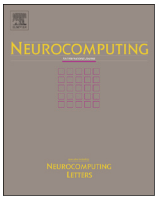
Neurocomputing 149 (2015) 295–307

Contents lists available at ScienceDirect

Neurocomputing

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



A fast learning method for feedforward neural networks

Shitong Wang[a,b,*], Fu-Lai Chung[b], Jun Wang[a], Jun Wu[a]

[a] School of Digital Media, Jiangnan University, Wuxi, Jiangsu, China
[b] Department of Computing, Hong Kong Polytechnic University, Hong Kong, China

ARTICLE INFO

Article history:
Received 8 August 2013
Received in revised form
21 December 2013
Accepted 19 January 2014
Available online 18 September 2014

Keywords:
Fast learning method
Feedforward neural network
Extreme learning machine
Hidden-feature-space ridge regression

ABSTRACT

In order to circumvent the weakness of very slow convergence of most traditional learning algorithms for single layer feedforward neural networks, the extreme learning machines (ELM) has been recently developed to achieve extremely fast learning with good performance by training only for the output weights. However, it cannot be applied to multiple-hidden layer feedforward neural networks (MLFN), which is a challenging bottleneck of ELM. In this work, the novel fast learning method (FLM) for feedforward neural networks is proposed. Firstly, based on the existing ridge regression theories, the hidden-feature-space ridge regression (HFSR) and centered ridge regression Centered-ELM are presented. Their connection with ELM is also theoretically revealed. As special kernel methods, they can inherently be used to propagate the prominent advantages of ELM into MLFN. Then, a novel fast learning method FLM for feedforward neural networks is proposed as a unified framework for HFSR and Centered-ELM. FLM can be applied for both SLFN and MLFN with a single or multiple outputs. In FLM, only the parameters in the last hidden layer require being adjusted while all the parameters in other hidden layers can be randomly assigned. The proposed FLM was tested against state of the art methods on real-world datasets and it provides better and more reliable results.

1. Introduction

The wide popularity of feedforward neural networks in many fields is mainly due to two factors: (1) the strong approximation capability for complex multivariate nonlinear function directly from input samples; (2) The strong modeling capability for a large class of natural and artificial phenomena which are very difficult to handle with classical parameter techniques. However, when applied to many application scenarios, feedforward neural networks often face a serious bottleneck issue: their traditional learning algorithms are usually much slower than required, for example, taking several hours, several days and even more.

From a mathematical viewpoint, researches about the approximation capability of feedforward neural networks can be categorized into two types: universal approximation on compact input sets and approximation of a finite set of samples. Theoretical results about the universal approximation of feedforward neural networks have been obtained by Hornik and Lesino, see [1,2]. In real-world applications, since feedforward neural networks are trained on a finite set of samples, much more endeavors should be taken for the approximation capability of the second type. Typically, gradient descent based learning algorithms like BP [3–7] of feedforward neural networks have been developed and extensively applied in the last decades. When these learning algorithms are used, all the parameters of the feedforward neural networks need to be adjusted in a backward way and thus there exists the dependence relationship between different layers of parameters in the network. Due to iterative learning steps, these learning algorithms generally converge very slowly and even to local minima. On the other hand, cross-validation and/or early stopping are sometimes adopted to circumvent the so called over-fitting phenomena.

In order to overcome these shortcomings of these learning algorithms, Huang et al. proposed the extreme learning machine (ELM) for single hidden layer feedforward neural networks (SLFN) [3–21]. They proved that the input weights and the hidden layer biases can be randomly assigned if the activation function in the hidden layer is infinitely differentiable. Once the input weights and the hidden layer biases are randomly assigned, SLFN can be considered as a linear system and the output weights of SLFN can be analytically solved by using the simple generalized inverse operation of the hidden layer output matrix. With its easy implementation, ELM can reach both the smallest training error and the smallest norm of weights and thus provide good generalization performance in extremely fast learning speed, for example, thousands of times faster than BP in many applications[7].

However, as stated in [7], *it should be worth pointing out that gradient-based learning algorithms like back-propagation can be used*

[*] Corresponding author at: School of Digital Media, Jiangnan University, Wuxi, Jiangsu, China.
E-mail address: wxwangst@aliyun.com (S. Wang).

http://dx.doi.org/10.1016/j.neucom.2014.01.065
0925-2312/© 2014 Elsevier B.V. All rights reserved.

for feedforward neural networks which have more than one hidden layers while the proposed ELM algorithm at its present form is still only valid for SLFN. In other words, ELM at its present form cannot be directly applied to multiple hidden layer feedforward neural networks. In this paper, we first propose the hidden-feature-space ridge regression HFSR and centered ridge regression Centered-ELM for both SLFN and MLFN, and then build the link between extreme learning machine (ELM) and them for SLFN. As the special kernel methods, the virtues of both HFSR and Centered-ELM exist in that rigorous Mercer's condition for kernel functions is not required and that it plays a bridging role in naturally propagating the prominent advantages of ELM into MLFN by using randomly assigned parameters and randomly-selected examplars for kernel activation functions. Through constructing the transformed data set from the training dataset in a forward layer-by-layer way, we can easily extend HFSR and Centered-ELM to MLFN. Accordingly, as the unified framework for HFSR and Centered-ELM, the fast learning machine (FLM) is proposed for both SLFN and MLFN with a single or multiple outputs. FLM keeps the same virtues of ELM only for SLFN, i.e., only the parameters in the last hidden layer require being adjusted, all the parameters in other hidden layers can be randomly assigned, and FLM is much faster than BP in training the sample sets. The experimental results clearly indicate the power of FLM.

The contributions of this paper exist in two aspects: (1) Through FLM, we can extend ELM to MLFN with keeping the same virtues of ELM only for SLFN; (2) FLM indeed gives a new forward encoding learning way rather than a backward gradient-descent learning way in the widely used learning algorithm BP. It views the behavior of MLFN between the last hidden layer and the input layer as the successive encoding procedure for the input data in a difficult-to-understand way. To large extent, this new understanding can also help us answer why MLFN behaves like a black box.

The remainder of this paper is organized as follows. In Section 2, we briefly review ELM for SLFN. In Section 3, we first propose the hidden-feature-space ridge regression HFSR and Centered-ELM, and then build the link between ELM and them for SLFN. Finally, we give the fast learning machine FLM as the unified framework of HFSR and Centered-ELM for SLFN and MLFN with a single or multiple outputs. In Section 4, we report the obtained experimental results about Centered-ELM for SLFN and FLM for MLFN on artificial or benchmarking datasets. Section 5 concludes the paper.

2. Elm for SLFN

In this section, we give a brief review of the extreme learning machine for a single hidden layer feedforward neural network. For easy interpretation and derivation hereafter and without loss of generality, we first consider a single hidden layer feedforward neural network (SLFN for brevity) with a single output here. Given$N$arbitrary distinct samples$(\mathbf{x}_j, t_j)$,$\mathbf{x}_j = \left[x_{j1}, x_{j2}, \ldots, x_{jn}\right]^T \in \mathbf{R}^n$, $t_j \in \mathbf{R}, j = 1, 2, \ldots\ldots, N$, SLFN with $\tilde{N}$hidden nodes and the activation function $g(\mathbf{x})$and a single output can be mathematically modeled as

$$\sum_{i=1\tilde{N}} \beta_i g_i(\mathbf{x_j}) = \sum_{i=1\tilde{N}} \beta_i g(\mathbf{w}_i^T \mathbf{x}_j + b_i) = O_j, j = 1, 2, \ldots, N \tag{1}$$

where $\mathbf{w}_i = [w_{i1}, w_{i2}, \ldots, w_{in}]^T$is the weight vector connecting the ith hidden node and the input nodes, $\boldsymbol{\beta} = [\beta_1, \beta_2, \ldots, \beta_{\tilde{N}}]^T$ is the weight vector connecting all the hidden nodes and the output node, $b_i$is the threshold of the ith hidden node, and $\mathbf{w}_i^T \mathbf{x}_j$denotes the inner product of $\mathbf{w}_i$ and $\mathbf{x}_j$.

We desire that the above SLFN with a single output can approximate these N samples with zero error, that is to say,

$$\sum_{j=1}^{N} \|O_j - t_j\|^2 = 0, \tag{2}$$

i.e. $\sum_{i=1\tilde{N}} \beta_i g\left(\mathbf{w}_i^T \mathbf{x}_j + b_i\right) = t_j, j = 1, 2, \ldots, N$

The above N equations can be compactly written as the following linear system

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

where

$$\mathbf{H}(\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_{\tilde{N}}, b_1, b_2, \ldots, b_{\tilde{N}}, \mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N) = \begin{bmatrix} g(\mathbf{w}_1^T \mathbf{x}_1 + b_1) & \cdots & g\left(\mathbf{w}_{\tilde{N}}^T \mathbf{x}_1 + b_{\tilde{N}}\right) \\ \vdots & \cdots & \vdots \\ g(\mathbf{w}_1^T \mathbf{x}_N + b_1) & \cdots & g\left(\mathbf{w}_{\tilde{N}}^T \mathbf{x}_N + b_{\tilde{N}}\right) \end{bmatrix}_{N \times \tilde{N}}$$

$$\boldsymbol{\beta} = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \cdots \\ \beta_{\tilde{N}} \end{bmatrix}_{\tilde{N} \times 1} \quad \text{and}\mathbf{T} = \begin{bmatrix} t_1 \\ t_2 \\ \cdots \\ t_N \end{bmatrix}_{N \times 1}. \tag{4}$$

Here $\mathbf{H}$ is called the hidden layer output matrix of SLFN, whose ith column is the ith hidden node output with respect to the inputs $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$.

Huang et al. [7] proposed the famous extreme learning machine based on the following two theorems.

Theorem 1. *Given the above single-output SLFN with$N$hidden nodes and the activation function$g : \mathbf{R} \to \mathbf{R}$which is infinitely differentiable in any interval, for arbitrary$N$distinct samples$(\mathbf{x}_j, \mathbf{t}_j), \mathbf{x}_j \in \mathbf{R}^n, t_j \in \mathbf{R}$, for any randomly selected weight vector and bias $\mathbf{w}_i \in \mathbf{R}^n, b_i \in \mathbf{R}$, according to any continuous probability distribution, then with probability one, the hidden layer output matrix*$\mathbf{H}$ *of the SLFN is invertible, and* $\|\mathbf{H}\boldsymbol{\beta} - \mathbf{T}\|_2^2 = 0$.

Theorem 2. *Given* ***any*** *small positive value $\varepsilon > 0$, there must exist the above single-output SLFN with $\tilde{N}$hidden nodes and $\tilde{N} \le N$and activation function$g : \mathbf{R} \to \mathbf{R}$which is infinitely differentiable in any interval, such that for arbitrary$N$distinct samples$(\mathbf{x}_j, t_j), \mathbf{x}_j \in \mathbf{R}^n, t_j \in \mathbf{R}$, for any randomly assigned weight vector and bias $\mathbf{w}_i \in \mathbf{R}^n, b_i \in \mathbf{R}$, according to any continuous probability distribution, then with probability one,* $\|\mathbf{H}_{N \times \tilde{N}} \boldsymbol{\beta}_{\tilde{N} \times 1} - \mathbf{T}_{N \times 1}\|_2^2 < \varepsilon$.

According to the above two theorems, for the linear system in Eq. (3), we can have its unique solution, i.e, the smallest norm least squares solution $\hat{\boldsymbol{\beta}}$as follows

$$\hat{\boldsymbol{\beta}} = \mathbf{H}^{\dagger} \mathbf{T} \tag{5}$$

where $\mathbf{H}^{\dagger}$is the Moore-penrose generalized inverse of the matrix $\mathbf{H}$. Accordingly, Huang et al. proposed the following extreme learning machine ELM [7].

```
Extreme learning machine ELM
Given the sample set$D = \{(\mathbf{x}_j, t_j) | \mathbf{x}_j \in \mathbf{R}^n, t_j \in \mathbf{R}, j = 1, 2, \ldots\ldots, N\}$,
  the infinitely differential activation function$g(\mathbf{x})$and the
  hidden node number $\tilde{N}$ of SLFN with a single output.
Step1: Randomly assign the weight vector and the bias
  $\mathbf{w}_i, b_i, i = 1, 2, \ldots \tilde{N}$
Step2: Compute the hidden layer output matrix$\mathbf{H}$
Step3: Compute the output weight vector of SLFN, i.e,$\hat{\boldsymbol{\beta}} = \mathbf{H}^{\dagger}\mathbf{T}$,
  where$\mathbf{T} = [t_1, t_2, \ldots t_N]^T$.
```

In fact, let $\boldsymbol{\beta} = \begin{bmatrix} \boldsymbol{\beta}_1^T \\ \boldsymbol{\beta}_2^T \\ \vdots \\ \boldsymbol{\beta}_m^T \end{bmatrix}$, $\boldsymbol{\beta}_i = [\beta_{i1}, \beta_{i2}, \ldots, \beta_{i\tilde{N}}]^T$, $\mathbf{T} = \begin{bmatrix} \mathbf{t}_1^T \\ \mathbf{t}_2^T \\ \vdots \\ \mathbf{t}_m^T \end{bmatrix}$, $\mathbf{t}_i = [t_{i1}, t_{i2}, \ldots, t_{iN}]^T$, $i = 1, 2, \ldots, m$ according to Huang's theory [7], the above ELM still holds for *SLFN* with m multiple outputs.