Model-population analysis and its applications in chemical and biological modeling

Hong-Dong Li, Yi-Zeng Liang, Qing-Song Xu, Dong-Sheng Cao

Model-population analysis (MPA) was recently proposed as a general framework for designing new types of chemometrics and bioinformatics algorithms, and it has found promising applications in chemistry and biology. The goal of MPA is to extract useful information from complex analytical systems, so as to lead to better understanding and better modeling of chemical and biological data.

To give an overall picture of MPA, we first review its key elements. Then, we describe the theories and the applications of selected methods that focus on the two fundamental aspects in chemical and biological modeling: outlier detection and variable selection. We highlight the key common principles of these methods and pinpoint the critical differences underlying each method.

Keywords: Algorithm; Bioinformatics; Chemometrics; Complex analytical system; Data modeling; Modeling; Model-population analysis (MPA); Monte Carlo sampling; Outlier detection; Variable selection

Hong-Dong Li, Yi-Zeng Liang*, Dong-Sheng Cao
College of Chemistry and Chemical Engineering, Central South University, Changsha 410083, PR China

Qing-Song Xu
School of Mathematic Sciences, Central South University, Changsha 410083, PR China

*Corresponding author.
Tel./Fax: +86 731 8830831;
E-mail: yizeng_liang@263.net

1. Introduction

A vast amount of data is routinely produced in chemistry and biology and often display high complexity (e.g., existence of outliers, a large number of irrelevant variables measured, and non-linearity), rendering their statistical modeling challenging [1–4]. Specifically, existence of outliers caused by experimental errors or other uncontrolled factors would make a predictive model misleading, so such models cannot be used to make reliable predictions [5,6]. Furthermore, chemical and biological data resulting from modern high-throughput analytical instruments have a large number of variables most of which are irrelevant to or would even interfere with the problem under investigation [7–10]. Also, the sample size is comparatively small. This is the so called ''large p, small n'' problem that has proved to be very challenging in statistical learning [2,11,12]. Predictive models built using all measured variables are quite difficult to interpret, usually of low prediction accuracy and therefore of little use in practice. Variable selection is an effective solution to solve this problem. Indeed, a large number of methods have been developed for variable selection and gained successful applications in chemistry and biology {e.g., LASSO [13], elastic net [11], target projection [14] and CARS [15]}. Summing up, outlier detection and variable selection are of substantial importance in mining information from complex analytical data. They are the two most fundamental issues relating to the statistical modeling of chemical and biological data.

In outlier detection, samples to be diagnosed as outliers are always based on a single number (e.g., Mahalanobis distance) or a prediction error from only one model [16]. We argue that characterizing a sample as an outlier by a single number is insufficient. A distribution of the criterion selected for assessing the outlying propensity of a sample, rather than a single number, should be used. To this end, we proposed the Monte Carlo (MC) method, where the distribution of prediction errors of a test sample resulting from a population of sub-models is used to evaluate whether a sample is likely to be an

0165-9936/$ - see front matter © 2012 Elsevier Ltd. All rights reserved. doi:10.1016/j.trac.2011.11.007

outlier or not. It was shown that cross-validated estimates of prediction errors after removal of outliers were significantly lowered [17]. Of note, cross validation plays a pivotal role in model assessment and selection, and is widely used in chemistry [18,19] and biology [20,21]. Important developments include double cross validation [22], MC cross validation [18,23] and repeated double cross validation [24]. The statistical characteristics (e.g., bias and consistency of cross validation) have been discussed elsewhere [25–27].

In variable selection, variable importance is often assessed by assigning an importance score to each variable using a selected criterion [14,28], followed by selecting variables that display high importance scores. We also argue that assessing variable importance by a single importance score resulting from a single model is unreliable. Indeed, a variety of methods have been proposed for variable selection {e.g., looking at the distribution of regression coefficients in uninformative variable elimination (UVE) [29], its MC extension [30], prediction errors in sub-window permutation analysis (SPA) [31] and noise-incorporated sub-window permutation analysis (NISPA) [10], and margin of support vector machines (SVMs) in margin influence analysis (MIA) [32]}. In addition, Wongravee et al. proposed to determine potentially discriminatory variables for supervised self-organizing maps (SOMs) by analyzing the distribution of variable rank obtained from 100 training sets generated using MC sampling [33]. Interestingly, Abeel et al. [34] showed that the robustness of biomarker selection can be improved significantly through the analysis of their complete (weighted) linear aggregation of variable ranks by running a linear SVM coupled with recursive feature elimination [35] on multiple bootstrap datasets. Of note, a Bayesian approach is proposed for gene selection by analyzing inclusion probability of each gene based on a large number of sub-models that are drawn from its posterior distribution using a Markov Chain MC (MCMC) method [36].

Here, we would like to highlight that these methods discussed above (e.g., UVE and SPA) implemented the idea of model-population analysis (MPA) developed in our previous work [37]. The core of MPA is establishing data-analysis methods by statistically analyzing the distribution of an interested outcome of a population of sub-models derived with the aid of MC sampling {e.g. jackknife or bootstrap [25,38]}. Indeed, MPA has gained wide, successful applications in a variety of fields [17,19–21,31,32,36]. However, a comprehensive description of MPA is lacking. To facilitate understanding of MPA, here we first review its basic elements. Then, we describe and discuss algorithms and applications of selected MPA-based methods in outlier detection and variable selections.

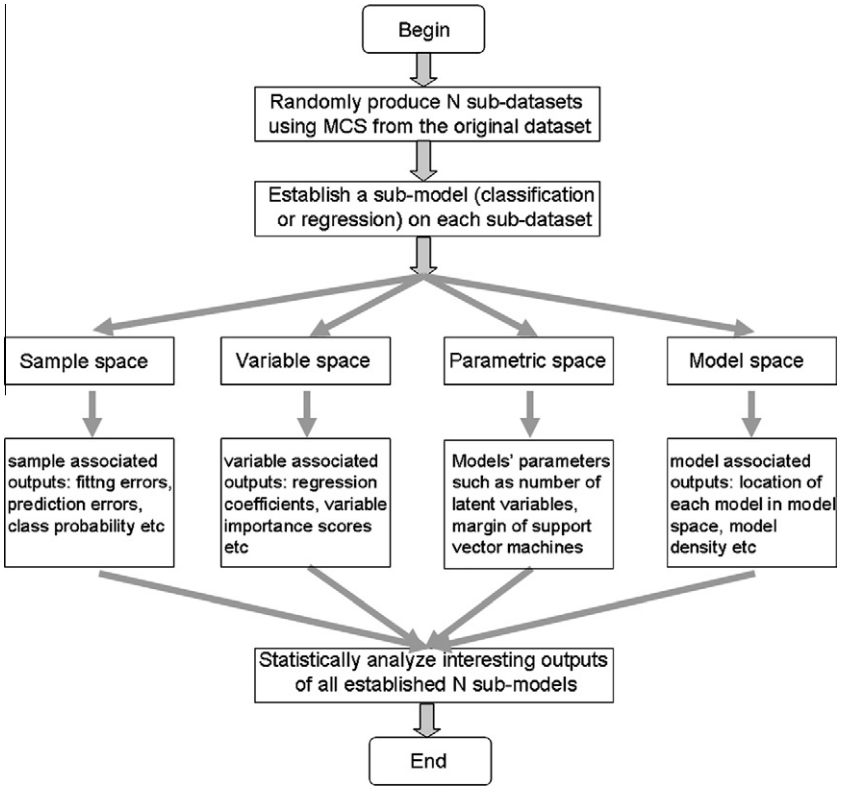

Figure 1. Model-population analysis.

2. Model-population analysis

MPA was recently proposed as a general framework for developing data-analysis methods. As depicted in Fig. 1, MPA works in three steps:

(1) MC sampling (MCS) is used to randomly draw N sub-datasets (e.g., 10,000);
(2) for each sub-dataset, a sub-model is built; and,
(3) last but not least, an outcome of interest (e.g., prediction errors) of all the N sub-models is statistically analyzed.

It is important to note that it is the third step, not the MC sampling, that is the key to MPA. As can be seen in Fig. 1, the parameters that can be statistically analyzed are put into four spaces: (1) sample space, (2) variable space, (3) parametric space and (4) model space. By analyzing an interesting parameter that is associated with one of the four spaces, a data-analysis algorithm can be developed. As an illustrative example, the MC method [17] for detecting outliers was designed by studying the distribution of prediction errors of each sample, which is a parameter associated with sample space.

2.1. Monte Carlo sampling for a sub-dataset

Sampling is a key tool in statistics, which allows creation of sub-datasets, from which an interested unknown parameter could be estimated. Given a dataset ($\mathbf{X}$, $\mathbf{y}$), assume that the design matrix $\mathbf{X}$ contains m samples in rows and p variables in columns, the response vector denoted by $\mathbf{y}$ is of size $m\times1$, and the number of MC sampling is set to N. At this setting, N sub-datasets can be drawn from N MC samplings with or without replacement. The N sub-datasets randomly sampled are denoted as $(\mathbf{X}_{sub}, \mathbf{y}_{sub})_i$, i = 1, 2, 3,...N.

2.2. Establishing a sub-model for each sub-dataset

For each sub-dataset $(\mathbf{X}_{sub}, \mathbf{y}_{sub})_i$, a sub-model can be built using a selected method [e.g., partial least squares (PLS) or SVMs or classification and regression tree (CART)]. Denote the sub-model established as f_i ($\mathbf{X}$). Then, all the sub-models can be gathered into a collection:

$$C = (f_1(X), f_2(X), f_3(X), \ldots, f_N(X)) \quad (1)$$

All these sub-models are expected jointly to provide comprehensive information on the original data.

2.3. Statistically analyzing an interesting output of all the sub-models

Statistical analysis of an interesting output (e.g., prediction errors or regression coefficients) of all the sub-models is the core of MPA. Different designs for the analysis of different outputs of all the sub-models will lead to different algorithms. As proof of principle, the analysis of the distribution of prediction errors has been shown to be effective in outlier detection [17], whereas the analysis of the distribution of prediction errors [31] or regression coefficients [29,30] proves to be useful in variable selection.

3. Applications of model-population analysis

3.1. Outlier detection

The recognition and the removal of outliers from measured data is a crucial step before modeling. The interpretability and the predictive performance of a calibration model built using data with outliers removed could be improved. A number of algorithms have been proposed for outlier detection and proved effective. These

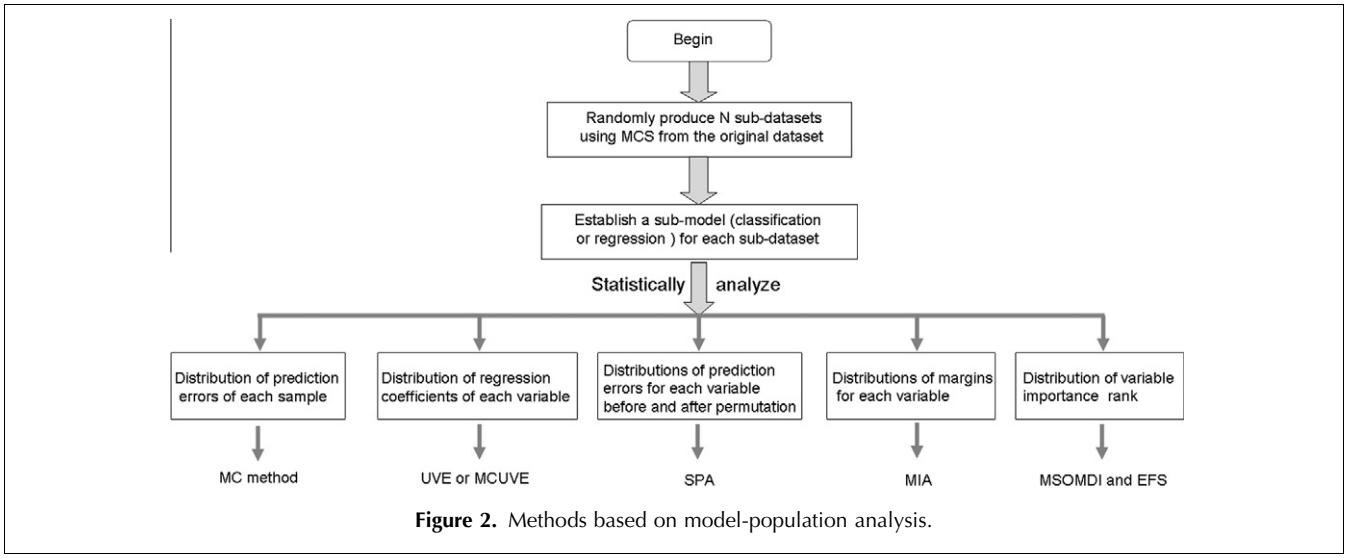

Figure 2. Methods based on model-population analysis.

algorithms include, but are not limited to, Mahalanobis distance, Cook's distance, minimum-volume ellipsoid (MVE), random sampling by half-means (RHM) [16], and the MC method [17]. Among these methods, the MC method implementing the idea of MPA has been shown to be promising in identifying both **X**-outliers and **Y**-outliers by analyzing the distribution of prediction errors of each sample. Fig. 2 shows the MC method, which comprises three steps, as described below.

(1) A percentage, denoted by r (e.g., $r = 0.80$), of samples is selected randomly as a training set. The remaining samples serve as an independent test set. This procedure is repeated N times, and N training sub-sets and N test sub-sets can be obtained.
(2) For each training sub-set, a sub-model is established and is then used to make predictions on the corresponding test sub-set.
(3) Each sample will be selected into a test set approximately $N(1-r)$ times by assuming that each sample is selected with equal probability. So, each sample is associated with around $N(1-r)$ prediction errors, of which the mean and the standard deviation can be computed. By plotting the standard deviation against the mean of the prediction errors of each sample, a diagnostic plot can be obtained that will be used for outlier detection.

Here, the mechanism of the MC method is illustrated using a benchmark near-infrared (NIR) dataset, the corn data (http://software.eigenvector.com/Data/index.html). The NIR spectra measured on an mp5 instrument is used and the chemical measurement to model is starch concentration. PLS is chosen for building a calibration model. The number of MC sampling N is set to 1000 and, at each sampling, 70% samples are randomly selected as a training sub-set to build a PLS model with nine latent variables determined using five-fold cross validation. Using the MC method, a diagnostic plot for outlier detection is obtained and shown in the left panel of Fig. 3.

As proof-of-principle, we selected three samples (Sample A, B and C in Fig. 3) that are most representative of a normal sample, an **X**-outlier and a **Y**-outlier, respectively. The distributions of prediction errors of these three samples are shown in the right panel. Clearly, the distribution of prediction errors of the normal sample has an approximately zero mean and a small standard deviation. For the **X**-outlier, the distribution exhibits a small absolute mean but a large standard deviation. By contrast, the absolute mean of the **Y**-outlier's prediction errors is much wider than that of both **X**-outlier and the normal sample. These results suggest that one prediction error from a single model cannot be used to conduct outlier detection and a distribution of prediction errors is much more informative to characterize the outlying propensity of a sample and therefore should be recommended for outlier detection.

To test whether the removal of outliers will improve prediction ability, three samples (Samples B, C and D, Fig. 3) with either a large mean or a large standard deviation of prediction errors are removed as outliers. The five-fold root mean squared error of cross validation (RMSECV) was calculated for comparison. The minimum RMSECVs achieved are 0.372 (nine PLS components) for

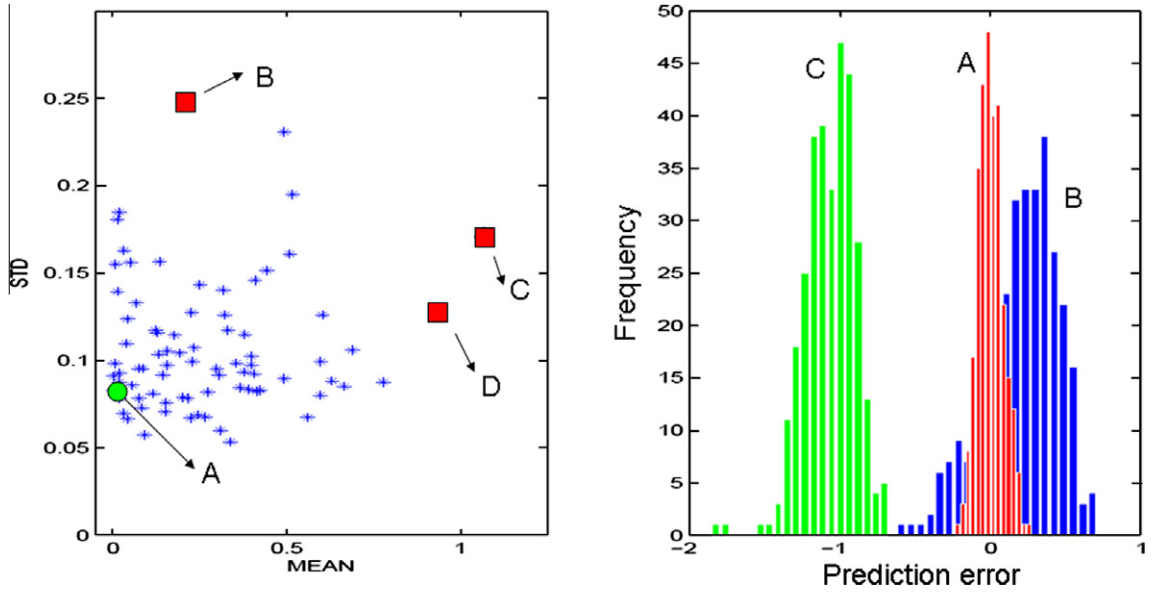

Figure 3. (Left) The diagnostic plot for outlier detection of the corn-starch data. Three samples that are most representative of (A) a normal sample, (B) an X-outlier, and (C) a Y-outlier are selected. (Right) The distributions of prediction errors of these three samples. Samples B, C and D are three outliers.

the original data and 0.305 (nine PLS components) for the reduced data with three outliers removed, indicating that removal of outliers indeed improves the predictive performance of NIR calibration models.

3.2. Variable selection

3.2.1. Monte Carlo uninformative variable elimination. Building upon the distribution of PLS regression coefficients of each variable resulting from the leave-one-out procedure, a reliability index, defined as the ratio of the mean to the standard deviation of this distribution, is used to assess variable importance in uninformative variable elimination (UVE). The use of the leave-one-out procedure limits the derivation of the distribution of regression coefficients. Recently, by borrowing the virtue of the MC technique, a modified version of UVE, called MC UVE (MC-UVE) was proposed [30]. No noise variables are used in MC-UVE, making it faster than UVE. Fig. 2 shows UVE or MC-UVE. And we detail the algorithm of MC-UVE below.

(i) A percentage, denoted by r (e.g., $r = 0.80$), of samples is selected randomly as a training sub-set. This procedure is repeated N times, and N training sub-sets are obtained.

(ii) For each training sub-set, a sub-model is established using, for example, PLS.

(iii) N regression coefficients are obtained and collected into a vector $\boldsymbol{c}$ for each variable and a distribution of these regression coefficients can be derived. The mean and the standard deviation of this distribution are denoted as mean($\boldsymbol{c}$) and sd($\boldsymbol{c}$), respectively. Then, a reliability index (RI), defined as the ratio of mean($\boldsymbol{c}$) to sd($\boldsymbol{c}$) in Equation (2), is used to assess the reliability of each variable. Based on this reliability, all variables are ranked. Then, these variables are sequentially added to build a PLS model whose performance is assessed using cross validation. The reliability index corresponding to the variable whose addition results in the minimum RMSECV value is chosen as the threshold. All variables that are associated with a reliability index lower than this threshold value can be eliminated.

$$\mathrm{RI} = \mathrm{mean}(\boldsymbol{c})/\mathrm{sd}(\boldsymbol{c}) \quad (2)$$

The mechanism of MC-UVE is illustrated with the corn-starch data, as used in Section 3.1. Nine PLS components, determined using five-fold cross validation, are used to build a PLS regression model. N and r are set to 1000 and 0.8, respectively. At these settings, 1000 regression coefficients are obtained for each variable. The reliability index calculated using Equation (2) is shown in the left panel of Fig. 4. From this plot, two wavelengths (marked A and C) of high reliability index and one wavelength with nearly zero-valued reliability index are selected and their distributions of regression coefficients are given in the right panel of Fig. 4. The fact that regression coefficients have a distribution (here caused by sample variation) indicates that assessment of variable importance using a single regression coefficient from only one model is unreliable. A distribution of regression coefficients reflects much more information about the data analyzed so we recommends using it.

To test the performance of MC-UVE, 67 wavelengths with the highest reliability index are selected using five-fold cross validation. The minimum five-fold RMSECV achieved using these selected 67 wavelengths is 0.358 (seven PLS components), indicating improvement over the full spectral model that has a minimum five-fold RMSECV 0.372 (nine PLS components), as shown

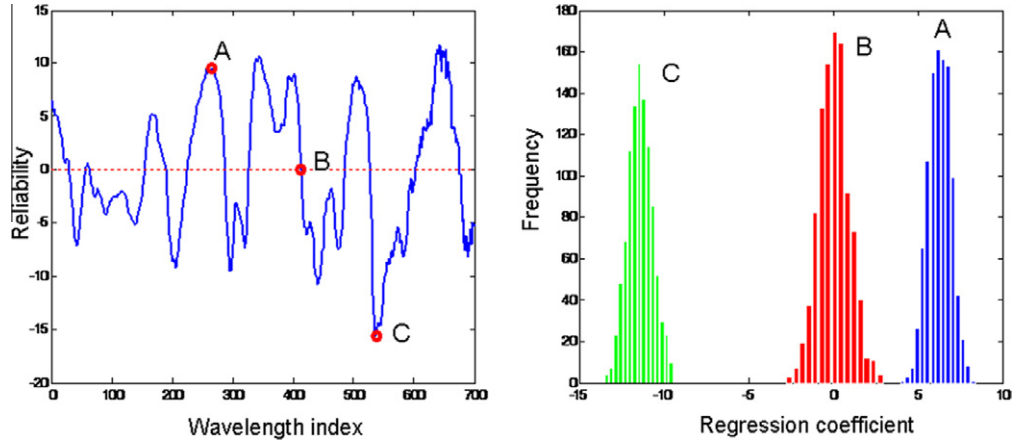

Figure 4. (Left panel) The reliability index of each wavelength resulting from Monte Carlo uninformative variable elimination for the corn-starch data. (Right panel) For illustration, the distributions of regression coefficients of three wavelengths (marked as A, B and C) resulting are displayed accordingly.

previously. Taken together, this modified UVE method proves to be effective in selecting informative variables and improving the prediction ability of NIR calibration models.

3.2.2. Subwindow permutation analysis. Aimed at exploring synergistic effects among multiple variables and motivated by the permutation technique for assessing variable importance in random forests (RF) [39], subwindow permutation analysis (SPA) was proposed for discriminant analysis in our previous work by following the framework of MPA [31]. A modified version of SPA, called noise-incorporated SPA (NISPA), was recently developed for variable selection of SVMs [10]. Only SPA is considered here, since the basic idea of NISPA is the same as that in SPA. As depicted in Fig. 2, SPA comprises three steps. Assuming that $\mathbf{X}$ is of size $n \times p$, we describe the algorithm of SPA below.

(1) At each iteration, Q ($<p$) out of the p variables are randomly selected and considered. A percentage denoted by r (e.g., $r = 0.80$) of samples is also selected randomly as a training sub-set with the remaining samples as a test sub-set. Repeating this procedure N times, N training sub-sets and N test sub-sets are obtained. Note that both training sets and test sets contains only Q variables.

(2) For each training sub-set, a sub-model for classification is established using, for example, partial least squares-linear discriminant analysis (PLS-LDA).

(3) For each established classification model, a normal prediction error (NPE) is first computed using the corresponding test sub-set. Then, only one out of Q variables in the test sub-set is permutated at a time and a permuted prediction error (PPE) is calculated. Thus, one NPE and Q PPEs are obtained by making predictions on the original as well as the permuted test sub-set. This procedure is repeated N times.

Without loss of generality, assuming that the jth variable has been selected J (approximately NQ/p) times, J PPEs as well as J NPEs can be obtained for the jth variable. Using a statistical test method, the difference between the distribution of NPEs and PPEs can be assessed, leading to a P value for each variable. This P value is transformed into a conditional synergetic score (COSS) for measuring variable importance using Equation (3):

$$\mathrm{COSS} = -\log_{10}(P) \tag{3}$$

Intuitively, if a variable is not random and is important, the prediction error will increase significantly when this variable is permuted, therefore resulting in a big difference between the distribution of NPEs and PPEs and hence a small P value (high COSS value). If a variable is random, no big difference is expected between distribution of NPEs and PPEs, thus giving a high P value (low COSS value). In this sense, variable importance can be assessed using the SPA method.

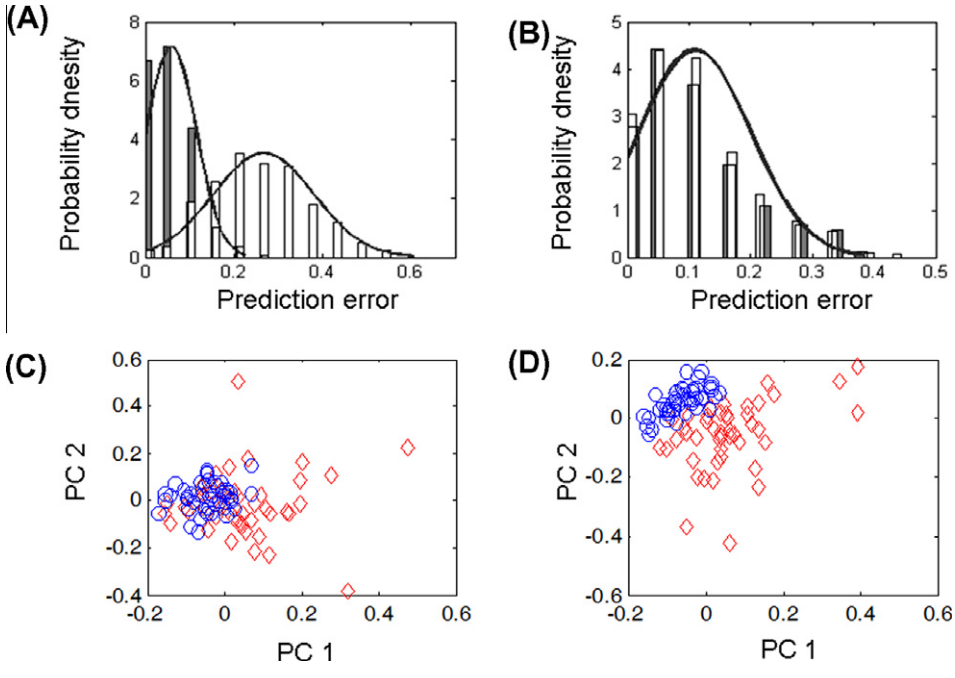

Figure 5. Plots A and B show the distributions of normal prediction errors (grey bar) and permuted prediction errors (white bar) of an informative metabolite (C18:1n-9, P = 0) and an uninformative one (C16:1n-7, P = 0.8791) for type 2 diabetes data, respectively. Samples projected onto the first two principal components (circle: normal, diamond: patients) using all the 21 metabolites and the 3 metabolites (OLA, ALA and EPA) selected by SPA are shown in Plots C and D, respectively.

In our previous work, SPA was applied to assess the importance of 21 metabolites in their association with type 2 diabetes using a dataset of 90 samples (45 healthy controls and 45 cases). The two tuning parameters (i.e. Q and N) were set to 10 and 1000, respectively. By comparing the distributions of NPEs and PPEs using Mann-Whitney-U test, the variable importance of each metabolite assessed by P value or COSS value was first calculated.

As an example, the distributions of NPEs and PPEs of an informative metabolite (C18:1n-9, $P = 0$) and an uninformative one (C16:1n-7, $P = 0.8791$) are presented in Fig. 5. Using 10-fold double cross validation, three metabolites [i.e. oleic acid (OLA), α-linolenic acid (ALA) and eicosapentaenoic acid (EPA)] were selected and collectively exhibited the lowest predictive error. A principal component analysis (PCA) was performed on both the original data and the reduced data with only these three metabolites. The resulting scores plots are shown in Fig. 5. By comparison, it was found that better separation is achieved using the three metabolites identified, suggesting that SPA is a good alternative for variable selection.

3.2.3. Margin influence analysis. SVMs are a kernel method originally developed for classification based on the principle of structural risk minimization [40,41]. SVMs have been gaining increasing applications in a variety of fields (e.g., NIR analysis, QSAR/QSPR, and gene-expression-based disease classification). It has been shown that prediction accuracy of the SVM-classification model could be improved by means of variable selection [32,35,42,43]. Based on MPA, a method dedicated to variable selection of SVMs, called MIA, was proposed in our previous work [32]. MIA is shown in Fig. 2. Assuming that $\mathbf{X}$ is of size $n \times p$, the algorithm of MIA is detailed below.

(1) At each iteration, Q ($<p$) out of the p variables are randomly selected to obtain a training sub-set of size $n \times Q$. N training sub-sets are attained by repeating this procedure N times.
(2) For each training sub-set, an SVM classifier with tuning parameters optimized using cross validation is built.
(3) The margin of each SVM model established is then calculated and recorded. In doing so, N margin values are obtained, denoted as m_i ($i = 1, 2, \ldots, N$). Without loss of generality, the jth variable is taken to illustrate the mechanism of MIA. Based on the jth variable, the N margins can be divided into two groups, denoted by Group A and Group B. Group A collects the margins associated with SVM model that includes this variable, while all the remaining margins belong to Group B.

Suppose that the numbers of margins in these two groups are $N_{j,A}$ and $N_{j,B}$, respectively. Thus, $N_{j,A} + N_{j,B} = N$. Further denote the means of these two groups of margins as $MEAN_{j,A}$ and $MEAN_{j,B}$, of which the difference is calculated as:

$$DMEAN_j = MEAN_{j,A} - MEAN_{j,B} \quad (4)$$

It can be inferred from Equation (4) that the inclusion of the jth variable in an SVM model increases the margin if $DMEAN_j > 0$ and *vice versa*. In this sense, variables with $DMEAN_j < 0$ are first removed and then the Mann-Whitney U test is employed to compare the distributions of the two groups of margins to examine whether the increment of margin in Group A over Group B is significant, leading to a P value for each variable. This P value is used to evaluate the variable important in MIA.

In our previous work, MIA was used to select gene-expression traits that can increase the margin of an SVM model for colon-cancer classification. Linear kernel was chosen for building SVM models. The penalizing factor C of SVM was optimized using cross validation. Q and N were chosen to be 200 and 10,000, respectively. In this setting, we obtained 10,000 margin values resulting from 10,000 SVM sub-models. The margin distributions of an informative gene and an uninformative one are shown in Fig. 6. For example, this selected informative gene can on average significantly increase the margin of

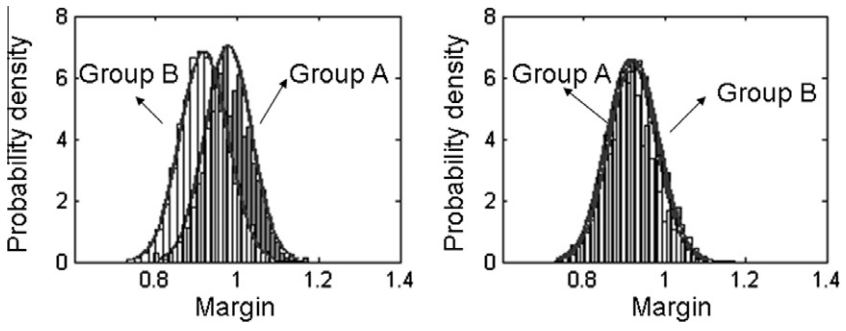

Figure 6. The margin distributions of SVM models of an informative gene expression trait (left, DMEAN = 0.060, $P = 5.64 \times 10^{-181}$) and an uninformative one (right, DMEAN = −0.009, $P = 0.11$).

an SVM model by 0.060. By contrast, a decrease in margin by 0.009 is observed for the uninformative gene.

To test the performance of MIA, we built SVM classification models using a certain number of informative genes and evaluated these models by leave-one-out cross validation (LOOCV). The minimum classification error from LOOCV achieved was 0.00 with 100 informative genes included, indicating significant improvement over the SVM model including all 2000 genes with LOOCV error 17.74. A comprehensive comparison of MIA with other methods was presented in Table 3 in the reference [32]. Summing up, MIA is by design an intuitive method for variable selection of SVM and was shown to be effective in singling out informative genes in our work.

3.2.4. Determination of significant variables for supervised self-organizing maps. SOMs [44], also called Kohonen neural networks, were developed by Kohonen for projecting samples in a high dimensional space onto a 2D plane, thus providing visualization of how samples get clustered. Compared to the most commonly used dimension-reduction method, PCA, SOMs are more robust to outliers and can handle non-linear problems. Later, supervised SOMs were introduced by incorporating the class information of training samples. As is known, for classification of high dimensional samples, variable selection can often improve classification results. Based on MC sampling, Wongravee et al. [33] described an extension of the SOMs discrimination index (SOMDI) by using supervised SOMs to determine potentially significant variables that are responsible for class separation. Since this method can be seen as an MPA approach, given two classes of samples, its algorithm is briefly reviewed as follows:

(1) At each iteration, a percentage denoted by r (e.g., $r = 0.67$) of samples is selected randomly as a training sub-set. Repeating this procedure N times, N training sub-sets are obtained.
(2) For each training sub-set, the SOMDI scores of all variables are calculated for both the ''in-group'' and ''out-group'' samples, which are stored into two p-dimensional vectors (i.e. $\mathbf{S}_{in}$ and $\mathbf{S}_{out}$) of which the difference is computed as $S\Delta = \mathbf{S}_{in} - \mathbf{S}_{out}$. Any variable associated with a negative value of $S\Delta$ is treated as an unranked variable, whereas those variables with positive $S\Delta$ values are assigned a rank.
(3) Compute the average rank of each variable over the N training sub-sets to provide an overall estimate of variable rank, which is then used to determine which variables are significant.

The proposed method was applied to a nuclear magnetic resonance (NMR)-based metabolic dataset of 96 saliva samples. The performance of this method was discussed [33].

3.2.5. Ensemble-based robust biomarker identification. Biomarker discovery is of particular use in biomedical applications for understanding biological data [8,28,35]. As stated by Abeel et al. [34], the selection stability of biomarkers with respect to sampling variation has received attention only recently, and it may greatly influence subsequent biological validations. In biomedical practice, it would therefore be very important to improve the reliability or the statistical significance of selected biomarkers. To this end, a method that was shown to be effective in improving robustness of biomarker identification based on ensemble was proposed by Abeel et al. [34]. This method falls into the category of MPA approaches, so we briefly introduce it here by analogy with previous methods described.

(1) At each iteration, the bootstrapping method is used to generate a training sub-set from the original training data. Repeating this procedure N times, N training sub-sets are obtained.
(2) For each training sub-set, linear SVMs coupled with recursive feature elimination is used to perform variable selection. Then, each variable is assigned a rank, which can be collected into a p-dimensional vector. After N iterations, a matrix of size $N \times p$ that records ranks of all variables in N iterations is obtained.
(3) The ensemble ranking is derived by summing the ranks over N iterations or taking a weighted summation of the ranks over N iterations with the AUC of each linear SVM model as weight.

The authors evaluated the proposed methodology using four microarray datasets and found an increase of up to almost 30% in robustness of the selected biomarkers along with an improvement of around 15% in prediction accuracy [34]. These results provided a good example that, with the aid of MC sampling, variable-selection stability can be improved significantly.

4. Concluding remarks

By highlighting the common principle of a spectrum of MC-based algorithms, MPA has been proposed as a general framework for data analysis.

In the context of chemical and biological modeling, the application of MPA-based methods in outlier detection and variable selection demonstrates that the use of a population of sub-models could provide comprehensive information of the data and hence should promise better understanding and modeling of the data analyzed. The methods and the applications selected in this review are limited, but we hope that they can represent a general, systematic framework for chemical and biological modeling.

Acknowledgements
This work was financially supported by the National Nature Foundation Committee of PR China (Grants No. 20875104 and No. 21075138) and the Graduate Degree Thesis Innovation Foundation of Central South University (CX2010B057). The study was approved by the review board of Central South University.

References

[1] B. Walczak, D.L. Massart, Chemometr. Intell. Lab. Syst. 41 (1998) 1.
[2] I.M. Johnstone, D.M. Titterington, Phil. Trans. R. Soc. A 367 (2009) 4237.
[3] Y. Saeys, I. Inza, P. Larranaga, Bioinformatics 23 (2007) 2507.
[4] K. Hasegawa, K. Funatsu, Curr. Comput.-Aid. Drug Des. 6 (2010) 1.
[5] R.J. Pell, Chemometr. Intell. Lab. Syst. 52 (2000) 87.
[6] M. Hubert, K. Vanden Branden, J. Chemometr. 17 (2003) 537.
[7] G.C. Cawley, N.L.C. Talbot, Bioinformatics 22 (2006) 2348.
[8] K.Y. Yeung, R.E. Bumgarner, A.E. Raftery, Bioinformatics 21 (2005) 2394.
[9] T. Chen, E. Martin, Anal. Chim. Acta 631 (2009) 13.
[10] Q. Wang, H.-D. Li, Q.-S. Xu, Y.-Z. Liang, Analyst (Cambridge, UK) 136 (2011) 1456.
[11] H. Zou, T. Hastie, J. R. Statist. Soc. B 67 (2005) 301.
[12] E. Candes, T. Tao, Ann. Statist. 35 (2007) 2313.
[13] R. Tibshirani, J. R. Stat. Soc. B 58 (1996) 267.
[14] T. Rajalahti, R. Arneberg, A.C. Kroksveen, M. Berle, K.-M. Myhr, O.M. Kvalheim, Anal. Chem. 81 (2009) 2581.
[15] H.-D. Li, Y.-Z. Liang, Q.-S. Xu, D.-S. Cao, Anal. Chim. Acta 648 (2009) 77.
[16] W.J. Egan, S.L. Morgan, Anal. Chem. 70 (1998) 2372.
[17] D.S. Cao, Y.Z. Liang, Q.S. Xu, H.D. Li, X. Chen, J. Comput. Chem. 31 (2010) 592.
[18] Q.-S. Xu, Y.-Z. Liang, Chemometr. Intell. Lab. Syst. 56 (2001) 1.
[19] A.T. Brunger, G.M. Clore, A.M. Gronenborn, R. Saffrich, M. Nilges, Science (Washington, DC) 261 (1993) 328.
[20] N. Shu, T. Zhou, S. Hovmoller, Bioinformatics 24 (2008) 775.
[21] T.S. Furey, N. Cristianini, N. Duffy, D.W. Bednarski, M. Schummer, D. Haussler, Bioinformatics 16 (2000) 906.
[22] V.E. De Monte, G.M. Geffen, C.R. May, K. McFarland, J. Clin. Exp. Neuropsych. 26 (2004) 628.
[23] Q.S. Xu, Y.Z. Liang, Y.P. Du, J. Chemometr. 18 (2004) 112.
[24] P. Filzmoser, B. Liebmann, K. Varmuza, J. Chemometr. 23 (2009) 160.
[25] B. Efron, G. Gong, Am. Stat. 37 (1983) 36.
[26] B. Efron, Techn. Rep. 78, Stanford University, Stanford, CA, USA, 1982.
[27] J. Shao, J. Am. Stat. Assoc. 88 (1993) 486.
[28] T. Rajalahti, R. Arneberg, F.S. Berven, K.-M. Myhr, R.J. Ulvik, O.M. Kvalheim, Chemometr. Intell. Lab. Syst. 95 (2009) 35.
[29] V. Centner, D.-L. Massart, O.E. de Noord, S. de Jong, B.M. Vandeginste, C. Sterna, Anal. Chem. 68 (1996) 3851.
[30] W. Cai, Y. Li, X. Shao, Chemometr. Intell. Lab. Syst. 90 (2008) 188.
[31] H.-D. Li, M.-M. Zeng, B.-B. Tan, Y.-Z. Liang, Q.-S. Xu, D.-S. Cao, Metabolomics 6 (2010) 353.
[32] H.-D. Li, Y.-Z. Liang, Q.-S. Xu, D.-S. Cao, B.-B. Tan, B.-C. Deng, C.-C. Lin, IEEE/ACM Trans. Comput. Biol. Bioinform. 8 (2011) 1633.
[33] K. Wongravee, G.R. Lloyd, C.J. Silwood, M. Grootveld, R.G. Brereton, Anal. Chem. 82 (2010) 628.
[34] T. Abeel, T. Helleputte, Y. Van de Peer, P. Dupont, Y. Saeys, Bioinformatics 26 (2010) 392.
[35] I. Guyon, J. Weston, S. Barnhill, V. Vapnik, Mach. Learn. 46 (2002) 389.
[36] Y. Ai-Jun, S. Xin-Yuan, Bioinformatics 26 (2009) 215.
[37] H.-D. Li, Y.-Z. Liang, Q.-S. Xu, D.-S. Cao, J. Chemometr. 24 (2009) 418.
[38] B. Efron, R. Tibshirani, An Introduction to the Bootstrap, Chapman & Hall, New York, USA, 1993.
[39] L. Breiman, Mach. Learn. 45 (2001) 5.
[40] H.-D. Li, Y.-Z. Liang, Q.-S. Xu, Chemometr. Intell. Lab. Syst. 95 (2009) 188.
[41] W.S. Noble, Nat. Biotechnol. 24 (2006) 1565.
[42] Y.X. Zhang, Y. Aksu, G. Kesidis, D.J. Miller, Y. Wang, SVM margin-based feature elimination applied to high-dimensional microarray gene expression data, IEEE Workshop Mach. Learn. Signal Process., 2008, p. 97.#.
[43] O. Gualdrón, J. Brezmes, E. Llobet, A. Amari, X. Vilanova, B. Bouchikhi, X. Correig, Sens. Actuators, B 122 (2007) 259.
[44] T. Kohonen, Construction of Similarity Diagrams for Phonemes by a Self- Organising Algorithm, Helsinki University of Technology, Espoo, Finland, 1981.