous or categorical random variable(s).
- The *inputs*, $\boldsymbol{X}$ ($p \times 1$), interpreted as a collection of random variables believed to influence the variation in the **response** across an experimental sample of $\boldsymbol{Y}$s.
- The *hidden variables*, $\boldsymbol{Z}$ ($d \times 1$), interpreted as a collection of unobserved random variables that form the hidden link between $\boldsymbol{X}$ and $\boldsymbol{Y}$s.

This structure can be generalized to incorporate multiple hidden layers, but we restrict attention to the single **hidden** layer case.

In a classical biological conception of the model, the hidden variables $\boldsymbol{Z}$ have some physical interpretation as nodes in a neuronal network in the brain, but this interpretation is not necessary. Mathematically, perhaps the most useful interpretation of the hidden layer is that as a *projection* of the $\boldsymbol{X}$ onto a lower dimensional space ($p > d$) of *features* that facilitates modelling of the variation in $\boldsymbol{Y}$. These features may or may not have a physical interpretation. Diagrammatically, such a network (with $q = 2, p = 6$ and $d = 3$ is represented in Figure 9.

Figure 9 about here

The arrows connecting nodes in Figure 9 represent functional links encompassed through mathematical functions. At the first stage, we represent the components of $\boldsymbol{Z}$ as weighted sums of functions of the input variables, typically

$$z_j = G_j \left(\alpha_j + \sum_{k=1}^{p} w_{jk} g_{jk}(x_k) \right) \qquad j = 1, \ldots, d \tag{2}$$

where $\alpha_j, j = 1, \ldots, d$ are unknown constants, $w_{jk}, k = 1, \ldots, p, j = 1, \ldots, d$ are weights satisfying

$$0 \leq w_{jk} \leq 1 \qquad \sum_{k=1}^{p} w_{jk} = 1$$

and $G_j, j = 1, \ldots, d$ and $g_{jk}, k = 1, \ldots, p, j = 1, \ldots, d$ are known *link* functions, the latter often chosen to be identity functions for convenience. At the second stage we have a similar structure modelling the dependence of $\boldsymbol{Y}$ on $\boldsymbol{Z}$

$$y_l = H_l \left(\beta_l + \sum_{j=1}^{d} \omega_{lj} z_j \right) \qquad l = 1, \ldots, p \tag{3}$$

for parameters $\beta_l, l = 1, \ldots, p$ and weights $\omega_{lj}, j = 1, \ldots, d, l = 1, \ldots, p$.

In summary, therefore, we have parameters $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, $\boldsymbol{w}_j, j = 1, \ldots, d$ and $\boldsymbol{\omega}_l, l = 1, \ldots, p$ to estimate from the data. This is achieved by optimization of some objective function characterizing discrepancy of fit, that is, for observed cases $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n$, we choose parameters $\widehat{\boldsymbol{\theta}}$ as

$$\widehat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} D(\boldsymbol{y}_i, \widehat{\boldsymbol{y}}_i) \tag{4}$$

where D measures the discrepancy between observation $\boldsymbol{y}_i$ and fitted value $\widehat{\boldsymbol{y}}_i$ given by using parameters $\boldsymbol{\theta}$ in equations (2) and (3). D is chosen to represent the continuous or categorical nature of the response variables, and the minimization in equation (4) is achieved using numerical methods.

As described above, ANNs are flexible non-linear regression models constructed from simple mathematical functions that are learned from the observation of cases. As such, they are ideal models for classification. However, their flexibility means that model parameters, such as those determining the architecture (number of input, hidden and output nodes etc.), are not predetermined and must be chosen based on the skill and experience of the user. In addition, while good classification performance for metabolic profiles has been obtained (El-Deredy, 1997), interpretation of the rules encoded by an ANN is not straightforward. This has limited their application in the metabolic profiling arena in recent years, since model interpretation is of prime importance in applications such as biomarker screening and chemical structure elucidation. ANNs have found metabolic profiling applications in diverse areas such as classification of tumours (Howells et al., 1992) pre-clinical toxicity prediction (Anthony et al., 1995), and determination of herbicide mode of action in plants (Ott et al., 2003). In all cases, the network weights could not be interpreted directly and other methods had to be used to determine the biochemical reasons for classification.

5.2 Kernel Density Estimates, PNNs and CLOUDS

Kernel density estimators (KDEs) are a well known class of probability density estimators which have been extensively applied in many different areas of science (Parzen, 1962; Duda et al., 2000). In classification problems, KDEs are usually applied to estimation of the class density (the conditional probability of observing an object, given that it is a member of the class). The estimators are especially useful when the class density deviates substantially from standard (e.g. Normal) forms and allow us to estimate the density in an essentially non-parametric way. Classification of test objects is then effected by calculation of the class density at the coordinates of the test object, and assigning the object to the class whose probability is highest. Using a Gaussian kernel leads to estimates of the form:

$$p\left(\underline{x} \mid \underline{x}_{i \in A}\right) = \frac{1}{N_A (2\pi\sigma^2)^{M/2}} \sum_{i \in A} \exp\left\{ -\frac{\|\underline{x} - \underline{x}_i\|^2}{2\sigma^2} \right\} \tag{5}$$

for the density of class A at point $\underline{x}$, where N_A is the number of training set objects $\underline{x}_i$ in the class, M is the number of dimensions and σ^2 is the width of the kernel. The kernel width regulates the smoothness of the density estimate and requires careful choice. It is often fixed by cross-validation methods at the value which maximises the likelihood of class membership for objects left out of the training set in each cross validation round.

Probabilistic neural networks (PNNs) are a special kind of ANN whose architecture enables them to compute a kernel density estimate of the above form. Each input node corresponds to a variable in the input space and when a test object is presented feeds the corresponding value to each of the nodes in the second layer. Each second layer node corresponds to one training set vector and computes the exponential of equation (5), feeding its output to a final layer of nodes which compute the summation. Classification of Unknowns by Density Superposition (CLOUDS) (Ebbels et al., 2003) uses the KDE / PNN framework to both perform point-wise classification and detect similarity between the estimated densities. The latter function uses the overlap integral of the two densities as a measure of similarity and was developed to compare high dimensional distributions with complex shape and topology.

The potential of PNNs to model complex high dimensional metabolic data was first recognised by Holmes et al. (2001) who developed a model to classify ^{1}H NMR spectra of urine from laboratory rats treated with well known toxins. When asked to classify test samples into one of 18 toxicity-related classes, the PNN obtained higher classification accuracies than back-propagation neural networks and soft independent modelling of class analogy (SIMCA). In the area of food processing, PNNs were recently applied to classifying ^{1}H NMR spectra of fish tissue extracts according to different processing treatments (Martinez et al., 2005). In order to avoid problems with the high dimensionality of the spectra, the scores from the first 20 principal components were used as input to the network which successfully classified 80% of the spectra. In metabonomic toxicology, CLOUDS was applied to several thousand ^{1}H NMR urine spectra from laboratory rats subjected to 19 different treatments known to cause toxic or other metabolic effects (Ebbels et al., 2003). The approach was able to classify samples according to liver or kidney toxicity with 77% and 90% success respectively, while experiencing a low 2% rate of confusion between the classes. The class probabilities were further combined into two parameters, the confidence and uniqueness of the classification, which allowed pre-selection of the best classified urinary profiles. Recently, the CLOUDS methodology was employed in the construction of an 'expert system' for pre-clinical toxicity screening using a large database of ^{1}H NMR urinary metabolic profiles from 80 different treatments (Ebbels et al., 2006). A novel measure of similarity between classes was developed based on the overlap integral of the density estimates. Although the system was unable to judge the likely toxicity of some of the treatments, where a decision could be made, over 92% were classified correctly. In addition, the system correctly determined the site of toxicity for two blinded treatments.

The densities estimated by CLOUDS and similar techniques can be visualised by computing contours or isosurfaces of constant probability density and viewing them with the help of dimension reduction techniques such as PCA. Figure 10 illustrates this with the metabonomic toxicology data of figure 2. The control class has a shape which might be well summarised by a multivariate normal, while the high dose cloud extends away from the controls in a hairpin-like trajectory. The part of the high dose density coinciding with controls is due to samples from animals which have either yet to react or have already recovered from the toxic episode.

Figure 10 about here

Overall, use of KDE methods, while versatile and well suited to the characteristics of metabolic profiling data, are subject to some drawbacks. The most important of these is the large amount of data required to accurately estimate the density when the number of dimensions is high. Along with the high storage requirement and the need for the ability to interpret models in terms of spectral features, these are areas which will be the subject of further research in the near future.

6 Evolutionary algorithms

Evolutionary computations have attracted increasing interest as a method of solving complex problems in medicine, industry and bioinformatics. Evolutionary algorithms (EA) are ideal strategies for mining metabolite data to build useful relationships, rules and predictions. The advantages of this methodology include conceptual simplicity, broad applicability, ability to optimize complex multimodal objective functions and outperform standard optimization procedures in real world applications, flexibility and ease of hybridization with other methods such as Neural Networks (Fogel and Corne, 2003). EAs are adaptive procedures, motivated by genetic processes as they evolve a population of *chromosomal structures* (possible solutions), in order to find the *fittest* individual. Candidate solutions to the optimization problem play the role of individuals in a population, that is evolved through generations undergoing processes inspired by biological evolution: reproduction, mutation, recombination, natural selection and survival of the fittest. Broadly speaking, the evolutionary optimization techniques use a population of possible solutions subjected to random *variation* and *selection* until some termination criterion is satisfied. The fitness of each individual in the population reflects the individual's worth in relation to some objective function.

The aim of the EA is to progressively develop better solutions to the problem under study by modifying previous solutions that exhibited good performance. Starting with an initial population of individuals, generated through some randomized process, each individual is then placed in a common environment where it competes and breeds with the other members of the population. In many applications, the environment is usually referred to as the design space and is the set of all possible solutions for a given problem. The individual's fitness shows how well it has adapted to its environment, i.e. larger fitness values correspond to better solutions. The fittest individuals have more chance to survive in the next generation and to be selected to reproduce. New individuals are generated through variation operators, the most common of which are mutation, the introduction of one or more random changes to a single parent individual, and recombination (commonly referred to as crossover), that consists in randomly taking components/characteristics from two or more parent individuals to create children. Variation operators are fundamental as they affect the algorithm's search power: when the search space has many local optima, large-scale mutations might be useful for escaping local optima, but less radical mutations might be important to proceed towards a global optimum. Following the application of variation operators, two sets of solutions exist: the parent population and the child population. Thus a selection for survival stage is required to form the new population. This step requires scoring each solution on its worth with respect to given goal, i.e. according to their fitness. It is essential that the fitness function reflects the problem characteristics although the specification of a suitable function can, in itself, be a difficult task. Once the solutions have been scored, then the new generation is formed according to various schemes: some schemes require that the new generation is formed only by child solutions, while in other cases the new population is formed by combining individuals from both the parent and the child population. Selection for survival is often a deterministic process, in which the best solutions are selected, however stochastic schemes are also possible. Most EAs work on a generational base, applying variation and selection operators to all members of the population at a given time and iterating this process for a pre-specified number of generations or until convergence to a local or global optimum has been reached.

EAs are becoming increasingly popular in chemometrics as they are useful supervised learning techniques that can be utilized in many scenarios, e.g. to identify metabolites associated with a particular phenotypes (Goodacre, 2005). These include genetic algorithms, evolution strategies, genetic programming and genomic computing. Amongst the evolutionary techniques, genetic programming (GP) has been most commonly used in metabolic profiling applications. In the GP framework, the solution is structured as a parse tree. For example, in symbolic regression the parse tree specifies the regression relationship between the response and the predictors. One of the earliest applications of GPs to metabolic data can be found in Gray et al. (1998) who proposed a genetic programming (GP) algorithm to classify tumours based on ^{1}H NMR spectra of biopsy extracts. Following successful use of genetic programming algorithms for the analysis of pyrolysis mass spectral data of fruit juice

(Gilbert et al., 1997), similar strategies have been developed to analyse metabolic profiling and fingerprinting data from several other spectroscopic and chromatographic techniques including FT-IR (Johnson et al., 2000; Goodacre, 2005), HPLC (Kell et al., 2001), nonlinear dielectric spectroscopy (Woodward et al., 1999) and various types of MS (Taylor et al., 1998; Goodacre et al., 2000, 2003). In most applications the authors highlight the ability of the GP to combine small numbers of explanatory metabolites in simple prediction rules to classify the response of interest. For example, as a test of the GP approach to modelling complex mixtures such as metabolic profiles, Goodacre (2005) developed a GP to quantify the levels of the antibiotic ampicillin in cultures of the bacterium *E. Coli.* Ampicilin was added to *E. Coli* cultures at different concentrations, observed with FT-IR spectroscopy and the GP used in 'symbolic regression' mode to quantify the ampicillin level using the FT-IR profiles as input. The GP was able to identify the 1767cm^{-}1 vibration (corresponding to the beta-lactam ring of ampicillin) as the important variable corresponding to ampicillin despite large signals from other *E. Coli* metabolites, thus highlighting its promise as a method for data mining complex metabolic profiles. Figure 11 shows the frequency of the number of times each input (wavenumber) was used for 10 evolved populations, i.e. 10 runs of the GP algorithm. The area of the spectra corresponding to vibration 1767cm^{-1} is clearly dominating for *E. Coli* treated with ampicilin, but absent from *E. Coli* alone.

Figure 11 about here

7 Conclusions

In this chapter we have described the nature of metabolic profiling data and some of the statistical approaches which are typically used to model it. Due to space considerations we have not undertaken a comprehensive review of all methods used in the field, but have highlighted the salient ones which have seen wide application. The reader is referred to Lindon et al. (2007) for a more thorough description of both the chemical and data analytic techniques used, along with in-depth reviews of many different application areas of the technology. It is clear that there are many problems still to be solved in the statistical analysis of metabolic profiles, such as peak shifts in NMR and the proper treatment of differential ionisation efficiencies in MS. The lack of structural identification of a large proportion of the signals remains a huge impediment to biological interpretation and given the diversity of the metabolome throughout the various kingdoms of life, it seems clear that this situation is unlikely to change any time soon. Therefore statistical approaches which can deal with a mixture of identified and unidentified peaks are likely to have a large impact in this area. Approaches based on mixed-models and wavelet approaches have proved extremely useful (Morris et al., 2006; Brown et al., 2001; Clyde et al., 2005). For example, the sparse representation of the NMR spectrum in terms of wavelet coefficients makes them an excellent tool in data compression and hence feature extraction. In addition, the wide range of chemical analytic techniques used in the generation of metabolic profiles will require statistical models which can integrate and combine information across multiple platforms. Overall, the field is a young but rapidly developing one, and looks set to reap many benefits from input of researchers in other fields, including those of machine learning and statistical analysis.

8 Acknowledgements

The authors would like to acknowledge the members of the Consortium for Metabonomic Toxicology (COMET) for generous permission to use data illustrating the methods in this chapter.

References

Anthony, M. L., Rose, V. S., Nicholson, J. K., and Lindon, J. C. (1995). Classification of toxin-induced changes in 1H NMR spectra of urine using an artificial neural network. *Pharmaceutical and Biomedical Analysis*, 13:205.

Beckonert, O., Bollard, E., Ebbels, T. M. D., Keun, H. C., Antti, H., Holmes, E., Lindon, J. C., and Nicholson, J. K. (2003). NMR-based metabonomic toxicity classification: hierarchical cluster analysis and k-nearest-neighbour approaches. *Analytica Chimica Acta*, 490(3).

Breiman, L. and Friedman, J. H. (1997). Predicting Multivariate Responses in Multiple Linear Regression. *Journal of the Royal Statistical Society, Series B (Methodological)*, 59:3–54.

Brown, P. J., Fearn, T., and Vannucci, M. (2001). Bayesian wavelet regression on curves with applications to a spectroscopic calibration problem. *Journal of the American Statistical Society*, 96:398–408.

Burnham, A. J., MacGregor, J. F., and Viveros, R. (1999). A Statistical Framework for Multivariate Latent Variable Regression Methods Based on Maximum Likelihood. *Journal of Chemometrics*, 13:49–65.

Butler, N. A. and Denham, M. C. (2000). The peculiar shrinkage properties of partial least squares regression. *Journal of the Royal Statistical Society, Series B (Methodological)*, 62:585–593.

Christoffersson, A. (1970). *The one component model with incomplete data*. PhD thesis, Uppsala University.

Cloarec, O., Dumas, M., Trygg, J., Craig, A., Barton, R., Lindon, J., Nicholson, J., and Holmes, E. (2005). Evaluation of the orthogonal projection to latent structure model limitations caused by chemical shift variability and improved visualization of biomarker changes in ^{1}H NMR spectroscopic metabonomic studies. *Analytical Chemistry*, 77:517–526.

Clyde, M. A., House, L. L., and Wolpert, R. L. (2005). Nonparametric models for proteomic peak identification and quantification. Technical report, Duke University.

Coen, M., Lenz, E. M., Nicholson, J. K., Wilson, I. D., Pognan, F., and Lindon, J. C. (2003). An integrated metabonomic investigation of acetaminophen toxicity in the mouse using NMR spectroscopy. *Chemical Research in Toxicology*, 16:295–303.

Davies, T. (1998). The new automated mass spectrometry deconvolution and identification system (AMDIS). *Spectroscopy*, 10(3):24–27.

Duda, R. O., Hart, P. E., and Stork, D. G. (2000). *Pattern Classification.* John Wiley & Sons, New York, NY, USA.

Dumas, M. E., Canlet, C., Andre, F., Vercauteren, J., and Paris, A. (2002). Metabonomic assessment of physiological disruptions using 1H-13C HMBC-NMR spectroscopy combined with pattern recognition procedures performed on filtered variables. *Analytic Chemistry*, 74(3):2261–2273.

Ebbels, T. M. D., Keun, H. C., Beckonert, O., Antti, H., Bollard, M., Holmes, E., Lindon, J. C., and Nicholson, J. K. (2003). Toxicity classification from metabonomic data using a density superposition approach: 'clouds'. *Analytica Chimica Acta*, 490:109.

Ebbels, T. M. D., Keun, H. C., Beckonert, O., Bollard, E., Lindon, J. C., Holmes, E., and Nicholson, J. K. (2006). Prediction and classification of drug toxicity using probabilistic modeling of temporal metabolic data. in preparation.

El-Deredy, W. (1997). Pattern recognition approaches in biomedical and clinical magnetic resonance spectroscopy: a review. *NMR Biomedicine*, 10:99–124.

Fogel, G. B. and Corne, D. W. (2003). An Introduction to Evolutionary Computation for Biologists. In Fogel, G. B. and Corne, D. W., editors, *Evolutionary Computation in Bioinformatics*, pages 19–38. Morgan Kaufmann Publishers.

Frank, I. E. and Friedman, J. H. (1993). A Statistical View of Some Chemometrics Regression Tools. *Technometrics*, 35:109–148.

Geladi, P., MacDougall, D., and Martens, H. (1985). Linearization and scatter-correction for near-infrared reflectance spectra of meat. *Applied Spectroscopy*, 3:491–500.

Gilbert, R. J., Goodacre, R., Woodward, A. M., and Kell, D. B. (1997). Genetic programming: A novel method for the quantitative analysis of pyrolysis mass spectral data. *Analytical Chemistry*, 69:4381–4389.

Goodacre, R. (2005). Making sense of the metabolome using evolutionary computation: seeing the wood with the trees. *Journal of Experimental Botany*, 56:245–254.

Goodacre, R., Shann, B., Gilbert, R. J., Timmins, E. M., McGovern, A. C., Alsberg, B. K., Kell, D. B., and Logan, N. A. (2000). Detection of the dipicolinic acid biomarker in bacillus spores using curie-point pyrolysis mass spectrometry and fourier transform infrared spectroscopy. *Analytical Chemistry*, 72:119–127.

Goodacre, R., York, E. V., Heald, J. K., and Scott, I. M. (2003). Chemometric discrimination of unfractionated plant extracts analyzed by electrospray mass spectrometry. *Phytochemistry*, 62:859–863.

Gray, H. F., Maxwell, R. J., Martinez-Perez, I., Arus, C., and Cerdan, S. (1998). Genetic programming for classification and feature selection: analysis of 1H nuclear magnetic resonance spectra from human brain tumour biopsies. *NMR in Biomedicine*, 11:217–224.

Hastie, T., Tibshirani, R., and Friedman, J. (2001). *The Elements of Statistical Learning*. Springer-Verlag, New York, NY, USA.

Holmes, E., Bonner, F. W., Sweatman, B. C., Lindon, J. C., Beddell, C. R., Rahr, E., and Nicholson, J. K. (1992). Nuclear-magnetic-resonance spectroscopy and pattern-recognition analysis of the biochemical processes associated with the progression of and recovery from nephrotoxic lesions in the rat induced by mercury(ii) chloride and 2-bromoethanamine. *Molecular Pharmacology, ,*, 42:922.

Holmes, E., Nicholson, J. K., and Tranter, G. (2001). Metabonomic characterization of genetic variations in toxicological and metabolic responses using probabilistic neural networks. *Chemical Research in Toxicology*, 14:182.

Howells, S. L., Maxwell, R. J., Peet, A. C., and Griffiths, J. R. (1992). An investigation of tumor 1H nuclear magnetic resonance spectra by the application of chemometric techniques. *Magnetic Resonance Medicine*, 28:214–36.

Johnson, H. E., Gilbert, R. J., Winson, M. K., Goodacre, R., Smith, A. R., Rowland, J. J., Hall, M. A., and Kell, D. B. (2000). Explanatory analysis of the metabolome using genetic programming of simple, interpretable rules. *Genetic Programming and Evolvable Machines*, 1:243.

Jolliffe, I. T. (1986). *Pricipal Component Analysis*. Springer-Verlag, New York, NY, USA.

Kell, D. B., Darby, R. M., and Draper, J. (2001). Genomic computing. explanatory analysis of plant expression profiling data using machine learning. *Plant Physiology*, 126:943–951.

Lindon, J. C., Holmes, J. L., and Tranter, G. E. (2007). *A Handbook of Metabonomics and Metabolomics*. Elsevier.

Lindon, J. C., Keun, H. C., Ebbels, T. M. D., Pearce, J. M., Holmes, E., and Nicholson, J. K. (2005). The consortium for metabonomic toxicology (COMET): aims, activities and achievements. *Pharmacogenomics*, 6:691–699.

Martens, H. and Naes, T. (1989). *Multivariate Calibration.* New York, NY, USA, New York, NY, USA.

Martinez, I., Bathen, T., Standal, I. B., Halvorsen, J., Aursand, M., Gribbestad, I. S., and Axelson, D. E. (2005). Bioactive compounds in cod (*gadus morhua*) products and suitability of 1H NMR metabolite profiling for classification of the products using multivariate data analyses. *Journal of Agricultural and Food Chemistry*, 53:6889–6895.

Massy, W. F. (1965). Principal Component Regression in Exploratory Statistical Research. *Journal of the American Statistical Association*, 60:234–246.

Morris, J., Brown, P., Baggerly, K., and Coombes, K. (2006). *Bayesian Inference for Gene Expression and Proteomics*, chapter Analysis of Mass Spectrometry Data Using Bayesian Wavelet-Based Functional Mixed Models., pages 269–292. Cambridge University Press.

Nicholson, J. K., Connelly, J., Lindon, J. C., and Holmes, E. (2002). Metabonomics: a platform for studying drug toxicity and gene function. *Nature Reviews Drug Discovery*, 1:153.

Nicholson, J. K., Lindon, J. C., and Holmes, E. (1999). Metabonomics': understanding the metabolic responses of living systems to pathophysiological stimuli via multivariate statistical analysis of biological NMR spectroscopic data. *Xenobiotica*, 29:1181.

Nicholson, J. K. and Wilson, I. D. (2003). Understanding 'global' systems biology: Metabonomics and the continuum of metabolism. *Nature Reviews Drug Discovery*, 2:668.

Ott, K. H., Aranibar, N., Singh, B., and Stockton, G. W. (2003). Metabonomics classifies pathways affected by bioactive compounds. artificial neural network classification of NMR spectra of plant extracts. *Phytochemistry*, 62:971–85.

Parzen, E. (1962). On estimation of a probability density function and mode. *The Annals of Mathematical Statistics*, 33:1065.

Raamsdonk, L. M., Teusink, B., Broadhurst, D., Zhang, N., Hayes, A., Walsh, M. C., Berden, J. A., Brindle, K. M., Kell, D. B., Rowland, J. J., Westerhoff, H. V., van Dam, K., and Oliver, S. G. (2001). A functional genomics strategy that uses metabolome data to reveal the phenotype of silent mutations. *Nature Biotechnology*, 19:45–50.

Ripley, B. D. (1996). *Pattern Recognition and Neural Networks.* Cambridge University Press, Cambridge.

Savitsky, A. and Golay, M. J. E. (1964). Smoothing and differentiation of data by simplified least squares procedures. *Analytical Chemistry*, 36:1627–1639.

Stein, S. E. (1999). An integrated method for spectrum extraction and compound identification from gc/ms data. *Journal of the American Society of Mass Spectrometry*, 10:770–871.

Stone, M. (1974). Cross-validatory Choice and Assessment of Statistical Predictions (with discussion). 36:111–147.

Stone, M. and Brooks, R. J. (1990). Continuum Regression: Cross-Validated Sequentially Constructed Prediction Embracing Ordinary Least Squares, Partial Least Squares and Principal Components Regression. 52:237–269.

Stoyanova, R., Nicholls, A. W., Nicholson, J. K., Lindon, J. C., and Brown, T. R. (2004). Automatic alignment of individual peaks in large high-resolution spectral data sets. *Journal of Magnetic Resonance*, 170:329–35.

Taylor, J., Goodacre, R., Wade, W. G., Rowland, J. J., and Kell, D. B. (1998). The deconvolution of pyrolysis mass spectra using genetic programming: application to the identification of some eubacterium species. *FEMS Microbiology Letters*, 160:237–246.

Trygg, J. and Wold, S. (2002). Orthogonal projections to latent structures (O-PLS). *Journal of Chemometrics*, 16:119–128.

Trygg, J. and Wold, S. (2003). O2-pls, a two-block (x-y) latent variable regression (LVR) method with an integral osc filter. *Journal of Chemometrics*, 17:53–64.

West, M. (2002). Bayesian Factor Regression Models in the “Large p, Small n” Paradigm. In Bernardo, J. M., Bayarri, M. J., Berger, J. O., Dawid, A. P., Heckerman, D., Smith, A. F. M., and West, M., editors, *Bayesian Statistics 7*, pages 723–732. Oxford: University Press.

Wilson, I. D., Plumb, R., Granger, J., Major, H., Williams, R., and Lenz, E. M. (2005). Hplc-ms-based methods for the study of metabonomics. *Chromatogr B Analyt Technol Biomed Life Sci*, 817:67–76.

Wold, H. (1966). Estimation of principal components and related models by iterative least squares. In Krishnaiah, P. R., editor, *Multivariate Analysis, Proceedings of the International Symposium, June 1965*, pages 391–420. Academic Press: New York.

Wold, H. (1975). Soft Modelling by Latent Variables; the Non-linear Iterative Partial Least Squares Approach. In Gani, J., editor, *Perspectives in Probability ans Statistics, Papers in Honour of M.S. Bartlett*. London: Academic Press.

Wold, S. (1978). Cross-validatory estimation of the number of components in factor and principal components models. *Technometrics*, 20:397–404.

Woodward, A. M., Gilbert, R. J., and Kell, D. B. (1999). Genetic programming as an analytical tool for non-linear dielectric spectroscopy. *Bioelectrochemistry and Bioenergetics*, 48:389.

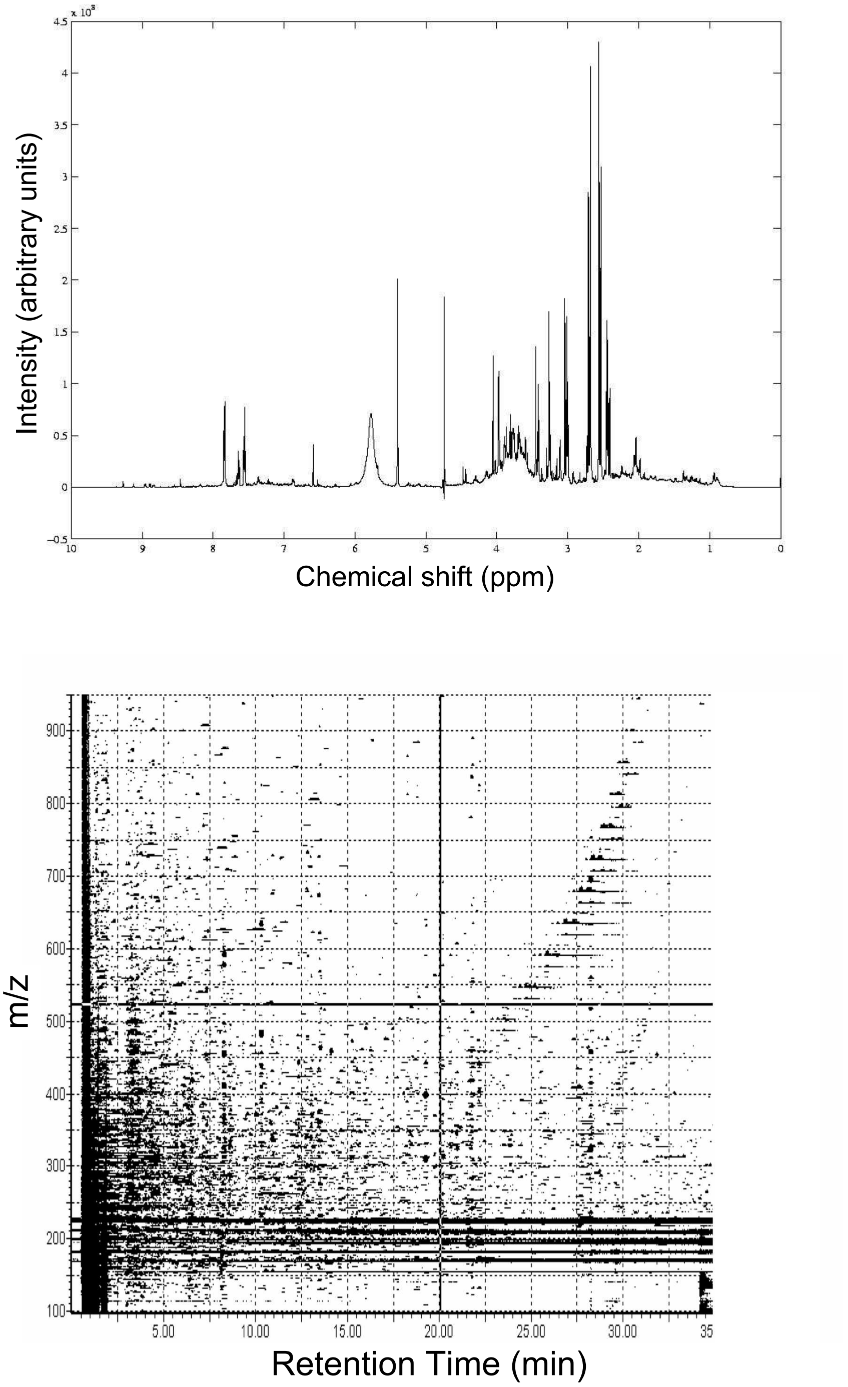

Figure 1: Typical metabolic profiles of rat urine. The top panel shows a one dimensional ^{1}H NMR spectrum, while the bottom panel shows a two dimensional contour plot of an LC-MS profile. Both the higher number of signals and level of noise are clear in the LC-MS profile.

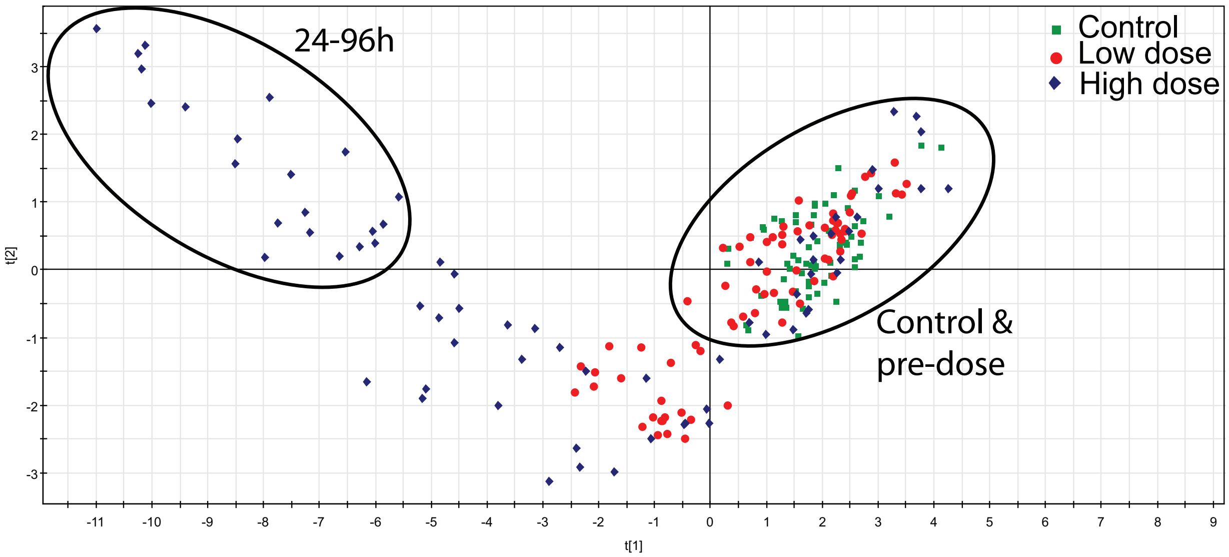

Figure 2: Scores plot of the first two principal components for the Hydrazine dataset. Each point represents one urine sample (metabolic profile) from one animal at one time point. The different colours indicate different treatment groups (green control, red low dose, blue high dose). The plot shows the characteristic L shaped trajectory of Hydrozine toxicity; high dose profiles initially resembles controls but over the time course move to a well separated region of the plot. The ellipses identify regions associated with control-like profiles (on the right) and profiles showing maximum toxicity (on the left).

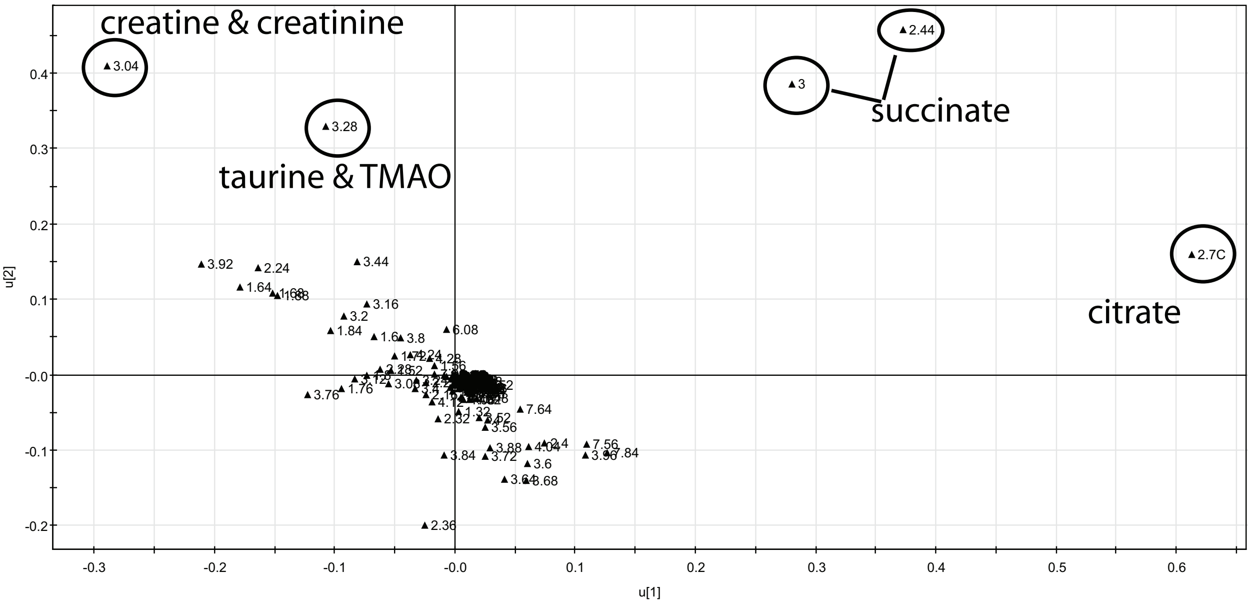

Figure 3: PCA loadings plot of the first two principal components for the Hydrazine dataset. Each point represents one spectral bin. Points lying far from the origin indicate those bins characterised by a high degree of variation in the data and thus explaining the pattern on the scores plot. Some analytes of interested are labelled on the plot.

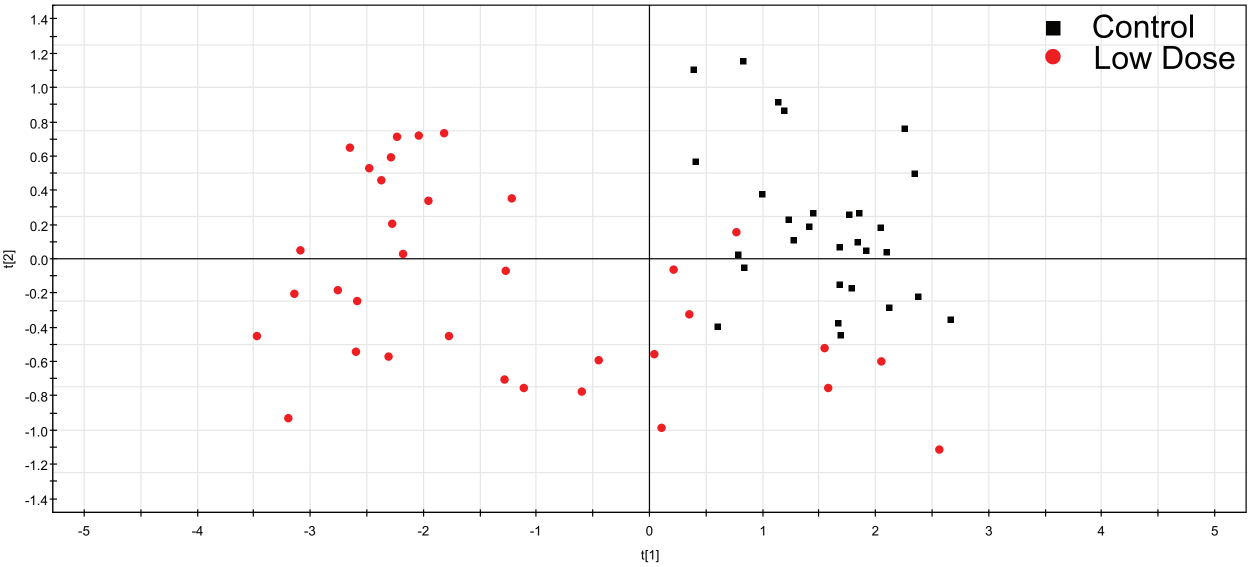

Figure 4: PLS-DA scores of the first two latent variables for a subset of the Hydrazine dataset (control and low dose 8 to 72 hours). The plot shows a clear separation between the two subgroups on the first latent variable. The metabolites responsible for the separation can be found from the corresponding loadings (results not shown).

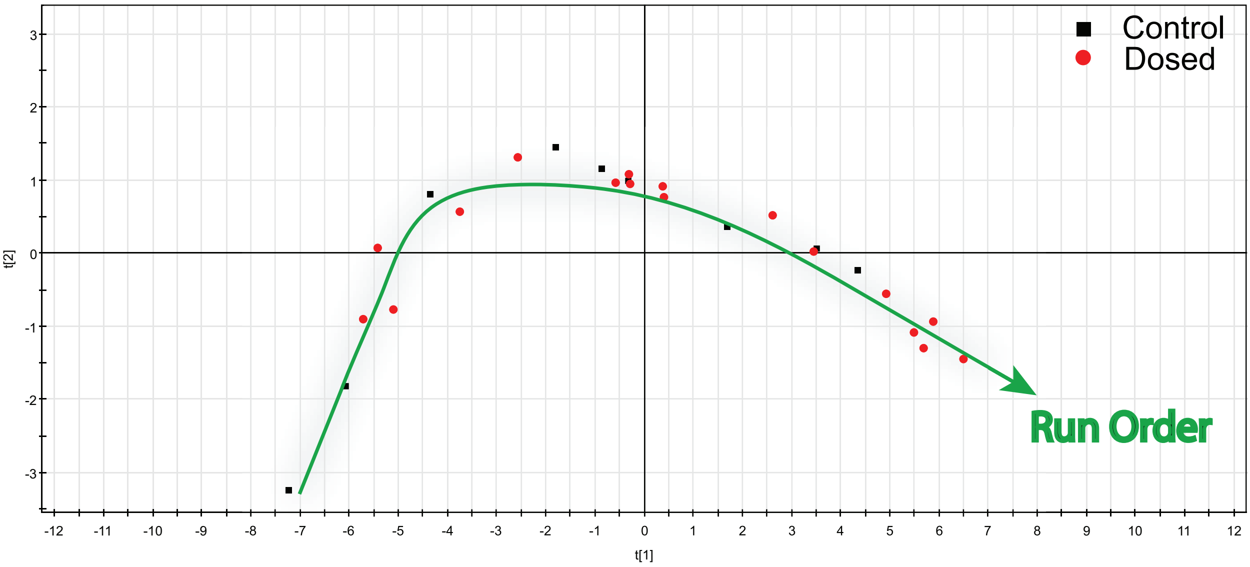

Figure 5: Scores plot of the first two principal components of the paracetamol dataset. The PCA score plot shows no separation and a clear trend due to run order is highlighted by the arrow.

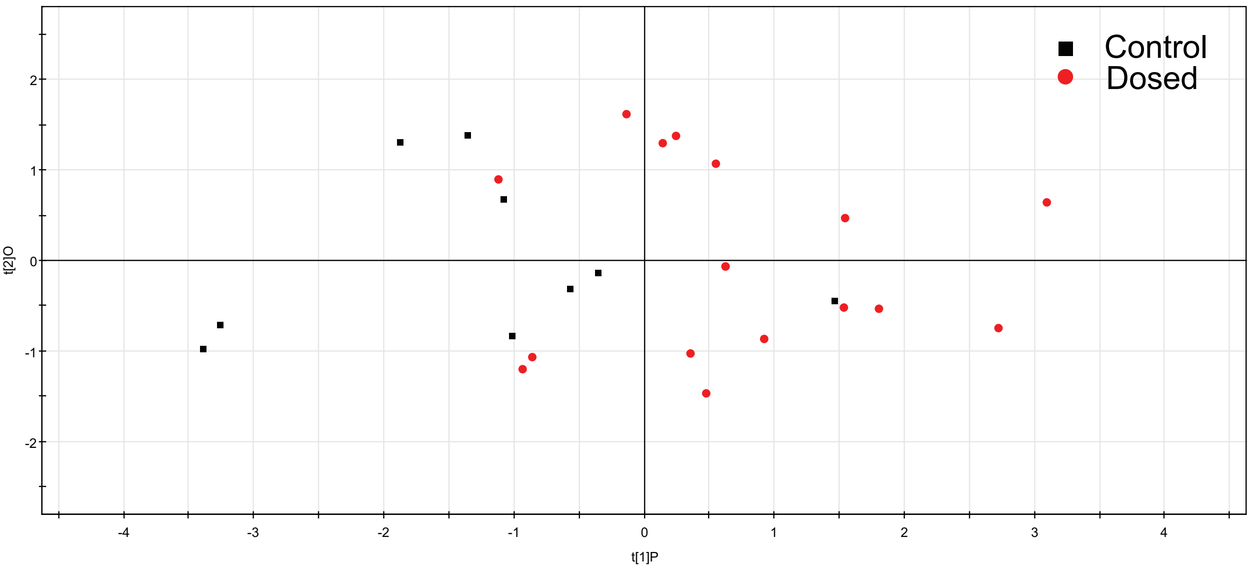

Figure 6: Scores plot of the first orthogonal component ($y-$axis) versus the first correlated component ($x-$axis) for the paracetamol dataset. The systematic variation visible in figure 5 has been captured by the orthogonal component and it is no longer confounding the interpretation of the correlated component.

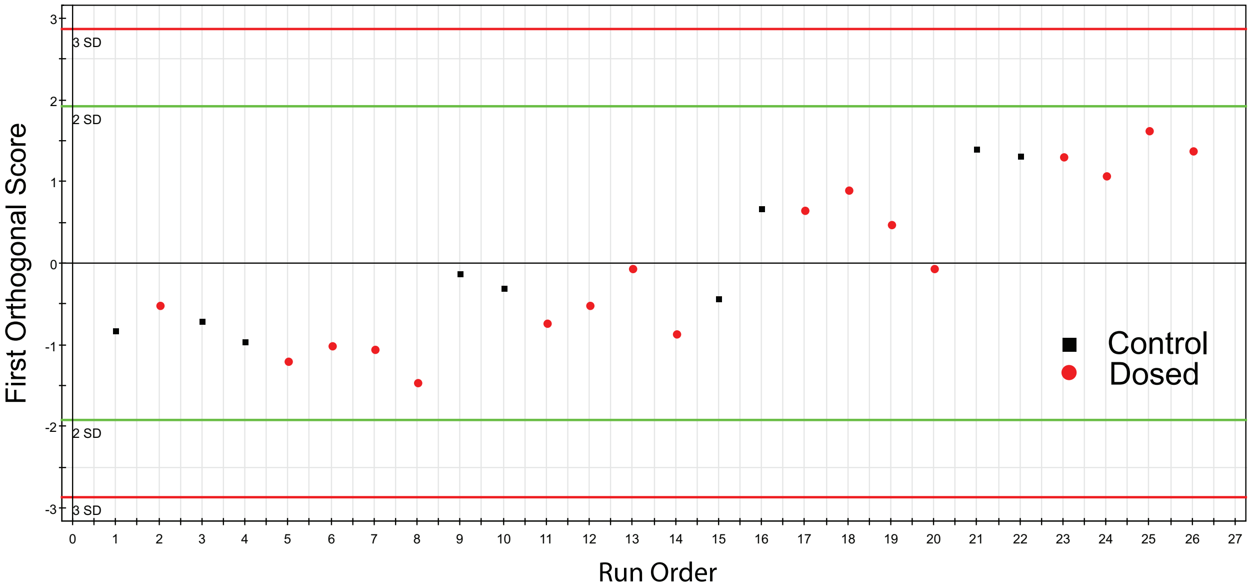

Figure 7: The plot shows the a strong relationship between the first orthogonal component scores ($y-$axis) and the experimental run order ($x-$axis). The orthogonal filtering captures most of the systematic variation due instrumental drift, visible in figure 5.

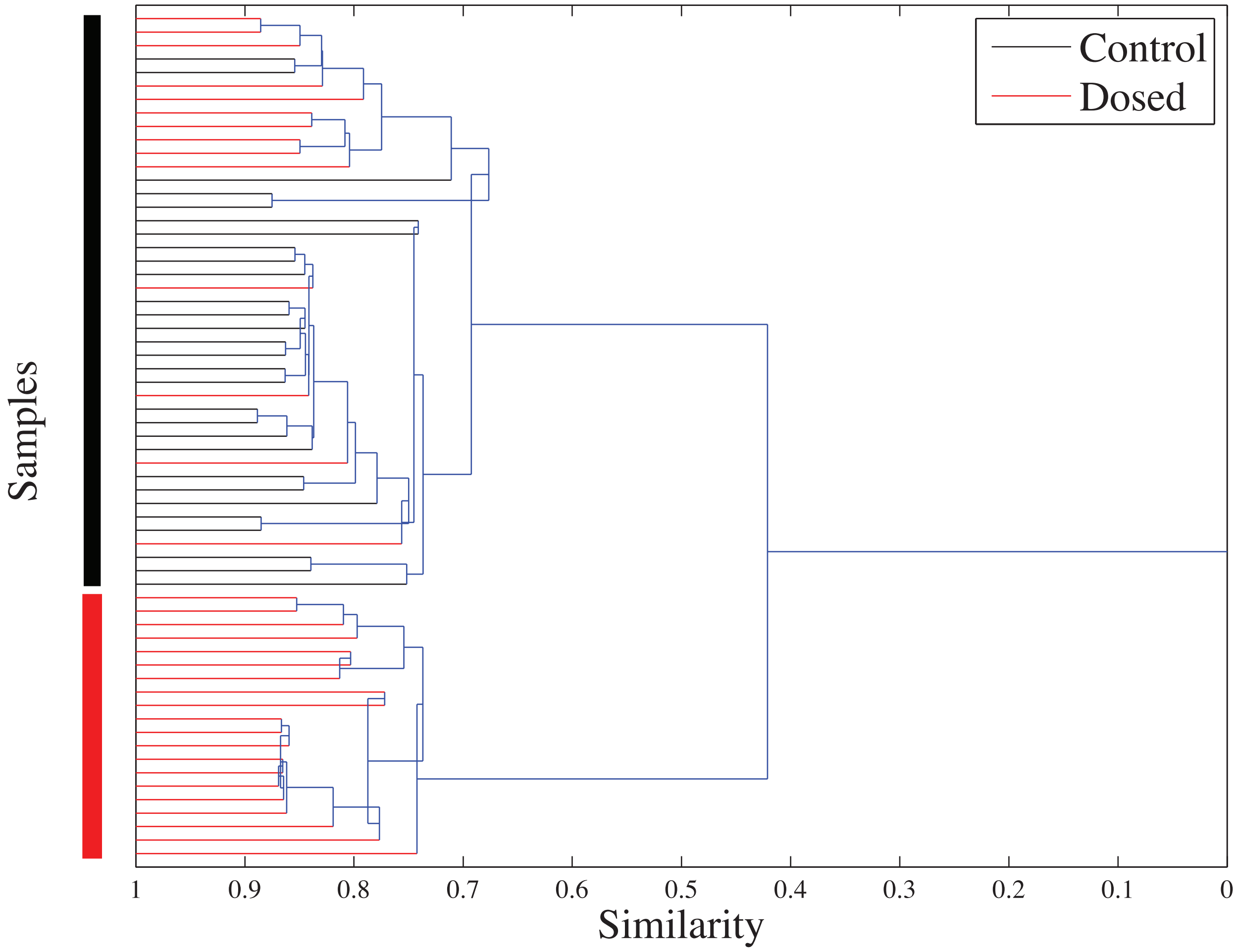

Figure 8: Hierarchical clustering for a subset of the Hydrazine dataset (control and low dose 8 to 72 hours). The control samples are shown in black while the dosed samples are shown in red. There are two main branches corresponding to control-like samples and dosed samples. Note some dosed samples appeared to be similar to controls because the relevant animals have either recovered from or not yet responded to the toxin.

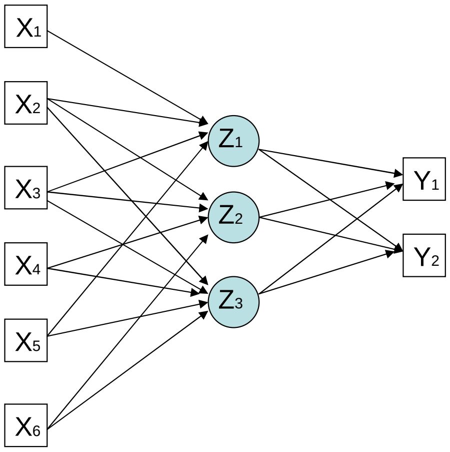

Figure 9: Three-level neural network with $q = 2$, $p = 6$ and $d = 3$. The hidden layer $\boldsymbol{Z} = (Z_1, Z_2, Z_3)'$ links inputs $\boldsymbol{X}$ to outputs $\boldsymbol{Y}$. Arrows indicate positive weights.

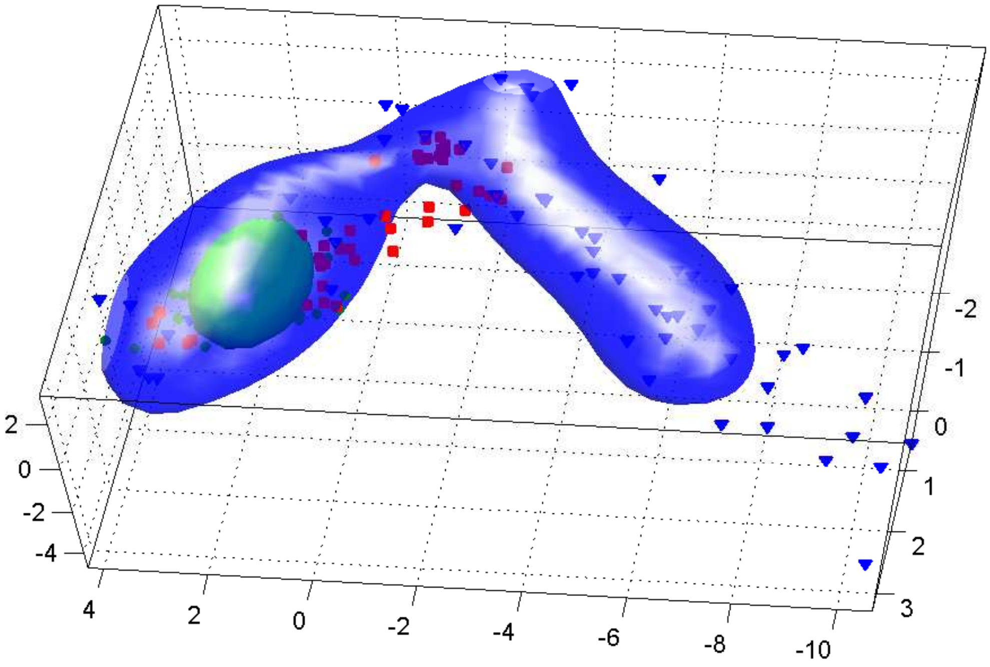

Figure 10: Visualisation using CLOUDS. The figure shows the toxicity data of figure 1 (first 3 PCs) superimposed with iso-surfaces of constant probability density estimated through CLOUDS. The surface for the high dose class (blue) has been rendered transparent so as to enable viewing of the control density (green). One can clearly see how irregular distributions can be modelled in this way.

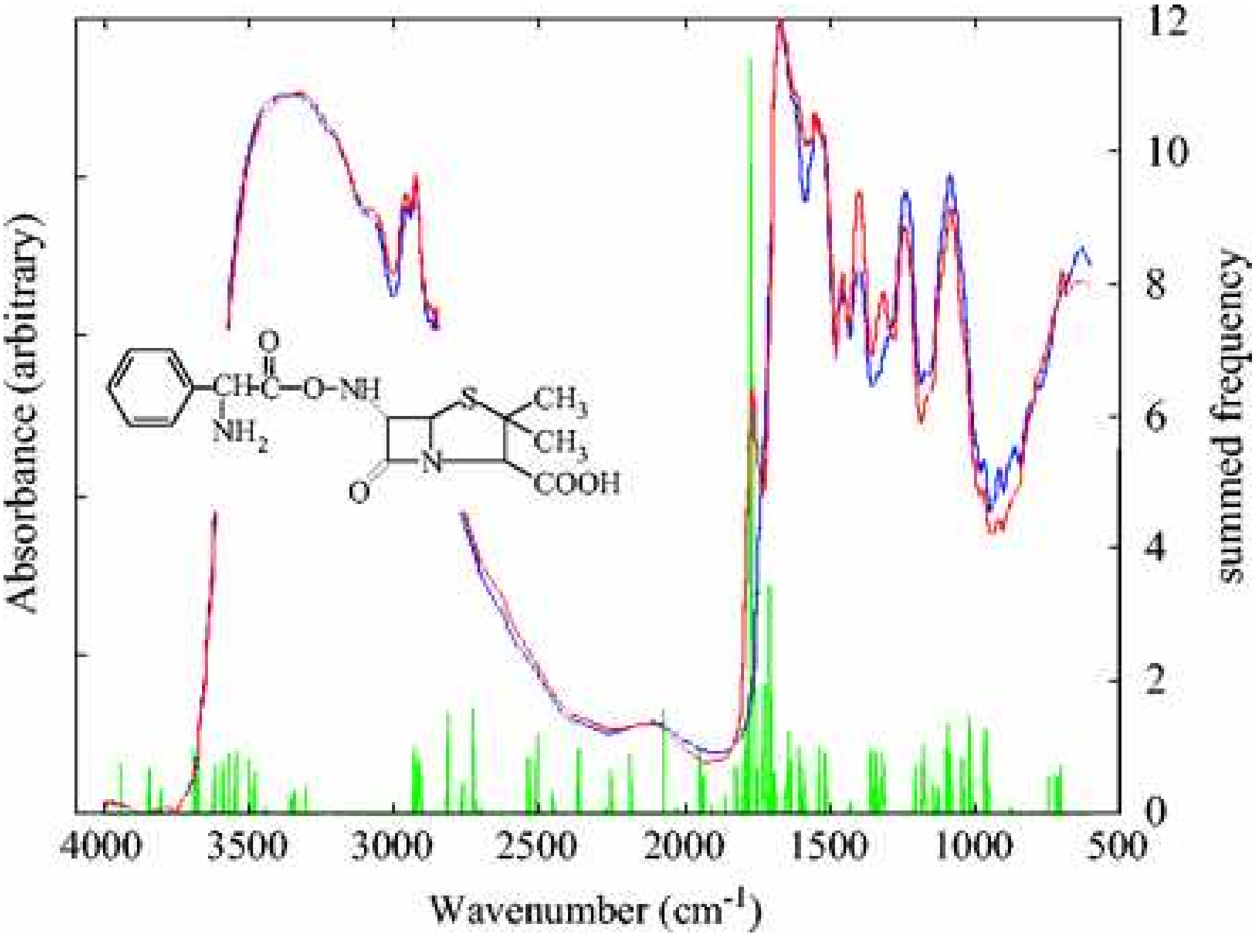

Figure 11: Summed frequency plot from GP analysis of the number of times each input (wavenumber) was used for the 10 evolved populations. Also shown are the normalized FT-IR spectra from *E. Coli* (blue trace) and *E. Coli*+5000 μg.ml^{-1} ampicillin (red trace), and the structure of ampicillin. (Reproduced with permission from Goodacre (2005). Copyright 2005 Oxford University Press)