Proceedings of the 2012 International Conference on Machine Learning and Cybernetics, Xian, 15-17 July, 2012

A WEIGHTED VOTING METHOD USING MINIMUM SQUARE ERROR BASED ON EXTREME LEARNING MACHINE

JING-JING CAO, SAM KWONG, RAN WANG, KE LI

Department of computer science, The City University of Hong Kong, Tat Chee Avenue, Kowloon, Hong Kong
E-MAIL: jingcao3@student.cityu.edu.hk, cssamk@cityu.edu.hk

Abstract:

Extreme Learning Machine (ELM) has become popular for solving classification problem due to its fast speed. However, the system of ELM may be unreliable since its performance often relies on random input hidden node parameters. The techniques of combining multiple classifiers are widely adopted to improve both reliability and accuracy of a single classifier. Thus, this paper presents a minimum square error (MSE) based weighted voting method to optimize the linear combination of multiple ELMs. The experimental results over ten UCI data sets show better classification performance than the original ELM and the voting based ELM classifiers.

Keywords:

Extreme learning machine; Weighted voting; Minimum square error

1. Introduction

Extreme Learning machine (ELM) is an extened approach derived from single-hidden layer feedforward networks (SLFNs). Different from neural networks (NNs) [3], which cost high computational complexity, ELM develops a least-square method to obtain the approximately optimized weights for each activation function of SLFNs. Thus, the process for tuning the hidden layer parameters of SLFNs is avoided which extremely enhance the speed of the learning. However, Cao et al. [2] pointed out that since the input hidden node parameters are randomly generated, it is easy to misclassify some patterns that are close to the boundary.

The idea of ensemble method is to combine multiple classifiers based on different training patterns or features. Typically, the classification performance of combined classifier is better than each of the base classifiers. Many successful applications can be found in various fields, such as bioinformatics [10] and image retrieval [12]. In some ensemble techniques, the linear combination of classifiers often adopts weighted voting method instead of simple voting.

Recently, Cao et al. [2] design a voting based ELM (V-ELM) by employing ELM as base classifier under the framework of majority voting scheme. However, they simply incorporate the classifiers without considering the different performances of each single one. To tackle this issue, a common technique is to evaluate the confidence degree of each component classifier, which can be considered as weighted voting method.

In this paper, we adopt the minimum square error (MSE) based weight optimization approach to obtain better performance than basic ELM and V-ELM. Section 2 briefly introduces the principal theory of ELM. Section 3 presents the voting based ELM and the proposed weighted voting based ELM. The experimental results are shown in Section 4. Finally, Section 5 gives a conclusion.

2. Preliminary

Many related ensemble works have been devoted to the development of ELM. In [7], Liang et al. study an online sequential ELM (OS-ELM), which shows better generalization behavior than the other sequential algorithms. Then in [6], Lan et al. extend OS-ELM to an ensemble version and improve the stability performance of the OS-ELM as well. In [8], Liu et al. point out that ELM might be prone to overfit since it approximates the training data excessively. To alleviate this problem, they present an ensemble based ELM (EN-ELM) and embedded the cross-validation into the training process. The V-ELM algorithm [2] is designed to alleviate the uncertainty of the patterns that are close to the classification boundary.

For a classification problem, we typically have a training dataset with patterns in a d dimensional space, and each pattern belong to one of the m classes. In this paper, let the

978-1-4673-1487-9/12/$31.00 ©2012 IEEE

dataset denoted as $\boldsymbol{z}_n = (\boldsymbol{x}_n, \boldsymbol{y}_n), n = 1, 2, ..., N$, where $\boldsymbol{x}_n \in \mathbf{R}^d, \boldsymbol{y}_n \in \mathbf{R}^m$. In neural network field, the task for supervise learning is transformed to minimize a regression cost function $\| \widehat{\mathbf{Y}} - \mathbf{Y} \|$, where $\mathbf{Y} = (\boldsymbol{y}_1, \boldsymbol{y}_2, ..., \boldsymbol{y}_N)$ is the target output matrix, and $\widehat{\mathbf{Y}} = (\widehat{\boldsymbol{y}}_1, \widehat{\boldsymbol{y}}_2, ..., \widehat{\boldsymbol{y}}_N)$ is the output of network with L hidden nodes:

$$\widehat{\boldsymbol{y}}_n = \sum_{i=1}^{L} \boldsymbol{\beta}_i g(\boldsymbol{w}_i \cdot \boldsymbol{x}_n + b_i) \quad (1)$$

Where $\boldsymbol{w}_i \in \mathbf{R}^d$ and $b_i \in R$ $(i = 1, 2, ..., L)$ are the weight vector and threshold of the ith hidden node. $\boldsymbol{\beta}_i$ is the weight vector connecting the ith hidden node and the output nodes, and $g(\boldsymbol{w}_i \cdot \boldsymbol{x}_n + b_i)$ is the activation function of additive nodes. Two types of activation functions: Radial basis function (RBF) and Sigmoid functions are utilized in this paper, and they are defined as RBF function: $g(\boldsymbol{w}_i \cdot \boldsymbol{x}_n + b_i) = exp(-b_i \|\boldsymbol{x}_n - \boldsymbol{w}_i\|^2)$ and Sigmoid function: $g(\boldsymbol{w}_i \cdot \boldsymbol{x}_n + b_i) = 1/(1 + exp(-(\boldsymbol{w}_i \cdot \boldsymbol{x}_n + b_i)))$.

Equivalently, a compact format of Eq. (1) can be written as $\mathbf{H}\boldsymbol{\beta} = \mathbf{Y}$, where $\mathbf{H}_{ni} = g(\boldsymbol{w}_i \cdot \boldsymbol{x}_n + b_i)$ denotes the hidden layer output matrix, and $\boldsymbol{\beta} = (\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, ..., \boldsymbol{\beta}_L)$. The unique minimum norm least squares solution of the above linear system is

$$\boldsymbol{\beta} = \mathbf{H}^{\dagger}\mathbf{Y} \quad (2)$$

Where $\mathbf{H}^{\dagger}$ is the Moore-Penrose generalized inverse [9] of output matrix $\mathbf{H}$. Theoretically, Huang et al. [5] proposed the interpolation theorem and proved that the hidden layer parameters can be randomly generated if the activation function g is infinitely differentiable in any interval. Furthermore, they also showed the universal approximation theorem [4] and proved that SLFNs with randomly generated additive or RBF nodes can universally approximate any continuous target functions over any compact subset $X \in \mathbf{R}^d$.

3. MSE Based Weighted Voting ELM

Though there are many different methods to calculate the importance of each classifier, they usually require some time on learning the optimized weight vector, such as cross-validation. The simple MSE approach needs much less time to calculate the linear weights for base classifiers.

Suppose that we have T base ELM classifiers, which are trained with different randomly generated hidden node parameters. Let $f_t^{(j)}(\boldsymbol{x}_n)$ represents the confidence degree of $\boldsymbol{x}_n$ belonging to the jth class decided by the tth ELM. The weighted linear combination of these T base classifiers can be written as follows:

$$f_{com}^{(j)}(\boldsymbol{x}_n) = \boldsymbol{\alpha}^{(j)\mathcal{T}} \boldsymbol{f}^{(j)}(\boldsymbol{x}_n). \quad (3)$$

Where $\mathcal{T}$ denotes the transpose and $\boldsymbol{f}^{(j)}(\boldsymbol{x}_n) = (f_1^{(j)}(\boldsymbol{x}_n), ..., f_T^{(j)}(\boldsymbol{x}_n))^{\mathcal{T}}$. $\boldsymbol{\alpha}^{(j)\mathcal{T}} = (\alpha_1^{(j)}, ..., \alpha_T^{(j)})^{\mathcal{T}}$ represents the weight vector of the linear combination which is independent to the class label j. It should be noted that the $\boldsymbol{\alpha}^{(j)}$ vector in our work requires that $\sum_{t=1}^{T} \alpha_t^{(j)} = 1$ and $\alpha_t^{(j)} \geq 0$.

Thus, to decide the class label of a testing example $\boldsymbol{x}_p$ in the voting system, the decision strategy is given by the rule (4). Further, the V-ELM algorithm can also be considered as a special case of the weighted majority voting scheme with equal weights for each classifier, that is, $\boldsymbol{\alpha}^{(j)} = (\frac{1}{T}, ..., \frac{1}{T})^{\mathcal{T}}$.

$$\textit{Decide } \boldsymbol{y}_p \in c_j \quad \textit{if } f_{com}^{(j)}(\boldsymbol{x}_p) = \max_k f_{com}^{(k)}(\boldsymbol{x}_p) \quad (4)$$

Since the training error of each individual classifier is obtained, the problem is then transformed to how to pick the appropriate weights so that the additive error of the ensemble is minimal? The simple MSE method [1, 11] is a classical optimal weighting approach used for linear combination of multiple classifiers. According to the Eq. (3), the weighted voting based algorithm to solve the classification issue can be changed as selecting optimal $\boldsymbol{\alpha}^{(j)}$ that can minimize the least square error $\| \boldsymbol{f}_{com}(\boldsymbol{x}) - \mathbf{Y} \|^2$, where $\boldsymbol{f}_{com}(\boldsymbol{x}) = (f_{com}^{(1)}(\boldsymbol{x}), ..., f_{com}^{(m)}(\boldsymbol{x}))$

$$\begin{aligned} \widehat{\boldsymbol{\alpha}}^{(j)} &= \arg\min_{\boldsymbol{\alpha}^{(j)}} \{\sum_{n=1}^{N} (f_{com}^{(j)}(\boldsymbol{x}_n) - y_{nj})^2\} \\ &= \arg\min_{\boldsymbol{\alpha}^{(j)}} \{\sum_{n=1}^{N} (\boldsymbol{\alpha}^{(j)\mathcal{T}} \boldsymbol{f}^{(j)}(\boldsymbol{x}_n) - y_{nj})^2\} \quad (5) \\ &= (\sum_{n=1}^{N} \boldsymbol{f}^{(j)}(\boldsymbol{x}_n) \boldsymbol{f}^{(j)}(\boldsymbol{x}_n)^{\mathcal{T}})^{-1} \sum_{n=1}^{N} y_{nj} \boldsymbol{f}^{(j)}(\boldsymbol{x}_n) \end{aligned}$$

In the above equation, $(\cdot)^{-1}$ denotes the inverse of the corresponding matrix. In practice, the estimated $\widehat{\boldsymbol{\alpha}}^{(j)}$ can hardly satisfy the condition $\sum_{t=1}^{T} \widehat{\alpha}_t^{(j)} = 1$ and $\widehat{\alpha}_t^{(j)} \geq 0$. Then we normalized their values into a range $[0, 1]$ by the following equation:

$$\widetilde{\alpha}_t^{(j)} = \frac{\widehat{\alpha}_t^{(j)} - \min_{q \in \{1,...,T\}} (\widehat{\alpha}_q^{(j)})}{\max_{q \in \{1,...,T\}} (\widehat{\alpha}_q^{(j)}) - \min_{q \in \{1,...,T\}} (\widehat{\alpha}_q^{(j)})} \quad (6)$$

Thus, a MSE based weighted majority voting method, called WV-ELM, can be summarized as Algorithm 1.

Algorithm 1: WV-ELM

Input: Given a training set $Z = \{(\boldsymbol{x}_n, \boldsymbol{y}_n) \mid \boldsymbol{x}_n \in \mathbf{R}^d, \boldsymbol{y}_n \in \mathbf{R}^m\}_{n=1}^N$, hidden node output function $g(\boldsymbol{w}_i \cdot \boldsymbol{x}_n + b_i)$, hidden node number L, learning iteration **T**.

Initialization: $t = 1$

while $t \leq T$ **do**

1. Randomly generate the learning parameters $(\boldsymbol{w}_i^t, b_i^t)$ $(i = 1, 2, ..., L)$ of the tth ELM.
2. Calculate the hidden layer output matrix $\mathbf{H}^t$
3. Calculate the output weight $\boldsymbol{\beta}^t$: $\boldsymbol{\beta}^t = (\mathbf{H}^t)^\dagger \mathbf{Y}$, where $\mathbf{Y}$ is the target output matrix.
4. $t = t + 1$.

5. Calculate the $\widetilde{\boldsymbol{\alpha}}^{(j)}$ of each class $j, j = 1, ..., m$ according to Eq. (6).

6. Normalization: $\widetilde{\boldsymbol{\alpha}}^{(j)} = \widetilde{\boldsymbol{\alpha}}^{(j)} / \sum_{j=1}^m \widetilde{\boldsymbol{\alpha}}^{(j)}$.

Output: Final class label $\widehat{\boldsymbol{y}}_p$ obtained by weighted majority voting for a testing instance $\boldsymbol{x}_p$: $\widehat{\boldsymbol{y}}_p = \arg\max_k \widetilde{\boldsymbol{\alpha}}^{(k)\mathcal{T}} \boldsymbol{f}^{(k)}(\boldsymbol{x}_n)$

4. Simulation

Table 1. Datasets Information

Datasets	# Patterns	# Features	# Classes
Car	1728	6	4
Chart	600	60	6
Glass	214	9	6
Heart	270	13	2
Ionosphere	351	33	2
Iris	150	4	3
Letter	20000	16	26
Libras	360	90	15
Sonar	208	60	2
Wine	178	13	3

The simulations are conducted with the aim of comparing the performance of our approach with the original ELM and the V-ELM. The WV-ELM are respectively performed on ten UCI datasets with Sigmoid and RBF functions. Table 1 exhibits the basic information of UCI datasets, and the experiments implement 10 fold cross-validation(CV) as a trail and repeat it 10 times. For all the comparison methods, the number of hidden nodes is fixed on 20, and for both V-ELM and WV-ELM, $T = 7$ base classifiers (ELM) are utilized.

Tables 2 and 3 give the mean and standard deviation of testing accuracies among the three algorithms with RBF and Sigmoid functions respectively. From table 2, it can be seen that the WV-ELM method has the best classification performance on 7 datasets out of 10, while the V-ELM approach shows better testing behavior than the original ELM and performs best on 3 data sets. We can also see that the smallest standard deviation is obtained by the WV-ELM on most of data sets. Similarly, from table 3, it is observed that WV-ELM can provide better generalization results than the other two methods on 8 data sets. but the standard deviation is similar.

Table 2. Comparisons of the average testing accuracy and standard deviation with RBF function

Datasets	ELM	V-ELM	WV-ELM
Car	80.85 ± 0.86	81.69 ± 0.57	**82.52** ± 0.41
Chart	69.93 ± 2.10	86.98 ± 1.30	**90.80** ± 1.23
Glass	65.19 ± 2.03	67.66 ± 1.55	**67.85** ± 1.37
Heart	78.93 ± 1.08	82.19 ± 1.41	**82.63** ± 0.93
Ionosphere	83.79 ± 2.39	88.52 ± 1.17	**89.83** ± 1.07
Iris	96.80 ± 0.98	**97.80** ± 0.45	97.00 ± 0.72
Letter	50.11 ± 0.42	57.64 ± 0.33	**61.38** ± 0.19
Libras	47.50 ± 3.83	61.56 ± 2.02	**66.00** ± 1.42
Sonar	68.89 ± 3.48	**76.88** ± 3.15	76.20 ± 1.29
Wine	95.79 ± 0.96	**98.26** ± 0.72	98.03 ± 0.81

Table 3. Comparisons of the average testing accuracy and standard deviation with Sigmoid function

Datasets	ELM	V-ELM	WV-ELM
Car	81.45 ± 0.94	82.05 ± 0.39	**82.47** ± 0.42
Chart	78.00 ± 1.11	88.52 ± 1.93	**91.02** ± 1.26
Glass	64.30 ± 1.79	65.70 ± 1.06	**67.66** ± 0.96
Heart	83.04 ± 0.85	**83.96** ± 0.61	83.81 ± 0.55
Ionosphere	85.81 ± 0.88	87.78 ± 1.10	**88.80** ± 1.14
Iris	96.47 ± 0.55	**97.13** ± 0.71	97.07 ± 0.64
Letter	55.52 ± 0.33	57.63 ± 0.22	**59.07** ± 0.22
Libras	57.78 ± 2.68	63.83 ± 1.70	**67.86** ± 2.45
Sonar	73.56 ± 3.22	77.60 ± 2.51	**78.70** ± 1.95
Wine	97.81 ± 1.07	98.82 ± 0.56	**99.04** ± 0.53

Fig. 1 shows the average testing accuracies among the three algorithms with different number of input hidden nodes ($L = 10, 20, 30, 40, 50$). It can be seen that with the increase of input hidden nodes, the corresponding testing accuracies are improved, and the WV-ELM approach can always outperform the others. Fig. 2 depicts that the ensemble size affects the

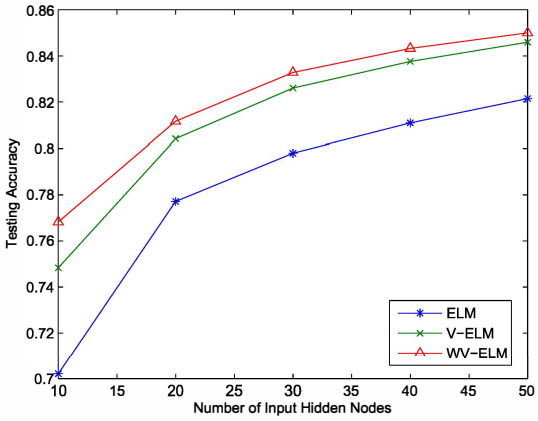

Figure 1. Average testing accuracy with different number of input hidden nodes

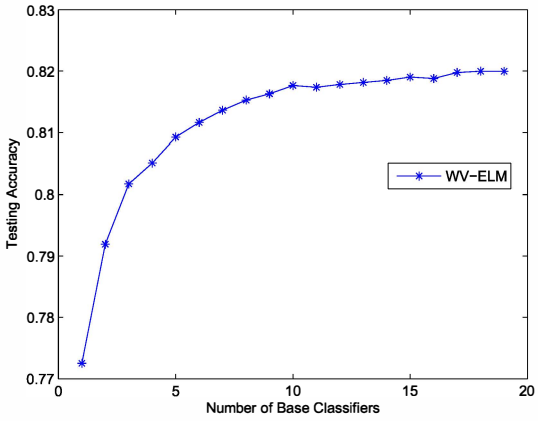

Figure 2. Average testing accuracy with different number of ensemble classifiers

classification behavior. However, this effect decreases with the ensemble size becomes large.

5. Conclusion

This paper deals with weighted voting methods based ELM, which implements the simple MSE method to optimize the weight coefficients of linearly combined classifiers. From the experiments, we can see that our weighted voting approaches show better performance than the original ELM and the V-ELM with regards to the mean and standard deviation.

Acknowledgements

This paper is supported by the City University of Hong Kong Grant 9610025.

References

[1] J.A. Benediktsson, J.R. Sveinsson, O.K. Ersoy, and P.H. Swain. Parallel consensual neural networks. *Neural Networks, IEEE Transactions on*, 8(1):54–64, 1997.

[2] Jiuwen Cao, Zhiping Lin, Guang-Bin Huang, and Nan Liu. Voting based extreme learning machine. *Inf. Sci.*, 185(1):66–77, 2012.

[3] M.T. Hagan, H.B. Demuth, M.H. Beale, and Boulder University of Colorado. *Neural network design.* PWS Pub, 1996.

[4] G.B. Huang, L. Chen, and C.K. Siew. Universal approximation using incremental constructive feedforward networks with random hidden nodes. *Neural Networks, IEEE Transactions on*, 17(4):879–892, 2006.

[5] G.B. Huang, Q.Y. Zhu, and C.K. Siew. Extreme learning machine: theory and applications. *Neurocomputing*, 70(1):489–501, 2006.

[6] Y. Lan, Y.C. Soh, and G.B. Huang. Ensemble of online sequential extreme learning machine. *Neurocomputing*, 72(13-15):3391–3395, 2009.

[7] N.Y. Liang, G.B. Huang, P. Saratchandran, and N. Sundararajan. A fast and accurate online sequential learning algorithm for feedforward networks. *Neural Networks, IEEE Transactions on*, 17(6):1411–1423, 2006.

[8] N. Liu and H. Wang. Ensemble based extreme learning machine. *Signal Processing Letters, IEEE*, 17(8):754–757, 2010.

[9] D. Serre. Matrices: Theory and applications. 2002.

[10] A. Tan, D. Gilbert, and Y. Deville. Multi-class protein fold classification using a new ensemble machine learning approach. 2003.

[11] N. Ueda. Optimal linear combination of neural networks for improving classification performance. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(2):207–215, 2000.

[12] L. Yang, R. Jin, L. Mummert, R. Sukthankar, A. Goode, B. Zheng, S.C.H. Hoi, and M. Satyanarayanan. A boosting framework for visuality-preserving distance metric learning and its application to medical image retrieval. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(1):30–44, 2010.