Information Sciences 185 (2012) 66–77

Contents lists available at SciVerse ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Voting based extreme learning machine

Jiuwen Cao [a,*], Zhiping Lin [a], Guang-Bin Huang [a], Nan Liu [b]

[a] School of Electrical and Electronic Engineering, Nanyang Technological University, Nanyang Avenue, Singapore 639798, Singapore
[b] Department of Emergency Medicine, Singapore General Hospital, Outram Road, Singapore 169608, Singapore

ARTICLE INFO

Article history:
Received 3 November 2010
Received in revised form 21 May 2011
Accepted 15 September 2011
Available online 22 September 2011

Keywords:
Classification
Extreme learning machine
Majority voting
Single hidden layer feedforward networks
Ensemble methods

ABSTRACT

This paper proposes an improved learning algorithm for classification which is referred to as voting based extreme learning machine. The proposed method incorporates the voting method into the popular extreme learning machine (ELM) in classification applications. Simulations on many real world classification datasets have demonstrated that this algorithm generally outperforms the original ELM algorithm as well as several recent classification algorithms.

1. Introduction

During past several decades, a lot of methods have been developed for classifications. The most representative approaches are the traditional Bayesian decision theory [8,21], support vector machine (SVM) and its variants [12,25], artificial neural network (ANN) and its variants [9–11,24,27,33], fuzzy method and its variants [5,13,20,23,36], etc. Among these methods, ANN provides several particular characteristics. First, by incorporating certain learning algorithms to change the network structure and parameters based on external or internal information that flows through the network, a neural network can adaptively fit to the data without any explicit specifications of the underlying model. Second, neural networks have the universal approximation characteristic [10,11]. It can be used to approximate any function to arbitrary accuracy. As in classification applications, the general procedure is to construct a functional relationship between several given attributes of an object and its class label. Therefore, the universal approximation feature can make the neural network to be an efficient classification tool. Finally, the changeable network structure and nonlinear basic computing neurons make neural networks flexible in modeling the complex functional relationship of real world applications.

Recently, a least square based learning algorithm named extreme learning machine (ELM) [14] was developed for single hidden layer feedforward networks (SLFNs). Using random computational nodes which are independent of training samples, ELM has several promising features. It is a tuning free algorithm and learns much faster than traditional gradient-based approaches, such as Back-Propagation [9] algorithm (BP) and Levenberg–Marquardt [24,27] algorithm. Moreover, ELM tends to reach the small norm of network output weights as Bartlett's theory [1] states that for feedforward neural networks reaching small training error, the smaller the norm of weights is, the better generalization performance the network tends to have. It has been further shown [15–18] that many types of computational hidden nodes which may not be neuron alike nodes can be used in ELM as long as they are piecewise nonlinear, such as radial basis function (RBF) hidden nodes [16], fully complex nodes [18], wavelets [3,4], etc. A number of real world applications based on ELM have been done in recent years [19,34,35].

* Corresponding author.
E-mail addresses: caoj0003@e.ntu.edu.sg, jiuwcao919@126.com (J. Cao).

0020-0255/$ - see front matter
doi:10.1016/j.ins.2011.09.015

ELM performs classification by mapping the signal label to a high dimensional vector and transforming the classification task to a multi-output function regression problem. An issue with ELM is that as the hidden node learning parameters in ELM are randomly assigned and they remain unchanged during the training procedure, the classification boundary may not be an optimal one. Some samples may be misclassified by ELM, especially for those which are near the classification boundary.

To reduce the number of such misclassified samples, we propose in this paper an improved algorithm called voting based extreme learning machine (in short, as V-ELM). The main idea in V-ELM is to perform multiple independent ELM training instead of a single ELM training, and then make the final decision based on the majority voting method [29]. Compared with the original ELM algorithm, the proposed V-ELM is able not only to enhance the classification performance and reduce the number of misclassified samples, but also to lower the variance among different realizations. Simulations on many real world classification datasets demonstrate that V-ELM outperforms several recent methods in general, including the original ELM [14], support vector machine (SVM) [12], optimally pruned extreme learning machine (OP-ELM) [28], Back-Propagation algorithm (BP) [9,24,27], K nearest neighbors algorithm (KNN) [2,7], robust fuzzy relational classifier (RFRC) [5], radial basis function neural network (RBFNN) [33] and multiobjective simultaneous learning framework (MSCC) [6].

The organization of this paper is as follows. In Section 2, we first briefly review the basic concept of ELM. Then, we analyze an issue with ELM in classification applications, and present the V-ELM. Simulation results and comparisons are provided in Section 3. In Section 4, discussions on the performance of V-ELM with respect to different independent training numbers and on three recent methods [22,26,32] which also exploit multiple classifiers in ELM are given. Conclusions are drawn in Section 5. An appendix is given at the end to illustrate the three propositions presented in Section 2.

2. Voting based extreme learning machine

In this section, we first review the basic concept of the ELM algorithm for SLFNs in Section 2.1. Then, we analyze an issue that may exist in ELM when performing classification applications in Section 2.2. Finally, the new proposed V-ELM algorithm will be presented in Section 2.3 to tackle the issue and enhance the classification performance of ELM.

2.1. Review of extreme learning machine

Different from traditional theories that all the parameters of the feedforward neural networks need to be tuned to minimize the cost function, ELM theories [14–16] claim that the hidden node learning parameters can be randomly assigned independently and the network output weights can be analytically determined by solving a linear system using the least square method. The training phase can be efficiently completed without time-consuming learning iterations and ELM can achieve a good generalization performance.

For N arbitrary training samples $\{(\boldsymbol{x}_i, \boldsymbol{t}_i)\}_{i=1}^{N}$, where $\boldsymbol{x}_i \in \mathbf{R}^d$ and $\boldsymbol{t}_i \in \mathbf{R}^m$, the output of a SLFN with L hidden nodes is

$$\boldsymbol{o}_i = \sum_{j=1}^{L} \boldsymbol{\beta}_j G(\boldsymbol{a}_j, b_j, \boldsymbol{x}_i), \quad i = 1, 2, \ldots, N \tag{1}$$

where $\boldsymbol{a}_j \in \mathbf{R}^d$ and $b_j \in \mathrm{R}$ ($j = 1,2,\ldots,L$) are learning parameters of the jth hidden node, respectively. $\boldsymbol{\beta}_j \in \mathbf{R}^m$ is the link connecting the jth hidden node to the output node. $G(\boldsymbol{a}_j, b_j, \boldsymbol{x}_i)$ is the output of the jth hidden node with respect to the input sample $\boldsymbol{x}_i$.

For all N samples, an equivalent compact form of (1) can be written as

$$\mathbf{O} = \mathbf{H}\boldsymbol{\beta} \tag{2}$$

where $\mathbf{H}_{ij} = G(\boldsymbol{a}_j, b_j, \boldsymbol{x}_i)$ represents the entry in the ith row and jth column of the hidden layer output matrix $\mathbf{H}$, and $\boldsymbol{\beta} = (\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \ldots, \boldsymbol{\beta}_L)$ and $\mathbf{O} = (\boldsymbol{o}_1, \boldsymbol{o}_2, \ldots, \boldsymbol{o}_N)$.

To minimize the network cost function $\|\mathbf{O} - \mathbf{T}\|$, where $\mathbf{T} = (\boldsymbol{t}_1, \boldsymbol{t}_2, \ldots, \boldsymbol{t}_N)$ is the target output matrix, ELM theories claim that the hidden node learning parameters $\boldsymbol{a}_j$ and b_j can be randomly assigned *a priori* without considering the input data. Thus, the system Eq. (1) becomes a linear model and the output weights can be analytically determined by finding a least-square solution of this linear system (1) as follows

$$\boldsymbol{\beta} = \mathbf{H}^{\dagger}\mathbf{T} \tag{3}$$

where $\mathbf{H}^{\dagger}$ is the Moore–Penrose generalized inverse [30] of the hidden layer output matrix $\mathbf{H}$.

ELM adopts an One-Against-All (OAA) method to decompose multi-classification applications into multiple binary classifiers and transforms the classification application to a multi-output function regression problem. Then, the one with the largest output value is used to represent the class label of the given sample. For a C-labels classification application, the output label $\boldsymbol{t}_i$ of a sample $\boldsymbol{x}_i$ is usually encoded to a C-dimensional vector $(t_{i1}, t_{i2}, \ldots, t_{iC})^T$ with $t_{ic} \in \{1, -1\}$ ($c = 1,2,\ldots,C$). In the OAA approach, if the class label $\boldsymbol{t}_i$ of the sample $\boldsymbol{x}_i$ is c, then t_{ic} will be set to be 1 and the others are set to be -1 in the new formed C-dimensional output vector. Therefore, the class label c^{test} of a testing sample $\boldsymbol{x}^{\text{test}}$ predicted by the ELM algorithm is the index of the largest entry in the corresponding output vector.