Chemometrics and Intelligent Laboratory Systems 118 (2012) 62–69

Contents lists available at SciVerse ScienceDirect

Chemometrics and Intelligent Laboratory Systems

journal homepage: www.elsevier.com/locate/chemolab

Review

A review of variable selection methods in Partial Least Squares Regression

Tahir Mehmood *, Kristian Hovde Liland, Lars Snipen, Solve Sæbø

Biostatistics, Department of Chemistry, Biotechnology and Food Sciences, Norwegian University of Life Sciences, Norway

ARTICLE INFO

Article history:
Received 12 April 2012
Received in revised form 4 July 2012
Accepted 25 July 2012
Available online 2 August 2012

Keywords:
Variable selection
PLS

ABSTRACT

With the increasing ease of measuring multiple variables per object the importance of variable selection for data reduction and for improved interpretability is gaining importance. There are numerous suggested methods for variable selection in the literature of data analysis and statistics, and it is a challenge to stay updated on all the possibilities. We therefore present a review of available methods for variable selection within one of the many modeling approaches for high-throughput data, Partial Least Squares Regression. The aim of this paper is mainly to collect and shortly present the methods in such a way that the reader easily can get an understanding of the characteristics of the methods and to get a basis for selecting an appropriate method for own use. For each method we also give references to its use in the literature for further reading, and also to software availability.

Contents

1. Introduction

The massive data generation which is experienced in many real world applications nowadays calls for multivariate methods for data analysis. The cost of measuring an increasing number of variables per sample is steadily decreasing with the advances in technology.

* Corresponding author.
E-mail addresses: tahir.mehmood@umb.no (T. Mehmood), kristian.liland@umb.no (K.H. Liland), lars.snipen@umb.no (L. Snipen), solve.sabo@umb.no (S. Sæbø).

0169-7439/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.chemolab.2012.07.010

Measuring gene expression in bioinformatics is a good example of how the technology has developed from more or less univariate methods like PCR (Polymerase Chain Reaction) for measuring EST's (Expressed Sequence Tags) to the highly multivariate microarrays for measuring of relative mRNA contents and finally to the newly developed sequencing technology facilitating high-speed sequencing of DNA/RNA. In order to gain insight into complex systems like metabolism and gene regulation multivariate considerations are necessary and in this respect the advances in technology have been of utmost importance to collect the information from all relevant variables. However, the downside of the technological expansion is of course the risk of including irrelevant variables in the statistical models. In order to minimize the influence of such noisy variables some data reduction is usually necessary, either through projection methods, variable selection or a combination of both.

The dimensionality problem described here is typical for many fields of science. Many authors, for example [1–5], have addressed this problem and there are numerous suggested approaches for dealing with the so-called large p small n problem [1], that is, many variables and few samples. Some approaches survives the wear of time by being adopted by data analysts, others are one-hit wonders which do not reach up in the competition. This is how scientific progress is obtained. However, every now and then it is necessary to pause and review the current status in a given field of research, and in this paper we will do so for variable selection methods only in Partial Least Squares regression (PLSR) [6,7].

PLSR has proven to be a very versatile method for multivariate data analysis and the number of applications is steadily increasing in research fields like bioinformatics, machine learning and chemometrics (see Fig. 1, source: http://wokinfo.com/ with search keyword "partial least squares" OR "projection to latent structures"). It is a supervised method specifically established to address the problem of making good predictions in multivariate problems, see [1]. PLSR in its original form has no implementation of variable selection, since the focus of the method is to find the relevant linear subspace of the explanatory variables, not the variables themselves, but a large number of methods for variable selection in PLSR has been proposed. Our aim with this paper is to give the reader an overview of the available methods for variable selection in PLSR and to address the potential uniqueness of each method and/or similarity to other methods.

Before we focus on the various methods it may be worthwhile to give some motivation for performing variable selection in PLSR. Partial Least Squares is a projection based method which in principle should ignore directions in the variable space which are spanned by irrelevant, noisy variables. Hence, for prediction purposes variable selection may seem unnecessary since up- and down-weighting of variables is an inherent property of the PLS estimator. However, a very large p and small n can still spoil the PLS regression results. For instance, in such cases there is a problem with the asymptotic consistency of the PLS estimators for univariate responses [2], and from a prediction perspective the large number of irrelevant variables may yield large variation on test set prediction [8]. These two deficiencies are likely related to the fact that the PLS-algorithm has an increasing problem finding the correct size of the relevant sub-space of the p-dimensional variable space when the number of variables increases. See e.g. [5] for more discussion on this. These examples motivate variable selection for improved estimation/prediction performance, and is also discussed by [3,4]. Variable selection may improve the model performance, but at the same time it may eliminate some useful redundancy from the model, and using a small number of variables for prediction means we are putting large influence of each variable in final model [9]. In this respect the consistency of selected variables is also important, as utilized by [10,11]. A second motivation for variable selection is improved model interpretation and understanding of the system studied. Hence, the motivation for the analysis may rather be to identify a set of important variables for further study, possibly by other technologies or methods. These two motivations may be somewhat contradictory and, for achieving better interpretation, it may be necessary to compromise the prediction performance of the PLSR model [11]. Hence variable selection is needed for providing a more observant analysis of the relationship between a modest number of explanatory variables and the response.

This paper is organized as follows. First, in Section 2 we present the most common PLSR algorithm, the orthogonal score PLSR, since we will refer repeatedly to this algorithm when discussing the various variable selection methods. Then, in Section 3 we present the variable selection methods which are organized into three main categories, filter methods, wrapper methods and embedded methods. In the discussion at the end we present the linkages between the variable selection methods.

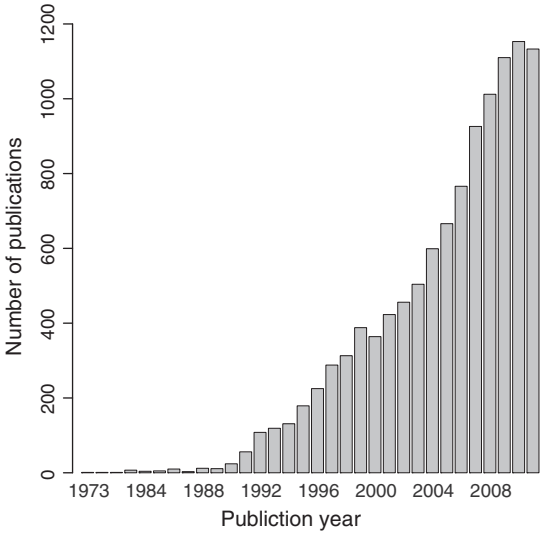

Fig. 1. An overview of growth of PLS and related work.

2. PLSR algorithm

There are many versions of the PLSR-algorithm available. For simplicity we here focus only on the orthogonal score PLSR [6] because most of the variable selection methods are originally based on this, and the rest should be straightforward to implement. We limit ourselves to PLS-regression in this paper, the situation where a set of explanatory variables $X_{(n,p)}$ are assumed to be linked to a response $y_{(n,1)}$ through the linear relationship $y=\alpha+X\beta+$, for some unknown regression parameters α and β and error term . For simplicity we will concentrate on the single response case called PLS1 or simply PLS, but the methods should be straight forward to generalize to multiple responses, called PLS2. PLS2 is suitable in a limited number of cases, as most of the times, it is better to optimize the number of latent variables for each y vector separately (PLS1). Initially the variables are centered (and optionally scaled) into $X_0 = X-1\bar{x}'$ and $y_0 = y-1\bar{y}$. Assume that some A (where $A \leq p$) is equal to the number of relevant components for prediction, following the definition by Naes and Helland [12]. Then for $a=1, 2, \ldots, A$ the algorithm runs:

1. Compute the loading weights by

$$w_a = X'_{a-1} y_{a-1}$$

The weights define the direction in the space spanned by X_{a-1} of maximum covariance with y_{a-1}. Normalize to loading weights to have length equal to 1 by

$$w_a \leftarrow w_a / w_a$$

2. Compute the score vector t_a by

$$t_a = X_{a-1} w_a$$

3. Compute the X-loadings p_a by regressing the variables in X_{a-1} on the score vector:

$$p_a = X'_{a-1} \frac{t_a}{t'_a t_a}$$

Similarly compute the Y-loading q_a by

$$q_a = y'_{a-1} \frac{t_a}{t'_a t_a}$$

4. Deflate X_{a-1} and y_{a-1} by subtracting the contribution of t_a:

$$X_a = X_{a-1} - t_a p'_a$$

$$y_a = y_{a-1} - t_a q_a$$

5. If $a<A$ return to 1.

Let the loading weights, scores and loadings computed at each step of the algorithm be stored in matrices/vectors $W=[w_1, w_2, \ldots, w_A]$, $T=[t_1, t_2, \ldots, t_A]$, $P=[p_1, p_2, \ldots, p_A]$ and $q=[q_1, q_2, \ldots, q_A]$. Then the PLSR-estimators for the regression coefficients for the linear model are found by: $\hat{\beta} = W\left(P'W\right)^{-1} q$ and $\hat{\alpha} = \bar{y} - \bar{x}\hat{\beta}$. In case of multiple responses the Y-loadings will be replaced by a loading matrix $Q=[q_1, q_2, \ldots, q_A]$.

3. Variable selection methods in PLS

Based on how variable selection is defined in PLSR we can categorize the variable selection methods into three main categories: filter-, wrapper-, and embedded methods. This categorization was also used by [13]. Before we go into details and look at the specific selection methods, we give a short explanation of these three categories.

- Filter methods: These methods use the (optionally modified) output from the PLSR-algorithm to purely identify a subset of important variables. The purpose is variable identification.
- Wrapper methods: The variables identified by the filter methods may be piped back into a re-fitting of the PLSR-model to yield reduced models in which case we have wrapper-methods. The methods are mainly distinguished by the choice of underlying filter-method and how the 'wrapping' is implemented. These methods are mainly based on procedures iterating between model fitting and variable selection.
- Embedded methods: The variable selection is an integrated part of a modified PLSR-algorithm. Hence, these methods do the variable selection at component level.

The differences between these methods are also illustrated in Fig. 2. In Fig. 3 an overview of variable selection methods discussed in this paper is shown.

3.1. Filter methods

The filter methods select variables in two steps, firstly the PLSR model is fitted to the data, and secondly the variable selection is carried out by introducing a threshold on some measure of relevancy obtained from the fitted PLS model. Normally these methods are fast and easy to compute, but with regard to prediction relevancy of the selected variables these methods give no indication. These methods require some sort of filter measure representing the response relation with the respective variable. A threshold on the filter measure is required to classify variables as selected or not, hence the selection is heavily affected by the chosen threshold and choosing a good threshold level may be problematic. Examples of filter measures in PLS include loading weight vectors w_a, PLS regression coefficients $\hat{\beta}$ and variable importance on projection.

3.1.1. Loading weights ($\boldsymbol{w}$)

From the fitted PLS model, which is optimized for some number of components, possibly through cross-validation, the loading weight ($w_{\mathbf{a}}$) can be used as a measure of importance to select variables [14,3]. For each component the variables with a loading weight above a certain threshold in absolute value may be selected. This is also known as hard thresholding and was suggested by [15]. For example, loading weights extracted from the optimum model are used for the determination of wavelengths effective for the discrimination of fruit vinegars [16] and for the selection of wavelengths correlated with concentration which results in stable results with respect to prediction [17]. Since defining the threshold is an issue, a possibility is to use *Hotelling's* T^2 test statistics [18] coupled with Jacknifing (ref) to test a joint significance of all A loading weights for a given variable [19]. In [20] this approach was used for significant gene selection from gene expression data. Loading weight is

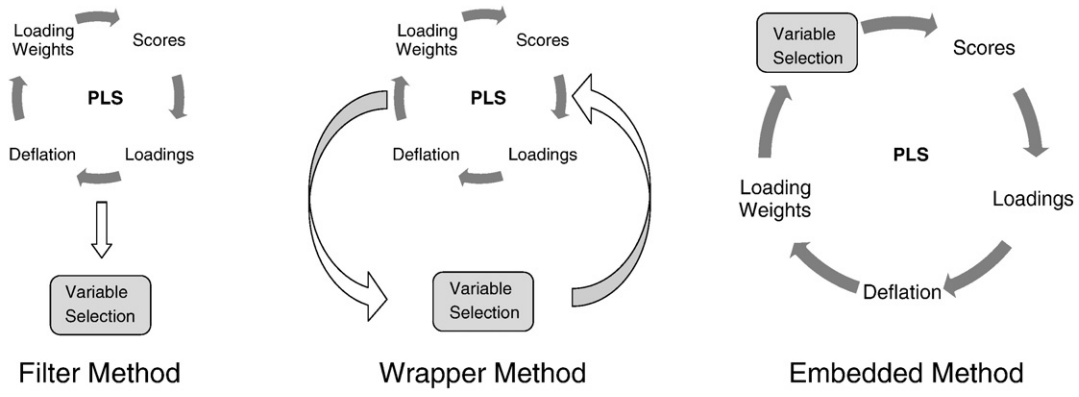

Fig. 2. Illustration for filter, wrapper and embedded methods.

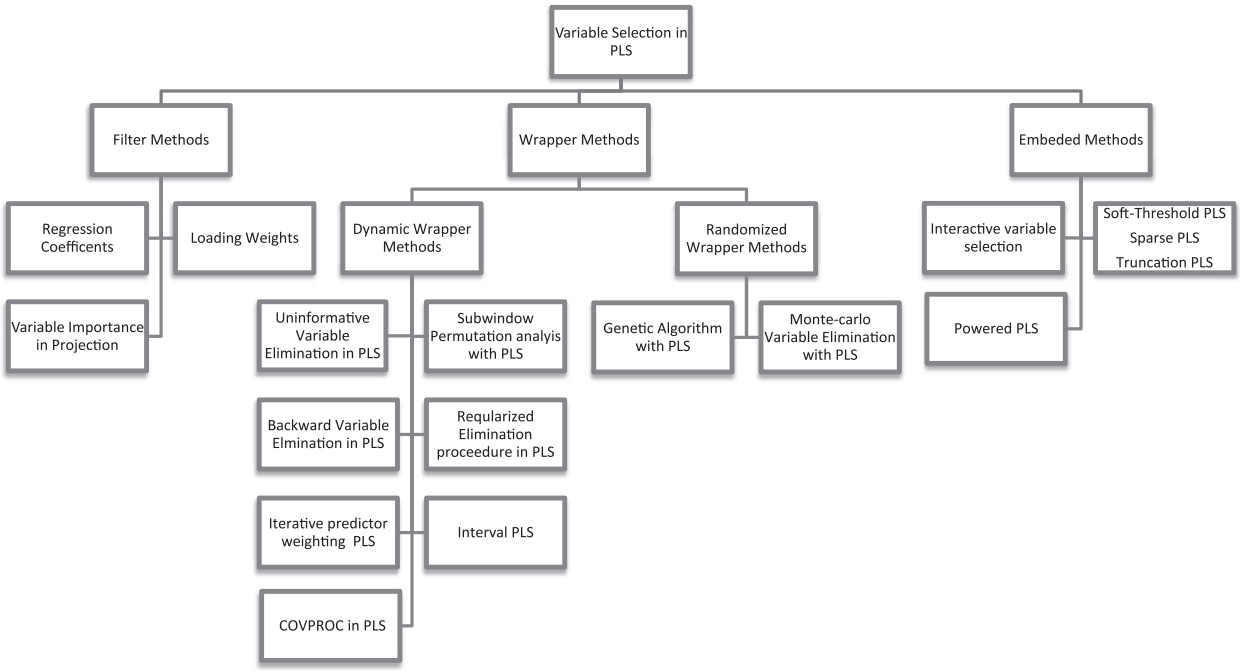

Fig. 3. An overview of methods used for variable selection with PLS.

an easily accessible output and filter measure from the PLSR model from most softwares, for example using the R-implementation (http://cran.r-project.org/web/packages/pls/).

3.1.2. Regression coefficients (β)

A second possibility is to use the vector of regression coefficients (β) which is a single measure of association between each variable and the response. Again, variables having small absolute value of this filter measure can be eliminated [4]. Also in this case thresholding may be based on significance considerations from jackknifing or bootstrapping [21] which has been adopted in a wide range of studies (e.g. [22,23]). In this way, instead of relying on the single observation of the filter measure, the distribution or variation is also considered. In this respect bootstrapping may give more efficient regression coefficients estimation than jackknifing, but at the cost of extra computation time [21]. Examples of the use of regression coefficients as filter measure include wavelength selection [24,23] and gene selection in modeling the relationship between the left mechanical ventricular assist device (LVAD) support time and gene expression changes in the human heart [25]. Regression coefficients are also standard output from most PLSR software, including R.

3.1.3. Variable importance in projection (VIP)

A third filter measure is the variable importance in PLS projections (VIP) introduced by [26] as 'Variable influence on projection' which is known now as 'Variable importance in projection' termed by [27]. The idea behind this measure is to accumulate the importance of each variable j being reflected by $\boldsymbol{w}$ from each component. The VIP measure v_j is defined as

$$v_j = \sqrt{p\sum_{a=1}^{A}\left[SS_a\left(w_{aj}/||w_a||^2\right)\right]/\sum_{a=1}^{A}(SS_a)}$$

where SS_a is the sum of squares explained by the ath component. Hence, the v_j weights is a measure of the contribution of each variable according to the variance explained by each PLS component where $(w_{aj}/||w_a||)^2$ represents the importance of the jth variable. Since the variance explained by each component can be computed by the expression $q_a^2 t'_a t_a$ [27], the v_j can alternatively be expressed as

$$v_j = \sqrt{p\sum_{a=1}^{A}\left[\left(q_a^2 t'_a t_a\right)\left(w_{aj}/||w_a||\right)^2\right]/\sum_{a=1}^{A}\left(q_a^2 t'_a t_a\right)}.$$

Variable j can be eliminated if $v_j < u$ for some user-defined threshold $u \in [0, \infty)$. It is generally accepted that a variable should be selected if $v_j > 1$, [27–29], but a proper threshold between 0.83 and 1.21 can yield more relevant variables according to [28]. Further, the importance of each variable can, if preferred, be expressed as a percentage. Also, if probabilistic considerations regarding the importance of $\boldsymbol{v}$ is required, a bootstrap procedure can be applied. This may improve the stability of the results compared to selection based on regression coefficients β [29]. For PLSR models the VIP is implemented in MATLAB (http://www.mathworks.com) that is codes are available on MATLAB central. As examples of its use we can mention the search of biologically relevant QSAR descriptors [30] and wavelength selection [31,29].

3.2. Wrapper methods

Wrapper methods mostly use the filter methods in an iterative way and are based on some supervised learning approach, where model refitting is wrapped within the variable search algorithm [13]. The search algorithm extracts the subset of relevant variables and evaluates each subset by fitting a model to the subset variable. For subset extraction that optimize the performance, the ideal approach would be to evaluate all possible subsets, but for large number of variables this is impractical since the number of subsets to evaluate increases exponentially with the number of variables. Therefore so-called greedy search algorithms of various kinds have been proposed in the literature. These methods interact with the model at the risk of over-fitting and intensive computational time. Wrapper methods can further be categorized [13] on the basis of search algorithm; deterministic or randomized. Randomized search algorithms utilize some kind of randomness in the selection of subset while deterministic do not. Examples from randomized search based wrapper methods in

PLSR include Genetic Algorithm PLSR [32] and the Monte-Carlo based UVE-PLS [10] for variable selection, while deterministic search based wrapper methods includes uninformative variable elimination in PLS (UVE-PLS) [33], sub-window permutation analysis coupled with PLS [34], backward variable elimination PLS [35] and regularized elimination procedure [11]. Deterministic wrapper methods are relatively simpler and require less computations, have lower risk of over-fitting and smaller number of parameters are required to tune than randomized wrapper methods. Because of a more greedy search the deterministic approaches carry a higher risk of coming up with local instead of global optima. All wrapper methods are computationally more expensive than filter methods.

3.2.1. Genetic algorithm combined with PLS regression (GA-PLS)

The Genetic Algorithm (GA) has become a widespread subset search algorithm, and Hasegawa et al. combine GA with PLS in the GA-PLS method [32] and further developed by Leardi et al. [36,37]. Genetic algorithms are inspired by biological evolution theory and natural selection in the sense that variables that yield fitted models showing high performance (or *fitness*) have higher probability to "survive" and to be included in variable sets in subsequent model refits. Further, a mutation step ensures a certain level of randomness in the algorithm. The steps involved are:

1. Building an initial population of variable sets by setting bits for each variable randomly, where bit '1' represents selection of corresponding variable while '0' presents non-selection. The approximate size of the variable sets must be set in advance.
2. Fitting a PLSR-model to each variable set and computing the performance by, for instance, a leave one out cross-validation procedure.
3. A collection of variable sets with higher performance are selected to survive until the next "generation".
4. Crossover and mutation: new variable sets are formed 1) by crossover of selected variables between the surviving variable sets, and 2) by changing (mutating) the bit value for each variable by small probability.
5. The surviving and modified variable sets form the population serving as input to point 2.

The steps 2–5 are repeated a preset number of times. Upon completion of the GA-algorithm the best variable set (or a combination of a collection of the best sets) in terms of performance is selected. The GA-PLS has been adopted in a number of studies, for instance: for the selection of topological descriptors to build quantitative structural property relationship (QSPRs) models [38,39], for selection of quantum topological molecular similarity indices (QTMS) for describing the quantitative effects of molecular electronic environments on the antagonistic activity of some dihydropyridine (DHP) derivatives [40], and for wavelength selection [41].

Codes for GA-PLS are implemented in MATLAB (http://www.mathworks.com/).

3.2.2. Uninformative variable elimination in PLS (UVE-PLS)

Centner et al. [33] introduce uninformative variable elimination in PLS (UVE-PLS), where artificial noise variables are added to the predictor set before the PLSR model is fitted. All the original variables having lower "importance" than the artificial noise variables are eliminated before the procedure is repeated until a stop criterion is reached. The steps involved in UVE-PLS are

1. Generate a noise matrix $\boldsymbol{N}$, having the same dimension as $\boldsymbol{X}$ where entries are randomly drawn from uniform distribution in the interval 0.0–1.0.
2. Combine $\boldsymbol{N}$ and $\boldsymbol{X}$ matrix in new matrix of variables $\boldsymbol{Z}=\boldsymbol{[X,N]}$.
3. Fit the PLSR model to the combined matrix $\boldsymbol{Z}$ and validate by means of leave-one-out cross-validation.
4. Cross-validation results are used in jack-knifing to compute a test statistic for each variable as $c_j = mean\left(\hat{\beta}_j\right)/sd\left(\hat{\beta}_j\right)$, for $j=1, 2, \ldots, 2p$.
5. Set the threshold *cmax* as the maximum of absolute value c among the noise variables. Original variables with an absolute value of c smaller than *cmax* are assumed to be noise variables and are eliminated.

The steps 2–6 are repeated unless the performance of the models start decreasing. Artificially added random variables could influence the model if random variables are not properly selected [42]. Examples of the use of UVE-PLS are the identification of regions contributing to the binding activity of Aromatase enzyme [43], and wavelength selection [44–47]. Code for UVE-PLS is available in MATLAB (http://www.mathworks.com/). Importantly, the noise variables $\boldsymbol{N}$ should have same variability as the original $\boldsymbol{X}$ ones, this motivates some kind of pre-scaling or standardization.

There exist alternative variants of UVE-PLS. The Monte-Carlo based variant MC-UVE-PLS is an example [10]. The main difference to the regular UVE-PLS is a repeated splitting of the sample set into training and test data, and random cross-validation on the training data with a final performance test on the test data. In this method the variables are selected directly from their stability instead of adding random noise variables. With regard to assessing the stability of the regression coefficients, there are alternatives to jack-knifing, for instance, resampling and bootstrapping methods as well as adding noise to the response as discussed in [48,49]. Further, the multiple models with different training sets from MC-UVE-PLS may be more efficient then the single model of UVE-PLS, so MC-UVE-PLS may decrease the risk of over-fitting. A similar concept was presented by Li et al. [50] where model population analysis was coupled with PLS (MPA-PLS) for variable selection, also including Monte Carlo sampling. Moving window PLS (MW-PLS) is another close variant of MC-UVE-PLS [51].

3.2.3. Backward variable elimination PLS (BVE-PLS)

A backward variable elimination procedure was introduced by Frank et al. [3] for elimination of non informative variables. Later Fernandez et al. [35] came with an updated version, which has been used for wavelength selection [52,53]. In general, variables are first sorted with respect to some importance measure, and usually one of the filter measures described above are used. Secondly, a threshold is used to eliminate a subset of the least informative variables. Then a model is fitted again to the remaining variables and performance is measured. The procedure is repeated until maximum model performance is achieved.

3.2.4. Sub-window permutation analysis coupled with PLS (SwPA-PLS)

Sub-window permutation analysis coupled with PLS (SwPA-PLS) [34] provides the influence of each variable without considering the influence of the rest of the variables. The use of subset of variables makes SwPA-PLS more efficient and fast for large datasets. Steps involved in SwPA-PLS are

1. Sub-dataset sampling in both sample and variable space into N test and N training data sets.
2. For each randomly sampled training set a PLS model is built and a normal prediction error, NPE, is measured on the corresponding test set. This NPE will be associated with all the predictors in the given training data set.
3. Prediction performance is also measured for each sampled training set by iteratively permuting each predictor variable in the training set to obtain a permuted prediction error, PPE, associated with each predictor.
4. The variable importance of each variable j is assessed upon completion of the sub-dataset sampling by comparing the distributions of normal prediction errors (*NPE*) and permuted prediction errors (*PPE*) of any given variable. Hence, for variable j the statistical assessment of importance is computed as $D_j = mean(PPE_j) - mean(NPE_j)$.

5. All variables for which $D_j > 0$ are considered informative in the sense that they will with large probability improve the prediction performance of a model if they are included.

In [34] the informative variables are subject to further significance testing comapring the distribution of NPE's and PPE's by a Mann–Whitney U-test ([54]). The code for SwPA-PLS is implemented in R and can be found at rlhttp://code.google.com/p/spa2010/downloads/list. A very close variant of SwPA-PLS is competitive adaptive reweighing sampling (CARS) [55], which has been used for identification of free fatty acids and exploring possible biomarkers [56] and for wavelength selection [57].

3.2.5. Iterative predictor weighting PLS (IPW-PLS)

Forina et al. [58] introduced an iterative procedure for variable elimination, called Iterative predictor weighting PLS (IPW-PLS). This is an iterative elimination procedure where a measure of predictor importance is computed after fitting a PLSR model (with complexity chosen based on predictive performance). The importance measure is used both to re-scale the original X-variables and to eliminate the least important variables before subsequent model re-fitting. Recently, the procedure is implemented in the field of mass spectrometry [59], which is relatively new and attractive technology getting in for disease and protein-based biomarker profiling.

3.2.6. Regularized elimination procedure in PLS

Mehmood et al. [11] introduced a regularized variable elimination procedure for parsimonious variable selection, where also a stepwise elimination is carried out. A stability based variable selection procedure is adopted, where the samples have been split randomly into a predefined number of training and test sets. For each split, g, the following stepwise procedure is adopted to select the variables: Let $Z_0 = X$ and s_j (e.g. w_j, β_j or r_j) be one of the filter criteria for variable j.

1. For iteration g run $\boldsymbol{Y}$ and Z_g through cross validated PLS. The matrix Z_g has p_g columns, and we get the same number of criterion values, sorted in ascending order as $s_{(1)}, \ldots, s_{(p_g)}$.
2. Assume there are M criterion values below some predefined cutoff u. If $M = 0$, terminate the algorithm here.
3. Else, let $N = \lfloor fM \rfloor$ for some fraction $f \in \langle 0, 1]$. Eliminate the variables corresponding to the N most unfavorable criterion values.
4. If there is still more than one variable left, let Z_{g+1} contain these variables, and return to 1).

The fraction f determines the 'steplength' of the elimination algorithm, where an f close to 0 will only eliminate a few variables in every iteration. The fraction f and the cutoff u can be determined through cross validation.

Allowing a marginal decrease in model performance may often result in a substantial decrease in the number of selected variables, which in turn may improve the interpretability of the model considerably, and therefore Mehmood et al. also adopt an additional test in order to find the simplest model, in terms of number of variables, which is not significantly worse than the optimal model. Mehmood et al. have tested and compared three different filter measures commonly used in the Partial Least Square modeling paradigm for variable selection; loading weights, regression coefficients and variable importance on projections, where variable importance on projection based elimination outperforms the others with respect to the consistency of the selected variables. The algorithm is applied to a problem of identifying codon variations discriminating different bacterial taxa, which is of particular interest in classifying metagenomics samples.

3.2.7. COVPROC in PLS

The COVPROC method (Covariance Procedure) is a forwards stepwise approach presented by [60]. For each component the predictor variables are added sequentially based on the absolute value of the loading weights. The variables are added and the model refitted until no improvement in fit is recorded. Upon deflating the predictor matrix by the contribution from the first component the COVPROC procedure continues on the next component, and so on. In [60] the COVPROC method is illustrated on a real data set from process control in industry.

3.2.8. Interval PLS (iPLS)

The interval PLS (iPLS) introduced by [9] is especially designed for wavelength selection, where spectra is split into smaller subintervals with equal distance, then a PLSR is fitted to each sub-interval. The sub-interval having the smallest prediction error is selected. The selected subintervals can also optimized by including or eliminating new variables. This has been used for wavelength selection [61].

3.3. Embedded methods

These methods combine variable selection and modeling in a one-step procedure. The search for an optimal subset of variables is conducted on each component of the PLS model. Embedded methods nest the variable selection within the PLS algorithm, and are based on one iterative procedure (which is common to standard PLS algorithm), while the wrapper methods in the previous section are based on a double iterative procedure where outer iterations for variable selection and inner iterations of model refitting works in an iterative way. Hence, the embedded methods are typically less time consuming than the wrapper methods [13].

3.3.1. Interactive variable selection (IVS) for PLS

The interval PLS (iPLS) introduced by [62,63] is especially designed for wavelength selection, where spectra are split into smaller subintervals of equal distance, then a PLSR is fitted to each sub-interval. In "forward" mode the sub-interval having the smallest cross-validated prediction error is selected. This can be repeated and extended to include more subintervals for better prediction. In "backward" mode the subintervals having the largest error are removed iteratively. The selected subintervals can also be optimized by including or eliminating single variables. This has been used for wavelength selection [62]. MATLAB code and a graphical toolbox can be downloaded from: (http://www.models.life.ku.dk/ipls).

3.3.2. Soft-Threshold PLS (ST-PLS) and Sparse-PLS

Sæbø et al. [64] introduced a soft-thresholding step in PLS algorithm (ST-PLS) based on ideas from the nearest shrunken centroid method [65]. The ST-PLS approach is more or less identical to the Sparse-PLS presented independently by Lê Cao et al. [66]. ST-PLS is further very similar to the IVS method described above. At each step of the sequential ST-PLS algorithm the loading-weights are modified as follows:

1. Scaling:
   $w_k \leftarrow w_k / max_j |w_{kj}|$, for $j = 1, \ldots, p$
2. Soft-thresholding:
   $w_{kj} \leftarrow \text{sign}(w_{kj})(|w_{kj}| - \delta)_+$, for $j = 1, \ldots, p$ and some $\delta \in [0, 1\rangle$. Here $(\ldots)_+$ means $max(0, \ldots)$
3. Normalizing:

$$w_k \leftarrow w_k / ||w_k||$$

The shrinkage $\delta \in [0, 1)$ sets the degree of thresholding, i.e. a larger δ gives a smaller selected set of variables. Cross validation is used to define this threshold [64]. The scarcity in $\boldsymbol{w}$ at each component for variable selection was also considered by [2,67,62,63,58,68]. ST-PLS has been used for selection of relevant genes for genotype phenotype mapping [69] and Sparse-PLS has been used for biomarker selection from omics data [70] and for SNPs selection [71], and codes for Sparse

PLS is implemented in R-package (http://cran.r-project.org/web/packages/spls/).

3.3.3. Powered PLS (PPLS)

PLSR aims at optimizing the covariance between response and explanatory variables at each step of the algorithm. However, if a large number of non-relevant noise variables are included in $\boldsymbol{X}$, then PLS gets deviated from its aim and starts to optimize the covariance between predictors mainly [2]. To cope with this, a new data compression method for estimating optimal latent variables by combining PLS methodology introduced in [72] and generalized in [73] by canonical correlation analysis (CCA). They introduce a flexible trade-off between the element wise correlations and variances specified by a power parameter γ, ranging from 0 to 1. Modified loading weights are defined as:

$$w(\gamma) = K_\gamma \left[h_1 |corr(x_1, y)|^{\frac{\gamma}{1-\gamma}} .std(x_1)^{\frac{1-\gamma}{\gamma}}, \ldots, h_p \left| corr\left(x_p, y\right)\right|^{\frac{\gamma}{1-\gamma}} .std\left(x_p\right)^{\frac{1-\gamma}{\gamma}} \right]'$$

where h_j denotes the sign of the jth correlation and K_γ is a scaling constant assuring unit length $w(\gamma)$. The γ value optimizing the canonical correlation between $\boldsymbol{Y}$ and $XW\gamma$ is always selected for each component of the PPLS algorithm. The PPLS algorithm is not a variable selection method *per se* since all variables will retain at least a minimum contribution through non-zero loading weights, but typically many weights are negligible and may be regarded as eliminated. Inspection of loading plots is a useful tool which may reveal the relative importance of the variables. PPLS has been used for bio-marker selection [74] and for spectrum selection [75], and codes for PPLS is implemented in R as part of the pls-package (cran.r-project.org/), and in the article describing PPLS MATLAB code for PPLS is given in an appendix.

4. Discussion

In this paper a range of variable selection methods for PLSR has been presented and classified into three main categories; filter methods, wrapper methods and embedded methods. The classification is based on the properties of the variable selection methods. Filter methods are simple and provide quickly a ranking of the variables with respect to some importance measure. A limitation of the filter methods is the fact that some 'threshold' must be specified in order to select a subset of variables, hence selection is threshold dependent and without proper cross-validated tuning it is hard to get reliable results. Using cross-validation for threshold selection quickly turns filter methods into wrapper methods, where variable selection is wrapped with the model. Wrapper methods run in an iterative way where in each run the model performance is measured, variables are sorted based on the filter measures and selection is carried out. Together with improved interpretability of the model these methods also provide performance measures. Wrapper methods are typically time consuming and need larger number of complexity parameters to be tuned. Embedded methods incorporate variable selection in a very nice structural form by making variable selection as an integrated step of modeling. Most of the embedded methods do, however, also run cross-validation internally in the PLS-algorithm to select component-wise threshold parameters, and this will inevitably slow down the algorithms. An exception here is the PPLS which instead of running cross-validation selects optimal power parameter based on maximization of canonical correlation.

At the end it is worth mentioning the importance of proper validation when it comes to variable selection. With a large number of variables there will always be irrelevant variables which turn up as important due to chance, even after cross-validation. Hence, the risk of over-fitting is very large in the process of variable selection. However, there are differences between the various variable selection methods with regard to the risk of over-fitting. The problem of over-fitting is discussed by several authors, for instance by Jouan-Rimbaud et al. [76] and Leardi [77] who discuss this problem in the context of a backward elimination from subsets of variables selected by GA. Some approaches tend to exploit the Y-information in data more than others in the search for important variables, and typically the more flexible approaches run a greater risk of over-fitting. To be on the safe side, regardless of which method one chooses, a test set of independent samples should ideally be used as final assessment of the selected variables.

Many authors have attempted to compare the various selection methods, usually in order to demonstrate the improved performance for a newly suggested method. Very few neutral comparisons have been conducted on a wide range of data sets which can give suggestions to what method to choose. Probably there is no such thing as an always best variable selection method since there is likely an interaction between method and the properties of the data. In this paper we therefore make no final recommendation to what the reader should choose to have in their statistical toolbox. However, we hope that this review gives the reader a better understanding of the similarities and differences of many of the methods suggested in literature, and it is always a good idea to try out approaches with different properties to a given data problem.

References

[1] H. Martens, T. Næs, Multivariate Calibration, Wiley, 1989.
[2] H. Chun, S. Keleş, Sparse partial least squares regression for simultaneous dimension reduction and variable selection, Journal of the Royal Statistical Society, Series B (Statistical Methodology) 72 (2010) 3–25.
[3] I. Frank, Intermediate least squares regression method, Chemometrics and Intelligent Laboratory Systems 1 (1987) 233–242.
[4] A. Frenich, D. Jouan-Rimbaud, D. Massart, S. Kuttatharmmakul, M. Galera, J. Vidal, Wavelength selection method for multicomponent spectrophotometric determinations using partial least squares, Analyst 120 (1995) 2787–2792.
[5] I. Helland, Some theoretical aspects of partial least squares regression, Chemometrics and Intelligent Laboratory Systems 58 (2001) 97–107.
[6] S. Wold, H. Martens, H. Wold, The Multivariate Calibration Problem in Chemistry Solved by the PLS Method, in: Conference Proceeding Matrix pencils, 1983, pp. 286–293.
[7] P. Geladi, B. Kowalski, Partial least-squares regression: a tutorial, Analytica Chimica Acta 185 (1986) 1–17.
[8] A. Höskuldsson, Variable and subset selection in PLS regression, Chemometrics and Intelligent Laboratory Systems 55 (2001) 23–38.
[9] L. Norgaard, A. Saudland, J. Wagner, J. Nielsen, L. Munck, S. Engelsen, Interval partial least-squares regression (iPLS): a comparative chemometric study with an example from near-infrared spectroscopy, Applied Spectroscopy 54 (2000) 413–419.
[10] W. Cai, Y. Li, X. Shao, A variable selection method based on uninformative variable elimination for multivariate calibration of near-infrared spectra, Chemometrics and Intelligent Laboratory Systems 90 (2008) 188–194.
[11] T. Mehmood, H. Martens, S. Sæbø, J. Warringer, L. Snipen, A partial least squares based algorithm for parsimonious variable selection, Algorithms for Molecular Biology 6 (2011).
[12] T. Naes, I. Helland, Relevant components in regression, Scandinavian Journal of Statistics 20 (1993) 239–250.
[13] Y. Saeys, I. Inza, P. Larrañaga, A review of feature selection techniques in bioinformatics, Bioinformatics 23 (2007) 2507.
[14] M. Martens, Sensory and chemical quality criteria for white cabbage studied by multivariate data analysis, Lebensmittel-Wissenschaft und Technologie 18 (1985) 100–104.
[15] R. Shao, F. Jia, E. Martin, A. Morris, Wavelets and non-linear principal components analysis for process monitoring, Control Engineering Practice 7 (1999) 865–879.
[16] F. Liu, Y. He, L. Wang, Determination of effective wavelengths for discrimination of fruit vinegars using near infrared spectroscopy and multivariate analysis, Analytica Chimica Acta 615 (2008) 10–17.
[17] D. Jouan-Rimbaud, B. Walczak, D. Massart, I. Last, K. Prebble, Comparison of multivariate methods based on latent vectors and methods based on wavelength selection for the analysis of near-infrared spectroscopic data, Analytica Chimica Acta 304 (1995) 285–295.
[18] M. Xiong, J. Zhao, E. Boerwinkle, Generalized T^2 test for genome association studies, The American Journal of Human Genetics 70 (2002) 1257–1268.
[19] H. Martens, M. Martens, Multivariate Analysis of Quality- An Introduction, Wiley, 2001.
[20] L. Gidskehaug, E. Anderssen, A. Flatberg, B. Alsberg, A framework for significance analysis of gene expression data using dimension reduction methods, BMC Bioinformatics 8 (2007) 346.
[21] B. Efron, R. Tibshirani, in: An Introduction to the Bootstrap, volume 57, Chapman & Hall/CRC, 1993.

[22] A. Ferreira, T. Alves, J. Menezes, Monitoring complex media fermentations with near-infrared spectroscopy: comparison of different variable selection methods, Biotechnology and Bioengineering 91 (2005) 474–481.
[23] H. Xu, Z. Liu, W. Cai, X. Shao, A wavelength selection method based on randomization test for near-infrared spectral analysis, Chemometrics and Intelligent Laboratory Systems 97 (2009) 189–193.
[24] S. Osborne, R. Künnemeyer, R. Jordan, Method of wavelength selection for partial least squares, Analyst 122 (1997) 1531–1537.
[25] X. Huang, W. Pan, S. Park, X. Han, L. Miller, J. Hall, Modeling the relationship between LVAD support time and gene expression changes in the human heart by penalized partial least squares, Bioinformatics 20 (2004) 888.
[26] S. Wold, E. Johansson, M. Cocchi, in: PLS: Partial Least Squares Projections to Latent Structures, 3D QSAR in drug design, 1, 1993, pp. 523–550.
[27] L. Eriksson, E. Johansson, N. Kettaneh-Wold, S. Wold, Multi-and megavariate data analysis, Umetrics, Umeå, 2001.
[28] G. Chong, C.H. Jun, Performance of some variable selection methods when multicollinearity is present, Chemometrics and Intelligent Laboratory Systems 78 (2005) 103–112.
[29] R. Gosselin, D. Rodrigue, C. Duchesne, A bootstrap-vip approach for selecting wavelength intervals in spectral imaging applications, Chemometrics and Intelligent Laboratory Systems 100 (2010) 12–21.
[30] M. Olah, C. Bologa, T. Oprea, An automated PLS search for biologically relevant QSAR descriptors, Journal of Computer-Aided Molecular Design 18 (2004) 437–449.
[31] G. ElMasry, N. Wang, C. Vigneault, J. Qiao, A. ElSayed, Early detection of apple bruises on different background colors using hyperspectral imaging, LWT- Food Science and Technology 41 (2008) 337–345.
[32] K. Hasegawa, Y. Miyashita, K. Funatsu, GA strategy for variable selection in QSAR studies: GA-based PLS analysis of calcium channel antagonists, Journal of Chemical Information and Computer Sciences 37 (1997) 306–310.
[33] V. Centner, D. Massart, O. de Noord, S. de Jong, B. Vandeginste, C. Sterna, Elimination of uninformative variables for multivariate calibration, Analytical Chemistry 68 (1996) 3851–3858.
[34] H. Li, M. Zeng, B. Tan, Y. Liang, Q. Xu, D. Cao, Recipe for revealing informative metabolites based on model population analysis, Metabolomics 6 (2010) 353–361.
[35] J. Fernández Pierna, O. Abbas, V. Baeten, P. Dardenne, A backward variable selection method for PLS regression (BVSPLS), Analytica Chimica Acta 642 (2009) 89–93.
[36] R. Leardi, A. Lupiáñez González, Genetic algorithms applied to feature selection in PLS regression: how and when to use them, Chemometrics and Intelligent Laboratory Systems 41 (1998) 195–207.
[37] R. Leardi, M. Seasholtz, R. Pell, Variable selection for multivariate calibration using a genetic algorithm: prediction of additive concentrations in polymer films from Fourier transform-infrared spectral data, Analytica Chimica Acta 461 (2002) 189–200.
[38] S. Riahi, M. Ganjali, P. Norouzi, F. Jafari, Application of GA-MLR, GA-PLS and the DFT quantum mechanical (QM) calculations for the prediction of the selectivity coefficients of a histamine-selective electrode, Sensors and Actuators B: Chemical 132 (2008) 13–19.
[39] H. Liu, R. Zhang, X. Yao, M. Liu, Z. Hu, B. Fan, Prediction of the isoelectric point of an amino acid based on GA-PLS and SVMs, Journal of Chemical Information and Computer Sciences 44 (2004) 161–167.
[40] A. Mohajeri, B. Hemmateenejad, A. Mehdipour, R. Miri, Modeling calcium channel antagonistic activity of dihydropyridine derivatives using QTMS indices analyzed by GA-PLS and PC-GA-PLS, Journal of Molecular Graphics and Modelling 26 (2008) 1057–1065.
[41] J. Ghasemi, A. Niazi, R. Leardi, Genetic-algorithm-based wavelength selection in multicomponent spectrophotometric determination by PLS: application on copper and zinc mixture, Talanta 59 (2003) 311–317.
[42] N. Faber, M. Meinders, P. Geladi, M. Sjostrom, L. Buydens, G. Kateman, Random error bias in principal component analysis. part I. Derivation of theoretical predictions, Analytica Chimica Acta 304 (1995) 257–271.
[43] J. Polanski, R. Gieleciak, The comparative molecular surface analysis (CoMSA) with modified uniformative variable elimination-PLS (UVE-PLS) method: application to the steroids binding the aromatase enzyme, Journal of Chemical Information and Computer Sciences 43 (2003) 656–666.
[44] J. Koshoubu, T. Iwata, S. Minami, Application of the modified UVE-PLS method for a mid-infrared absorption spectral data set of water-ethanol mixtures, Applied Spectroscopy 54 (2000) 148–152.
[45] J. Koshoubu, T. Iwata, S. Minami, Elimination of the uninformative calibration sample subset in the modified UVE (uninformative variable elimination)-PLS (partial least squares) method, Analytical Sciences 17 (2001) 319–322.
[46] C. Abrahamsson, J. Johansson, A. Sparén, F. Lindgren, Comparison of different variable selection methods conducted on NIR transmission measurements on intact tablets, Chemometrics and Intelligent Laboratory Systems 69 (2003) 3–12.
[47] C. Spiegelman, M. McShane, M. Goetz, M. Motamedi, Q. Yue, G. Coté, Theoretical justification of wavelength selection in PLS calibration: development of a new algorithm, Analytical Chemistry 70 (1998) 35–44.
[48] N. Faber, Uncertainty estimation for multivariate regression coefficients, Chemometrics and Intelligent Laboratory Systems 64 (2002) 169–179.
[49] N.K.M. Faber, Improved computation of the standard error in the regression coefficient estimates of a multivariate calibration model, Analytical Chemistry 72 (2000) 4675–4676.
[50] H. Li, Y. Liang, Q. Xu, D. Cao, Model population analysis for variable selection, Journal of Chemometrics 24 (2010) 418–423.
[51] J. Jiang, R. Berry, H. Siesler, Y. Ozaki, Wavelength interval selection in multicomponent spectral analysis by moving window partial least-squares regression with applications to mid-infrared and near-infrared spectroscopic data, Analytical Chemistry 74 (2002) 3555–3565.
[52] E. Guzmán, V. Baeten, J. Fernández Pierna, J. García-Mesa, Application of low-resolution raman spectroscopy for the analysis of oxidized olive oil, Food Control 86 (2011) 1507–1515.
[53] A. Lazraq, R. Cleroux, J. Gauchi, Selecting both latent and explanatory variables in the PLS1 regression model, Chemometrics and Intelligent Laboratory Systems 66 (2003) 117–126.
[54] H. Mann, D. Whitney, On a test of whether one of two random variables is stochastically larger than the other, Annals of Mathematical Statistics 18 (1947) 50–60.
[55] H. Li, Y. Liang, Q. Xu, D. Cao, Key wavelengths screening using competitive adaptive reweighted sampling method for multivariate calibration, Analytica Chimica Acta 648 (2009) 77–84.
[56] B. Tan, Y. Liang, L. Yi, H. Li, Z. Zhou, X. Ji, J. Deng, Identification of free fatty acids profiling of type 2 diabetes mellitus and exploring possible biomarkers by GC–MS coupled with chemometrics, Metabolomics 6 (2010) 219–228.
[57] W. Fan, H. Li, Y. Shan, H. Lv, H. Zhang, Y. Liang, Classification of vinegar samples based on near infrared spectroscopy combined with wavelength selection, Analytical Methods 3 (2011) 1872–1876.
[58] M. Forina, C. Casolino, C. Pizarro Millan, Iterative predictor weighting (IPW) PLS: a technique for the elimination of useless predictors in regression problems, Journal of Chemometrics 13 (1999) 165–184.
[59] D. Chen, B. Hu, X. Shao, Q. Su, Variable selection by modified IPW (iterative predictor weighting)-PLS (partial least squares) in continuous wavelet regression models, Analyst 129 (2004) 664–669.
[60] S. Reinikainen, A. Höskuldsson, Covproc method: strategy in modeling dynamic systems, Journal of Chemometrics 17 (2003) 130–139.
[61] R. Leardi, L. Nørgaard, Sequential application of backward interval partial least squares and genetic algorithms for the selection of relevant spectral regions, Journal of Chemometrics 18 (2004) 486–497.
[62] F. Lindgren, P. Geladi, S. Rännar, S. Wold, Interactive variable selection (IVS) for pls. Part 1: theory and algorithms, Journal of Chemometrics 8 (1994) 349–363.
[63] F. Lindgren, P. Geladi, A. Berglund, M. Sjöström, S. Wold, Interactive variable selection (IVS) for PLS. part II: Chemical applications, Journal of Chemometrics 9 (1995) 331–342.
[64] S. Sæbø, T. Almøy, J. Aarøe, A.H. Aastveit, St-pls: a multi-dimensional nearest shrunken centroid type classifier via pls, Journal of Chemometrics 20 (2007) 54–62.
[65] R. Tibshirani, T. Hastie, B. Narasimhan, G. Chu, Class prediction by nearest shrunken centroids, with applications to DNA microarrays, Statistical Science (2003) 104–117.
[66] K. Lê Cao, D. Rossouw, C. Robert-Granié, P. Besse, A sparse PLS for variable selection when integrating omics data, Statistical Applications in Genetics and Molecular Biology 7 (2008) 35.
[67] B. Alsberg, A. Woodward, M. Winson, J. Rowland, D. Kell, Variable selection in wavelet regression models, Analytica Chimica Acta 368 (1998) 29–44.
[68] B. Hu, X. Shao, Q. Su, Variable selection by modified IPW (iterative predictor weighting)-PLS (partial least squares) in continuous wavelet regression models, Analyst 129 (2004) 664–669.
[69] T. Mehmood, H. Martens, S. Sæbø, J. Warringer, L. Snipen, Mining for genotype–phenotype relations in *Saccharomyces* using partial least squares, BMC Bioinformatics 12 (2011).
[70] D. Rossouw, C. Robert-Granié, P. Besse, A sparse PLS for variable selection when integrating omics data, Genetics and Molecular Biology 7 (2008) 35.
[71] K. Lê Cao, S. Boitard, P. Besse, Sparse PLS discriminant analysis: biologically relevant feature selection and graphical displays for multiclass problems, BMC Bioinformatics 12 (2011) 253.
[72] U. Indahl, A twist to partial least squares regression, Journal of Chemometrics 19 (2005) 32–44.
[73] U. Indahl, K. Liland, T. Næs, Canonical partial least squares — a unified PLS approach to classification and regression problems, Journal of Chemometrics 23 (2009) 495–504.
[74] N. Gerd, et al., Finding biomarker signatures in pooled sample designs: A simulation framework for methodological comparisons, Advances in Bioinformatics 2010 (2010).
[75] K. Liland, B. Mevik, E. Rukke, T. Almøy, T. Isaksson, Quantitative whole spectrum analysis with MALDI-TOF MS, Part II: Determining the concentration of milk in mixtures, Chemometrics and Intelligent Laboratory Systems 99 (2009) 39–48.
[76] D. Jouan-Rimbaud, D. Massart, O. De Noord, Random correlation in variable selection for multivariate calibration with a genetic algorithm, Chemometrics and Intelligent Laboratory Systems 35 (1996) 213–220.
[77] R. Leardi, Genetic algorithms in feature selection, Academic Press, New York, 1996.