Bagging for robust non-linear multivariate calibration of spectroscopy

Ke Wang, Tao Chen*, Raymond Lau

School of Chemical and Biomedical Engineering, Nanyang Technological University, 62 Nanyang Drive, Singapore 637459, Singapore

Abstract
This paper presents the application of the bagging technique for non-linear regression models to obtain more accurate and robust calibration of spectroscopy. Bagging refers to the combination of multiple models obtained by bootstrap re-sampling with replacement into an ensemble model to reduce prediction errors. It is well suited to "non-robust" models, such as the non-linear calibration methods of artificial neural network (ANN) and Gaussian process regression (GPR), in which small changes in data or model parameters can result in significant change in model predictions. A specific variant of bagging, based on sub-sampling without replacement and named subagging, is also investigated, since it has been reported to possess similar prediction capability to bagging but requires less computation. However, this work shows that the calibration performance of subagging is sensitive to the amount of sub-sampled data, which needs to be determined by computationally intensive cross-validation. Therefore, we suggest that bagging is preferred to subagging in practice. Application study on two near infrared datasets demonstrates the effectiveness of the presented approach.

Keywords: Bootstrap aggregating; Ensemble modelling; Near infrared spectroscopy; Non-linear calibration; Robust model.

1. Introduction
Recently, spectroscopy in combination with multivariate calibration models has become an important tool for fast and non-invasive analysis in pharmaceutical, petrochemical and food sectors [1-4]. Principal component regression (PCR) and partial least squares (PLS) [5] are traditional linear calibration techniques. In practice, non-linear variations can be induced to the spectra in the presence of external disturbances, such as light scattering effect [6], temperature and pressure variation [7-9], and instrumental variation of background noise and baseline drift [10-11]. Under these conditions, linear calibration techniques may give inaccurate predictions. Several strategies have been proposed to deal with the non-linearity, such as pre-processing [12] (standard normal variate (SNV) [13], extended multiplicative signal correction (EMSC) [14], etc.), non-linear calibration techniques (artificial neural network (ANN) [15-17], Gaussian processes regression (GPR) [18-19]), and variable selection [20-21]. However, these models may be non-robust in the sense that small change in the calibration data and/or model parameters can result in significant change in model predictions. Ensemble modelling is an effective way to improve model robustness [22]. The general idea of ensemble modelling is to construct multiple models and combine their predictions into a single value using certain rules [23]. Bagging is among the most important ensemble modelling techniques and will be further investigated in this study.

Bagging, short for ‘bootstrap aggregating’, was originally contrived by Breiman [24-25]. It generates different models from a series of training sets obtained by a bootstrap re-sampling strategy. Then, the models

* Corresponding author. Tel.: +65 6513 8267; Fax: +65 6794 7553 (T. Chen).

Email addresses: chentao@ntu.edu.sg (T. Chen).
are combined in a certain way to make prediction. The bootstrap, developed by Efron [26-27], is a technique of forming different training sets by randomly selecting a fixed number of data points from the original training set with replacement. Many studies have confirmed the efficiency of bagging for reducing prediction errors [28-30]. It has also been demonstrated that bagging works especially well for non-robust models [24].

In the regression field, bagging was initially applied to regression trees [24]. Subsequently, attention has been focused on bagging neural networks [31-36]. Recently, bagging was developed for other regression methods, such as partial least squares (PLS), multiple linear regression (MLR) [37], and Gaussian process regression (GPR) [38]. Over the years, bagging has been modified in several ways such as “nice” bagging [39], iterated bagging [40], subagging (sub-sample aggregating, based on sub-sampling without replacement) [41], and trimmed bagging [42]. Among these modified methods, subagging was reported to provide similar performance to bagging with less computation, since it uses a subset of the data for model development [41].

For the purpose of multivariate spectroscopic calibration, subagging has already been applied to linear methods as PLS and MLR with variable selection [37]; yet its role in improving non-linear calibration methods (such as ANN and GPR) has been under-explored. Previous studies demonstrated that non-linear regression for spectroscopic calibration can achieve accurate prediction of analyte properties [15-18]. However, their prediction performance may be sensitive to small change in calibration data and/or model parameters. Therefore, the major contribution of this work is to apply bagging and subagging for non-linear calibration methods (specifically ANN and GPR) to obtain more accurate and robust predictions. In addition, we examine the effect of the amount of sub-sampled training data on the prediction performance of subagging, a topic that was not carefully examined in the literature of chemometric calibration. The results indicate that subagging is sensitive to the amount of sub-sampled data, which needs to be determined by the computationally intensive cross-validation method. Therefore, we suggest that bagging is preferred to subagging in practice.

Two publicly available near infrared (NIR) datasets will form the basis of the current case study. During the development of the calibration models, each dataset needs to be divided into training and testing data. It is crucial to ensure that the training/testing data are representative of the whole dataset in order to conduct reliable and meaningful evaluation. This objective can be implemented by the application of the SPXY algorithm (sample set partitioning based on joint x-y distances) [43], which is a development of the classic Kennard-Stone algorithm [44]. The predictive accuracy, quantified by root mean square error of prediction (RMSEP), is used to evaluate the performance of the investigated regression techniques.

2. Non-linear calibration techniques

This section briefly presents the theory of ANN and GPR. Throughout this paper, $x_{ij}$ denotes the $i^{th}$ ($i = 1, \ldots, I$) measured spectra at the $j^{th}$ ($j = 1, \ldots, J$) wavelength (variable), and $x_i = [x_{i1}, x_{i2}, \ldots, x_{ij}]$; $y_i$ represents the corresponding response variable for the $i^{th}$ sample; $y^*$ indicates the predicted response variable.

2.1. Artificial neural network (ANN)

As a flexible modelling tool, ANN has a wide application in many areas such as pattern recognition, optimization and control [45]. In the 1990s, ANN was introduced in the chemometrics community for spectroscopic calibration [16-17]. A typical feed-forward ANN model consists of three layers (input, hidden and output layer), each layer comprising multiple neurons. In spectroscopic calibration, the response variable to be predicted $y_i$ can be expressed mathematically as [46]:

2
\[ y_i = f \left[ \sum_{n=1}^{N} \omega_n g \left( \sum_{j=1}^{I} (w_j x_{ij} + \phi_j) \right) + \epsilon_n \right] \]  

where \( N \) is the number of hidden-layer neurons, \( w_j \) represents the weights connecting the input- and hidden-layer neurons, \( \omega_n \) indicates the weights connecting the hidden- and output-layer neurons, \( \phi_j \) and \( \epsilon_n \) are the biases in the hidden and output layers, respectively. The two “transfer functions”, \( f(\cdot) \) and \( g(\cdot) \), are typically taken as linear and sigmoid functions, respectively, since such a neural network is capable of approximating any function to arbitrary accuracy under certain assumptions [45].

The parameters in an ANN are estimated through optimization algorithm to minimize the output error \( \sum_{i=1}^{I} (\hat{y}_i - y_i)^2 \) by adjusting the weights [46]. “Back-propagation” algorithm is a typical estimation method of calculating the gradient information efficiently for this non-linear optimization problem [15]. In this study, we adopt a Bayesian regularized version of back-propagation algorithm. Bayesian back-propagation was initially introduced by MacKay [47-48], based on a Gaussian approximation to the posterior distribution of parameters. Previous study has shown that the Bayesian approach to parameter estimation typically attains more robust and accurate ANN models [49].

2.2. Gaussian process regression (GPR)

GPR was originally presented by O’Hagan [19] and it can be derived from the perspectives of ANN and Bayesian regression [50-51]. GPR has recently been applied to spectroscopic calibration modelling [18], with outstanding prediction accuracy being achieved. In GPR model, the response variable \( y_i \) is modelled by a joint Gaussian distribution with zero mean:

\[ y = (y_1, ..., y_I) \sim G(0, \Sigma) \]  

where \( \Sigma \) is an \( I \times I \) covariance matrix whose elements are defined by the covariance function: \( \Sigma_{ik} = \Sigma(x_i, x_k) \). The following covariance function, widely used in the literature, is adopted in this study:

\[ \Sigma(x_i, x_k) = a_0 + a_1 \sum_{j=1}^{I} x_{ij} x_{kj} + \nu_0 \exp \left( -\sum_{j=1}^{I} w_j (x_{ij} - x_{kj})^2 \right) + \sigma^2 \delta_{ik} \]  

where the first two terms represent the constant bias and linear correlation, respectively. The third term is similar to the form of the radial basis function and the fourth term corresponds to the random error. The use of both linear and non-linear terms in the covariance function allows GPR to deal with both linear and non-linear problems.

The model parameters \( \theta = (a_0, a_1, \nu_0, w_1, ..., w_I, \sigma^2) \), defining the covariance function in eq. (3), can be estimated by maximizing the following log-likelihood function:

\[ \log p(y|\theta) = -\frac{1}{2} \log |\Sigma| - \frac{1}{2} y^T \Sigma^{-1} y - \frac{I}{2} \log (2\pi) \]  

Given the estimated parameters, for the newly measured spectra \( x^* \), the prediction of response variable \( \hat{y} \) is also Gaussian distributed with the following mean:

\[ E(\hat{y}) = k^T(x^*) \Sigma^{-1} y \]  

where \( k(x^*) = [\Sigma(x^*, x_1), ..., \Sigma(x^*, x_I)]^T \).

3. Bagging and subagging for non-linear multivariate calibration

The principle of bagging is conceptually straightforward. In bagging, a number of models are developed from a re-sampling process on the original training (calibration) data with replacement. Suppose the original training data are \( Z = \{x_i, y_i\} \), where \( i = 1, ..., I \). A new re-sampled training set is constructed by randomly
selecting \( I \) data points from \( Z \) with replacement. This process is repeated for \( K \) times to obtain \( K \) different training sets. Then, \( K \) calibration models can be built from the \( K \) re-sampled training sets. For a testing data point, \( K \) predicted values can be obtained from the \( K \) models, and these \( K \) values are combined in a certain way to form the ultimate prediction. A simple averaging rule [24] was recommended and is adopted in this study. Specifically, the final prediction \( \hat{y} \) of the ensemble model is calculated as

\[
\hat{y} = \frac{1}{K} \sum_{k=1}^{K} \hat{y}(k)
\]

where \( \hat{y}(k) \) is the prediction of the response variable by the \( k^{th} \) model developed from the \( k^{th} \) training set.

The principle of subagging is similar to bagging. In subagging, \( K \) new calibration models are constructed by randomly sub-sampling \( P \ (P \leq I) \) data points from \( Z \) without replacement for \( K \) times. Subsequently, the \( K \) models are combined to make prediction.

In summary, the process of bagging/subagging for non-linear calibration method includes two steps: 1) obtaining \( K \) calibration models from the \( K \) re-sampled/sub-sampled training datasets; 2) combining the resulting models to generate an ensemble model and making prediction. The fundamental reason for the effectiveness of bagging unstable models is that the predictive capability of the regression models is based on the bias/variance trade-off, and the variance of the aggregated predictor is reduced and close-to-constant bias is maintained [52].

4. Application study

4.1. Datasets

These proposed techniques of bagging/subagging are evaluated on two NIR spectral datasets. The first dataset is related to the transmittance spectra of wheat kernels [53]. In this dataset, 523 samples from three different locations were analyzed at 100 wavelengths in the range of 850-1050nm. The data are available at http://www.models.life.ku.dk/research/data/wheat_kernels/. The objective of the analysis is to predict the protein content of wheat kernels. This dataset is divided into 415 and 108 samples for training and testing, respectively.

The second dataset, named “meat”, is publicly available at http://lib.stat.cmu.edu/datasets/tecator. The data were collected on a Tecator Infratec Food and Feed Analyzer, consisting of 215 samples with 100 wavelengths in the range of 850-1050 nm. The objective of the analysis is to predict the moisture content (water), fat and protein in finely chopped meat. In this study, the training data consist of 172 samples and the rest 43 samples are used for testing.

4.2. Implemental details

Root mean square error of prediction (RMSEP) is used to evaluate the performance of the investigated calibration methods. To better evaluate the advantage of bagging/subagging, a relative RMSEP reduction was also calculated as [37]

\[
\frac{\langle \text{RMSEP}_{\text{individual}} \rangle - \langle \text{RMSEP}_{\text{(su)bagging}} \rangle}{\langle \text{RMSEP}_{\text{individual}} \rangle} \times 100\%
\]

which indicates the improvement of (su)bagging procedure with respect to ANN and GPR.

When dividing the limited dataset into training and testing data, the training and testing data should be representative of the entire dataset. For this purpose, the SPXY algorithm [43], developed from the classic Kennard-Stone algorithm [44], is employed to extract the representative training data, while the rest are used for testing. The actual partition based on SPXY is dependent on the initially selected training sample. Hence,
the SPXY-partition is repeated 20 times with different random initializations, and the average RMSEP of the 20 repeats is reported to evaluate the prediction performance. It is to note that the SPXY algorithm is based on a linear distance function; nevertheless its effectiveness for non-linear regression models has been reported in the literature [54].

For benchmark comparison, bagging/subagging is also applied to the linear PLS method by selecting the number of latent variables (LVs) using five-fold random-splitting cross-validation. The influence of the amount of sub-sampled training data on the prediction performance of subagging is determined by using different ratios of sub-sampled/whole training data (i.e. \( P/I \)) from 0.1 to 1 with a step of 0.1. In practice, this ratio should be selected automatically, and a four-fold random-splitting cross-validation is used in this study. For the purpose of bagging/subagging, 20 models are developed for PLS, ANN and GPR. The selection of 20 models is supported by the preliminary study to be explained in the next section. Prior to developing the non-linear calibration methods (ANN and GPR), PLS is first applied to reduce the dimensions of the data, which is a usual method to reduce the computational cost of parameter estimation in the non-linear models [15, 18]. The number of LVs in this PLS-based “pre-processing” is also selected by five-fold random-splitting cross-validation. Subsequently, ANN and GPR are developed from the PLS scores, i.e. variable \( x \) denotes PLS scores as opposed to the spectra. A more rigorous approach may be used to select the number of LVs by minimizing the cross-validation errors of the hybrid PLS-ANN (or PLS-GPR) model. However, this method significantly increases the computational cost. In addition, our other study (details not reported here) shows that the performance of ANN and GPR is insensitive to the number of selected LVs, since the cross-validation procedure on PLS alone has selected a sufficient number of LVs.

One important consideration in calibration is outlier detection. It is well known that including outliers in model development has adverse impact on the prediction accuracy. Due to its importance, outlier detection has been extensively discussed in the literature; an overview of the current status and some recent development can be found in [55-57]. In this work, the built-in outlier detection capability of PLS [5] is utilized and no obvious outliers are found in the two datasets under investigation.

All computation was carried out in Matlab. The Matlab PLS toolbox version 5.2 (Eigenvector Research, Inc., Wenatchee, WA, USA) was used to perform PLS. The Matlab Neural Network toolbox was employed for the implementation of ANN. The Matlab code for Gaussian process was described in [51] and is available at http://www.gaussianprocess.org/gpml/code/matlab/doc/. The computation time was based on a Pentium 2.4 GHz desktop computer with 1 GB memory running Windows Vista.

5. Results and discussions

Fig. 1 illustrates the capability of bagging to reduce the model instability on the “wheat kernel” dataset. Fig. 1(a)(b)(c) display the RMSEPs of 50 individual GPR, ANN and PLS models, respectively. The prediction performances of 50 models, developed from re-sampled training data with replacement, are quite different, indicating the instability of a single GPR, ANN or PLS model. Fig 1(d)(e)(f) show that bagging improves the calibration accuracy and robustness. It appears that the combination of 10 or more models results in satisfactory prediction. (Results similar to Fig. 1 are obtained for subagging and thus not repeated here.) However, in order to obtain additional assurance, bagging/subagging 20 individual models are adopted subsequently.

(Fig. 1 about here)
Table 1 shows the RMSEPs of individual calibration methods (PLS, GPR and ANN) and subagging GPR, ANN and PLS, and the relative improvement of the prediction accuracy by using subagging with different sub-sampling ratios for the “wheat kernel” data. All results (including RMSEP and its improvement, number of LVs and computational time) are averaged over 20 repeated partitions of the dataset via the SPXY algorithm. For this dataset, individual GPR and ANN give better performance than PLS, and their RMSEPs are 0.45 and 0.48, respectively. Then, we investigate the effect of different sub-sampling ratios of training data on the predictive performance. Table 1 shows that for the ratios of 0.1 to 0.6, the RMSEP of subagging GPR is higher than that of individual GPR because the sub-sampled training data is not sufficient to cover the variation of the entire dataset. From the ratio of 0.7, the RMSEPs of subagging GPR start to be lower than that of individual GPR. At the ratios of 0.8 and 0.9, it attains the greatest improvement of 10.4%. Subagging ANN gives the similar prediction performance to subagging GPR, and also achieves satisfactory improvement. However, subagging PLS does not achieve better performance than PLS at any ratio. Fig. 2 displays the RMSEP trend of subagging GPR, ANN and PLS. It can be observed that in general, the RMSEP of subagging decreases with the increased sub-samples. The only exception is at “ratio=1” for subagging GPR and ANN, where the RMSEP is higher than at the ratio of 0.9. In fact, “ratio=1” essentially means no sub-sampling and all the 20 models are developed from the same original data but with different initialization of the parameter values. For non-linear regression, the parameter estimation is a non-linear optimization problem and different estimates may be obtained. Combining these 20 models helps to reduce the model sensitivity to the variation of parameters (but not to the variation of data). Therefore, subagging at “ratio=1” is not preferred.

Previous study indicated that subagging is a useful variant of the original bagging algorithm with similar prediction performance but improved computational efficiency, because it uses less data [41]. However, results in Table 1 suggest that the selection of sub-sampling ratio is crucial to the success of subagging. The selection of sub-sampling ratio should be accomplished by cross-validation, which requires additional computation time. For comparison, the prediction accuracy and the computation time of various methods on the “wheat kernel” dataset are summarized in Table 2. Comparing Table 1 and 2, it is clear that if the optimal ratio is known a priori, subagging gives similar accuracy but requires less computation than bagging. This advantage is especially remarkable for GPR since its computation increases in cubic order with the increase of data [51]. However, the need of cross-validation dramatically increases the computation time of subagging. For example, subagging GPR and ANN by cross validation took approximately 385 and 24.6 minutes, respectively, as compared with 76.5 and 0.6 minutes for bagging. Hence, in practice bagging is preferred to subagging. For the other dataset of “meat”, only bagging is further investigated.

The “meat” dataset is further used to validate the performance of bagging, whereby the results are summarized in Table 3. The results on the “meat” dataset are similar to those on the “wheat kernel”: the prediction accuracy of GPR and ANN is better than PLS; bagging GPR and ANN gives superior accuracy to individual methods in most cases; bagging PLS does not achieve significant improvement. More specifically, for moisture content, bagging attains excellent improvement of 16.9% and 11.1% on GPR and ANN, respectively. For the prediction of fat, bagging GPR shows outstanding prediction (relative improvement of 39.8%), while bagging ANN also performs satisfactorily in the term of 7.2% improvement. For protein,
bagging GPR does not show any advantage; but the results of bagging ANN are favourable with 14.5% improvement.

(Table 3 about here)

6. Conclusions

This paper explores the application of bagging/subagging for non-linear calibration of NIR spectroscopy with the aim to improve the prediction accuracy and robustness. The results have confirmed the effectiveness of bagging for the investigated non-linear models (GPR and ANN), while bagging linear PLS does not show significant advantage. Subagging possesses similar prediction performance to bagging. However, in practice the sub-sampling ratio of subagging needs to be optimized by cross-validation, which greatly increases the computation load. Therefore, bagging is recommended in practice. Clearly, by using multiple models, the computational cost at the model development stage inevitably multiplies. However, the improved prediction accuracy as demonstrated in this study may well justify the additional computation, which is becoming an inexpensive resource with the rapid development of computers.

In the future, we expect to explore alternative ensemble modelling techniques (e.g. boosting), as well as the practical application to in-line real-time monitoring of chemical and pharmaceutical processes.

References


Fig. 1. Prediction errors of single (a) GPR (b) ANN and (c) PLS model, and bagging (d) GPR (e) ANN and (f) PLS model on “wheat kernel” set.
Fig 2. RMSEP of subagging GPR, ANN and PLS at different sub-sampling ratios on the “wheat kernel” dataset.

Table 1. Results of individual calibration methods (GPR, ANN, PLS) and those of subagging with different ratios of sub-sampled training data on the “wheat kernel” dataset.

<table>
<thead>
<tr>
<th>Ratio</th>
<th>Subag GPR</th>
<th>Subag ANN</th>
<th>Subag PLS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RMSEP</td>
<td>Improvement</td>
<td>No. of LVs</td>
</tr>
<tr>
<td>0.1</td>
<td>0.76</td>
<td>-</td>
<td>9.8</td>
</tr>
<tr>
<td>0.2</td>
<td>0.74</td>
<td>-</td>
<td>10.0</td>
</tr>
<tr>
<td>0.3</td>
<td>0.72</td>
<td>-</td>
<td>10.5</td>
</tr>
<tr>
<td>0.4</td>
<td>0.68</td>
<td>-</td>
<td>11.1</td>
</tr>
<tr>
<td>0.5</td>
<td>0.59</td>
<td>-</td>
<td>11.6</td>
</tr>
<tr>
<td>0.6</td>
<td>0.51</td>
<td>-</td>
<td>11.7</td>
</tr>
<tr>
<td>0.7</td>
<td>0.44</td>
<td>-</td>
<td>11.9</td>
</tr>
<tr>
<td>0.8</td>
<td>0.43</td>
<td>-</td>
<td>11.9</td>
</tr>
<tr>
<td>0.9</td>
<td>0.43</td>
<td>-</td>
<td>11.9</td>
</tr>
<tr>
<td>1</td>
<td>0.48</td>
<td>-</td>
<td>11.9</td>
</tr>
</tbody>
</table>

RMSEP_{GPR}=0.48; RMSEP_{ANN}=0.45; RMSEP_{PLS}=0.49.
Table 2. A summary of results for individual calibration methods, bagging, and subagging by cross-validation on the “wheat kernel” dataset.

<table>
<thead>
<tr>
<th>Methods</th>
<th>GPR</th>
<th>Subag_GPR^{(CV)}</th>
<th>Bag_GPR</th>
<th>ANN</th>
<th>Subag_ANN^{(CV)}</th>
<th>Bag_ANN</th>
<th>PLS</th>
<th>Subag_PLS^{(CV)}</th>
<th>Bag_PLS</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMSEP</td>
<td>0.48</td>
<td>0.43</td>
<td>0.41</td>
<td>0.45</td>
<td>0.42</td>
<td>0.42</td>
<td>0.49</td>
<td>0.49</td>
<td>0.49</td>
</tr>
<tr>
<td>Improvement</td>
<td>-</td>
<td>2.1%</td>
<td>14.6%</td>
<td>-</td>
<td>6.7%</td>
<td>6.7%</td>
<td>-</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>No. of LVs</td>
<td>12.0</td>
<td>11.9</td>
<td>18.0</td>
<td>12.0</td>
<td>11.8</td>
<td>18.9</td>
<td>11.9</td>
<td>11.7</td>
<td>18.6</td>
</tr>
<tr>
<td>Time (min)</td>
<td>2.82</td>
<td>385.05</td>
<td>76.51</td>
<td>0.03</td>
<td>24.63</td>
<td>0.60</td>
<td>0.02</td>
<td>7.47</td>
<td>0.30</td>
</tr>
</tbody>
</table>

Table 3. Summary of results for individual calibration methods (GPR, ANN, PLS) and bagging GPR, ANN, PLS on the “meat” dataset. (a) Moisture; (b) Fat; (c) Protein.

(a)

<table>
<thead>
<tr>
<th>Methods</th>
<th>GPR</th>
<th>Bag_GPR</th>
<th>ANN</th>
<th>Bag_ANN</th>
<th>PLS</th>
<th>Bag_PLS</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMSEP</td>
<td>0.77</td>
<td>0.64</td>
<td>0.99</td>
<td>0.88</td>
<td>2.68</td>
<td>2.61</td>
</tr>
<tr>
<td>Improvement</td>
<td>-</td>
<td>16.9%</td>
<td>-</td>
<td>11.1%</td>
<td>-</td>
<td>2.6%</td>
</tr>
<tr>
<td>No. of LVs</td>
<td>16.2</td>
<td>14.9</td>
<td>16.0</td>
<td>19.0</td>
<td>17.1</td>
<td>19.5</td>
</tr>
<tr>
<td>Time (min)</td>
<td>0.30</td>
<td>9.96</td>
<td>0.03</td>
<td>0.72</td>
<td>0.01</td>
<td>0.25</td>
</tr>
</tbody>
</table>

(b)

<table>
<thead>
<tr>
<th>Methods</th>
<th>GPR</th>
<th>Bag_GPR</th>
<th>ANN</th>
<th>Bag_ANN</th>
<th>PLS</th>
<th>Bag_PLS</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMSEP</td>
<td>0.83</td>
<td>0.50</td>
<td>0.83</td>
<td>0.77</td>
<td>3.02</td>
<td>2.96</td>
</tr>
<tr>
<td>Improvement</td>
<td>-</td>
<td>39.8%</td>
<td>-</td>
<td>7.2%</td>
<td>-</td>
<td>2.0%</td>
</tr>
<tr>
<td>No. of LVs</td>
<td>16.2</td>
<td>14.7</td>
<td>16.6</td>
<td>19.3</td>
<td>16.5</td>
<td>18.4</td>
</tr>
<tr>
<td>Time (min)</td>
<td>0.30</td>
<td>9.15</td>
<td>0.03</td>
<td>0.78</td>
<td>0.01</td>
<td>0.24</td>
</tr>
</tbody>
</table>

(c)

<table>
<thead>
<tr>
<th>Methods</th>
<th>GPR</th>
<th>Bag_GPR</th>
<th>ANN</th>
<th>Bag_ANN</th>
<th>PLS</th>
<th>Bag_PLS</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMSEP</td>
<td>0.40</td>
<td>0.41</td>
<td>0.55</td>
<td>0.47</td>
<td>0.54</td>
<td>0.54</td>
</tr>
<tr>
<td>Improvement</td>
<td>-</td>
<td>-2.5%</td>
<td>-</td>
<td>14.5%</td>
<td>-</td>
<td>0%</td>
</tr>
<tr>
<td>No. of LVs</td>
<td>14.3</td>
<td>14.0</td>
<td>14.4</td>
<td>16.4</td>
<td>14.3</td>
<td>16.1</td>
</tr>
<tr>
<td>Time (min)</td>
<td>0.27</td>
<td>9.63</td>
<td>0.03</td>
<td>0.63</td>
<td>0.01</td>
<td>0.24</td>
</tr>
</tbody>
</table>