Analytical Methods

PAPER

View Article Online
View Journal | View Issue

Cite this: *Anal. Methods*, 2017, **9**, 2983

A boosting extreme learning machine for near-infrared spectral quantitative analysis of diesel fuel and edible blend oil samples

Xihui Bian,*[ab] Caixia Zhang,[ab] Xiaoyao Tan,[ab] Michal Dymek,[bc] Yugao Guo,[b] Ligang Lin,[a] Bowen Cheng[a] and Xiaoyu Hu[a]

Received 10th February 2017
Accepted 24th April 2017

DOI: 10.1039/c7ay00353f

rsc.li/methods

Extreme learning machines (ELMs) have drawn increasing attention due to their characteristics of simple structure, high learning speed and excellent performance. However, a single ELM tends to low predictive accuracy and instability in dealing with quantitative analysis of complex samples. To further improve the predictive accuracy and stability of ELMs, a new quantitative model, called the boosting ELM is proposed. In this approach, a large number of ELM sub-models are sequentially built by selecting a certain number of samples from the original training set according to the distribution of the sampling weights, and then their predictions are aggregated using the weighted median. The activation function and the number of hidden nodes of ELM sub-models are determined simultaneously by the ratio of mean value and standard deviation of correlation coefficients (MSR). The performance of the proposed method is tested with diesel fuel and blended edible oil samples. Compared with partial least squares (PLS) and ELMs, our results demonstrate that the boosting ELM is an efficient ensemble model and has obvious superiorities in predictive accuracy and stability. Therefore, the proposed method may be an alternative for near-infrared (NIR) spectral quantitative analysis of complex samples.

1. Introduction

Near-infrared (NIR) spectroscopy has been widely used in various fields including food, fuel, medicine and soil because it is fast, convenient, green, and requires little or no sample pre-processing.[1–4] Nevertheless, the presence of the relatively weak and highly overlapping spectral bands originating from the overtones and combinations of fundamental vibrations of hydrogen bonds like C–H, N–H, and O–H[5,6] makes NIR spectroscopy heavily rely on the use of chemometrics to construct an appropriate calibration model for quantitative analysis of the target component in complex samples.

The most commonly used multivariate calibration model is partial least squares (PLS)[7–9] due to its advantages of rapidity, simplicity and ease-of-use. However, PLS is essentially a linear modeling method and would produce large errors when the system exhibits strong nonlinear behaviors, such as deviations from the Beer–Lambert law, nonlinear detector response, scattering of light, changes in temperature, interaction between analytes, *etc.*[5,6,10] Many nonlinear techniques[7,10,11] such as artificial neural network (ANN) and support vector regression (SVR) are developed. These methods may perform well on nonlinear data but are time-consuming compared with linear methods, easy to trap in local optimum and require many parameters. Recently, a new supervised algorithm called extreme learning machines (ELMs) for single-hidden layer feed-forward neural networks (SLFNs) was proposed by Huang *et al.*[12–14] ELMs combine the superiorities of linear and non-linear methods[15] and have attracted increasing attention in spectral quantitative analysis of complex samples.[10,15–18] Compared with traditional multivariate calibration methods,[7–11] the superiority of ELMs is attributed to their inherent characteristics of simple structure, high learning speed and excellent predictive performance.

To achieve good predictive performance, ELMs minimize training error on the whole training dataset. Thus they might suffer from overfitting[19] and give unsatisfactory prediction for the unknown samples. In addition, the weights and biases for hidden nodes can be randomly generated without iteratively adjusting ELMs.[15,20] This increases the learning speed and reduces the number of parameters that need to be optimized. However, the random initialization of the input weights and hidden layer biases may make ELMs unstable in practice,[20,21] that is, different runs of the ELM model will lead to fluctuation in the predicted results. Therefore, it is highly demanded to

[a] *State Key Laboratory of Separation Membranes and Membrane Processes, Tianjin Polytechnic University, Tianjin, 300387, P. R. China. E-mail: bianxihui@163.com; Fax: +86-22-83955663; Tel: +86-22-83955663*

[b] *School of Environmental and Chemical Engineering, Tianjin Polytechnic University, Tianjin, 300387, P. R. China*

[c] *Department of Chemical Technology, Faculty of Chemical Engineering and Technology, Tadeusz Kosciuszko Cracow University of Technology, Cracow, 31-155, Poland*

develop a new approach for improving the predictive accuracy and stability of ELMs.

Considerable research studies have demonstrated that ensemble modeling is one of the best ways to prevent overfitting[19,22] and improve the accuracy and stability of a single model.[7,23,24] The most popular ensemble strategies are bagging and boosting. It has been reported that boosting is usually superior to bagging.[25] Boosting, which originated from machine learning,[26] has attracted increasing interest in chemometrics.[7,25,27] Boosting provides an accurate prediction by combining a series of rough and inaccurate sub-models. Such a series of sub-models is developed by using the training subsets selected from the original training set according to the distribution of the sampling weights. As a prominent ensemble strategy, boosting has been used to improve the performance of a single multivariate calibration model such as PLS,[27–33] ANN,[34,35] and SVR.[36–38] Recently, the boosting ELM has been developed[19,39,40] for classification problems in the fields of mathematics and computer science. However, until now, few studies have been performed to combine boosting with ELMs for multivariate calibration.

Inspired by the attractive feature of boosting and ELMs, a novel method named the boosting ELM is proposed to improve the performance of ELMs. As comparisons to the boosting ELM, the conventional PLS and ELM have been investigated. To assess the predictive performance and computational efficiency of the boosting ELM model, two NIR spectral datasets are utilized. One is used to calibrate the total aromatic contents of diesel fuel samples, which directly affects the combustion degree, engine life and automobile exhaust emissions.[41] The other is used to calibrate the corn oil content of edible blend oil samples. The contents and compositions of edible blend oil affect its quality, price and nutritional value.[42]

2. Theory and algorithm

2.1 Extreme learning machine (ELM)

The ELM is a simple and efficient learning algorithm for SLFNs.[12–14] Conventional learning algorithms for neural networks, such as back propagation and gradient descent algorithms, have some disadvantages including the difficulty in adjusting many parameters, the tendency to get trapped in the local minimum and the slow learning speed. Fortunately, the above disadvantages can be avoided in the ELM. The topology structure of the ELM is shown in Fig. 1. From the figure, it can be seen that the ELM has two appealing characteristics. One is that the learning parameters of hidden nodes, including input weights and hidden layer biases, can be randomly assigned independently. The other is that the weights between the hidden nodes and the output layer nodes are calculated by using the least square method (Moore-Penrose generalized inverse). These two characteristics provide the method with the advantages of fast learning speed, high generalization performance and a few involved parameters.

The ELM algorithm consists of four simple steps. Firstly, the input weights and hidden layer biases are randomly initialized. Secondly, the appropriate activation function and the number of hidden nodes are chosen. Thirdly, the hidden layer output matrix is calculated. Finally, the output weights are obtained by the Moore-Penrose generalized inverse of the hidden layer output matrix and target values. More detailed descriptions of ELM are provided in ref. 12, 15, 20 and 43.

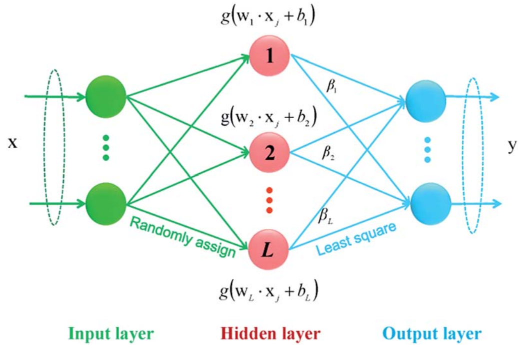

Fig. 1 The structure of the ELM.

From the above mentioned steps, only two parameters, *i.e.*, the activation function and the number of hidden nodes need to be optimized for the ELM. Considering the predictive accuracy and stability of the model simultaneously, the ratio of mean value and standard deviation (SD) of correlation coefficient (R), which is abbreviated as MSR, was introduced in our previous work.[15] In this study, the activation function and the number of hidden nodes are determined simultaneously according to the variation of MSR with the two parameters. The optimal activation function and the number of hidden nodes of ELM are selected simultaneously corresponding to the maximum value of MSR.

2.2 Boosting extreme learning machine (boosting ELM)

The ELM has many advantages due to its two characteristics. However, the ELM also tends to instability and inaccuracy to some extent because of the random set of input weights and hidden layer biases[20,21] and the use of least squares to calculate the output weights.[44] To improve the predictive accuracy and stability of the ELM, a new algorithm named the boosting ELM is proposed by combining the boosting strategy with the ELM.

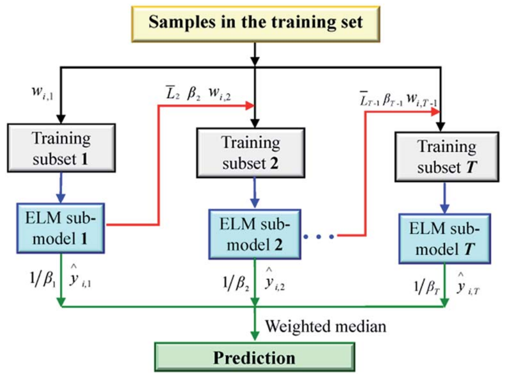
Samples in the training set
$w_{i,1}$
$\bar{L}_2$ β_2 $w_{i,2}$
$\bar{L}_{T\text{-}1}$ $\beta_{T\text{-}1}$ $w_{i,T\text{-}1}$
Training subset 1
Training subset 2
Training subset T
ELM sub-model 1
ELM sub-model 2
ELM sub-model T
$1/\beta_1$ $\hat{y}_{i,1}$
$1/\beta_2$ $\hat{y}_{i,2}$
$1/\beta_T$ $\hat{y}_{i,T}$
Weighted median
Prediction

Fig. 2 The flowchart of boosting ELM modeling.

The boosting ELM is implemented by sequentially developing multiple sub-models using the training subsets selected from the original training set according to the sampling probability formed by the previous iteration. The sub-models are trained by the ELM and then their predictions are combined using the weighted median to get the final prediction for unknown samples. The brief flowchart of the method is given in Fig. 2 and the corresponding detailed procedure is described as follows.

First, all the samples in the training set are given the equal sampling weights, $w_{i,1} = 1/m$ ($i = 1, 2,\cdots, m$), where m is the number of samples in the training set.

Then, for $t = 1, 2,\cdots T$ (T is the number of iterations), perform the following steps.

(1) Select a certain number of samples from the training set to form the training subset according to the probability distributions of $w_{i,t}$.

(2) Establish an ELM sub-model by using the training subset with the optimal activation function and the number of hidden nodes, and then use this model to predict the target value of the training set. Then the predicted values $y_{i,t}$ ($i = 1, 2,\cdots, m$) are obtained.

(3) Calculate the absolute error $e_{i,t}$ of each sample in the training set.

$$e_{i,t} = |y_{i,t} - \hat{y}_i| \tag{1}$$

where $\hat{y}_i$ is the reference value of the sample i in the training set.

(4) Compute the loss function $L_{i,t}$ by using the exponential form.[25,33]

$$L_{i,t} = 1 - \exp\left[-\frac{e_{i,t}}{\max(e_{i,t})}\right] \tag{2}$$

(5) Compute the mean loss function $\bar{L}_t$.

$$\bar{L}_t = \sum_{i=1}^{m} L_{i,t} w_{i,t} \tag{3}$$

(6) Calculate the confidence indicator β_t.

$$\beta_t = \bar{L}_t/(1 - \bar{L}_t) \tag{4}$$

(7) Update the new sampling weight $w_{i,t+1}$ of each sample in the training set.

$$w_{i,t+1} = w_{i,t}{\beta_t}^{-L_{i,t}} \tag{5}$$

The new sampling weights should be normalized to ensure that $\sum_{i=1}^{m} w_{i,t+1} = 1$.

Finally, after performing T iterations, T ELM sub-models are established. For the prediction stage, each ELM sub-model gives a prediction $y_{i,t}$ and a corresponding β_t for the ith unknown sample. The final prediction is obtained by combining these T predictions using the weighted median according to the following process.

Sort these T predictions $y_{i,t}$ ($t = 1, 2,\cdots, T$) for the ith sample in the incremental order and simultaneously obtain the corresponding β_{n_t} to y_{i,n_t}.

$$y_{i,n_1} \le y_{i,n_2} \le \cdots \le y_{i,n_t} \tag{6}$$

where n_t ($t = 1, 2,\cdots, T$) represents a permutation of $1, 2,\cdots, T$. Take the sum of $\log(1/\beta_{n_t})$ over t, until the rth prediction satisfies the following inequality, *i.e.*,

$$\sum_{t=1}^{r} \log(1/\beta_{n_t}) \ge \frac{1}{2}\sum_{t=1}^{T} \log(1/\beta_{n_t}) \tag{7}$$

where r is the n_rth prediction by sorting in the ascending order. Then the prediction y_{i,n_r} from the n_rth ELM sub-model is identified as the ensemble prediction for the ith sample.

3. Experimental

To evaluate the performance of the boosting ELM, two spectral datasets were used for this study. The diesel fuel dataset consists of NIR spectra of 256 samples and six physical properties including boiling point at 50% recovery, cetane number, density, freezing temperature, total aromatics and the viscosity of those fuels.[45] The spectra of diesel fuel samples were measured directly at Southwest Research Institute (SWRI) on a project sponsored by the U.S. Army. First-derivative spectral data at 401 wavelengths in the NIR region were supplied. Each spectrum is composed of 401 variables recorded in the wavelength range 750–1550 nm. The six properties of the same diesel fuel sample were independently measured by the American Society of Testing and Materials (ASTM) standard method. The dataset was provided by SWRI, San Antonio, TX through Eigenvector Research, Inc. (Manson, Washington) and can be downloaded freely from http://www.eigenvector.com/Data/SWRI. The spectra and total aromatic contents were investigated for this study. Fig. 3(a) displays the NIR first-derivative spectra of the 256 samples.

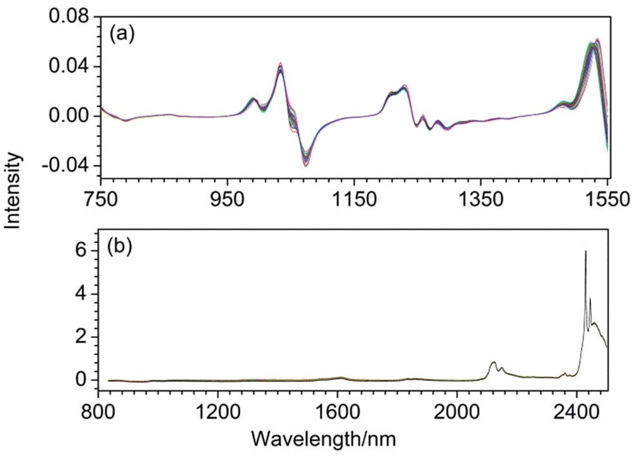

Fig. 3 NIR spectra of the diesel fuel dataset (a) and the edible blend oil dataset (b).

The edible blend oil dataset, measured by us, contains the NIR spectra of 50 samples and the contents of ternary components including soybean oil, corn oil and sesame oil of those samples. The spectra of edible blend oil samples were measured directly on a Vertex70 spectrometer (Bruker Optics, Ettlingen, Germany) over 4148 channels recorded in the wavelength range 833.3–2500 nm. The spectrometer was balanced at *ca.* 25 °C for 60 min before measurement. To increase the signal to noise ratio, the spectra were measured with scan number 64, and the average spectrum of three parallel measurements was used for each sample. The spectra and corn oil contents were used for this study. Fig. 3(b) displays the NIR original spectra of the 50 samples, which are of high quality and used for calibration directly without processing.

Before calculation, the datasets were divided into the training and prediction sets. For the fuel oil dataset, the two datasets were divided according to the dataset grouping in the website. 138 samples including the spectra (t_sd_hl and t_sd_ll_b sets) and the total aromatic contents (t_y_hl and t_y_ll_b sets) were used as the training set. 118 samples including the spectra (t_sd_ll_a set) and the total aromatic contents (t_y_ll_a set) were taken as the prediction set. For the edible blend oil dataset, 33 samples were selected as the training set and the remaining 17 samples were used as the prediction set by using the Kennard–Stone (KS) grouping algorithm.[46] For the PLS model, the number of latent variables (LV) is determined to be 8 and 9 by Monte Carlo cross-validation (MCCV) combined with Osten's F criterion[47] for diesel fuel and edible blend oil datasets, respectively.

4. Results and discussion

4.1 Optimization of the ELM parameters

As mentioned above, the activation function and the number of hidden nodes are two important parameters required to be optimized, which have great effects on the predictive accuracy and stability of the ELM model. Thus, the optimal activation function and the number of hidden nodes should be selected simultaneously at first. The number of hidden nodes was set from 1 to 100 with the interval 1, and sigmoidal (sig), sine (sin), hardlim, triangular basis (tribas) and radial basis (radbas) functions were used as activation functions, respectively. For each activation function and each hidden node number, the ELM model was established. The process was repeated 500 times and 500 R between the predicted and measured values were calculated. Then the ratio of mean value and SD of 500R, *i.e.*, MSR, was obtained, which increases with the increase of mean value and the decrease of SD of R.

Taking the diesel fuel dataset as an example, Fig. 4 shows the variation of MSR values with activation functions and the number of hidden nodes. It can be clearly seen that the MSR values of sig, sin, tribas and radbas functions are much bigger than those of hardlim function overall. The sig, sin, tribas and radbas functions produce similar predictive accuracy and stability with the same number of hidden nodes. Specifically, MSR values first increase obviously with the increase of the number of hidden nodes, and reach the maximum value at 44, 44, 46 and 47 for sig, sin, tribas and radbas functions, respectively. After the maximum values, they decrease to the end. For further comparison, the sin function has the best predictive performance. Thus, the optimal activation function and the number of hidden nodes are sin and 44, respectively. Similar analysis can be performed for the edible blend oil dataset and sig and 18 are determined as the optimal activation function and the number of hidden nodes for this dataset.

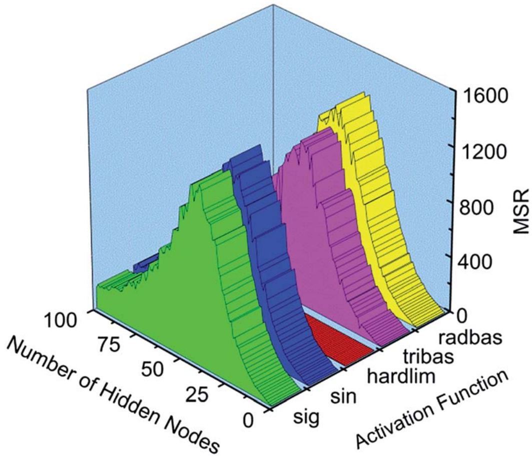

Fig. 4 Variation of MSR values with activation functions and the number of hidden nodes for the diesel fuel dataset.

4.2 Iteration number

The iteration number (T), *i.e.*, the number of ELM sub-models, is an important parameter involved in the boosting ELM. A certain number of T is needed to improve the predictive accuracy and stability of the proposed model. However, with too large T, the complexity of the boosting model is increased and the computation will be time-consuming. Therefore, an appropriate iteration number should be selected.

In this study, T from 1 to 600 was investigated. For each T, a boosting ELM model is developed and then the model was used to predict the prediction set. The above process is performed 50 times independently and then the mean root mean square error of prediction (RMSEP) and its SD are calculated for each T. Fig. 5(a) shows the variation of mean RMSEP and its SD with the iteration number T for the diesel fuel dataset. From the figure, it can be clearly seen that the mean RMSEP and its SD are both comparatively large at the beginning. With the increase of T, the mean RMSEP and its SD first tend to decrease sharply and then descend gradually before 100. When it is above 100, the decrease of mean RMSEP and its SD becomes slow. For further distinguishing the difference, the variations of mean RMSEP and its SD with T in the range of 100–600 are plotted in Fig. 5(b). Because the range of the vertical coordinate becomes smaller, it is easy to find that the mean RMSEP and its SD still go down until 500. After that, both the mean RMSEP and its SD maintain a stable value. Considering the predictive ability and efficiency, 500 is used for the optimal iteration number for diesel fuel. The edible blend oil dataset has similar results. Accordingly, iteration number T is set as 500 for both datasets.

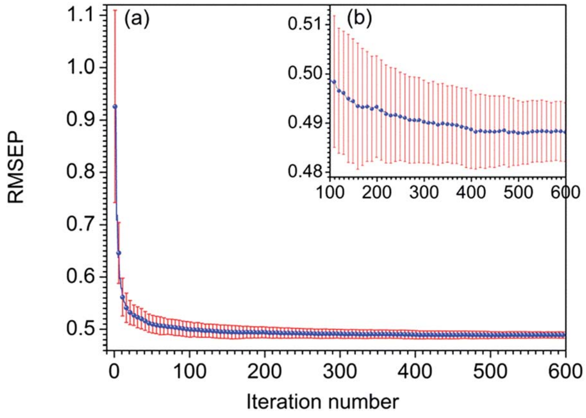

Fig. 5 Variation of the mean RMSEP and its standard deviation over 50 runs with the iteration number for the diesel fuel dataset (a) and its enlargement in the range of 100–600 (b).

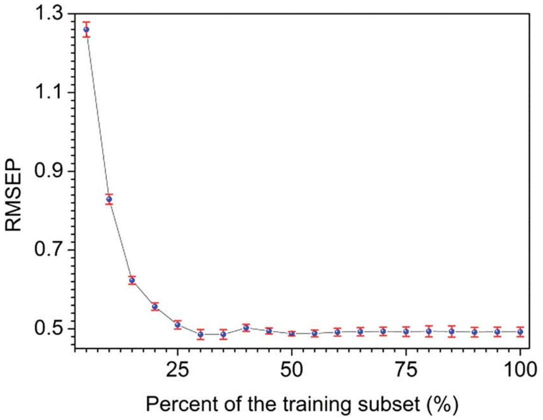

Fig. 6 Variation of the mean RMSEP and its standard deviation over 50 runs with the size of the training subset for the diesel fuel dataset.

4.3 Size of the training subset for ELM sub-models

In the boosting ELM method, a series of ELM sub-models is developed by using a certain number of samples selected from the original training set. Therefore another key parameter of the proposed method is the number of the selected samples, *i.e.*, the size of the training subset. It is important that the optimal parameter is the tradeoff between the accuracy and diversity of the sub-models. Thus, the size of the training subset needs to be investigated prior to establishment of the boosting ELM model. In order to show the relative size of the training subset, the percentage of the total number of training samples from 5% to 100% with an interval of 5% is investigated.

Fig. 6(a) shows the variation of mean RMSEP and its SD with the size of the training subset for the diesel fuel dataset. It is clear that the mean RMSEP is very large at first and decreases quickly before 30%, and then remains relatively stable in the range of 30–100%. For more details, in the range of 35–45%, the mean RMSEP has a slight rising and then falling trend. After 50%, the mean RMSEP maintains almost a line. For further considering the stability of the model, SD first decreases with the increase of the training subset. The smallest SD is located at 50%. After that, the SD increases gradually. Therefore, for considering the predictive accuracy and stability simultaneously, 50% of the total training samples are used as the optimal size of the training subset for the diesel fuel dataset in the following discussion. Similar analysis indicates that the optimal size of the training subset is 70% for the edible blend oil dataset.

4.4 Comparison of the prediction results

With the optimal parameters determined above, the boosting ELM model can be developed to predict unknown samples. The predictive ability of the proposed method is estimated by RMSE, the ratio of the standard error of prediction to the SD of the reference values (RPD) and R. Apparently, a better model should have a lower RMSE, a bigger RPD and R. Due to the instability of ELM sub-models and probability sampling involved in the proposed method, 50 independent runs are performed. The mean RMSEC, RPD, R and their SDs for the training set and the mean RMSEP, RPD, R and their SDs for the prediction set are listed in Table 1. As a comparison, the statistical results of PLS and the ELM with the same training set and prediction set are also calculated and summarized in Table 1.

Table 1 Prediction results of the two datasets obtained by boosting ELM, ELM and PLS methods

	RMSE[a]		RPD[a]		R[a]		Time[b]	
Method	RMSEC	RMSEP	Calibration	Prediction	Calibration	Prediction	Calibration	Prediction
Diesel fuel dataset								
PLS	0.7283(0)	0.8075(0)	9.1811(0)	7.3485(0)	0.9941(0)	0.9907(0)	0.0050	0.0003
ELM	0.4263(0.0249)	0.5874(0.0457)	15.7384(0.9312)	10.1897(0.7930)	0.9980(0.0002)	0.9951(0.0008)	0.0026	0.0005
Boosting ELM	0.3146(0.0032)	0.4876(0.0055)	18.7847(0.1905)	12.2586(0.1398)	0.9989(0.0000)	0.9967(0.0001)	2.7725	1.1714
Edible blend oil dataset								
PLS	5.2625(0)	7.0492(0)	2.7965(0)	1.9920(0)	0.9362(0)	0.8899(0)	0.0038	0.0004
ELM	3.4813(0.6486)	5.9650(0.8508)	4.3616(0.8478)	2.4445(0.3847)	0.9703(0.0101)	0.9291(0.0172)	0.0018	0.0004
Boosting ELM	1.4308(0.0428)	4.9404(0.0681)	9.9663(0.2971)	2.8431(0.0399)	0.9954(0.0002)	0.9422(0.0020)	3.0551	0.2024

[a] Average value over 50 runs is listed, and the value in the parentheses is the standard deviation. [b] Calculation was performed on a PC with an Intel (*R*) Core (TM) i7-4790 processor + 3.60 GHz CPU and 8 GB memory.

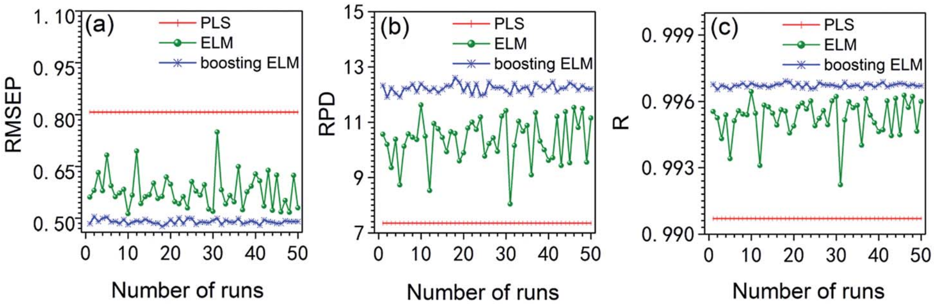

Fig. 7 RMSEPs (a), RPDs (b) and Rs (c) of the prediction set obtained by PLS, ELM and boosting ELM in 50 runs of prediction for the diesel fuel dataset.

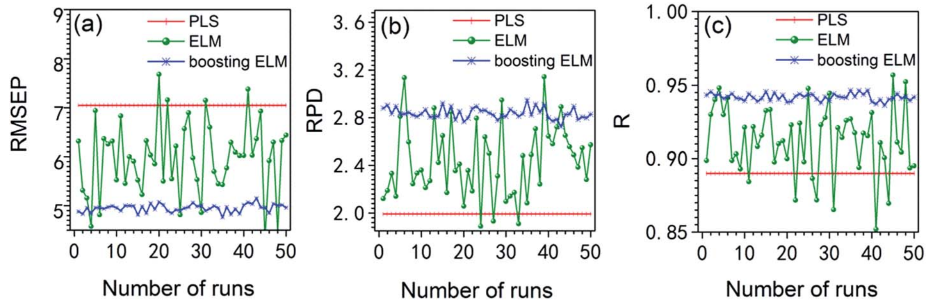

Fig. 8 RMSEPs (a), RPDs (b) and Rs (c) of the prediction set obtained by PLS, ELM and boosting ELM in 50 runs of prediction for the edible blend oil dataset.

From Table 1, the ELM and the boosting ELM have lower mean RMSEC, bigger mean RPD and R than PLS for the two datasets, suggesting that ELM based methods can achieve better calibration accuracy than PLS. Moreover, the SDs of RMSECs, RPDs and Rs in the boosting ELM are obviously smaller than those in the ELM. This demonstrates that the boosting strategy can improve the stability of the ELM greatly. The SDs of PLS are zero since no random process is involved. The mean RMSEP, RPD, R and their SDs for the prediction set have a similar changing law to those of the training set. To further display the predictive result of each run, the variation of RMSEPs (a), RPDs (b) and Rs (c) for the prediction set obtained by PLS, ELM and boosting ELM in 50 runs for the fuel oil dataset and the edible blend oil dataset is shown in Fig. 7 and 8, respectively. For the fuel oil dataset (Fig. 7), the ELM and the boosting ELM have lower RMSEPs, bigger RPDs and Rs in all runs than PLS. Moreover, the variation of RMSEPs, RPDs and Rs is all with large fluctuation for the ELM, with small fluctuation for the boosting ELM and no change for PLS. For the edible blend oil dataset (Fig. 8), 47, 46 and 43 in 50 RMSEPs, RPDs and Rs of the ELM are actually better than those of PLS, respectively. The three parameters and their mean values of the boosting ELM in all runs are all better than those of PLS. Herein, the ELM provides better prediction in some runs and worse prediction in other runs than PLS, since it is unstable in replicate runs. The boosting ELM can further improve the predictive accuracy and stability of the ELM and provide better prediction than PLS in all runs for the two datasets. Although the replicate results of the boosting ELM are not reproducible, the method can be used independently since the SD is very small compared with the mean value.

Table 1 also lists the computational times in calibration and prediction steps of the three methods for the two datasets. The ELM is faster than PLS in the calibration step and as efficient as PLS in the prediction step. The higher rapidity of the ELM is attributed to the arbitrarily assigning of input weights and hidden layer biases and calculating the output weights using the generalized inverse in the ELM. The boosting ELM needs more time than the ELM and PLS both in calibration and prediction steps, because *T* iterations are involved in the boosting ELM. In each iteration, the sampling weights, ELM sub-model and prediction should be calculated again. However, due to the rapidity of the ELM sub-model, the calibration and prediction steps of the boosting ELM only take several seconds (2.7725 and 1.1714 seconds for the diesel fuel dataset, 3.0551 and 0.2024 seconds for the edible blend oil dataset, respectively). Thus, the boosting ELM can be considered as an efficient ensemble method.

5. Conclusions

In order to improve the predictive accuracy and stability of the ELM, a new ensemble model named the boosting ELM was proposed. This method establishes a series of ELM sub-models sequentially by selecting a certain number of samples from the original training set and then the final prediction is obtained by taking the weighted median of the predictions from ELM sub-models. To validate the effectiveness of the boosting ELM model for NIR spectral multivariate calibration, two datasets of complex samples are used. Moreover, the predictive performance and computational efficiency of the proposed method are also compared with those of the ELM and PLS. The results show that the boosting ELM can greatly improve the predictive accuracy and stability compared with the ELM. Although it takes more time to calibrate a model and predict unknown samples than those of the ELM and PLS, the boosting ELM is an efficient ensemble method. Therefore, the combination of boosting with the ELM offered a promising tool for improving the NIR spectral analysis of complex samples.

Acknowledgements

Thanks are due to Prof. Xueguang Shao for providing instrumental support. This study is supported by the National Basic Research Program of China (No. 2014CB660813), the National Natural Science Foundation of China (Nos. 21405110 and 21676199) and the Science and Technology Plans of Tianjin (Nos. 15PTSYJC00230 and 16JCTPJC44700).

References

1 L. Wang, D. W. Sun, H. B. Pu and J. H. Cheng, *Crit. Rev. Food Sci. Nutr.*, 2017, **57**, 1524–1538.
2 A. Sakudo, *Clin. Chim. Acta*, 2016, **455**, 181–188.
3 J. J. Roberts and D. Cozzolino, *TrAC, Trends Anal. Chem.*, 2016, **83**, 25–30.
4 X. Y. Cui, X. W. Liu, X. M. Yu, W. S. Cai and X. G. Shao, *Anal. Chim. Acta*, 2017, **957**, 47–54.
5 C. Tan, H. Chen, Z. H. Xu, T. Wu, L. Wang and W. P. Zhu, *Spectrochim. Acta, Part A*, 2012, **96**, 526–531.
6 C. Tan, X. Qin and M. L. Li, *Anal. Bioanal. Chem.*, 2008, **392**, 515–521.
7 X. G. Shao, X. H. Bian, J. J. Liu, M. Zhang and W. S. Cai, *Anal. Methods*, 2010, **2**, 1662–1666.
8 S. Wold, M. Sjostrom and L. Eriksson, *Chemom. Intell. Lab. Syst.*, 2001, **58**, 109–130.
9 J. P. M. Andries, Y. Vander Heyden and L. M. C. Buydens, *Anal. Chem.*, 2013, **85**, 5444–5453.
10 J. T. Peng, L. Q. Li and Y. Y. Tang, *Chemom. Intell. Lab. Syst.*, 2013, **120**, 53–58.
11 R. M. Balabin and E. I. Lomakina, *Analyst*, 2011, **136**, 1703–1712.
12 G. B. Huang, Q. Y. Zhu and C. K. Siew, *Neurocomputing*, 2006, **70**, 489–501.
13 G. B. Huang, L. Chen and C. K. Siew, *IEEE Transactions on Neural Networks*, 2006, **17**, 879–892.
14 W. Y. Deng, Z. Bai, G. B. Huang and Q. H. Zheng, *Neural Network.*, 2016, **77**, 14–28.
15 X. H. Bian, S. J. Li, M. R. Meng, Y. G. Guo, N. Chang and J. J. Wang, *Anal. Methods*, 2016, **8**, 4674–46799.
16 H. Jiang, G. H. Liu, C. L. Mei and Q. S. Chen, *Anal. Methods*, 2013, **5**, 1872–1880.
17 H. AlHichri, Y. Bazi, N. Alajlan, F. Melgani, S. Malek and R. R. Yager, *J. Chemom.*, 2013, **27**, 447–456.
18 D. Xiao, J. C. Wang and Z. Z. Mao, *Chemom. Intell. Lab. Syst.*, 2014, **134**, 118–122.
19 H. M. Feng and X. Z. Wang, *Neural Network.*, 2015, **63**, 87–93.
20 W. R. Chen, J. Bin, H. M. Lu, Z. M. Zhang and Y. Z. Liang, *Analyst*, 2016, **141**, 1973–1980.
21 H. J. Lu, C. L. An, E. H. Zheng and Y. Lu, *Neurocomputing*, 2014, **128**, 22–30.
22 B. C. Deng, Y. H. Yun, Y. Z. Liang, D. S. Cao, Q. S. Xu, L. Z. Yi and X. Huang, *Anal. Chim. Acta*, 2015, **880**, 32–41.
23 Y. K. Li, X. G. Shao and W. S. Cai, *Talanta*, 2007, **72**, 217–222.
24 X. H. Bian, S. J. Li, L. G. Lin, X. Y. Tan, Q. J. Fan and M. Li, *Anal. Chim. Acta*, 2016, **925**, 16–22.
25 J. Jiao, S. M. Tan, R. M. Luo and Y. P. Zhou, *J. Chem. Inf. Model.*, 2011, **51**, 816–828.
26 R. E. Schapire, *Machine Learning*, 1990, **5**, 197–227.
27 M. H. Zhang, Q. S. Xu and D. L. Massart, *Anal. Chem.*, 2005, **77**, 1423–1431.
28 Y. P. Zhou, C. B. Cai, S. Huan, J. H. Jiang, H. L. Wu, G. L. Shen and R. Q. Yu, *Anal. Chim. Acta*, 2007, **593**, 68–74.
29 X. G. Shao, X. H. Bian and W. S. Cai, *Anal. Chim. Acta*, 2010, **666**, 32–37.
30 D. S. Cao, Q. S. Xu, Y. Z. Liang, L. X. Zhang and H. D. Li, *Chemom. Intell. Lab. Syst.*, 2010, **100**, 1–11.
31 Y. K. Li, *Anal. Methods*, 2012, **4**, 254–258.
32 Q. S. Xu, J. Xu, D. S. Cao and Y. Z. Liang, *Chemom. Intell. Lab. Syst.*, 2016, **152**, 134–139.
33 X. H. Bian, S. J. Li, X. G. Shao and P. Liu, *Chemom. Intell. Lab. Syst.*, 2016, **158**, 174–179.
34 W. X. Pan, J. W. Zhao, Q. S. Chen and D. L. Zhang, *Food Analytical Methods*, 2015, **8**, 749–757.
35 Q. Ouyang, Q. S. Chen and J. W. Zhao, *Spectrochim. Acta, Part A*, 2016, **154**, 42–46.
36 Y. P. Zhou, L. Xu, L. J. Tang, J. H. Jiang, G. L. Shen, R. Q. Yu and Y. Ozaki, *Anal. Sci.*, 2007, **23**, 793–798.
37 R. M. Luo, S. M. Tan, Y. P. Zhou, S. J. Liu, H. Xu, D. D. Song, Y. F. Cui, H. Y. Fu and T. M. Yang, *J. Chemom.*, 2013, **27**, 198–206.
38 P. R. Filgueras, L. A. Terra, E. V. R. Castro, L. M. S. L. Olivera, J. C. M. Dias and R. J. Poppi, *Talanta*, 2015, **142**, 197–205.
39 K. Li, X. F. Kong, Z. Lu, W. Y. Liu and J. P. Yin, *Neurocomputing*, 2014, **128**, 15–21.
40 Y. L. Jiang, Y. F. Shen, Y. Liu and W. C. Liu, *Math. Probl. Eng.*, 2015, 918105.
41 J. C. Laurentino Alves and R. J. Poppi, *Analyst*, 2013, **138**, 6477–6648.
42 O. Jovic, T. Smolic, I. Primozic and T. Hrenar, *Anal. Chem.*, 2016, **88**, 4516–4524.
43 Y. L. He, Z. Q. Geng and Q. X. Zhu, *Chemom. Intell. Lab. Syst.*, 2016, **151**, 78–88.
44 G. A. Barreto and A. L. S. P. Barros, *Neurocomputing*, 2016, **176**, 3–13.
45 O. O. Soyemi, M. A. Busch and K. W. Busch, *J. Chem. Inf. Comput. Sci.*, 2000, **40**, 1093–1100.
46 R. W. Kennard and L. A. Stone, *Technometrics*, 1969, **11**, 137–148.
47 Q. S. Xu and Y. Z. Liang, *Chemom. Intell. Lab. Syst.*, 2001, **56**, 1–11.