Available online at www.sciencedirect.com
ScienceDirect
Journal of Chromatography B, 866 (2008) 77–88
JOURNAL OF CHROMATOGRAPHY B
www.elsevier.com/locate/chromb

Review

Statistical data processing in clinical proteomics☆

Suzanne Smit*, Huub C.J. Hoefsloot, Age K. Smilde

Swammerdam Institute for Life Sciences, Universiteit van Amsterdam – Nieuwe Achtergracht 166, 1018 WV Amsterdam, The Netherlands

Received 20 July 2007; accepted 18 October 2007
Available online 4 November 2007

Abstract

This review discusses data analysis strategies for the discovery of biomarkers in clinical proteomics. Proteomics studies produce large amounts of data, characterized by few samples of which many variables are measured. A wealth of classification methods exists for extracting information from the data. Feature selection plays an important role in reducing the dimensionality of the data prior to classification and in discovering biomarker leads. The question which classification strategy works best is yet unanswered. Validation is a crucial step for biomarker leads towards clinical use. Here we only discuss statistical validation, recognizing that biological and clinical validation is of utmost importance. First, there is the need for validated model selection to develop a generalized classifier that predicts new samples correctly. A cross-validation loop that is wrapped around the model development procedure assesses the performance using unseen data. The significance of the model should be tested; we use permutations of the data for comparison with uninformative data. This procedure also tests the correctness of the performance validation. Preferably, a new set of samples is measured to test the classifier and rule out results specific for a machine, analyst, laboratory or the first set of samples. This is not yet standard practice. We present a modular framework that combines feature selection, classification, biomarker discovery and statistical validation; these data analysis aspects are all discussed in this review. The feature selection, classification and biomarker discovery modules can be incorporated or omitted to the preference of the researcher. The validation modules, however, should not be optional. In each module, the researcher can select from a wide range of methods, since there is not one unique way that leads to the correct model and proper validation. We discuss many possibilities for feature selection, classification and biomarker discovery. For validation we advice a combination of cross-validation and permutation testing, a validation strategy supported in the literature.

Keywords: Statistical validation; Permutation test; Classification; Biomarker discovery; Double cross-validation; Feature selection; Curse of dimensionality; Multivariate data analysis; Proteomics

Contents

1. Introduction 78
2. Feature selection 79
   2.1. Independent feature selection 79
   2.2. Wrappers 80
3. Classification methods 80
   3.1. Discriminant analysis 80
   3.2. Partial least squares 80
   3.3. Support vector machines 81
   3.4. Logistic regression 81
   3.5. Nearest shrunken centroids 81
   3.6. Artificial neural networks 81

☆ This paper is part of a Special Issue dedicated to the 50th anniversary of Journal of Chromatography.

* Corresponding author.
E-mail address: ssmit@science.uva.nl (S. Smit).

1570-0232/$ – see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.jchromb.2007.10.042

3.7. Classification trees 81
3.8. Ensemble classifiers 82
4. Biomarker candidate selection 82
5. Comparison studies 82
6. Statistical validation 83
6.1. Performance measures 83
6.2. Cross-validation for performance estimation 83
6.3. Cross-validation for metaparameters/feature selection 84
6.4. Double cross-validation for metaparameter selection and performance estimation 84
6.5. Permutation test 85
6.6. Strategies and applications 85
7. Proteomics data analysis: a framework 86
8. Black spots and open issues 86
8.1. External test set 86
8.2. Power calculations 86
8.3. Increasing complexity of data sets 86
9. Conclusions 86
Acknowledgement 87
References 87

1. Introduction

Modern developments in analytical techniques like mass spectrometry (MS) created the opportunity to measure protein concentrations on a large scale; this area of research is called proteomics. The hope is that proteomics studies can contribute to healthcare. In clinical proteomics thousands of proteins or peptides can be measured in a single experiment. This review describes how information is obtained from pre-processed clinical proteomics data and how to validate the information using statistical procedures. The clinical proteomics experiments that we discuss in this paper can be seen as a discovery tool for biomarkers. A possible workflow for biomarker discovery is given in Fig. 1. It starts with a biological question, which leads to a carefully designed experiment, sampling and measurements. Preprocessing of the data is necessary to remove instrumental noise and make the measurements of the samples comparable.

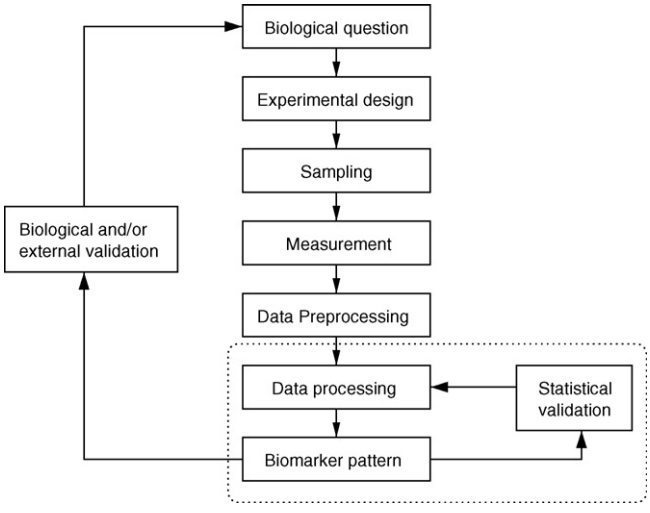

Fig. 1. Biomarker discovery workflow. From biological question to biomarker leads. The blocks data processing, biomarker pattern, and statistical validation form the subject of this review.

A preliminary answer to the biological question is obtained in the three blocks that are encircled in Fig. 1: data processing, biomarker pattern, and statistical validation. After the discovery of statistically valid biomarker leads, external testing and biological validation will show whether they truly answer the biological question.

Biomarkers can be used to predict the state of a patient, in diagnosis, to monitor the response to treatment, and to determine the stage of a disease. For diagnosis, but not essentially different for the other goals, samples from cases and controls are measured. The measurements are usually stored in a data matrix and class labels are stored in a response vector. Data analysis tools try to find the differences in measurements that predict the state of a patient. This information is preferably in just a few proteins (biomarkers) that are indicative for the biological state. Alternatively, the interplay of multivariate data can provide the desired information. Results should be subjected to validation: statistical as well as biological. The statistical validation should investigate the performance of the biomarker, as well as the relevance of the results. The biological validation is concerned with the question whether the biomarkers are involved in processes that can be related to the disease. If the result of both validation processes is satisfactory a putative biomarker is established. Many more steps have to be taken before this leads to an established biomarker [1].

MS is not the only technique used for proteomics investigations. Protein arrays and 2D gels also play an important role in the field [2]. However, when mining the literature on data analysis in clinical proteomics, most hits we encountered were on MS studies. Reviews on the application of MS in proteomics are available [3,4]; the current review does not discuss the many types of MS experiments. We restrict ourselves mainly to data analysis in single MS experiments (such as liquid chromatography–MS, matrix assisted laser desorption/ionisation MS and surface enhanced laser desorption/ionisation) although our conclusions also hold for other types of (omics) experiments.

In single MS experiments many different issues play a role. Among these are experimental design, selection of patients, sample handling, preprocessing of the spectra and biological validation [5–12]. In this review we are not taking up these issues but focus on classification methods for proteomics studies and the statistical validation tools that are used in combination with the classification methods.

Classification methods applied in proteomics are developed in different sciences, such as machine learning, chemometrics, data mining and statistics. A wide range of methods is available, with many different characteristics. We try to give an overview of the methods that are popular in proteomics.

The reason that validation in classification methods is an important and still open issue is mainly caused by the characteristics of a proteomics data set. Usually, a mass spectrum contains thousands of different mass/charge (m/z) ratios. The sample size, e.g. the number of patients, is relatively small. This results in a so-called high dimensionality small sample problem. This type of problem suffers from the curse of dimensionality [13], which means that the number of samples needed to accurately describe a (discrimination) problem increases exponentially with the number of dimensions (variables). In proteomics studies, the number of samples is usually low compared to the number of variables, due to the limited availability or the cost of measurements. This undersampling leads to the possibility of discovering a discriminating pattern between two populations, even when these two populations are statistically not distinct. Working with high dimensional data can easily lead to overfitting: the derived model is specific for the training data and does not perform well on new samples.

Literature provides several approaches to overcome these problems. One approach is to reduce the dimensionality of the data. This can be done before a classification is performed or it can be combined with a classifier. Other techniques to cope with high dimensional data are statistical validation strategies, such as cross-validation and permutation tests.

This review starts with an overview of the most encountered methods for classification and biomarker discovery in clinical proteomics. We present a framework in which most of the methods fall. And finally a strategy is put forward for a thorough statistical assessment of the entire data analysis procedure.

2. Feature selection

Feature selection plays an important role in clinical data analysis for three reasons. First, using all features in forming the classification rule in general does not give the best performance. Increasing the number of features from zero enhances performance to some point, after which adding more feature leads to a deteriorating performance, because many features are uninformative and they can conceal information in relevant features. This is called the peaking phenomenon [14–16]. The second reason is a technical one: some classification methods require the number of objects to be larger or equal to the number of features. Since proteomics data sets usually consist of far more features than samples, a selection has to be made before constructing the classification rule. Third, one of the goals of a proteomics study is to find leads for potential markers for disease. So the number of variables in the final model should be small to enhance the interpretability of the model. To this end, finding a good classifier is combined with selection of variables that discriminate well.

We distinguish different categories of feature selection methods. Filter methods and variable transformation reduce the number of features independent of a classification method (unsupervised), while wrappers select variables in concert with a classification method (supervised). Sometimes, feature selection is intrinsic to a classification method, for example in classification trees. Another category is variable selection after classification, where the information in the classification rule is used to find the most informative variables.

Filters, variable transformation and wrappers are discussed in this section, and the section Biomarker candidate selection describes variable selection intrinsic to classification and after classification. This division reflects that wrappers, filters and variable transformation are mostly used to deal with the peaking phenomenon and to solve the technical issues, while leads for biomarkers are often sought in the classification rule.

We realize that this is by no means a strict distinction. Wrapper [17] and filter [18] methods have also been used for biomarker selection, and vice versa: some intrinsic methods are used for pre-selection to provide input for other classification methods [19,20]. We would like to point out that statistical validation is of crucial importance in variable selection, as it is throughout the entire data analysis. In undersampled data sets, with fewer samples than variables, it may very well be possible to select a set of features that discriminate between cases and controls, but that turn out to be uninformative when new samples are classified. Thorough statistical validation can prevent overfitting, and we discuss it in the section Statistical validation.

2.1. Independent feature selection

Filter methods are applied to the pre-processed data before the construction of the classifier. Examples are significance tests such as the t-test, which compares differences in means between the case and the control groups. When the measurements for a variable differ significantly between the two groups, it is retained. The t-test assumes normality of the data. The Wilcoxon–Mann–Whitney test assesses differences between two groups without making this assumption.

These significance tests are designed to deal with univariate data, and a variable is considered to differ significantly when its test statistic is smaller than some value for α (generally, $\alpha = 0.05$ or $\alpha = 0.01$). Since proteomics data involves testing many individual variables simultaneously, applying the same value for α leads to many false positives [21]. The Bonferroni correction sets an α-value for the entire set, so that the test statistic for each individual variable is compared to a value of α/(number of variables) and the false positive rate or family wise error rate (FWER) is controlled.

A less conservative correction for multiple testing is controlling the false discovery rate (FDR): the number of false positives among all positives [22,23]. Significance analysis of microarrays

(SAM) uses a *t*-test with a threshold to select features. The false discovery rate is obtained by comparing the results with results in permutations [24].

Like filter methods, variable transformation is performed before classification. Projection methods reduce the dimensionality of the data in a multivariate approach. Principal component analysis (PCA) looks for linear combinations of the original variables that describe the largest amount of variation in the data [25]. The linear combinations (principal components) become new features that describe the data in a lower dimensional space.

2.2. Wrappers

Wrappers are feature selection methods that work in concert with a classification method. The classification method is used to test relevance of the variables. Variables that lead to good performance are selected. Forward selection starts with an empty set and selects the variable that gives the best classification result. Given this first variable, another variable is added that realizes the largest improvement of performance [13]. Variables are added until the performance does not improve or a set criterion is met. Backward elimination works similarly, starting with the full set of features and sequentially removing features from the set [13]. Genetic algorithms create many feature sets that are tested simultaneously for performance, given a classification method. The best sets are recombined to create a new generation of improved feature sets. The algorithm is stopped when the performance does not improve over several generations or when a preset performance measure is achieved [26].

3. Classification methods

3.1. Discriminant analysis

Discriminant analysis (DA) was first introduced by Fisher, who used it to discriminate between different Iris species [27]. In the feature space, a direction is sought that maximizes the differences between the classes with respect to the covariance within the control and case classes (Fig. 2). This direction, the discriminant vector, can be used to classify new samples. DA uses the covariance matrix to find the discriminant vector. Linear discriminant analysis (LDA) assumes the within-class covariance matrices to be equal, which leads to linear decision boundaries. When the covariance matrices are unequal, quadratic discriminant analysis (QDA) is applied. The decision boundary in QDA is quadratic.

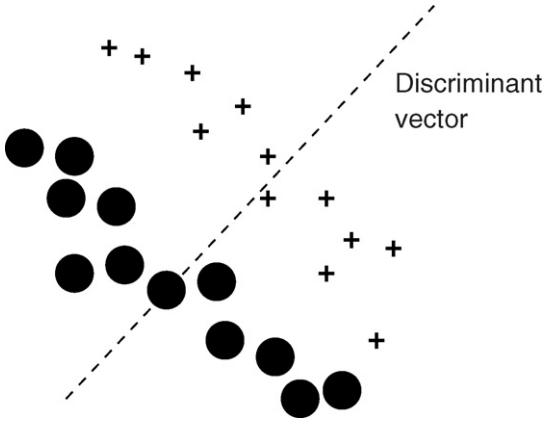

Fig. 2. Linear discriminant vector.

Usually, in proteomics data, undersampling causes the within-class covariance matrix to be singular, which makes it impossible to find the discriminant vector. This can be solved by filtering features [19] or by selecting features with a wrapper method as described in the previous section. Other solutions lie in adjusting the DA algorithm to repair the singularity of the covariance matrix. Regularized discriminant analysis (RDA) [28] shrinks the covariance matrix towards a multiple of the identity matrix [28]. In diagonal discriminant analysis the covariance matrix is assumed to be diagonal, setting all off-diagonal elements to zero (see for example [29]).

A popular variant of DA in omics studies is principal component discriminant analysis (PCDA) [30]. It solves the singularity by reducing the dimensionality of the data with PCA, after which DA is performed on the PCA scores.

PCDA has been used for omics data analysis under a variety of names. As uncorrelated discriminant analysis, Ye et al. used it for the analysis of several publicly available gene expression data sets [31]. The maximum number of principal components is used in the classifier. In a proteomics study of SELDI-TOF MS data concerning ovarian cancer and prostate cancer, Lilien et al. used the Q5 algorithm, also a combination of PCA and LDA to discriminate healthy from diseased [32]. Again, the maximum number of principal components is retained. The classification probability is calculated from the distance on the discriminant vector between the spectrum and the nearest class mean. Spectra with classification probabilities smaller than a threshold are not classified. Smit et al. applied PCDA to SELDI TOF MS measurements of serums to discriminate Gaucher from healthy samples [33]. The number of components was tuned with cross-validation, which showed that the maximum number of components does not always lead to the best model.

3.2. Partial least squares

Partial least squares (PLS) [34] is similar to PCA, but in extracting the new features, PLS also takes the covariance of the data with the response vector (vector of class labels) into account. PLS tries to find the relations between the data matrix and the vector of class labels, i.e. a latent variable approach to modelling the covariance structure of the data and the class labels. A PLS model will try to find the multidimensional direction in the space of the data matrix that explains the maximum variance in the class label space. When it is used for classification, it is referred to as partial least squares discriminant analysis (PLSDA) [35].

PLSDA is a much used method in metabolomics studies. It has for example been applied in a human metabolomics study into obesity to differentiate between obese and lean individuals [36]. In a proteomics dementia data set, Gottfries et al. employed PLSDA for discrimination between different classes

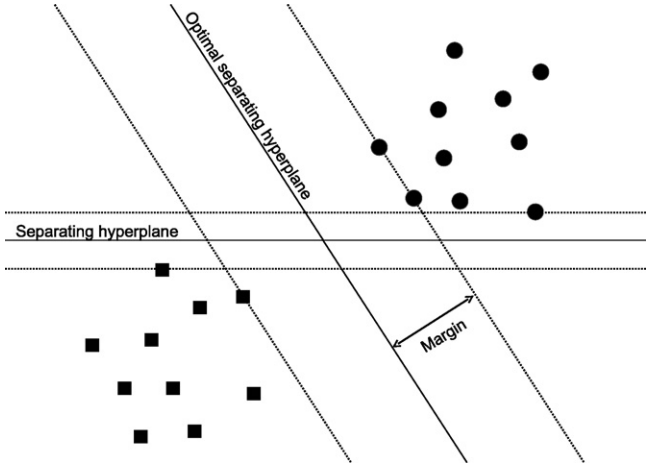

Fig. 3. The optimal separating hyperplane separates the classes with the widest margin.

of dementia and healthy individuals [37]. More examples of PLSDA applications in clinical metabolomics studies can be found in an overview by Trygg et al. [38].

3.3. Support vector machines

The support vector classifier constructs a hyperplane that separates the cases from the controls. When the classes are linearly separable, the optimal hyperplane maximizes the distance from the closest objects to the hyperplane, as is shown in Fig. 3. This distance is called the margin. The class assignment of new samples depends on which side of the hyperplane they are. In the case that the classes are not perfectly separable, some objects will be on the wrong side of the hyperplane (misclassification). The amount to which objects are allowed to be on the wrong side of the hyperplane is bound by a penalty. A high value for the penalty means it is very costly to cross the hyperplane. Consequently, in the original feature space the boundary will be wiggly to accommodate all samples; this may result in overfit. Small values can lead to hyperplanes that are not very effective in separating the classes [13,39].

In support vector machines (SVM), the data are transformed to a larger feature space. This makes it possible to accommodate discrimination problems for which a linear decision boundary is inappropriate. A nonlinear transformation of the data can be chosen in such a way that the classes are (almost) separable by a hyperplane in the higher dimensional feature space. The linear separation in the high dimensional feature space translates to a nonlinear decision boundary in the original feature space. The new, higher dimensional feature space does not have to be considered explicitly, the hyperplane can be computed using a kernel function. There are many possibilities for transforming the data, which makes SVM a versatile method [39]. The same data transformations could also be coupled to other classifiers, such as PCDA and PLSDA.

The SVM methodology is a popular method for classification in clinical proteomics. Among recent applications are studies of tuberculosis [40], ovarian and prostate cancer [41], response to therapy in rectal cancer patients [42], heart failure [43], and breast cancer [44].

3.4. Logistic regression

The odds is defined as the probability of a sample being a member of one class to the probability that the sample is outside that class. Logistic regression models use linear regression to fit the data to the natural logarithm of the odds. It ensures that the probabilities are between zero and one and that they sum to one. Logistic regression is similar to LDA, but it makes fewer assumptions about the underlying distributions. Like in DA, the large number of variables in proteomics data constitutes a problem, which can be tackled in several ways. Variable selection prior to modelling was used by Bhattacharyya et al. in a proteomics study of pancreatic cancer [45] and by Zhu and Hastie on microarray data in three cancer diagnosis data sets [46]. Others have combined PLS with logistic regression [47,48]. In penalized logistic regression, a penalty is set on the regression coefficients. As a result, some coefficients become zero, which effectively reduces the number of features [49,50].

3.5. Nearest shrunken centroids

In nearest centroid classification, a sample is assigned to the class with the nearest class mean. To accommodate classification of gene expression data, Tibshirani et al. developed the nearest shrunken centroids (NSC) method [51]. It shrinks the class centroids towards the overall centroid, thereby selecting genes. NSC, like diagonal discriminant analysis assumes a diagonal within-class covariance matrix. Tibshirani employed NSC for the discrimination of different cancer types. To predict the tissue of origin of 60 cancer cell lines, Shankavaram et al. applied NSC to gene expression profiles [52]. In a proteomics study of kidney patients with and without proteinuria, Kemperman et al. selected discriminating proteins using NSC [53].

3.6. Artificial neural networks

Artificial neural networks (ANN) refer to a class of nonlinear modelling methods. Three parts can be discerned in an ANN: the neurons in the input layer (data), neurons in one or more hidden layers, and the output layer neurons (predicted responses). The neurons in the hidden layer are formed by basis transformations of the input. The parameters of the basis transformations are learned from the data, as are the weights assigned to the hidden neurons to create the output [13]. Bloom applied ANN for the detection of the tissue of origin of adenocarcinomas, which were analyzed by 2D gel electrophoresis [54]. Other applications are prediction in breast cancer [55] and kidney disease [56].

3.7. Classification trees

A classification tree algorithm recursively splits the data in a parent node into two subsets called child nodes. The decision for the split is based on the value for one protein. The aim is to maximize homogeneity in the child nodes and the pro-

tein that gives the largest decrease in heterogeneity is chosen. The child nodes then become parent nodes and new variables are selected to split these nodes in turn. This process continues until all variables have been used or all terminal nodes are homogeneous. The last step is pruning of the tree to avoid overfitting. Several measures of heterogeneity are employed in different tree algorithms [13]. Some applications of decision trees in proteomics are clinical studies of pancreatic cancer [45], clinical behaviour after treatment in leukaemia patients [57], and ectopic pregnancy [58]. In this last study, Gerton et al. first built two trees to optimize separately for sensitivity and specificity, which they then combined to form one classification model.

3.8. Ensemble classifiers

Ensemble classifiers are formed by combining several single classification rules (base classifiers), with the goal to construct a predictor with superior performance. A new sample is classified by all individual classifiers and the ensemble prediction can be made by majority voting. The ensemble method is successful when each individual rule makes correct prediction for more than half of the samples and if the rules are diverse (give independent predictions) [59].

Different types of ensemble methods exist. Using several different classification methods to construct the base classifiers is one way to create diverse rules [60]. Alternatively, the rules can all be constructed with the same classification method, for example ANN [61]. Diversity of the rules can then be introduced by resampling the subjects with cross-validation [62], bootstrapping [61,63–65], and boosting [66,67]. A combination of bagging and boosting is used by Dettling in BagBoosting, where in each boosting step a bagged classifier is constructed [68]. Alternatively, resampling of the variables also leads to diverse base classifiers [69–72]. After construction of the base classifiers, their diversity can be evaluated by comparing their predictions [60,69] or the structure of the individual classifiers [63]. The final step is the combination of the base classifiers to arrive at one prediction for a sample. Several fusion methods exist [73], of which weighted voting and majority voting are much applied [60,62,64].

A well known ensemble classifier is the classification forest. The classification forest is an extension of the classification tree, where multiple trees are constructed and used in an ensemble to predict new samples. Examples of forest classifiers are random forest (RF) [74], for applications see [75,76], and decision forest [77,78].

4. Biomarker candidate selection

With biomarker candidate selection we refer to feature selection with the aim to discover which proteins are promising leads for biomarkers. We place this module after the classification methods, because the classification rules contain information about the contribution of each variable to the classification. This information reveals the proteins of interest, which may prove to be biomarkers. Two methods that determine the interesting variables directly are the classification tree [13], which classifies samples based on their values for a small number of proteins and the NSC algorithm, which as a by-product of constructing a classification rule selects variables [51].

Other classification methods carry relevant information about the variables in the form of weights and regression coefficients (linear SVM, DA). This information is used in many applications to select relevant sets of proteins. Guyon developed recursive feature elimination (RFE), a backward feature selection method, which eliminates the feature with the smallest weight in a linear SVM rule [17]. Rank products was initially designed for gene selection using gene expression differences between two groups directly [79], but it has also been employed for selection of proteins using a PCDA classification rule [33].

Bijlsma et al. used a threshold on the regression coefficients in PLSDA to select potential metabolite biomarkers [36]. Another feature extraction method for PLSDA is variable importance in the projection (VIP). The VIP value of a variable reflects its importance in the model with respect to the response vector as well as to the projected data [80]. It has been used in the selection of metabolites in studies of liver function in Hepatitis B [81] and intestinal fistulas [82].

Variable selection in ensemble methods is perhaps less straightforward, due to the amount of information that comes from using multiple classification rules. The random forest algorithm estimates the importance of a variable by permuting the measurements for that variable, leaving the rest of the data intact and classifying new samples [74].

It is also possible to use the information from significance tests (*t*-test, Wilcoxon–Mann–Whitney test) to select disease markers, without running a classification algorithm [18].

5. Comparison studies

Many more classification algorithms are available; the list of classifiers and variable selection methods we discuss is not exhaustive. The question arises which method is best suited for classification of proteomics data. It is hard to compare results from different studies because conditions vary. This is due to the fact that preprocessing, reporting of performance and validation schemes are not the same. There are some studies that describe performance of several classification methods applied to the same data set, with the aim to compare classifiers.

Liu et al. investigated six feature selection methods on leukaemia gene expression data and on ovarian cancer MS data [83]. After feature selection, four classifiers were applied to the reduced data. For the gene expression set entropy feature selection, which selects the features based on their discriminatory power, came out first. A correlation based feature selection (this method selects a subset of features that correlate with response but not with one another) led to the best performance in the ovarian cancer data.

A special issue of Proteomics in 2003 covered the data analysis efforts of several research groups on the same lung cancer data set [84]. Many strategies are applied in this issue to obtain a classifier. Due to the use of different validation schemes and different preprocessing it is very difficult to compare the performance.

In a comparison study of simple DA classifiers with aggregated classification trees (as representative for more sophisticated machine learning approaches) on three gene expression data sets, Dudoit et al. found that the DA methods performed very well [29].

Wagner et al. compared several linear and nonlinear DA methods and a linear SVM for classification of prostate cancer MS data [85]. Although the performances of the methods were comparable, the linear DA and linear SVM performed slightly better than nonlinear DA methods.

Wu et al. combined two feature selection methods and several classification algorithms to classify ovarian cancer MS data [19]. They concluded that RF outperformed the other methods (among which SVM, DA, bagged and boosted classification trees), but their conclusion was mainly based on the results after feature selection with RF. Feature selection based on the *t*-statistic resulted in superior performance of SVM and linear DA, closely followed by RF.

For classification of MS data of Gaucher disease, Hendriks et al. applied six classification methods [67]. The most successful were SVM, penalized logistic regression and PCDA.

The previous paragraphs show there is no consensus in what the best classifier is. This is due to the fact that different data sets have different characteristics and therefore no classifier will have optimal performance for all data sets. The performance not only depends on the data but also on the feature selection step and on the individual experience and taste of the data analyst. Experience with a method is likely to give better results. We have found no papers with general guidance in which situation to use a certain classifier.

6. Statistical validation

The next step towards clinical utility is validation. First, the results of a preliminary clinical proteomics study should be subjected to thorough statistical assessment. Next, a new set of samples should be measured independently in time and/or place from the first data set to test the classifier. If the preliminary results warrant the investment, the following step would be identification of the relevant proteins to determine biological validity.

In this section we describe two tools, permutation tests and cross-validation, to assess the statistical validity of the classifier, based on the first preliminary data set only. An overview of validation strategies in proteomics literature is given. We start by discussing different performance measures that are used in clinical proteomics.

6.1. Performance measures

The performance of a classifier in clinical applications is usually given in two measures. The sensitivity is the fraction of cases that are classified as cases. The specificity is the fraction of controls that is correctly identified. The sensitivity and specificity can take values between zero and one, where zero means all samples in that class are misclassified and one means that they are all correctly identified. They are both reported, because they each show a different characteristic of the classifier and can be very different [86]. The sensitivity and specificity can be altered by shifting the threshold for assignment to the case or control class. This may lead to a classifier with more desirable characteristics, such as a higher sensitivity, usually at the cost of specificity. The sensitivity and specificity can be plotted together in a receiver operating characteristic (ROC) curve. An example of an ROC curve is given in Fig. 4. The sensitivity is plotted on the *y*-axis and the *x*-axis represents the false positive fraction (1 - specificity). The lower left corner represents the case where all controls are correctly classified (specificity is 1), but all the cases are classified as controls (sensitivity is zero). The opposite case occurs in the upper right corner, where the sensitivity is 1 and the specificity is zero. Both corners are always part of the ROC curve. In between, the sensitivity and false positive fractions for different values of the threshold are plotted. Ideally, the resulting curve would go from the lower left corner to the upper left corner and then to the upper right corner. This represents a classifier that is able to perfectly distinguish between cases and control for some value for the threshold. The information in an ROC plot is summarized by the area under the curve (AUC). The AUC of a perfect classifier is one, whereas an uninformative classifier has an AUC of 0.5 [21,86].

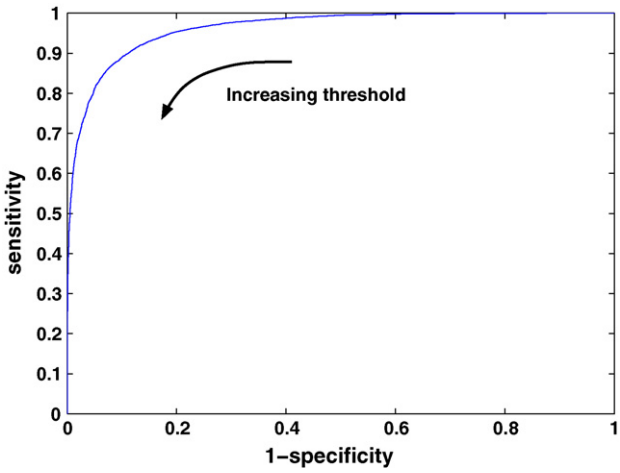

Fig. 4. ROC curve.

6.2. Cross-validation for performance estimation

A classifier is trained on a limited data set at some point in time, with the objective to correctly classify samples that will be measured in the future. At the time of construction, it is not possible to foresee how well a classifier will perform on freshly acquired samples, because the samples are not available. Therefore, the performance is estimated on data that is available. Nevertheless, the performance estimate should be based on an unseen set of samples, which are not in any way used in creating the classifier. If the performance is estimated using samples that have somehow been used in the modelling procedure, the estimate will be overly optimistic [13].

A second requirement of the performance estimate is that it should take into account the variability of the classifier. The data

set from which the parameters of the classifier are estimated is a sample from the entire population and therefore this classifier is one possible realisation. Other samples from the same population would result in different parameter estimates. The variability of the classifier should be reflected in the performance estimator.

Both requirements are met in cross-validation. Cross-validation makes efficient use of the available data, which is especially helpful in small datasets. The general idea is to split the data into several approximately equal-size parts. Each part is masked in turn (test set), while the remaining parts combined are used to train the classifier (training set). The classifier is then applied to the masked set for prediction. This is repeated until all parts have been masked once, and then the error made in the blinded test sets is combined to give an independent estimate of the performance of the classifier. Because the training sets are different in each repetition, the cross-validated performance estimate incorporates the variability of the classifier.

There are different variants of cross-validation. When the test set is made up of one sample it is called leave-one-out (LOO) cross-validation. In *k*-fold cross-validation, the data is divided in *k* parts. If *k* equals the number of samples it is leave-one-out cross-validation. A variant of *k*-fold cross-validation is leave-multiple-out cross-validation, where repetitions are allowed in the test sets [87]. Often, the ratio of the class sizes is preserved in the training and test sets, making them accurate representations of the original data. This is called stratified cross-validation [33,88,89].

6.3. Cross-validation for metaparameters/feature selection

Many of the classification methods described in the previous section require the optimization of model tuning parameters. For example, in PCDA and PLSDA, the number of retained latent variables should not be too low, because valuable information would be discarded. On the other hand, incorporating too many latent variables means uninformative noise is incorporated in the model. Care has to be taken to avoid overfitting of the model to the available data, as the data are typically highly undersampled. The choice of these parameters should be such that the generalization error of the resulting model (the error made in new samples) is low. This is also true for the selection of (a subset of) proteins for prediction. The selection should not only give good predictions for the available data, but also on newly acquired data. The tuning parameters and protein selection are called metaparameters.

Cross-validation is a much employed method to tune metaparameters in proteomics, as well as in other 'omics' studies, chemometrics, and Quantitative Structure–Activity Relationship research. In this section we will borrow from research on cross-validation in these fields and transfer relevant findings to clinical proteomics. For metaparameter tuning, the cross-validation procedure is repeated for different choices of the metaparameter. The performances of classifiers with different values for the metaparameters are compared to choose the parameter with the lowest cross validation error. Because the test sets are not used in creating the classifiers, overfitting of the model is prevented.

In the previous section we mentioned that cross-validation reflects the variability of the classifier that is due to the data being a sample from a population. This is also of importance for the selection of a metaparameter, since the goal is to construct a representative classifier. In LOO cross-validation, the training sets are very similar to the full data set and to each other. This means that the classifiers constructed on the training sets will not vary much and there is still a risk of overfitting. *k*-Fold cross-validation introduces more variability, because the training sets are smaller and less similar [13]. This forces the selection procedure to recognize general patterns, rather than individual data points [87]. A good value for *k* depends on the data: with smaller values for *k*, the test sets are larger and the training sets in undersampled datasets may become too small for building meaningful models. Moreover, the bias inherent to cross-validation increases with smaller values for *k*. This bias results from the training sets being smaller than the full data [13]. Generally, 5 or 10-fold cross-validation is used [90]. There are many ways to split the data into different parts in *k*-fold cross-validation. The estimate of the performance may depend on the choice of split [88]. Therefore, it is recommended to repeat the cross-validation several times with different splits of the data. Kohavi and John let the number of repeats depend on the standard deviation of the performance estimate [91]. They repeat until the standard deviation becomes sufficiently small. This way, large datasets are cross validated fewer times than small ones, in which the variance will be higher. It saves computing time and it gives a criterion for the number of repeats of cross-validation necessary.

Cross-validation can be performed with restrictions. Baumann restricts the number of variables (proteins) or latent variables to be selected [87]. However, this requires *a priori* knowledge of the data. Kohavi and John implement a complexity penalty in their evaluation to favour smaller subsets of variables [91].

6.4. Double cross-validation for metaparameter selection and performance estimation

When selecting a model with cross-validation, the corresponding cross-validation error is an inappropriate estimate of the prediction error of the model. In that case the cross-validation error is not based on an independent test set, because with the choice for a certain model, all of the data – the test samples as well as the training samples – is used. To solve this, Stone introduced the cross validatory paradigm: the cross validated choice of parameters requires cross validatory assessment to avoid overly optimistic performance estimates [92]. This means a nested cross-validation scheme is needed to estimate the prediction error, where the parameter optimization is executed in an internal loop and the prediction error is estimated in an external loop on a completely independent set of samples. Pseudocode for this cross-validation scheme is given in Fig. 5. It is often called cross-model validation or double cross-validation; in this review we refer to this scheme as double cross-validation. (For modelling procedures in which parameters are tuned in another way than with cross-validation, for example by bootstrapping, all

```
Split the data into K equal parts.

For k=1 to K

    Remove the kth part from the data (test set).

    On the remaining (K−1) parts (training set), perform an n−fold
    cross−validation for every metaparameter. Use the
    metaparameter with the lowest cross−validation error to build
    a classifier on the training set.

    Predict the test set with the classifier.

End

Combine the predictions for the K test sets to estimate the performance.
```

Fig. 5. Pseudocode for double cross-validation.

these training steps have to be taken into account in the validation of the performance.)

Several researchers have investigated the extent of the bias of the cross-validation error when not all model training steps are evaluated within the cross-validation. Taking two microarray datasets as an example (using SVM with RFE), Ambroise and McLachlan showed that, while single cross-validation suggests that the error rate was negligible, the test error was far from that [93]. Double cross-validation error is a much better estimate of the performance. In addition, they calculated the single and double cross-validation error rates for 20 permutations of the data. Although no information is present in the permuted data sets, the cross-validation error that is obtained with the selection of genes was almost zero. In contrast, double cross-validation error estimates were much more realistic, between 40% and 45%. Similar results were reported by Simon et al. [94], Varma and Simon [95] and Smit et al. [33].

The bias that is introduced in the performance estimate by ignoring the metaparameter selection in the validation process is called the parameter selection bias. Double cross-validation removes the parameter selection bias, but it does have the slight bias inherent to cross-validation that is the result of the lower number of samples in the training set than in the full data set [95].

It may seem a bit unclear what model is validated with double cross-validation, because the internal loop returns different tuning parameters for different training sets [96]. This is very much the same as what we described for cross-validation in the previous section. The variability of the classifier – in this case, the variability of the metaparameters as well as the estimated parameters – is taken into account in estimating the performance with double cross-validation [33]. Consequently, in double cross-validation the entire model optimization procedure is validated [95]. The ultimate classifier can be constructed in several ways. Stone chooses the tuning parameter with a cross-validation and uses this parameter to build a model on the full dataset [92,96]. Other possibilities are retaining all k classification rules from the double cross-validation and use them together as an ensemble classifier for new samples or using the most frequently selected parameter in the internal loop on the full data set [97].

6.5. Permutation test

In a permutation test the class labels are repeatedly removed and randomly reassigned to samples to create an uninformative data set of the same size as the data under study. One application of permutation tests is determining the relevance of a model. Building and testing a classifier on many permutations of the data gives a distribution of the performance found by chance, to which the performance of the classifier on the original data can be compared. The same classifier building protocol that is applied to the data is applied to the permutations, including any filtering or other selection of variables and parameter tuning [87].

Permutation testing was already mentioned in the previous section where it appeared as a tool to investigate the bias of different cross-validation methods [93]. Others have used it for the same purpose [97]. The rationale behind the use of the permutation test in this manner is that with uninformative data that is divided into two groups, a classifier would on average assign 50% to the wrong class. A validation method that returns an error rate that is on average much deviating from the expected 50% error rate is biased. Permutation tests thus answer two questions: whether the information in the data is truly relevant and whether the performance estimation is carried out properly.

In the literature, the number of executed permutations varies substantially. Ambroise et al. use 20 permutations to investigate the bias of incomplete cross-validation [93], while Bijlsma et al. and Smit et al. use as many as 10,000 permutations to determine the significance of the performance of a classifier [33,36,67].

So how many permutations are needed? For very small data sets it may be feasible to perform an exhaustive permutation test in which all possible permutations are considered. The number of possible permutations quickly rises, even for moderate class sizes. As an alternative, a test can be performed with only a subset of all permutations. The number of permutations determines accuracy and the lower bound of the p-value; with 100 permutations the lowest possible p-value is 0.01. Since the variance of the performance in permutations can be very large, a large number of permutations are needed to obtain a reliable result.

6.6. Strategies and applications

In this section we provide some examples of validation strategies applied in transcriptomics, metabolomics and proteomics literature.

A microarray data analysis workflow is suggested by Wessels et al. [88]. Their validation protocol consists of 100 repeats of a stratified double cross-validation, where the outer loop is a three fold cross-validation and the inner loop is 10 fold. They report the average of the sensitivity and the specificity.

For a metabolomics obesity study, Bijlsma et al. developed a strategy for data preprocessing, processing and validation [36]. The PLSDA classifier performance is evaluated with single cross-validation and 10,000 permutations. Potential biomarkers

are selected that have regression coefficients above a certain threshold. The information carried in the biomarker selection is tested by building models with only the selected variables. Additionally, non-informative models are build on the data without the selected variables to test if all relevant information is captured in the selected variables.

In proteomics research there are also several examples of statistical validation strategies. Lee validated PLSDA results on MS data with double cross-validation and by comparing the performance with 20 permutations of the original data [98]. Similar statistical strategies in clinical proteomics studies are used by Tong [78] and Smit et al. [33], but they consider thousands of permutations.

7. Proteomics data analysis: a framework

Data analysis methods extract information from the data to predict the class. As shown, there are many methods for feature selection, classification, biomarker candidate selection and statistical validation. It is possible to combine methods in different ways, leading to many data analysis approaches. We propose a modular data analysis framework (Fig. 6), in which most data analysis strategies fit. Some of the modules are optional, but validation is not! For each module the researcher can use his or her method of choice. In the remainder of this section we will discus the modules and their interactions.

Module 1 is the feature selection. This module is optional, but for high dimensional data the choice of classification method sometimes demands feature selection, for example when discriminant analysis or logistic regression is used. Module 2 is the classification method, this module is only necessary if one of the aims is to obtain a classification rule. Module 3 represents the biomarker selection, it is to be used if biomarker discovery is the purpose of the study and the biomarker selection is not intrinsic to the classification method.

The next three modules are statistical validation methods that are all discussed in the section Statistical validation. From a statistical point of view it is recommendable to use these modules if possible since they give generalizable models (module 4), performance estimates (module 5) and insight in the relevance of the model and the data (module 6). Invoking these validation tools enhances the trustworthiness of the model and the biomarkers.

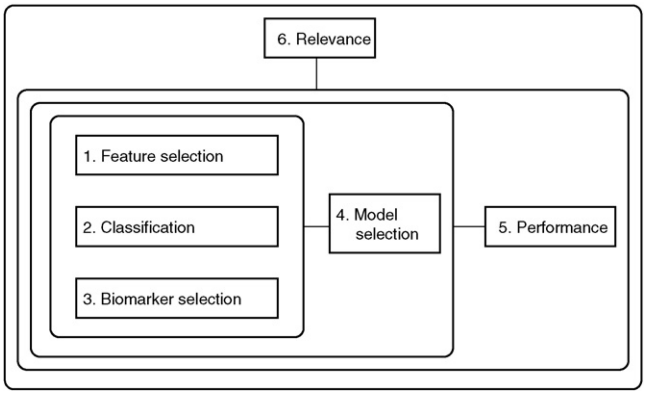

Fig. 6. Modular view of proteomics data analysis.

8. Black spots and open issues

8.1. External test set

If there is only one data set available a cross-validation approach makes efficient use of the data [97]. However, an external test set is always of added value [94]. An external data set obtained in a different way can show whether the model is not too specific for the data set that is used to construct the classification rule. For example the measurement could be performed on another instrument, by a different person, and the samples could have been obtained from a different population of patients. In the omics literature several examples of the use of external test set can be found [76,99].

8.2. Power calculations

An issue that we have not yet addressed in this review is power calculations. A power calculation determines the sample size necessary to observe a known effect. Such calculations are standard in clinical trials [100], but are not yet developed for clinical proteomics. There are two problems involved in power calculations for clinical proteomics: (i) unknown effect size, (ii) highly multivariate data. For power calculations the expected effect size (or the minimal wanted effect size) has to be known a priori. This is problematic in clinical proteomics. Moreover, power calculations are well developed for univariate analysis, but the results for multivariate analysis are very limited [101].

Obviously, the larger the sample sets, the more accurate the result. Unfortunately, the number of measurements is usually limited due to the cost of measurements or the limited availability of suitable samples. Validation strategies help overcome some problems. However, Rubingh et al. show that statistical tests become unreliable for data sets with small sample size [102].

8.3. Increasing complexity of data sets

The technology of mass spectrometry is improving, see for example the developments in hyphenated techniques, such as the combination of liquid chromatography and mass spectrometry (LC/MS). This implicates that the data sets, which are already complex, will be even more complex in the future. We see a tendency in the literature to analyze combinations of different types of omics data [3].

9. Conclusions

Proteomics research, despite the large effort in recent years, knows many issues that are still subject to debate. This review discussed some issues related to the analysis of proteomics data. Due to the complex nature and the high dimensionality of the data it generates it is easy to find differences between groups. But these differences are possibly just chance results. The goal is to develop classifiers and/or biomarkers that perform well on new data. Furthermore, a proper estimate of the performance is desirable for forming realistic expectations for the prediction of

future samples. Additionally, the relevance of the model should be investigated.

In this review we have shown that there are some good examples of performing statistical validation. We urge to set some standards in reporting results from models derived from proteomics data. Such a standard could include that sensitivity and specificity are only to be reported on test sets that have not been used during model building. Furthermore, also a *p*-value, possibly obtained from a permutation test, should be reported in order to assess the probability of a chance result.

A statistically valid biomarker should always be subjected to biological validation. This answers the question whether the biomarkers are specific for the disease. A statistical valid biomarker can be biologically irrelevant. To give an example: if the experiment is on a healthy control group and a group with cancer, the biomarker might be indicative for a secondary effect like inflammation that is not specific for cancer. Even the most thorough statistical procedure cannot safe-guard against this type of findings.

Acknowledgement

We thank Daniel Vis for careful reading of this manuscript.

References

[1] M.S. Pepe, R. Etzioni, Z.D. Feng, J.D. Potter, M.L. Thompson, M. Thornquist, M. Winget, Y. Yasui, J. Natl. Cancer Inst. 93 (2001) 1054.
[2] S. Hanash, Nature 422 (2003) 226.
[3] R. Aebersold, M. Mann, Nature 422 (2003) 198.
[4] B. Domon, R. Aebersold, Science 312 (2006) 212.
[5] G.S. Omenn, Proteomics 6 (2006) 5662.
[6] J. Villanueva, J. Philip, C.A. Chaparro, Y.B. Li, R. Toledo-Crow, L. DeNoyer, M. Fleisher, R.J. Robbins, P. Tempst, J. Proteome Res. 4 (2005) 1060.
[7] R.A.R. Bowen, Y. Chan, J. Cohen, N.N. Rehak, G.L. Hortin, G. Csako, A.T. Remaley, Clin. Chem. 51 (2005) 424.
[8] A.J. Rai, F. Vitzthum, Expert Rev. Proteomics 3 (2006) 409.
[9] M. Dijkstra, R.J. Vonk, R.C. Jansen, J. Chromatogr. B 847 (2007) 12.
[10] M. West-Nielsen, E.V. Hogdall, E. Marchiori, C.K. Hogdall, C. Schou, N.H.H. Heegaard, Anal. Chem. 77 (2005) 5114.
[11] A.E. Pelzer, I. Feuerstein, C. Fuchsberger, S. Ongarello, J. Bektic, C. Schwentner, H. Klocker, G. Bartsch, G.K. Bonn, Bju Int. 99 (2007) 658.
[12] M. Hilario, A. Kalousis, C. Pellegrini, M. Muller, Mass Spectrom. Rev. 25 (2006) 409.
[13] T. Hastie, J. Friedman, R. Tibshiranie, The Elements of Statistical Learning. Data mining, Inference and Prediction, Springer, New York, 2001.
[14] L. Kanal, B. Chandrasekaran, Pattern Recogn. 3 (1971) 225.
[15] A. Choudhary, M. Brun, J.P. Hua, J. Lowey, E. Suh, E.R. Dougherty, Bioinformatics 22 (2006) 837.
[16] E.R. Dougherty, J.P. Hua, M.L. Bittner, Curr. Genomics 8 (2007) 1.
[17] I. Guyon, J. Weston, S. Barnhill, V. Vapnik, Mach. Learn. 46 (2002) 389.
[18] M.K. Titulaer, I. Siccama, L.J. Dekker, A.L.C.T. van Rijswilk, R.M.A. Heeren, P.A.S. Smitt, T.M. Luider, BMC Bioinform. 7 (2006).
[19] B.L. Wu, T. Abbott, D. Fishman, W. McMurray, G. Mor, K. Stone, D. Ward, K. Williams, H.Y. Zhao, Bioinformatics 19 (2003) 1636.
[20] I. Levner, BMC Bioinform. 6 (2005).
[21] D.I. Broadhurst, D.B. Kell, Metabolomics 2 (2006) 171.
[22] Y. Benjamini, Y. Hochberg, J.R. Statist, Soc. B 57 (1995) 289.
[23] J.D. Storey, J.R. Statist. Soc. B 64 (2002) 479.
[24] V.G. Tusher, R. Tibshirani, G. Chu, Proc. Natl. Acad. Sci. U.S.A. 98 (2001) 5116.
[25] I.T. Joliffe, Principal Component Analysis, Springer, New York, 2002.
[26] R. Wehrens, L.M.C. Buydens, TrAC, Trends Anal. Chem. 17 (1998) 193.
[27] R.A. Fisher, Ann. Eugen. 7 (1936) 179.
[28] J.H. Friedman, J. Am. Stat. Assoc. 84 (1989) 165.
[29] S. Dudoit, J. Fridlyand, T.P. Speed, J. Am. Stat. Assoc. 97 (2002) 77.
[30] R. Hoogerbrugge, S.J. Willig, P.G. Kistemaker, Anal. Chem. 55 (1983) 1710.
[31] J. Ye, T. Li, T. Xiong, R. Janardan, IEEE/ACM Trans. Comput. Biol. Bioinform. 1 (2004) 181.
[32] R.H. Lilien, H. Farid, B.R. Donald, J. Comput. Biol. 10 (2003) 925.
[33] S. Smit, M.J. van Breemen, H.C.J. Hoefsloot, A.K. Smilde, J.M.F.G. Aerts, C.G. de Koster, Anal. Chim. Acta 592 (2007) 210.
[34] S. Wold, A. Ruhe, H. Wold, W.J. Dunn III, Siam J. Sci. Stat. Comput. 5 (1984) 735.
[35] M. Barker, W. Rayens, J. Chemom. 17 (2003) 166.
[36] S. Bijlsma, L. Bobeldijk, E.R. Verheij, R. Ramaker, S. Kochhar, I.A. Macdonald, B. van Ommen, A.K. Smilde, Anal. Chem. 78 (2006) 567.
[37] J. Gottfries, M. Sjogren, B. Holmberg, L. Rosengren, P. Davidsson, K. Blennow, Chemom. Intell. Lab. Syst. 73 (2004) 47.
[38] J. Trygg, E. Holmes, T. Lundstedt, J. Proteome Res. 6 (2007) 469.
[39] V.N. Vapnik, The Nature of Statistical Learning Theory, Springer-Verlag, New York, 2000.
[40] D. Agranoff, D. Fernandez-Reyes, M.C. Papadopoulos, S.A. Rojas, M. Herbster, A. Loosemore, E. Tarelli, J. Sheldon, A. Schwenk, R. Pollak, C.F.J. Rayner, S. Krishna, Lancet 368 (2006) 1012.
[41] K. Jong, E. Marchiori, A. van der Vaart, Applications of Evolutionary Computing 3005 (2004) 41.
[42] F.M. Smith, W.M. Gallagher, E. Fox, R.B. Stephens, E. Rexhepaj, E.F. Petricoin, L. Liotta, M.J. Kennedy, J.V. Reynolds, Ann. Surg. 245 (2007) 259.
[43] R. Willingale, D.J.L. Jones, J.H. Lamb, P. Quinn, P.B. Farmer, L.L. Ng, Proteomics 6 (2006) 5903.
[44] X.G. Zhang, X. Lu, Q. Shi, X.Q. Xu, H.C.E. Leung, L.N. Harris, J.D. Iglehart, A. Miron, J.S. Liu, W.H. Wong, BMC Bioinform. 7 (2006).
[45] S. Bhattacharyya, E.R. Siegel, G.M. Petersen, S.T. Chari, L.J. Suva, R.S. Haun, Neoplasia 6 (2004) 674.
[46] J. Zhu, T. Hastie, Biostatistics 5 (2004) 427.
[47] A. Goncalves, B. Esterni, F. Bertucci, R. Sauvan, C. Chabannon, M. Cubizolles, V.J. Bardou, G. Houvenaegel, J. Jacquemier, S. Granjeaud, X.Y. Meng, E.T. Fung, D. Birnbaum, D. Maraninchi, P. Viens, J.P. Borg, Oncogene 25 (2006) 981.
[48] L. Shen, E.C. Tan, IEEE/ACM Trans. Comput. Biol. Bioinform. 2 (2005) 166.
[49] P.H.C. Eilers, J. Boer, G.J.B. van Ommen, J.C. van Houwelingen, Prog. Biomed. Optics Imag. 2 (2001) 187.
[50] M. Dettling, P. Buhlmann, J. Multivar. Anal. 90 (2004) 106.
[51] R. Tibshirani, T. Hastie, B. Narasimhan, G. Chu, Proc. Natl. Acad. Sci. U.S.A. 99 (2002) 6567.
[52] U.T. Shankavaram, W.C. Reinhold, S. Nishizuka, S. Major, D. Morita, K.K. Chary, M.A. Reimers, U. Scherf, A. Kahn, D. Dolginow, J. Cossman, E.P. Kaldjian, D.A. Scudiero, E. Petricoin, L. Liotta, J.K. Lee, J.N. Weinstein, Mol. Cancer Ther. 6 (2007) 820.
[53] R.F.J. Kemperman, P.L. Horvatovich, B. Hoekman, T.H. Reijmers, F.A.J. Muskiet, R. Bischoff, J. Proteome Res. 6 (2007) 194.
[54] G.C. Bloom, S. Eschrich, J.X. Zhou, D. Coppola, T.J. Yeatman, Int. J. Cancer 120 (2006) 769.
[55] Q.H.C. Ru, L.W.A. Zhu, J. Silberman, C.D. Shriver, Mol. Cell. Proteomics 5 (2006) 1095.
[56] J.C. Oates, S. Varghese, A.M. Bland, T.P. Taylor, S.E. Self, R. Stanislaus, J.S. Almeida, J.M. Arthur, Kidney Int. 68 (2005) 2588.
[57] M. Albitar, S.J. Potts, F.J. Giles, S. O'Brien, M. Keating, D. Thomas, C. Clarke, I. Jilani, C. Aguilar, E. Estey, H. Kantarjian, Cancer 106 (2006) 1587.
[58] G.L. Gerton, X.J. Fan, J. Chittams, M. Sammel, A. Hummel, J.F. Strauss, K. Barnhart, Ann. N.Y. Acad. Sci. 1022 (2004) 306.
[59] L.K. Hansen, P. Salamon, IEEE Trans. Pattern Anal. Mach. Intell. 12 (1990) 993.

[60] G. Bhanot, G. Alexe, B. Venkataraghavan, A.J. Levine, Proteomics 6 (2006) 592.
[61] B. Liu, Q.H. Cui, T.Z. Jiang, S.D. Ma, BMC Bioinform. 5 (2004) 136.
[62] J. Khan, J.S. Wei, M. Ringner, L.H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C.R. Antonescu, C. Peterson, P.S. Meltzer, Nat. Med. 7 (2001) 673.
[63] J.H. Hong, S.B. Cho, Artif. Intell. Med. 36 (2006) 43.
[64] M.P.A. Ebert, J. Meuer, J.C. Wiemer, H.U. Schulz, M.A. Reymond, U. Traugott, P. Malfertheiner, C. Rocken, J. Proteome Res. 3 (2004) 1261.
[65] G. Valentini, M. Muselli, F. Ruffino, Neurocomputing 56 (2004) 461.
[66] E. Tamoto, M. Tada, K. Murakawa, M. Takada, G. Shindo, K. Teramoto, A. Matsunaga, K. Komuro, M. Kanai, A. Kawakami, Y. Fujiwara, N. Kobayashi, K. Shirata, N. Nishimura, S.I. Okushiba, S. Kondo, J. Hamada, T. Yoshiki, T. Moriuchi, H. Katoh, Clin. Cancer Res. 10 (2004) 3629.
[67] M.M.W.B. Hendriks, S. Smit, L.M.W. Akkermans, T.H. Reijmers, P.H.C. Eilers, H.C.J. Hoefsloot, C.M. Rubingh, C.G. de Koster, J.M. Aerts, A.K. Smilde, Proteomics 7 (2007) 3672.
[68] M. Dettling, Bioinformatics 20 (2004) 3583.
[69] Y.H. Peng, Int. J. Syst. Sci. 37 (2006) 931.
[70] A. Bertoni, R. Folgieri, G. Valentini, Neurocomputing 63 (2005) 535.
[71] K.J. Kim, S.B. Cho, Neurocomputing 70 (2006) 187.
[72] H.H. Won, S.B. Cho, Lecture Notes In Computer Science 2714 (2003) 1143.
[73] L.I. Kuncheva, IEEE Trans. Pattern Anal. Mach. Intell. 24 (2002) 281.
[74] L. Breiman, Mach. Learn. 45 (2001) 5.
[75] E.C. Gunther, D.J. Stone, R.W. Gerwien, P. Bento, M.P. Heyes, Proc. Natl. Acad. Sci. U.S.A. 100 (2003) 9608.
[76] K. Hoffmann, M.J. Firth, A.H. Beesley, N.H. de Klerk, U.R. Kees, BMC Cancer 6 (2006) 229.
[77] W.D. Tong, H.X. Hong, H. Fang, Q. Xie, R. Perkins, J. Chem. Inf. Comput. Sci. 43 (2003) 525.
[78] W.D. Tong, W. Xie, H.X. Hong, H. Fang, L.M. Shi, R. Perkins, E.F. Petricoin, Environ. Health Perspect. 112 (2004) 1622.
[79] R. Breitling, P. Armengaud, A. Amtmann, P. Herzyk, FEBS Lett. 573 (2004) 83.
[80] A.B. Umetrics, User's Guide to SIMCA-P, SIMCA-P + (2005), www.umetrics.com/pdfs/userguides/SIMCA-P_11_UG.pdf.
[81] J. Yang, X.J. Zhao, X.L. Liu, C. Wang, P. Gao, J.S. Wang, L.J. Li, J.R. Gu, S.L. Yang, G.W. Xu, J. Proteome Res. 5 (2006) 554.
[82] P.Y. Yin, X.J. Zhao, Q.R. Li, J.S. Wang, J.S. Li, G.W. Xu, J. Proteome Res. 5 (2006) 2135.
[83] H. Liu, J. Li, L. Wong, Genome Inform. 13 (2002) 51.
[84] Proteomics 3 (2003) 1667.
[85] M. Wagner, D.N. Naik, A. Pothen, S. Kasukurti, R.R. Devineni, B.L. Adam, O.J. Semmes, G.L. Wright, BMC Bioinform. 5 (2004) 26.
[86] M.S. Pepe, Stat. Med. 24 (2005) 3687.
[87] K. Baumann, TrAC Trends Anal. Chem. 22 (2003) 395.
[88] L.F.A. Wessels, M.J.T. Reinders, A.A.M. Hart, C.J. Veenman, H. Dai, Y.D. He, L.J. van't Veer, Bioinformatics 21 (2005) 3755.
[89] R. Kohavi, International Joint Conference on Artificial Intelligence, 1995, p. 1137.
[90] P. Zhang, Ann. Stat. 21 (1993) 299.
[91] R. Kohavi, G.H. John, Artif. Intell. 97 (1997) 273.
[92] M. Stone, J. R. Statist. Soc. B 36 (1974) 111.
[93] C. Ambroise, G.J. McLachlan, Proc. Natl. Acad. Sci. U.S.A. 99 (2002) 6562.
[94] R. Simon, M.D. Radmacher, K. Dobbin, L.M. McShane, J. Natl. Cancer Inst. 95 (2003) 14.
[95] S. Varma, R. Simon, BMC Bioinform. 7 (2006).
[96] R.G. Brereton, TrAC Trends Anal. Chem. 25 (2006) 1103.
[97] B.J.A. Mertens, M.E. De Noo, R.A.E.M. Tollenaar, A.M. Deelder, J. Comput. Biol. 13 (2006) 1591.
[98] K.R. Lee, X.W. Lin, D.C. Park, S. Eslava, Proteomics 3 (2003) 1680.
[99] N.P. Munro, D.A. Cairns, P. Clarke, M. Rogers, A.J. Stanley, J.H. Barrett, P. Harnden, D. Thompson, I. Eardley, R.E. Banks, M.A. Knowles, Int. J. Cancer 119 (2006) 2642.
[100] W.J. Dixon, F.J. Massey, Introduction to Statistical Analysis, McGraw-Hill, New York, 1983.
[101] J.A. Ferreira, A. Zwinderman, Stat. Appl. Genet. Mol. Biol. 5 (2006).
[102] C.M. Rubingh, S. Bijlsma, E.P.P.A. Derks, I. Bobeldijk, E.R. Verheij, S. Kochhar, A.K. Smilde, Metabolomics 2 (2006) 53.