Published in final edited form as:
Anal Chem. 2009 July 1; 81(13): 5204–5217. doi:10.1021/ac900251c.

Variable Selection using Iterative Reformulation of Training Set Models for Discrimination of Samples: Application to Gas Chromatography Mass Spectrometry of Mouse Urinary Metabolites

Kanet Wongravee[a], **Nina Heinrich**[b], **Maria Holmboe**[b], **Michele L. Schaefer**[c], **Randall R. Reed**[c], **Jose Trevejo**[b], and **Richard G. Brereton**[a]

[a] Centre for Chemometrics, School of Chemistry, University of Bristol, Cantocks Close, Bristol BS8 1TS, UK

[b] The Charles Stark Draper Laboratory, Inc., 555 Technology Square, Cambridge, MA 02139-3563, USA

[c] Johns Hopkins University, School of Medicine, Dept Neuroscience, Center for Sensory Biology, Department of Molecular Biology & Genetics, 725 North Wolfe St, Baltimore, MD 21205, USA

Abstract

The paper discusses variable selection as used in large metabolomic studies, exemplified by mouse urinary gas chromatography of 441 mice in three experiments to detect the influence of age, diet and stress on their chemosignal. Partial Least Squares Discriminant Analysis (PLS-DA) was applied to obtain class models, using a procedure of 20,000 iterations including the bootstrap for model optimisation and random splits into test and training sets for validation. Variables are selected using PLS regression coefficients on the training set using an optimised number of components obtained from the bootstrap. The variables are ranked in order of significance and the overall optimal variables are selected as those that appear as highly significant over 100 different test and training set splits. Cost benefit analysis of performing the model on a reduced number of variables is also illustrated. This paper provides a strategy for properly validated methods for determining which variables are most significant for discriminating between two groups in large metabolomic datasets avoiding the common pitfall of overfitting if variables are selected on a combined training and test set, and also taking into account that different variables may be selected each time the samples are split into training and test sets using iterative procedures.

1 Introduction

Over the past few years with the ready availability of large metabolomic datasets, especially as obtained using chromatography, there has been a significant growth in the use of pattern recognition studies for both discrimination and variable or feature selection. Partial Least Squares Discriminant Analysis [1–5], has been one of the most widespread techniques.

However in most studies, a large number of variables (or unique compounds) can be found, often several thousands or more [6–7]. In order to provide adequate sample sizes for effective modelling which is necessary [8] in many cases, the number of variables increases still further with compounds found sometimes in only a small subset of samples, as every extra sample

Correspondence to: Richard G. Brereton.

usually adds variables. However most variables are redundant and not useful to model the factor of interest. They may arise from uninteresting factors, environmental and analytical procedures etc. Perhaps only a small percentage of the measured variables is relevant to the study of interest. A flawed but common philosophy is to select these variables in advance on the entire dataset. By analogy we may toss a coin 20 times (corresponding for example to performing analyses on 20 samples). Sometimes an unbiased coin will turn Hs 17 or 18 times. If this experiment is repeated 1000 times (representing 1000 variables), then we will obtain some results for which there will be significantly more Hs than Ts, by analogy these represent potential marker compounds. If we then select only those results for which there are at least 15 Hs, and reform our model using these it will, falsely, look as if the coin is biased. In analogy in metabolomic studies, there will always be some variables that appear to have an unequal distribution between two or more groups, but it is a mistake to then select just these variables and reject the remainder, and form a model only on this subset. It can be shown that even for a random dataset, using such an approach results in apparent separation between two groups[8]. Hence great caution and strong safeguards are required.

In the past when computing power has been more limited, methods such as cross-validation [9] have been used to assess models but with modern computing we can produce large numbers of models (in the case of this paper 20,000) to assess average effects of variables. It can be shown that the bootstrap provides a more conclusive indicator of how many components are useful for the model than cross-validation[10] which allows one to reform test sets and bootstrap sets[11–12] many times over and see which variables or features occur regularly – which gives a much more robust indication of which are the significant variables. There are of course, other alternatives to the bootstrap for determining the optimum number of components, but on the whole iterative or permutation approaches tend to perform better than more traditional methods such as cross-validation. In addition each time the data are split into test and training sets, the model changes, being dependent on the samples included in the training set and therefore it is important to reform the training/test set split several times (in this paper we recommend 100 times): it is important to realise that the variables that appear most significant may vary each time the data is split into test and training sets, but the main aim of this paper is to develop a strategy that overcomes this whilst protecting against over optimistic and potentially flawed predictions. The methods described in the paper, can be applied to any indicator of significance of variables, which is used to rank each potential discriminator in order of significance, such as PLS regression coefficients, PLS weights, t-statistics or ANOVA based approaches. However in this paper we employ only PLS regression coefficients for brevity.

In addition when the final model is obtained, we may wish to reduce the number of variables, because each variable is often expensive and difficult to measure, for example it may not be necessary to measure 1,000 different biomarkers, but can an adequate model be obtained with just 10 biomarkers? What is the benefit of increasing the number to 15 biomarkers? This allows very efficient selection of variables, and in this paper we study the benefit of increasing number of variables on the quality of the model. It should be noted that more variables than are found by the methods below could be statistically significant but the aim of this paper is not to identify all variables that are potential markers, but an optimal subset of variables that are sufficient for discrimination between groups. For example we may find that there are 100 statistically significant markers than can be used to distinguish between males and females, but we could obtain an adequate model when using the top 20 of these variables, and retaining a further 80 involves more effort (eg expensive sensors).

The methods are illustrated by a large mouse dataset where Gas Chromatography Mass Spectrometry (GCMS) of volatiles extracted from urine which is used to profile the mice according to age, whether they have been on a diet or whether they are stressed.

It is important to recognise that most metabolomic methods for pattern recognition are exploratory and whereas the methods discussed in this paper provide good criteria for statistically sound predictions of which compounds are potentially discriminatory, they do not provide any information about why : for example we may be trying to discriminate males and females and in our population there may be more males with blue eyes than females, simply because of way the sampling is performed – this does not necessarily imply that blue eyes are linked genetically to maleness and in order to demonstrate whether there is causality, further experiments would need to be performed.

2 Experimental

2.1 Datasets

The dataset in the entire study consisted of 721 mouse urine samples. In this paper only 441 of these samples from three different studies which involve the effect of age (dataset 1), stress (dataset 2) and diet (dataset 3) on urinary chemicals are reported. The datasets consisted of urine samples from inbred mouse strains maintained in controlled environmental housing conditions with regulated light-dark (12hr/12hr light on at 06:00) cycles. Urine from individual male mice was collected to minimize sex-dependent chemical patterns and cycling in females. All mice, with the exception of those from the diet study, were allowed free access to a standard diet and water. Each sample consisted of urine from a single animal (minimum 50μl) and was stored at −80°C. The datasets used in the analyses and details of the samples are presented in Table 1, to study the effect of diet, stress and age on the mouse urinary chemosignal. The number of samples collected per mouse differs slightly because in some cases the amount of urine collected was insufficient and in other cases the quality of the GCMS chromatogram was inadequate. For the diet study, control and altered diet animals were maintained on the regular diet or one in which the fat component was increased from 5.3% to 15.3% for two weeks before and the 2 weeks during sample collection. The dataset for the stress study was generated by introducing a brief restraint protocol followed by collection of samples within 90 min. The protocol was repeated over 10 consecutive days. In the age study, mice were sampled over a period of several weeks: in this paper, the data are analysed as a classification problem, attempting to distinguish mice aged 4 weeks from those aged 8 weeks and above : this division into age groups is between very young mice (4 weeks) and sexually mature adolescent/adult mice and is most suited for classification studies as there is a sharp difference in metabolism as mice reach maturity.. Urine was collected from two cohorts of 10 mice as outlined in Table 1. Cohort 1 was born on Apr 5, 2006 and followed for 30 weeks. Cohort 2 was born on Aug 10, 2006 and followed for 12 weeks. Urine samples for each age group were obtained as morning and late afternoon samples (approximately 6 hours apart) over a 5 day period. The dataset consisted of a total of 48 samples from 4 week old mice and 144 from older mice. One advantage for using two cohorts was to eliminate the problems of instrumental variability with time which can be confounded with storage problems, therefore all samples were analysed by GCMS (Gas Chromatography Mass Spectrometry) close in time between Nov 6 and Nov 22, 2006; since two cohorts are studied, the storage time of samples from mice of the same age group is different for each cohort and so we can distinguish storage time from age, and be sure that the effects we are observing are age related. In this paper we define an “in group” as the mice containing a characteristic which we wish to detect e.g. whether a mouse is on a diet or not, and the “out group” are in practice controls.

2.2 Extraction and Instrumentation

Volatiles from urine sample headspace were concentrated on 4-divinylbenzene/carboxen/polydimethysiloxane (DVB/CAR/PDMS) SPME (Solid Phase Micro Extraction) fibers and subjected to GCMS. Volatiles were released from the SPME fiber by heating and entered a

DBwax GC column (0.25 mm i.d. × 30 m × 0.25 m film thickness, Agilent, Santa Rosa, California, USA). The samples were fractionated on the column by a temperature profile of 60°C held for 2 min, ramped to 230°C at 5.0°C/min, and then held for 9 min with helium (2ml/min) as the carrier gas.

2.3 Software

The GCMS data were exported to AIA/netCDF (network Common Data Format) format [14] and imported into MATLAB (The Mathworks, Inc., Natick, MA) using freely available tools, MEXNC and SNCTOOLS [14]. All data processing and analysis programs were written in-house using MATLAB version 7.0.4.365, Release 14, Service Pack 2.

3 Data Analysis

3.1 Data Preparation

3.1.1 Peak Tables—In this paper we use peak table methods [15–16], which aim to determine the presence or absence of analytes in each sample, together with their intensities. A peak table is a matrix whose rows represent samples, and whose columns each uniquely correspond to one detected compound (or variable) present in one or usually more samples, and whose elements represent integrated chromatographic intensity over all masses detected above a predefined threshold for this peak in each sample.

Because there are a large number of samples consisting of many peaks, manual approaches of visually identifying peaks in each sample and then determining whether peaks are common to more than one sample via their elution time and spectra are impracticable because of the size of the problem. We use the method described in [16] to produce the peak table, over a study consisting of 721 mouse urine samples. In order to account for variability in retention behaviour and mass spectral response, peaks that eluted within 37.5 and had a mass spectral similarity of 0.90 or more were deemed to have a common origin. Peak detection was performed on all 721 samples so that peaks in different subsets of the data can be matched. The peak detection algorithm detected initially 13,581 potential unique variables (ideally corresponding to unique compounds) over the entire dataset of mouse urine samples. In this paper, only 441 mouse urine samples out of 721 samples were retained to produce a peak table for the three datasets; the remaining samples were used for another study (genetics) but not reported below. Most variables corresponding to compounds that arise from the analytical process (e.g. SPME fibres and vial septa) include siloxanes (with characteristic m/z values of 147, 207, 221, 281, 327 and 341), involving 124 compounds, were removed. The final step is to develop local subsets of this peak table for each of the local models (corresponding to specific dataset). Variables that occur in less than 5 samples for each of the three datasets were removed because they are not likely to be significant markers from groups of samples. Whereas the rejection threshold of 5 samples may be adjected adjustable, it is clear that peaks detected in only 1 or 2 samples are not going to be valuable markers and probably these are artefacts of the analytical procedure, environment or peak detection algorithm; for example for the dataset consisting of 192 samples used to study the effect of age on the mouse urinary chemosignal, 12,481 of the original peaks are found in less than 5 out of the 192 samples, and a further 61 are attributed to siloxanes, so only 1039 are retained. The rejected peaks represent 5830 detections over all the 192 samples, whereas those retained represent 28861 detections, or around 80% of the detections over these samples. Hence our threshold we feel is a reasonable compromise between retaining the majority of peaks detected in each chromatogram, whilst removing rare ones. This creates data matrices for each experiment of sizes as shown in Table 2, to give a local datamatrix $\boldsymbol{Z_e}$ for each experiment e of dimensions $M_e \times N_e$ as indicated.

3.1.2 Preprocessing—The next stage is to prepare the data for pattern recognition. In this paper we report results using only one approach for simplicity. The rationale has been described previously in more detail [16,17] involving the following steps applied to the reduced peak tables based on the data of Table 2.

1. Peak areas were square root scaled.
2. The square root of all peak areas is summed to a total of 1 in each chromatogram, often called normalisation.
3. Finally each variable in the peak table was standardised over all samples in each local peak table. Where the data are separated into test and training sets (see Section 3.2.3) or the bootstrap (Section 3.2.2) this is performed on the coresponding training set samples as appropriate; the test set are scaled according to parameters obtained from the training set. Standardisation is a very important step for the data of this paper, as some compounds might be abundant in all samples, but their variation is not very interesting or significant [18].

The result of these steps is to obtain a datamatrix $\boldsymbol{X_e}$ for each experiment containing square root, normalised and standardised local data matrices, each consisting of a peak table for one of the three datasets.

3.2 Partial Least Squares Discriminant Analysis

3.2.1 Model—Partial Least Squares (PLS) [1–5] is one of the most common regression and supervised linear modelling techniques in the field of chemometrics in general, and spectroscopy and chromatography in particular [19–21]. PLS can be employed to find significant variables and/or to create empirical models from multivariate datasets

In this section, we discuss PLS in the context of class prediction [10] often called PLS-DA (Discriminant Analysis). The equations are given by

$$\mathbf{X}=\mathbf{TP}+\mathbf{E} \text{ and } \mathbf{c}=\mathbf{Tq}+\mathbf{f}$$

In this study, the value of elements of the $\boldsymbol{c}$ vector are set to +1 (the "in-group") or −1 (the "out-group") according to which class a sample belongs to and so contains the values +/−1; $\boldsymbol{X}$ is the experimental data matrix. $\boldsymbol{T}$ are the scores (in PLS these are common to both "X" block corresponding to the peak tables and "c" block corresponding to the classifier), $\boldsymbol{P}$ and $\boldsymbol{q}$ correspond to the loadings of the two blocks, and $\boldsymbol{E}$ and $\boldsymbol{f}$ contain the residuals. In this paper, we aim to obtain an optimal model for the descriptors for the group represented by +1 in the $\boldsymbol{c}$ vector, so for example are asking how best can we determine whether a mouse is on a diet rather than ask how best can we model both the control and dieting mice: the aim is to maximise the number of true positives. It is important to recognize that models could be optimised for different criteria. For example if we model mice on a diet and controls, there are three choices (1) to predict best whether a mouse is on a diet (to maximise true positives) (2) to predict best whether a mouse is not on a diet (to maximise true negatives) and (3) to predict best all the samples together. Option (3) is the most appropriate criterion for optimisation and validation if there are no specific overriding factors to choose options 1 and 2 (which is sometimes the case in forensic or medical diagnosis) which we use in this paper. However although the optimisation of the model is performed on both groups, we discuss primarily the results for the "in-group" this paper.

3.2.2 Number of components—The next step involves determining the number of PLS components. In this paper, we use the bootstrap [10–12] and try to optimise the %CC (correctly

classified) for the “in-group”. A sample is classified as belonging to the “in-group” if the predicted value of *c* is positive; otherwise it is assigned to the “out-group”. It is not necessary to employ a cut-off threshold of 0, and the optimal threshold can be selected using ROC curves [22] but in our case we use equal sample sizes for the training set for both “in-group” and “out-group”, even if the total number of samples for each group differs. This is a good compromise to optimise the model for this cut-off value.

In other studies, cross-validation methods [9] can be used as an alternative especially leave-one-out cross-validation. However, we find that the bootstrap [10–12], in which more samples are left out simultaneously and the calculation is repeated many times, provides more stable models. The bootstrap is more computationally intensive compared to cross-validation, but with new and more powerful computers available on desktops, the bootstrap method is now feasible to be applied large datasets in real time. The bootstrap was performed 200 times on a training set (Section 3.2.3), each time a different model is obtained, with different number of significant components. The training set samples were split into bootstrap training set and bootstrap test samples. In the bootstrap, the bootstrap training set was created by selecting samples which are equal to the number of samples in the training set randomly with replacement, so several samples are selected more than once and some samples not selected at all. Then, the bootstrap test set was formed by all samples which are not selected for the bootstrap training set and is not of a fixed size. This bootstrap procedure was performed with 200 repetitions. In each case, a PLS model was made from the bootstrap training set, and then used to predict the bootstrap test set to find the number of significant PLS components: the numbers of components which give the highest overall %CC on the bootstrap test set were determined in each repetition. The number of PLS components chosen with the most frequency in the bootstrap was used for building the overall PLS model for the full training set.

Note that as discussed in Section 3.2.3, the training set is generated 100 times, and each time there may be a different answer for the number of significant PLS components, according to the bootstrap.

3.2.3 Validation—Validation is used to estimate the performance of the classifier by dividing the dataset into training set and test set. An optimised PLS model is constructed from the training set to predict the test set. The strategy is as follows. The dataset of *M* samples was split into a training set as follows. Two thirds of the samples of the “in-group” were randomly selected and the same number of samples was randomly selected from the “out-group” so the training sets for both classes also had equal numbers of samples; the remaining samples were in the test set; this procedure allows for unequal numbers of samples in each group, but ensures that the training set is always balanced.

The training and test sets were created 100 times using different random splits of the data, each time constructing different models for the training set, resulting in 20,000 models overall, involving 200 bootstrap iterations for each of the 100 test and training set splits, called a procedure. The methodology is illustrated in Figure 1.

In a procedure, the higher the number of iterations and repetitions, the more stable the model appears. However if the number of iterations and repetitions is too high, calculation times will be too long. To determine how many iterations/repetitions are required it is necessary to see how much deviation there is in the predictive ability if an entire procedure is repeated, if this is unacceptable more iterations/repetitions are required and the optimal number of iterations and repetitions depends on size of data matrix, complexity and class separation.

3.3 Significant variables

A common problem in many areas of modern chemometrics is that there is often a very large number of variables in a dataset but the most of them are irrelevant for the particular classification problem. Typically several thousand or more unique variables can be identified in each biological dataset [23–24]. Many variables may originate from the analytical background or factors that are irrelevant to the problem in hand: for example there will be numerous factors that influence the occurrence of secondary metabolites in urine samples, but only a small number of these metabolites may be relevant to a specific experiment. Therefore, it is an important task to search for a small combination of variables that are necessary to model a specific factor or classifier.

There are several approaches to determining the most significant variables in supervised classification models [25–26]. The method employed in this paper is based on the calculation of the PLS regression coefficient vector (***b***) [27]. In its simplest form, the regression model specifies the relationship of data matrix and classifier is expressed by

$$\mathbf{c}=\mathbf{Xb}+\mathbf{f}=\mathbf{Tq}+\mathbf{f}$$

where ***b*** is a regression coefficient vector, estimated as follows

$$\mathbf{b}=\mathbf{Hq}$$

where ***H*** is the PLS weight matrix. The number of PLS components are determined using bootstrap.

The magnitude the coefficients of ***b*** can be used to determine which variables are most influential in the model[10]. Each variable *n* has a corresponding coefficient b_n : the higher its magnitude, the more likely it is to be a good marker. Variables are ranked from 1 (best) to *N* (worst) according to the magnitude of b_n. The sign of the regression coefficient can also be employed to determine which group the variable is a marker for.

Variable ranking can either be performed on the overall dataset (for a specific experiment $\boldsymbol{X_e}$) or on the training set. Selecting variables for a combined test and training set can result in overoptimistic predictions[8]. However an advantage of this is that the selection of variables is stable as there is just one overall set of variables for the entire dataset. Alternatively, variable selection can also be done on the training set, but if there are several training/test set splits, different variables may be selected as significant each time. In this paper we select variables only from the training set, so for each iteration there may be different subsets of variables selected, but describe in Section 3.5 how this criterion can be employed to provide an overall view of which variables are important for a specific dataset.

3.4 Assessment of Performance

Samples are assigned to the “in-group” if the predicted value of *c*, obtained using predictive PLS-DA (using the equation $\boldsymbol{c} = \boldsymbol{X}\,\boldsymbol{b}$) after forming a model on the training set, is positive and to the “out-group” otherwise. It is important to realize that the model has been already been optimised for both groups and for a cut-off decision threshold of 0.

3.4.1 Percent correctly classified—Three values for the %CC can be obtained.

a. Autoprediction on the training set using the optimal number of components as determined by the bootstrap criterion. This is an average for each of the 100 iterations. This value is an indicator as to how well the model is optimised.

b. Bootstrap on the bootstrap test set being an average of 200 bootstrap repetitions and 100 iterations. Samples are not always chosen for the bootstrap test set, but when they are they contribute to the %CC. There will be 20,000 such evaluations, and the bootstrap %CC is an average of these.

c. Test set from 100 iterations. This is an average from 100 iterations. Each sample will be included on average one time in three for each iteration, so is an average of approximately 33 predictions per sample. The higher the test set %CC, the better the model.

The number of PLS components is chosen via the bootstrap criterion as described in Section 3.2.2. A high %CC for autoprediction and high %CC for bootstrap test set suggests that the model is well optimised with correct number of significant PLS components. In case of test set, high %CC suggests that the model is appropriate for the study. It is possible to obtain a high autopredictive or bootstrap test set %CC and low test set %CC which indicates that the model is overfitted but well optimised.

3.4.2 Area Under the Curve—The PLS-DA algorithm as implemented above uses a decision threshold of predicted *c* equal to 0. It is possible also to investigate the performance of this algorithm as this threshold is varied [22].

Given any threshold, a contingency table (or confusion matrix) can be calculated, using any of the three types of %CC as described in Section 3.4.1. In case of the “in-group”, if the value of predicted *c* is lower than a specific threshold, this is counted as a FP (false positive). A FN (false negative) results in a sample having a value over the threshold (so predicted as belonging to the “in-group”) but, in fact, belonging to the “out-group”. As the threshold is reduced, the number of samples predicted as belonging to the “in-group” increases, so the number of TPs (true positives) and FPs rise, whilst the number of TNs (true negatives) and FNs reduces. The “in-group” %CC is equivalent to %TP and “out-group” %CC is equivalent to %TN. A ROC (Receiver Operator Characteristic) [11] curve can be produced by plotting the proportion of TPs against the proportion of FPs as the threshold changes.

Although the ROC curve gives a good visual summary of the performance of model, it is difficult to compare two ROC curves visually. A common way to summarize the quality of a ROC curve in a single number is to calculate the area under the ROC curve (AUC) [28] as follows.

$$\mathrm{AUC}=\sum_{i=1}^{I}\left\{\mathrm{TP_i}(\mathrm{FP_i}-\mathrm{FP_{i-1}})+\frac{1}{2}(\mathrm{TP_i}-\mathrm{TP_{i-1}})\mathrm{FP_i}\right\}$$

where *I* is the number of thresholds; TP_i and FP_i is the number of TP and FP at threshold *i*, respectively. A perfect ROC curve, in which a single threshold can be found that perfectly classifies all samples, has an AUC of 1. The higher the value of AUC, the better the classifier. The average AUC of autoprediction and test set criteria is calculated from 100 training set and test set splits, while the average bootstrap AUC is calculated from 20,000 bootstrap iterations.

3.5 Cost/Benefit Analysis

Of many thousand possible variables, only a few may be useful for forming a specific model. Using the methods of Section 3.3, variables can be ranked according to their significance. Models can then be constructed as the number of variables is increased, for example a model can be formed on the top 10 ranked variables, and performance than can be assessed by one or more of the indicators described in Section 3.4. The number of variables can be increased and the model performance is assessed on the new subset of variables and so on. This procedure can be called forward selection. Ideally the model performance should increase until an optimal number of variables is found, and then decrease afterwards. However sometimes the improvement of the model is not very great as the number of variables is increased above a certain number. If variables are expensive to measure (e.g. by constructing specific sensors), the cost of, for example, using 30 rather than 5 variables could be six fold increase, yet the benefit of including these additional variables may be limited, perhaps just by a few percent. So it is important not only to select the optimal number of variables, but also determine the benefit of including them.

For every training/test set split, variables were ranked according to PLS regression coefficients, using the optimal number of PLS components as determined by the bootstrap.. In this paper, the models were formed with increasing number of variables: the top variables were selected by increasing the number of variables (ordered in rank according to the size of the PLS Regression Coefficient) one at a time from the training set only as discussed above [8].

The procedure of full predictive modeling using these top variables is as follows.

1. The dataset of *M* samples was split into a training set and test set. The numbers of samples picked for the training set of each class are equal. The criterion to make training set of both classes equal in size was described in Section 3.2.3.
2. The top variables were selected from the training set for each iteration. Note that for each of the 100 training and test set iterations variables have slightly different ranks because the training set is different each time. The variables were ranked according to their PLS regression coefficients, ***b***. The predictive model was built from the training set using only the top variables. The optimal number of PLS components in the model was determined as described in Section 3.2.2.
3. In the validation step, the variables in the test set were reduced to the top variables selected from the training set in step 2. The predictive model from step 2 was used to predict the test set using the reduced subset of variables.

A separate issue involves selecting those variables that appear most diagnostic. The three steps above allow us to determine the influence of including more variables, but because the training set differs in each iteration, they may vary slightly according to which samples are included. To obtain a list of the best candidate marker variables, the variables selected in each iteration were scored according to their rank, the lower rank, the higher the score. The average score for each variable was calculated over 100 iterations. Variables with a high average score are regarded as potential markers.

The %CC and AUC was calculated to show the performance of model and can be plotted against the number of variables, to determine the behavior as the number of variables is increased.

In addition, the percentage change in prediction as the number of variables is increased from *l* to *l*+*v* can be computed. The relative improvement can be determined by the reduction in the classification error so, for example, a change from 80% CC to 85% CC is an

improvement of 5/20 as predictive error is reduced from 20% to 15%, involving a 25% improvement in performance. However as predictions become close to 100%, very small changes in predictive power (which may be due to one or two samples randomly being predicted correctly over several iterations) could have a correspondingly large influence over the calculated improvement, so an offset is used in the calculation

$$\%\text{improvement}_{l,v} = \frac{a_{l+v} - a_l}{(100 - a_l) + (\text{offset}) \times (100 - \min(a_{1..l}))}$$

a_l is the test set %CC for the "in-group" using the best l variables - in this paper, the offset was set to 1, and $\min(a_{1..l})$ is the lowest %CC (maximum misclassification error) when there are l variables or less.

4 Results and Discussion

4.1 Prediction using full variables

In this paper we use the bootstrap rather than cross-validation to determine the number of significant components, as the model is more stable. In order to demonstrate this we perform both leave-one-out cross-validation and bootstrap each time the dataset is split into training and test sets (100 iterations) and present graphs of the number of significant components found using each of these approaches in Figure 2. It can be seen that the bootstrap estimate is more stable for datasets 1 and 3 and of approximate similar stability for dataset 2.

The importance of performing several training and test set iterations on the stability of the estimate of %CC for the test set is illustrated in Figure 3. In this figure we iteratively reform training and test set splits a given number of times, and then repeat this calculation 20 times for different number of iterations to provide a mean and standard deviation for the test set %CC. The standard deviation of the overall test set %CC is 0.12%, 0.46% and 0.42% after 100 iterations for datasets 1 to 3 respectively, but 0.97%, 5.72% and 3.46% if there is only a single training test set split, suggesting that several training test set splits are required to obtain a stable estimate of %CC.

Table 3 shows the %CC and AUC for each model built using the full set of N_e variables for each dataset using the optimal number of PLS components as appropriate. In this paper, the "in-group" %CC and "out-group" %CC are equivalent to %TP and %TN, respectively. The test set values are for the overall test set using a training set model where the number of PLS components for the training set are determined using the bootstrap on the training set as illustrated in Figure 1. hence there is also a bootstrap test set %CC as well as a test set %CC.

In order to determine the quality of prediction, the average %CC was computed for all three criteria, as presented in Table 3. It can be seen the autoprediction %CC of all three models was around 100%. Note that if there is a high variable to sample ratio it is almost always possible to have very high autopredictive %CCs on any dataset [8]: note that under such circumstances the autopredictive %CC has little meaning as to how well the model can predict the data but is an indication of how well the model is optimised. For dataset 1, the "in-group" %CC for the bootstrap is 99.33% and for the test set is 95.13%, for dataset 2, the "in-group" %CC for the bootstrap is 96.15% and for the test set is 87.52% and for dataset 3, the "in-group" %CC for the bootstrap is 94.40% and for the test set is 88.30%. From Table 3, it can be seen there is a good balance between the predictions of both classes in the each of the 3 datasets: this suggests that the models are not biased towards either group and that the threshold of 0 has been well chosen. If the threshold is inappropriate often there are big

differences between the predictive quality of models for each class. Values for AUC also confirm the trends obtained using %CC as the indicator of success.

4.2 Cost/Benefit analysis

Classification accuracy (%CC and AUC) using the maximum number of variables is quite high for all three models. The prediction results suggest there are some discriminatory compounds which are significant markers for all three datasets. In the calculations of Section 4.1 around 1,000 variables were used to form models for each dataset. However many of these variables may be influenced by other factors that are not related to the problem of interest, as there are likely to be a large number of factors that influence the chemosignal, of which the ones studied only are responsible for a small fraction of the peaks detected in GCMS. It is possible to select a small subset of variables which appear to be the best discriminatory variables, as described in Section 3.5, which is performed by ranking the variables in order of significant using PLS-DA regression coefficients. It may be anticipated that only a small number of variables are necessary, and including more degrades the predictive power as assessed by the test set criterion. Note that only by using proper independent test sets is it possible to assess the influence of changing the number of variables, with autopredictive models including extra variables will almost always improve the model [8]. Cost/Benefit analysis was used to determine the benefit of including extra variables and search for the set of variables that gave the highest predictive ability.

The set of variables suggested as good markers in the models was selected using the procedure described in Section 3.5. The variable ranked 1 is the most significant variable using PLS-DA regression coefficients and is selected as the best marker in the model. The “in-group” %CC, AUC and %improvement of the age, stress and diet models of only the test set when the number of variables were changed to use to build models are shown in Figure 4. Initially the %CC and AUC increase when additional variables are added until the maximum is reached and the models either stabilise or the indicator of success decreases when more variables are included in the model. The optimal points for the datasets 2 and 3 are 10 and 8 variables respectively. For dataset l, %CC and AUC increase again when more than 100 variables were added and the optimal point cannot be determined directly from %CC and AUC. Instead, the %improvement was calculated and shown in Figure 4c. From this it can be determined that %CC and AUC using 10 variables is the optimal point because after this point %improvement does not increase and the improvements in %CC and AUC are quite stable.

The number of significant PLS components, average %CC (both of the “in-group” and “out-group”) and AUC for each model built from only the optimal number of variables and a detection threshold of 0 are shown in Table 4.

At the optimal point, it can be seen that the autoprediction %CC for the “in-group” is reduced to 99.91%, 96.14% and 95.44 % for the three datasets. Because the number of variables has reduced, lower autopredictive %CCs are expected, however the autopredictive %CC has very limited significance, if variable to sample ratios are high this is almost always high. For dataset 1, the “in-group” %CC is 99.06% for the bootstrap and 97.44% for the test set: the predictive ability using only the best variables is not very different from the model using all the variables. In the case of the dataset 2, the “in-group” %CC of the bootstrap is 96.47% and for the test set is 92.05% for the test set, an increase of around 10% over the model of Table 3. The model using all variables was dominated by the extra variables influenced by uninteresting or confounding factors because when these additional variables were eliminated from model, the predictive ability increases. For dataset 3, the “in-group” %CC is 92.39% for the bootstrap and 84.62% for the test set. For all three models, the gap between autoprediction and test set predictive ability is smaller than when all variables are

included. In addition when the optimum subset of variables is used, the gap between the predictive ability of the in group and out group increases, suggesting a more selective model.

Similar trends can be made when examining the AUC.

4.3 Markers

One major dilemma is that each time the data is split into test and training set, different models will be formed. Whereas this approach is essential for proper validation of the data, it poses problems in determining which variables are the best potential markers, as different compounds may be selected in each iteration. Therefore some extra steps are required to determine which markers are most significant, as these are likely to be selected most frequently.

From cost/benefit analysis, the optimal number of variables defined in datasets 1, 2 and 3 are 10, 10 and 8, respectively. Bar charts are presented in Figure 5 to show how many times the variables in the age, stress and diet models were found in the optimal subset of variables for each of the 100 iterations. So for datasets 1 and 2 the bar represents how frequently each variable (from the full peak table) is found in the top 10 variables over all iterations, and for dataset 3 in the top 8. For dataset 1 only 36 of the variables are ever found in the top 10 whereas 75 appear in the top 10 for dataset 2 and 99 different variables in the top 8 for dataset 3. A variable with a frequency of 100% is always selected in all 100 models.

The most significant 3 variables with highest frequency of selection for each of the three datasets, including the intensity distribution and mass spectra are presented in Figure 6, with their tentative identities as determined using the NIST spectral database.

5 Conclusions

The methods in this paper can have general applicability to metabolomic studies. Despite the widespread interest in metabolomics and use of chemometrics approaches such as PLS-DA, there are still some fundamental dilemmas, especially in the area of pattern recognition and variable selection. If variables are selected just from the overall dataset it is possible to obtain single models using the variables that appear to be most discriminatory, however this may bias the test set predictions because test set samples are included in the variable selection and it can be shown[8] that if this procedure is followed, random data can falsely suggest that there appears to be discrimination between groups even if correct approaches for validation are performed using these variables. Therefore the correct approach is to select variables just from the training set, but this poses a dilemma that each time a training set is formed there may be different variables selected, as models are not reproducible and depend on which samples are included in the training set. We show that there can be considerable variability in predictions if only a single training set is employed, and recommend that around 100 training/test set splits are employed in order to obtain stable predictions, however this poses a dilemma as to which variables should be chosen for the overall model assessment.

We demonstrate several approaches for overcoming these dilemmas, whilst studying the benefit of increasing the number of variables. This allows both an optimal choice of variables without overoptimistic modelling predictions, and permits the study of what the benefit is of increasing the number of variables. In this paper we use PLS-DA regression coefficients for ranking variables, but other approaches [28] could be used as alternatives for ranking the variables.

With the growth of bioanalytical pattern recognition in clinical as well as biological studies there is an urgent need to develop new chemometric procedures for selecting variables which can take advantage of the rapid increase in computing power over the past decade.

Acknowledgments

We thank Dr. Sarah Dixon, Hejun Duan and Dong Li of the centre of Chemometrics for help in data organisation. This work was sponsored by ARO Contract DAAD19-03-1-0215. Opinions, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by the United States Government. This work is approved for public release, distribution unlimited.

References

1. Hoskuldsson A. J Chemom 1988;2:211–228.
2. Geladi P, Kowalski BR. Anal Chim Acta 1986;185:1–17.
3. Wold S, Geladi P, Esbensen K, Ohman J. J Chemom 1987;1:41–56.
4. Martens, H.; Martens, M. Multivariate Analysis of Quality. Wiley; Chichester: 2001.
5. Gasteiger, J., editor. Handbook of Chemoinformatics. 1. Wiley; Weinheim: 2003.
6. Bijlsma S, Bobeldijk I, Verheij ER, Ramaker R, Kochhar S, Macdonald IA, van Ommen B, Smilde AK. Anal Chem 2006;78:567–574. [PubMed: 16408941]
7. van der Greef J, Smilde AK. J Chemom 2005;19:376–386.
8. Brereton RG. Trends Anal Chem 2006;25:1103–1111.
9. Brereton, RG. Applied Chemometrics for Scientists. Wiley; Chichester: 2007.
10. Dixon SJ, Xu Y, Brereton RG, Soini HA, Novotny MV, Oberzaucher E, Grammer K, Penn DJ. Chemom Intell Lab Syst 2007;87:161–172.
11. Wehrens R, Putter H, Buydens LMC. Chemom Intell Lab Syst 2000;54:35–52.
12. Wehrens R, Van Der Linden WE. J Chemom 1997;11:157–171.
13. Dixon SJ, Heinrich N, Holmboe M, Schaefer ML, Reed RR, Trevejo J, Brereton RG. J Chemom 2009;23:19–31.
14. Sourceforge.net. http://mexcdf.sourceforge.net/
15. Jonsson P, Johansson AI, Gullberg J, Trygg J, Grung B, Marklund S, Sjostrom M, Antti H, Moritz T. Anal Chem 2005;77:5635–5642. [PubMed: 16131076]
16. Dixon SJ, Brereton RG, Soini HA, Novotny MV, Penn DJ. J Chemom 2006;20:325–340.
17. Xu Y, Gong F, Dixon SJ, Brereton RG, Soini HA, Novotny MV, Oberzaucher E, Grammer K, Penn DJ. Anal Chem 2007;79:5633–5641. [PubMed: 17602669]
18. Brereton, RG. Chemometrics: Data Analysis for the Laboratory and Chemical Plant. Wiley; Chichester: 2003.
19. Cirovic DA, Brereton RG, Walsh PT, Ellwood JA, Scobbie E. Analyst 1996;121:575–580.
20. Geladi P, Sethson B, Nystrom J, Lillhonga T, Lestander T, Burger J. Spectrochim Acta B 2004;59:1347–1357.
21. Brereton RG. Analyst 2000;125:2125–2154.
22. Brown CD, Davis HT. Chemom Intell Lab Syst 2006;80:24–38.
23. Jarvis RM, Goodacre R. Bioinformatics 2005;21:860–868. [PubMed: 15513990]
24. Xu L, Jiang JH, Wu HL, Shen GL, Yu RQ. Chemom Intell Lab Syst 2007;85:140–143.
25. Ramadan Z, Song XH, Hopke PK, Johnson MJ, Scow KM. Anal Chim Acta 2001;446:223–244.
26. Lima SLT, Mello C, Poppi RJ. Chemom Intell Lab Syst 2005;76:73–78.
27. Baldovin A, Wu W, Centner V, Jouan-Rimbaud D, Massart DL, Favretto L, Turello A. Analyst 1996;121:1603–1608.
28. Bradley AP. Pattern Recognition 1997;30:1145–1159.
29. Alsberg BK, Woodward AM, Winson MK, Rowland JJ, Kell DB. Anal Chim Acta 1998;368:29–44.

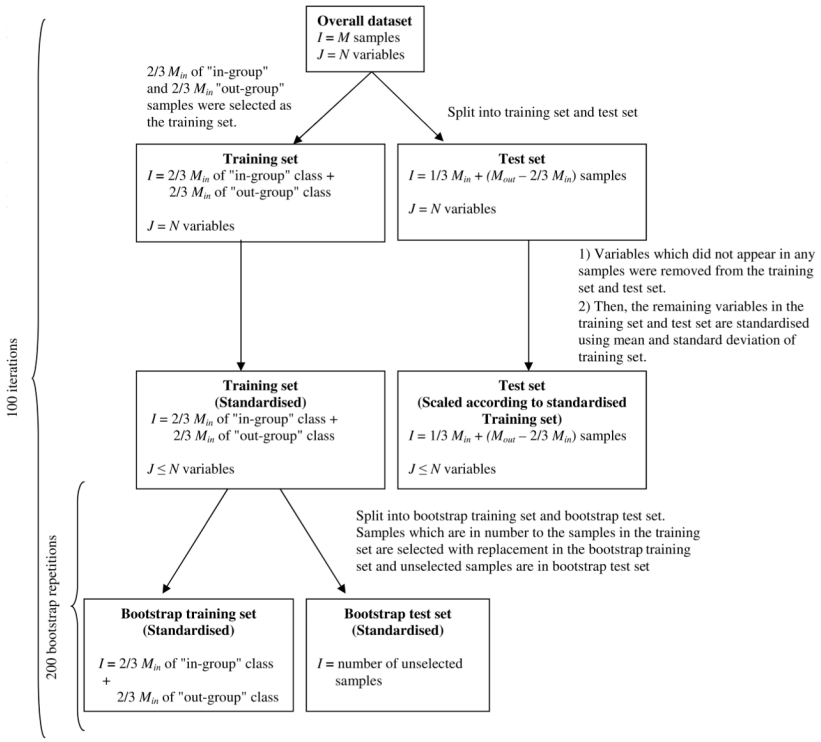


Figure 1.
Flowchart of the procedure to assess the predictive ability of a model. M_{in} and M_{out} are the number of samples in the “in-group” and “out-group”, respectively.

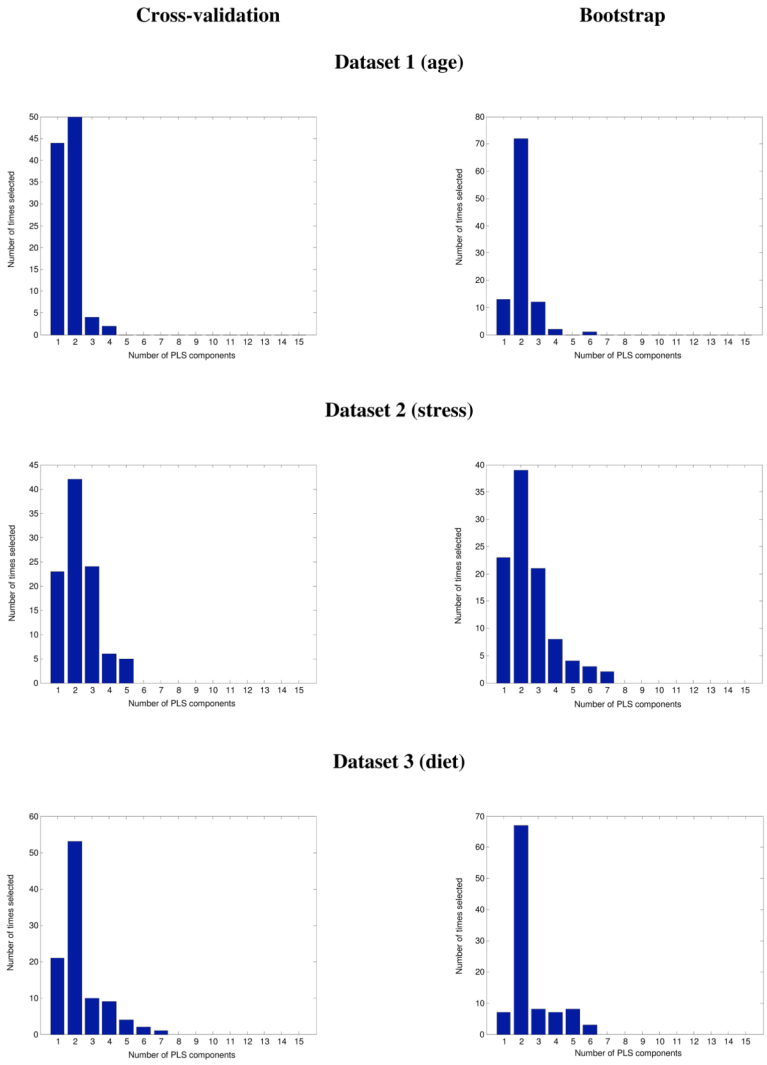


Figure 2.
Comparison of leave-one-out cross-validation and the bootstrap, for 100 training / test set splits, the optimal number of PLS components obtained using each iteration is presented.

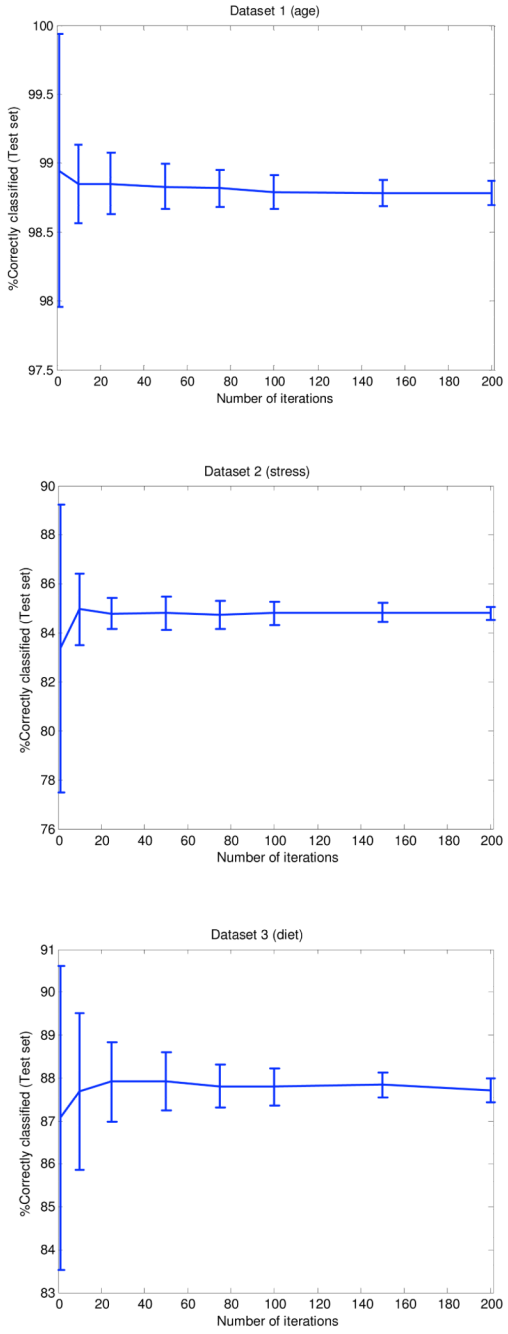
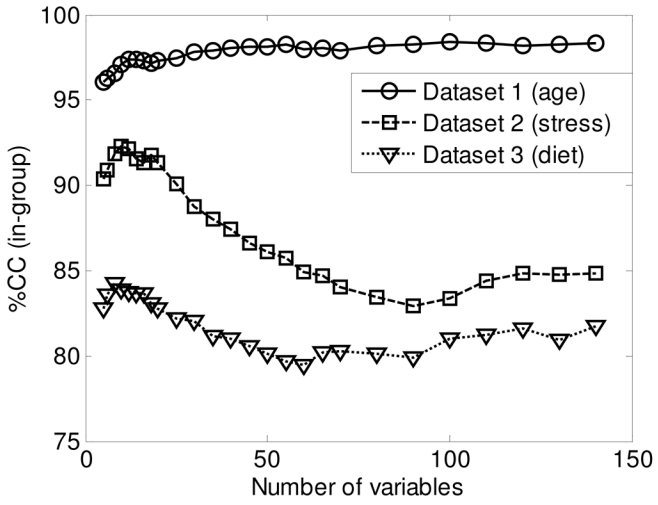
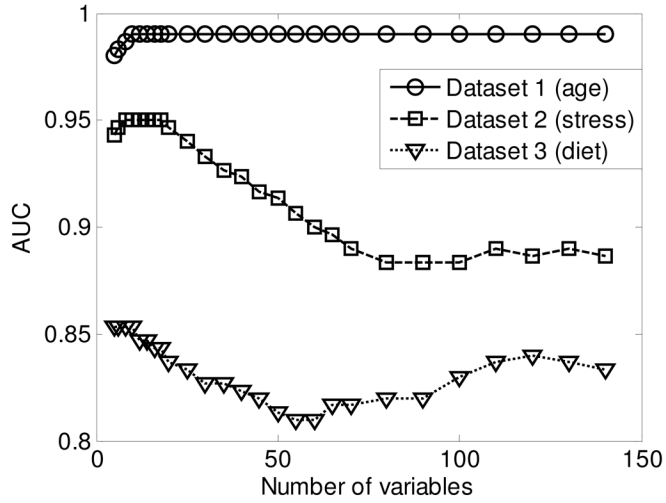


Figure 3.
Standard deviation and mean of %CC for test set using 20 repeat estimates of %CC, the bars represent the standard deviation.

a) %CC



b) AUC



c) % improvement

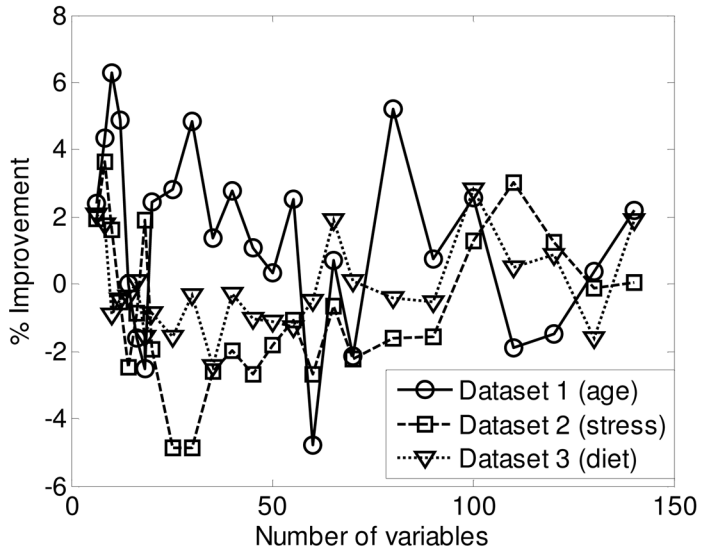
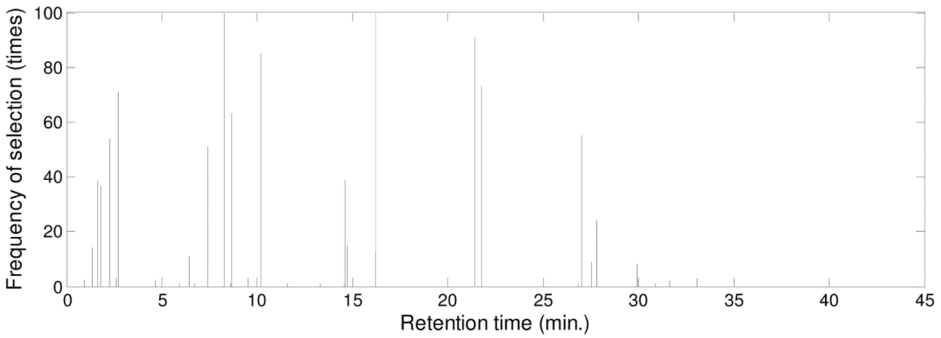
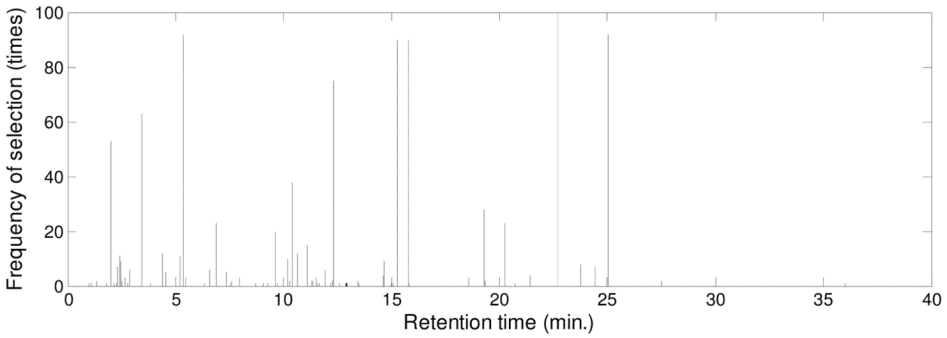


Figure 4.
"in-group" %CC, AUC and %improvement of the test set criterion including % improvement change with number of variables used to build model. (a) "in-group" %CC (b) AUC and (c) %improvement

(a) Dataset 1 (age)



(b) Dataset 2 (stress)



(c) Dataset 3 (diet)

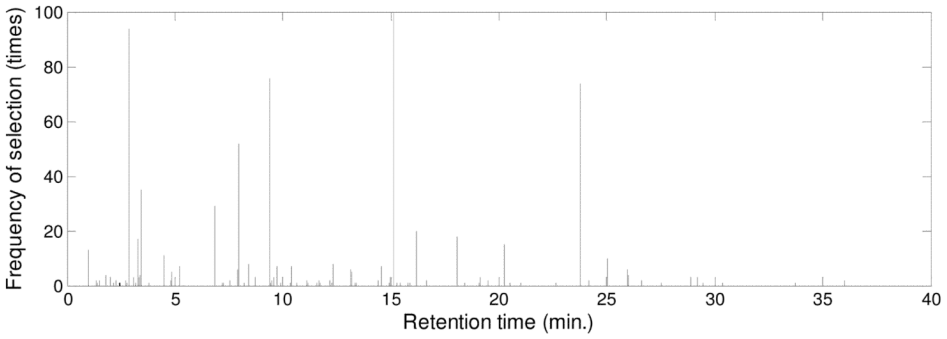
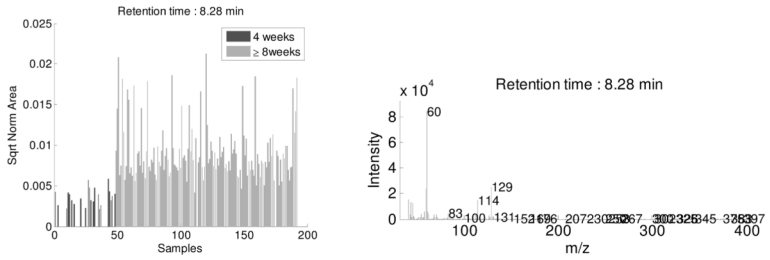


Figure 5.

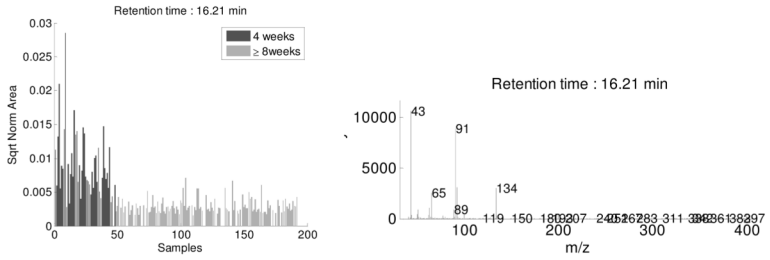
Bar charts showing the number times each variable (compound) represented by retention time (min) were found in all the test and training set split (100 iterations) for a) dataset 1 (age), b) dataset 2 (stress) and c) dataset 3 (diet) in the top 10, 10 and 8 variables for each dataset respectively.

(a) Dataset 1 (age)

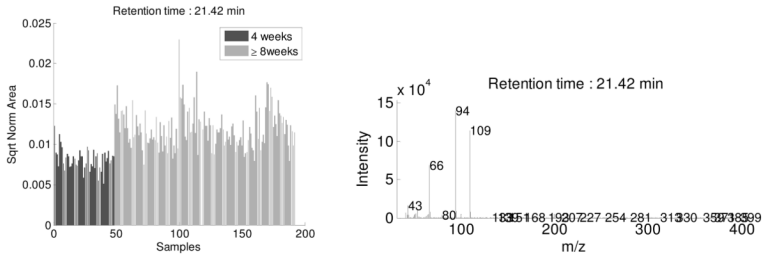
2-isopropyl-4,5-dihydrothiazole(IPT)



Benzyl methyl ketone

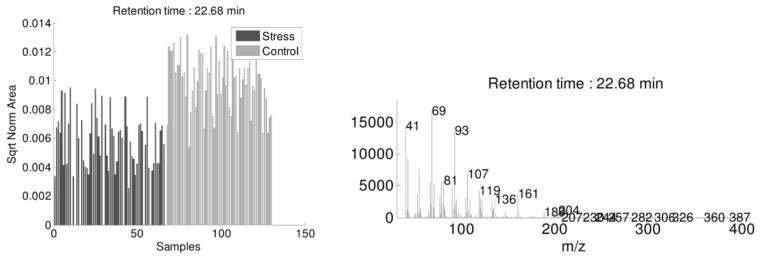


1-(1Hpyrrol-2-yl)-(2-acetyl-pyrrole)

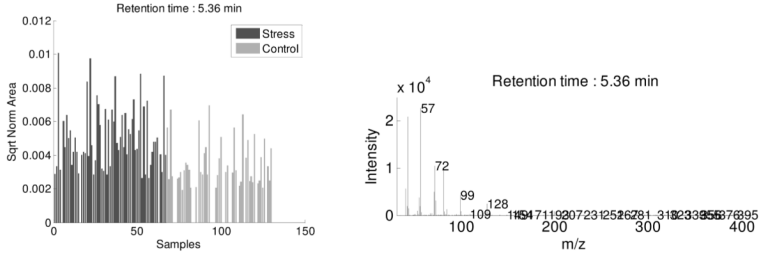


(b) Dataset 2 (stress)

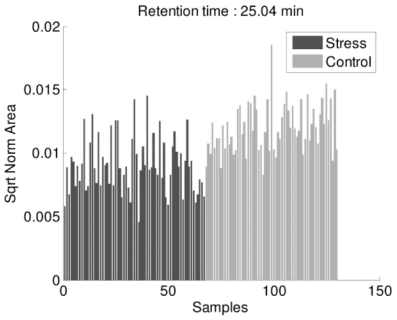
Nerolidol

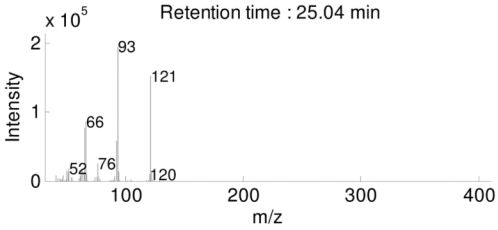


6-methyl-3-heptanone



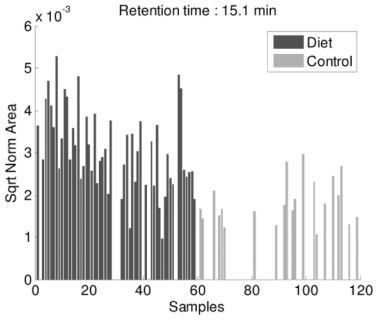
N-phenyl formamide

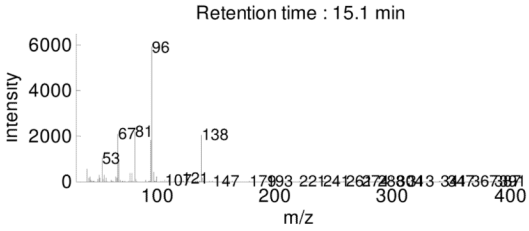




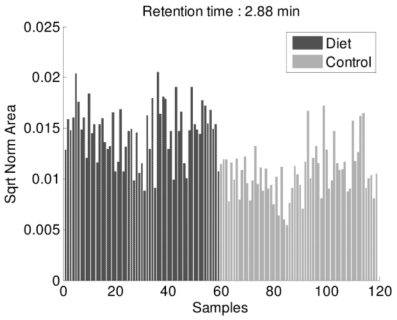
(c) Dataset 3 (diet)

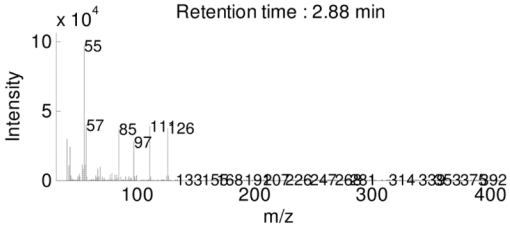
4,4,5-trimethyl-2-cyclohexanone





Dihydrofuran (DHF)





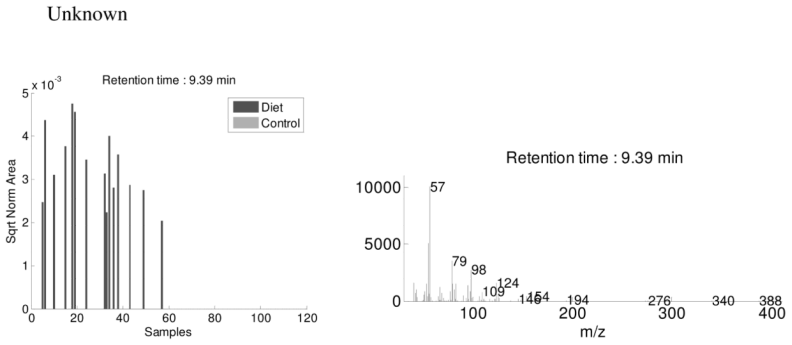


Figure 6.

Square root and row scaled areas and mass spectra of the top three candidate markers for (a) dataset 1 (age), (b) dataset 2 (stress) and (c) dataset 3 (diet) together with their tentative identities.

Table 1

The details of samples: (a) Age, (b) Stress and (c) Diet

	(a) Dataset 1 : Age study													
	Individual mouse (#Tag)	**526**	**532**	**533**	**534**	**539**	**568**	**569**	**617**	**620**	**622**	**Total samples**	**Samples in group**	
	4 weeks													
	Morning (am.)	3	2	3	2	4	2	3	2	0	2	23	**48**	**Young mice (“in-group”)**
	Afternoon (pm.)	2	3	2	3	1	2	2	4	4	2	25		
	8 weeks													
	Morning (am.)	-	-	-	-	-	3	2	3	2	3	13		
	Afternoon (pm.)	-	-	-	-	-	2	3	2	3	2	12		
	12 weeks													
	Morning (am.)	-	-	-	-	-	1	3	2	2	2	10		
	Afternoon (pm.)	-	-	-	-	-	2	2	3	3	3	13		
	15 weeks													
Time of Sampling	Morning (am.)	2	3	3	2	3	-	-	-	-	-	13		
	Afternoon (pm.)	3	2	2	3	2	-	-	-	-	-	12		
													144	**Old mice (“out-group”)**
	20 weeks													
	Morning (am.)	2	1	2	2	3	-	-	-	-	-	10		
	Afternoon (pm.)	3	3	3	4	1	-	-	-	-	-	14		
	26 weeks													
	Morning (am.)	3	2	3	2	3	-	-	-	-	-	13		
	Afternoon (pm.)	2	2	2	2	2	-	-	-	-	-	10		
	30 weeks													
	Morning (am.)	2	3	2	3	1	-	-	-	-	-	11		
	Afternoon (pm.)	3	2	3	2	3	-	-	-	-	-	13		

(b) Dataset 2 : Stress study

		Stress mice ("in-group")											Controls ("out-group")								
	Individual mouse (#Tag)	**571**	**572**	**573**	**598**	**599**	**600**	**639**	**640**	**657**	**582**	**583**	**584**	**585**	**586**	**658**	**659**	**660**	**661**	**662**
Time of Sampling	Morning	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	Afternoon	7	6	8	6	7	5	10	9	9	6	7	6	6	7	6	6	7	6	6
	Total samples	7	6	8	6	7	5	10	9	9	6	7	6	6	7	6	6	7	6	6
	Group samples					67									63					

(c) Dataset 3 : Diet study

		Mice on High Fat Diet ("in-group")											Controls ("out-group")							
	Individual mouse (#Tag)	**641**	**642**	**643**	**644**	**645**	**646**	**647**	**648**	**649**	**650**	**587**	**588**	**589**	**590**	**591**	**593**	**594**	**595**	**596**
Time of Sampling	Morning	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	Afternoon	4	3	7	3	9	8	6	8	6	5	6	8	7	7	6	5	8	9	4
	Total samples	4	3	7	3	9	8	6	8	6	5	6	8	7	7	6	5	8	9	4
	Group samples					59									60					

Table 2

Size of local data matrices

Dataset (number (*e*) and type)	Number of samples (M_e)		Number of variables (N_e)
	"in-group"	"out-group"	
1 (age)	48	144	1039
2 (stress)	67	63	1056
3 (diet)	59	60	996

Table 3

Percent Correctly Classified (%CC) and Area Under Curve (AUC) for autoprediction, bootstrap test set and test sets using the maximum variable model.

					Autoprediction				Bootstrap test set				Test set			
					%CC			AUC	%CC			AUC	%CC			AUC
Dataset	**“in-group”**	**“out-group”**	**No of variables**		**“in-group”**	**“out-group”**	**overall**		**“in-group”**	**“out-group”**	**overall**		**“in-group”**	**“out-group”**	**overall**	
1 (age)	Young (4 weeks)	Old (≥8 weeks)	1039	**“in-group”**	100	0	100	1	99.33	0.08	99.52	0.99	95.13	0.97	98.62	0.98
				“out-group”	0	100			0.67	99.92			4.87	99.03		
2 (stress)	Stress	Control	1056	**“in-group”**	100	0	100	1	96.15	12.77	90.45	0.95	87.52	18.52	84.76	0.92
				“out-group”	0	100			3.85	87.23			12.48	81.48		
3 (diet)	Diet	Control	996	**“in-group”**	99.97	0	99.98	1	94.4	10.02	90.91	0.96	88.3	17.55	86.34	0.93
				“out-group”	0.03	100			5.6	89.98			11.7	82.45		

Table 4

Percent Correctly Classified (%CC) and Area Under Curve (AUC) for autoprediction, bootstrap test set and test sets using the optimum variable model.

					Autoprediction				Bootstrap test set				Test set			
					%CC			AUC	%CC			AUC	%CC			AUC
Dataset	**“in-group”**	**“out-group”**	**No of variables**		**“in-group”**	**“out-group”**	**overall**		**“in-group”**	**“out-group”**	**overall**		**“in-group”**	**“out-group”**	**overall**	
1 (age)	Young (4 weeks)	Old (≥8 weeks)	10	**“in-group”**	99.91	1.16	99.57	1	99.06	0.71	99.25	1	97.44	4.99	95.31	0.99
				“out-group”	0.09	98.84			0.94	99.29			2.56	95.01		
2 (stress)	Stress	Control	10	**“in-group”**	96.14	5.12	95.53	0.99	96.47	12.1	90.58	0.96	92.05	13.46	89.46	0.95
				“out-group”	3.86	94.88			3.53	87.9			7.95	86.54		
3 (diet)	Diet	Control	8	**“in-group”**	95.44	11.67	90.79	0.96	92.39	11.65	90.35	0.95	84.62	11.9	80.95	0.86
				“out-group”	4.56	88.33			7.61	88.35			15.38	78.1		

Anal Chem. Author manuscript; available in PMC 2010 July 27.