Neurocomputing 144 (2014) 318–328

Contents lists available at ScienceDirect

Neurocomputing

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

Robust activation function and its application: Semi-supervised kernel extreme learning method

Shenglan Liu [a,b], Lin Feng [a,b,*], Yao Xiao [a,b], Huibing Wang [b]

[a] School of Computer Science and Technology, Faculty of Electronic Information and Electrical Engineering, Dalian University of Technology, Dalian 116024, China

[b] School of Innovation Experiment, Dalian University of Technology, Dalian 116024, China

ARTICLE INFO

Article history:
Received 13 August 2013
Received in revised form
11 April 2014
Accepted 27 April 2014
Communicated by G.-B. Huang
Available online 11 June 2014

Keywords:
Semi-supervised classification
Extreme Learning Machine
Robust activation function
Kernel method

ABSTRACT

Semi-supervised learning is a hot topic in the field of pattern recognition, this paper analyzes an effective classification algorithm – Extreme Learning Machine (ELM). ELM has been widely used in the applications of pattern recognition and data mining for its extremely fast training speed and highly recognition rate. But in most of real-world applications, there are irregular distributions and outlier problems which lower the classification rate of ELM (kernel ELM). This is mainly because: (1) Overfitting caused by outliers and unreasonable selections of activation function and kernel function and (2) the labeled sample size is small and we do not making full use of the information of unlabeled data either. Against problem one, this paper proposes a robust activation function (RAF) based on analyzing several different activation functions in-depth. RAF keeps the output of activation function away from zero as much as possible and minimizes the impacts of outliers to the algorithm. Thus, it improves the performance of ELM (kernel ELM); simultaneously, RAF can be applied to other kernel methods and a neural network learning algorithm. Against problem two, we propose a semi-supervised kernel ELM (SK-ELM). Experimental results on synthetic and real-world datasets demonstrate that RAF and SK-ELM outperform the ELM which use other activation functions and semi-supervised (kernel) ELM methods.

1. Introduction

The widespread popularity of single-hidden layer feedforward neural networks (SLFNs) in classification and prediction fields is mainly due to their excellent capability of approximating complex nonlinear functions. However, conventional learning methods of SLFNs utilize gradient-based algorithms to get the iterative solution. In this case, we need to consider the convergence problem to avoid multiple iterations and local optimal solution. Therefore, the learning speed of SLFNs becomes the bottleneck in applications. Recently, Huang et al. [1] mentioned that appropriate output weights were not dependent on the input weights and hidden layer neurons' biases. According to this discovery, they proposed a variety of SLFNs without an iterative calculation process called Extreme Learning Machine (ELM). ELM randomly chooses the input weights and the hidden neurons' biases and determines the output weights through a simple linear system at an extremely fast training speed. At the same time, ELM can avoid problems of the local optimal solution and the slow speed of convergence. Furthermore, when dealing with classification problems, ELM can get better solutions than SVM. In order to extend the learning capability of ELM, many researchers have improved ELM. Huang et al. have proved that SLFNs can approximate any continuous target function by randomly adding nodes of activation functions like RBF. Based on that, they proposed an incremental ELM algorithm [3] and a convex incremental ELM algorithm [4]. Tang et al. used the local Lanczos bidiagonalization method to calculate the output weights of ELM to enhance the stability of calculating a generalized inverse matrix. However, the number of iteration in Lanczos has a close relationship with the features of the target matrix and might have a great effect on computation efficiency.

When there are outliers in training datasets, the accuracy of ELM will be greatly affected. For this reason, Hortata et al. [7] proposed a Robust ELM algorithm, this algorithm constructs an Extended Complete Orthogonal Decomposition to get the output weights and calculates the final output weights by iterating the initial output weights. The time complexity of Robust ELM equals ELM which calculates output weights with SVD. However, Robust ELM is more robust to outliers. This is because Robust ELM lowers the empirical risk of ELM through a reasonable numerical calculation. Thus, we can know that the robustness of ELM has a close relationship with empirical risk. Deng et al. [8] proposed a weighted Regular ELM (RELM), this algorithm lowered the empirical risk

* Corresponding author at: School of Computer Science and Technology, Faculty of Electronic Information and Electrical Engineering, Dalian University of Technology, Dalian 116024, China.

http://dx.doi.org/10.1016/j.neucom.2014.04.041
0925-2312/

weights of outliers to enhance the robustness of RELM to outliers by weighting. However, this method is hard to be extended to a kernel method.

Huang et al. [9,10] have discussed the case of ELM with RBF and have proved that selections of input weights and bias do not influence the network's performance in theory. In the ELM algorithm, the features of hidden layer output matrix affect the output weights greatly. When dealing with high-dimensional data, the hidden layer output matrix may include lots of elements close to zero and the final output weights will be affected. Taking the Yale face datasets as an example, when the dimensions of Yale face images are reduced to 100 with linear dimensionality reduction methods, the 2 condition number of hidden layer output matrix is 1.0409e+13 with RBF(Data has been normalized), but with Sigmoid kernel the 2 condition number is only 114.4732. Therefore, RBF seriously affects the features of hidden layer output matrix which leads to inaccurate output weights. When there are outliers or noises, this phenomenon is more serious. To solve this problem, this paper proposes a Robust Activation Function (RAF). RAF translates the Euclidean distance measure in the Gaussian kernel function to Cosine measure. The Cosine measure can avoid the influences of outliers and the overfitting phenomenon as much as possible. Furthermore (see Fig. 1), more details can be seen in our previous work [11]. Simultaneously, RAF kernel has good features of activation functions and will not cause the ill-posed problem of hidden layer output matrix. Therefore, the performance of ELM can be improved greatly.

Although, the methods mentioned above have done a lot of improvements on ELM, but in real-world datasets, there are always few labeled samples. Therefore, in recent years, semi-supervised learning (SSL) attracts much attention of many researchers [12,13,15] and many excellent results appear. Such as Belkin et al. [12] proposed a Laplace graph SSL framework; Yan et al. [13] proposed a l 1 graph which is successfully applied in SSL based on sparse representation theory; Wang et al. translate LLE to a graph to implement the SSL based on the idea of LLE.

To improve the generalization capability of ELM and making full use of unlabeled samples, Liu et al. [5] proposed a semi-supervised ELM. This algorithm extends ELM to a semi-supervised version based on the graph theory and the semi-supervised learning framework proposed by Belkin. They also applied the semi-supervised ELM to a Wi-Fi positioning problem. But this method is only a particular solution of ELM in the semi-supervised environment. Besides, Huang et al. [16] proposed another excellent semi-supervised ELM version.

This paper utilizes RAF to implement ELM, the robustness and stability of ELM are improved. Meanwhile, we extend ELM to SK-ELM. SK-ELM can learn nonlinear distribution of datasets better and have a better generalization ability than ELM and SELM. Furthermore, we apply RAF to SK-ELM to strengthen the stability of SK-ELM and avoid the generalized inverse problem of ill-posed matrix. The main contributions of this paper are as follows:

(1) We propose a robust activation function (RAF) and prove that RAF can promote the performance of ELM in theory and experiments. RAF can be used in any kernel methods.

(2) We discuss the solutions of SELM more intensively and point out the suitable scope of SELM's solution.

(3) Reference [8] indicated that Kernel ELM could not find the appropriate model when dealing with outliers. This paper extends ELM to SK-ELM and applies RAF to SK-ELM to improve the robustness of the ELM kernel method.

2. A brief of ELM

For N arbitrary distinct samples (x_i, t_i), where $X = [x_1, x_2, \ldots, x_N]^T \in R^{D\times N}$, $t_i = [t_{i1}, t_{t2}, \ldots, t_{im}]^T \in R^m$, with $\tilde{N}$ being the hidden neurons in the network and the activation function is $g(x)$:

$$\sum_{i=1}^{\tilde{N}} \beta_i g(a_i x_j + b_j) = o_j \tag{1}$$

where $j = 1, \ldots, N$, $a_i = [a_{i1}, a_{i2}, \ldots, a_{in}]^T$ is the input weight vector connecting input neurons and the ith hidden neuron, b_i is the bias of the ith hidden node, $a_i \cdot x_i$ is the inner product of a_i and x_i.

Given hidden neurons $\tilde{N}$, Eq. (1) can be rewritten in a matrix form $H\beta = T$, where the network hidden layer output matrix is

$$H = \begin{bmatrix} g(a_1, x_1, b_1) & \cdots & g(a_{\tilde{N}}, x_1, b_{\tilde{N}}) \\ \vdots & \cdots & \vdots \\ g(a_1, x_N, b_1) & \cdots & g(a_{\tilde{N}}, x_N, b_{\tilde{N}}) \end{bmatrix}_{N\times\tilde{N}}, \quad \beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_{\tilde{N}}^T \end{bmatrix}_{\tilde{N}\times m}$$

and

$$T = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix} = \begin{bmatrix} t_{11} & \cdots & t_{1m} \\ \vdots & \vdots & \vdots \\ t_{N1} & \cdots & t_{Nm} \end{bmatrix}.$$

The standard SLFNs aim to find some appropriate $\hat{a}_i$, $\hat{b}_i$ and $\hat{\beta}$ $(i = 1, \ldots, \tilde{N})$ to satisfy

$$\|H(\hat{a}_1, \ldots, \hat{a}_{\tilde{N}}, \hat{b}_1, \ldots, \hat{b}_{\tilde{N}})\beta - T\| = \min_{a_i, b_i, \beta} \|H(a_1, \ldots, a_{\tilde{N}}, b_1, \ldots, b_{\tilde{N}})\beta - T\| \tag{2}$$

Eq. (2) can be solved with gradient-based algorithms, Huang et al. [1] have proved that the weights between input layer and

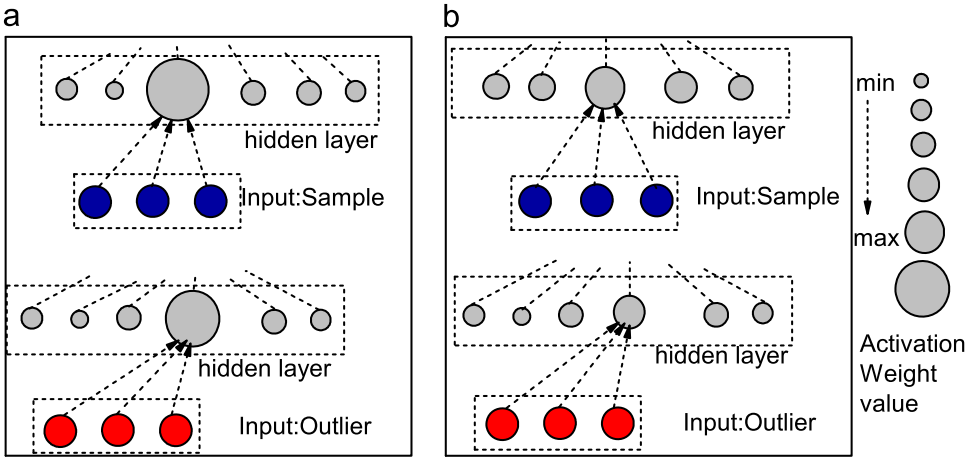

Fig. 1. Effects of different activation functions to the hidden layer. (a) RBF and (b) RAF.