Class-specific soft voting based multiple extreme learning machines ensemble

Jingjing Cao [a,b], Sam Kwong [a,*], Ran Wang [a], Xiaodong Li [a], Ke Li [a], Xiangfei Kong [a]

[a] Department of Computer Science, City University of Hong Kong, Tat Chee Avenue, Kowloon, SAR 999077, Hong Kong
[b] School of Logistics Engineering, Wuhan University of Technology, Wuhan 430070, China

ARTICLE INFO

ABSTRACT

Compared with conventional weighted voting methods, class-specific soft voting (CSSV) system has several advantages. On one hand, it not only deals with the soft class probability outputs but also refines the weights from classifiers to classes. On the other hand, the class-specific weights can be used to improve the combinative performance without increasing much computational load. This paper proposes two weight optimization based ensemble methods (CSSV-ELM and SpaCSSV-ELM) under the framework of CSSV scheme for multiple extreme learning machines (ELMs). The designed two models are in terms of accuracy and sparsity aspects, respectively. Firstly, CSSV-ELM takes advantage of the condition number of matrix, which reveals the stability of linear equation, to determine the weights of base ELM classifiers. This model can reduce the unreliability induced by randomly input parameters of a single ELM, and solve the ill-conditioned problem caused by linear system structure of ELM simultaneously. Secondly, sparse ensemble methods can lower memory requirement and speed up the classification process, but only for classifier-specific weight level. Therefore, a SpaCSSV-ELM method is proposed by transforming the weight optimization problem to a sparse coding problem, which uses the sparse representation technique for maintaining classification performance with less nonzero weight coefficients. Experiments are carried out on twenty UCI data sets and Finance event series data and the experimental results show the superior performance of the CSSV based ELM algorithms by comparing with the state-of-the-art algorithms.

© 2014 Elsevier B.V. All rights reserved.

1. Introduction

Extreme learning machine (ELM) becomes popular for solving classification problem due to its light computational requirements. It is an extension of the single-hidden layer feedforward networks (SLFNs). By making use of a least-square method, it analytically obtains the output weights of SLFNs [1]. Moreover, ELM emphasizes on achieving both the smallest norm of output weights and the least training error, which is different from conventional neural type of SLFNs. Essentially, ELM is originally designed by utilizing random computational nodes, which are independent of the training data. The process for tuning the hidden layer parameters is avoided, which significantly shortens the learning time. A great many of ELM based algorithms have been done in recent years [1–3].

However, since the input hidden nodes are randomly generated, it is easy to misclassify patterns that are close to the boundary [3,4]. In order to improve the classification performance, a number of real world applications based on ensemble learning have been done in

previous research [2,5–7]. Different from designing a single classifier in traditional pattern recognition field, ensemble learning consists of a group of machine learning algorithms that aims at constructing multiple classifiers to form a hybrid predictive model. Generally speaking, the overall classification performance of ensemble classifier could be better than using a single classifier. The ensemble learning aims at a high accurate prediction at the expense of increased complexity. In multiple classifier system (MCS), the field of ensemble learning usually employs homogeneous base learners. In the past few decades, many ensemble techniques [8–10] are proposed to enhance the reliability of multiple models. Besides, ensemble methods are also successfully applied into applications from a wide range of fields [11–13] due to their remarkable capability in increasing the classification performance of a learning model.

Numerous works have been proposed regarding to ensemble ELMs in recent years. In [14], Liang et al. proposed the online sequential extreme learning machine (OS-ELM), which shows better generalization behavior than the other sequential algorithms. Then in [5], Lan et al. extended OS-ELM to an ensemble version and improved the stability. In [6], Liu and Wang pointed out that ELM might be prone to overfit since it approximates the training data in

learning phase. To alleviate this problem, they presented an ensemble based ELM (EN-ELM) and embedded the cross-validation into the training process. Wang and Li [7] designed a dynamic Adaboost ensemble method by using multiple ELMs with fuzzy activation function to deal with large data sets. Different from this method, Zhai et al. [2] developed the sample entropy based dynamic ensemble to handle the instability and overfitting problems of ELM. Wang et al. [4] regarded the upper integral as a base classifier and constructed an upper integral network through the learning mechanism of ELM. van Heeswijk et al. [15] introduced an adaptive ensemble models of ELMs for time series prediction. The proposed algorithm aims at performing well on both nonstationary and stationary time series. Besides, Heeswijk et al. [16] also proposed the GPU-accelerated and parallelized ELM ensemble to perform regression on large data sets.

As aforementioned, the randomly selected parameters in ELM will lead to unstable training accuracy. Therefore, Cao et al. [3] designed a voting based extreme learning machine (V-ELM) by employing ELM as base classifier under the framework of majority voting. Despite the demonstrated reliability and stability, the fact that different base classifiers have different classification performances has not been considered in their work. In ensemble learning techniques, the combinative linear classifier adopts weighted voting instead of simple voting when the accuracies of base classifiers are unequal [17]. In view of this reason, weighted fusion methods can be utilized to evaluate the confidence degree of each base classifier. The weighted voting methods mainly include the weighted majority vote schemes [18] and weighting methods [19,20] with classifier-specific weights. In previous research, the minimum square error (MSE) based method [21] was proposed as a class-specific optimal weighting approach used for linear combination of multiple classifiers. It is easy to be implemented but the weights are not optimized. Recently, Zhang and Zhou [22] proposed three new weighted combination approaches that was inspired by the idea of sparse ensembles. They employed linear programming (LP) algorithm to select classifiers and tuned their weights simultaneously. This approach made use of optimization tool to get satisfactory results. However, the weights they used are also defined on the classifier-specific level. One cannot delicately tune or optimize the weight assignment distribution on this level. Therefore, in this work, we apply the class-specific weight based soft voting for ELM classifiers (CSSV-ELM) by optimizing a class-specific weight based model. Further, since the latter steps of ELM can be regarded as solving a linear equation problem, it may suffer from the ill-conditioned problem. Thus, the condition number of the inverse of the weight matrix between hidden nodes and output nodes could be considered as part of the constraints in the optimization model. Particularly, this model takes into account both the best-worst weighted voting measure and the condition number simultaneously.

To improve computational efficiency and increase test speed, another interesting problem is how to construct sparse ensemble for class-specific soft voting scheme. Concretely, sparse ensemble aims at finding a sparse weight vector to sparsely represent the outputs of multiple classifiers. In classifier-specific weight, the sparse ensemble concept is equivalent to ensemble pruning, which aims at selecting an optimal sub-ensemble (a subset of classifiers) from a weight vector. However, in class-specific soft voting scheme, a class-based weight matrix should be determined. The pruned ensemble methods [9,22–24] are not well suited to obtain the sparse ensemble with class weight matrix. Compared with these conventional pruning methods, sparse representation based methods have been popular recently due to its flexibility to construct various optimization models based on diverse problems. The other advantage is that once the model is built by selecting an appropriate over-complete dictionary, the corresponding solution algorithm is well prepared.

Furthermore, sparse representation has shown strong relationship to classification and face recognition [25–28]. The main technique for sparse representation is sparse coding and its variants has been successfully used in face recognition. Therefore, the sparse coding (SC) techniques are applied so as to represent class weight coefficients sparsely for multiple ELMs in this paper. The problem of SC origins from sparse representations of signals and the goal is to find a linear decomposition of a signal with a few atoms of a over-complete dictionary [28]. However, the objective function for optimizing the class-specific based weights does not naturally fit to the sparse coding condition since the "dictionary matrix" is not over-complete. In face recognition field, a common way for solving this problem is to map the high-dimensional data to low-dimensional spaces by using feature extraction techniques. Thus, in this work, we apply an iterative optimization algorithm for adapting the feature extraction projection matrix $\boldsymbol{P}$ and the weight coefficients $\boldsymbol{\alpha}_k$ for each class k simultaneously to exploit more robust and efficient classifier. We named the proposed model as sparse based class-specific soft voting ELM (SpaCSSV-ELM). On one hand, for fixed $\boldsymbol{\alpha}_k$, the update of $\boldsymbol{P}$ is to exploit more appropriate transformation for original conditional probability outputs of base classifiers. On the other hand, the weight $\boldsymbol{\alpha}_k$ can be refined when $\boldsymbol{P}$ is fixed. In this way, the learned $\boldsymbol{P}$ and $\boldsymbol{\alpha}_k$ can improve the robustness of sparse representation of the proposed model.

In this work, two new optimization methods based on class-specific weight are proposed for multiple ELMs with three contributions:

- The first contribution is that a convex optimization model (CSSV-ELM) is designed based on the class-specific soft voting scheme.
- The second contribution lies in constructing the constraints of optimization model (CSSV-ELM). Besides the constraint $\sum_{t=1}^{T} \alpha_t^j = 1$ and $\alpha_t^j \geq 0$ for each class, the weight constraint for each classifier is formed by combining the worst-best weighted voting and the condition number of ELM, which guarantees the importance and stability of each component ELM.
- The third contribution focuses on learning the sparse weight vector based on class-specific soft voting method with ELM as base classifier.

Overall, under the framework of class-specific weight based soft voting, two models that are related to ELM characteristic and sparse ensemble aspects, are designed to improve performance in terms of accuracy and sparsity. The rest of this paper is organized as follows. Section 2 briefly reviews ELM and some background on ensemble learning. Section 3 describes the proposed CSSV-ELM and SpaCSSV-ELM algorithms. Experimental results are shown in Section 4. Finally, Section 5 gives a conclusion.

2. Preliminary on extreme learning machine and related knowledge

2.1. Extreme learning machine

For a classification problem, we typically have a d-dimensional training data set with patterns that belong to one of m classes each. In this paper, let the data set 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 supervised 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:

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