Information Sciences 328 (2016) 546–557

Contents lists available at ScienceDirect

Information Sciences

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

An iterative learning algorithm for feedforward neural networks with random weights

Feilong Cao[a], Dianhui Wang[b,c,*], Houying Zhu[d], Yuguang Wang[d]

[a] *Department of Applied Mathematics, China Jiliang University, Hangzhou 310018, Zhejiang, China*
[b] *Department of Computer Science and Information Technology, La Trobe University, Melbourne, VIC 3086, Australia*
[c] *State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang 110819, Liaoning, China*
[d] *School of Mathematics and Statistics, The University of New South Wales, Sydney 2052, Australia*

ARTICLE INFO

Article history:
Received 15 June 2015
Revised 14 August 2015
Accepted 2 September 2015
Available online 11 September 2015

Keywords:
Neural networks with random weights
Learning algorithm
Stability
Convergence

ABSTRACT

Feedforward neural networks with random weights (FNNRWs), as random basis function approximators, have received considerable attention due to their potential applications in dealing with large scale datasets. Special characteristics of such a learner model come from weights specification, that is, the input weights and biases are randomly assigned and the output weights can be analytically evaluated by a Moore–Penrose generalized inverse of the hidden output matrix. When the size of data samples becomes very large, such a learning scheme is infeasible for problem solving. This paper aims to develop an iterative solution for training FNNRWs with large scale datasets, where a regularization model is employed to potentially produce a learner model with improved generalization capability. Theoretical results on the convergence and stability of the proposed learning algorithm are established. Experiments on some UCI benchmark datasets and a face recognition dataset are carried out, and the results and comparisons indicate the applicability and effectiveness of our proposed learning algorithm for dealing with large scale datasets.

1. Introduction

Feedforward neural networks with random weights (FNNRWs) were proposed by Schmidt and his co-workers in [21], which can be mathematically described by

$$G_N(\mathbf{x}) = \sum_{i=1}^{L} \beta_i g(\langle \omega_i, \mathbf{x} \rangle + b_i), \tag{1.1}$$

where $\mathbf{x} = [x_1, x_2, \ldots, x_d]^\top \in \mathbf{R}^d$, g is an activation function, b_i is a bias, $\omega_i = [\omega_{i1}, \omega_{i2}, \ldots, \omega_{id}] \in \mathbf{R}^d$ and $\beta_i \in \mathbf{R}$ are the input and output weights, respectively; $\langle \omega_i, x \rangle = \sum_{j=1}^{d} \omega_{ij} x_j$ denotes the Euclidean inner product.

The input weights and biases are assigned randomly with uniform distribution in $[-1, 1]$, and the output weights can be determined analytically by using the well-known least mean squares method [21]. Unfortunately, such a randomized learner model

[*] Corresponding author at: Department of Computer Science and Information Technology, La Trobe University, Melbourne, VIC 3086, Australia. Tel.: +61 3 9479 3034; fax: +61 3 9479 3060.

E-mail addresses: icteam@163.com (F. Cao), dh.wang@latrobe.edu.au (D. Wang), zhuhouying87@gmail.com (H. Zhu), wangyg85@gmail.com (Y. G. Wang).

http://dx.doi.org/10.1016/j.ins.2015.09.002
0020-0255/© 2015 Elsevier Inc. All rights reserved.

in [21] was not proposed as a working algorithm, but simply as a tool to investigate some characteristics of feedforward neural networks. From approximation theory, it is obvious that such a way to randomly assign the input weights and biases cannot guarantee the universal approximation capability (in the sense of probability one) of the resulting random learner models. This disability can be verified by many counter examples. Thus, statements on its approximation capability and good generalization in the literature are all misleading and lack scientific justification. A similar randomized learner model, termed as random vector functional-link nets (RVFLs), and associated algorithms were proposed by Pao and Takefuji in [15], where a direct link from the input layer to the output layer is added. In [13], a theoretical justification of the universal approximation capability of RVFLs (without direct link case) was established, where the scope of randomly assigned input weights and biases are data dependent and specified in a constructive manner. Therefore, a careful estimation of the parameters characterizing the scope in algorithm implementation should be done for each dataset in practice.

Due to some good learning characteristics [16], this type of randomized training scheme for feedforward neural networks has been widely applied in data modeling [9]. Obviously, the most appealing property of randomized techniques for feedforward neural networks lies in the possibility of handling large scale datasets with real-time requirement. Recently, some advanced randomized learning algorithms have been developed in [1,5–7,20]. Also, some theoretical results on the approximation power of randomized learners were investigated in [11,18,19]. It has been recognized that the way the output weights are computed can result in memory problems for desktop computers as the size of data samples and the number of nodes at the hidden layer of neural networks become very large. Motivated by this crucial drawback and potential applications of such a machine learning technique for big data, we aim to overcome the difficulty of computing the generalized inverse in least mean squares method, and present a feasible solution for the large scale datasets. This work is built on a framework for resolving the linear inverse problems in [8]. Note that our objective in this study is not to investigate properties of the randomized learner model from statistical learning theory, but an iterative scheme for building an instance learner model. It has been recognized that some machine learning techniques, such as ensemble learning [1] and sequential learning [14,17], are related to large scale data modeling problems, however, our focus in this work is neither on sampling-based ensemble techniques nor streaming data modeling.

The remainder of this paper is organized as follows. Section 2 provides some supportive results for algorithm development. Section 3 proposes an iterative learning algorithm with analyses of its convergence and stability. The performance evaluation is presented in Section 4, where the experimental results on some benchmark datasets and a face recognition dataset are reported. Section 5 concludes this paper with some remarks. Finally, mathematical proofs of the theoretical results are given in the Appendix.

2. Supportive results

Given a set of training samples $\mathscr{T} = \{(\mathbf{x}_i, t_i) : i = 1, 2, \dots, N\}$, let ω_i and b_i be chosen randomly from the uniform distribution and fixed in advance. Then, the output weights can be obtained by solving the following linear equation system:

$$\mathbf{H}\beta = \mathbf{T}, \tag{2.1}$$

where $\beta = [\beta_1, \beta_2, \dots, \beta_L]^\intercal$, $\mathbf{T} = [t_1, t_2, \dots, t_N]^\intercal$,

$$\mathbf{H} = \begin{bmatrix} g(\langle\omega_1, \mathbf{x}_1\rangle + b_1) & \dots & g(\langle\omega_L, \mathbf{x}_1\rangle + b_L) \\ \vdots & \dots & \vdots \\ g(\langle\omega_1, \mathbf{x}_N\rangle + b_1) & \dots & g(\langle\omega_L, \mathbf{x}_N\rangle + b_L) \end{bmatrix}. \tag{2.2}$$

A least mean squares solution of (2.1) can be expressed by $\beta = \mathbf{H}^\dagger\mathbf{T}$, where $\mathbf{H}^\dagger$ is the Moore–Penrose generalized inverse of $\mathbf{H}$. However, the least squares problem is usually ill-posed and one can employ a regularization model to find a solution, that is,

$$L_\mu(\beta) = \|\mathbf{H}\beta - \mathbf{T}\|_2^2 + \mu J(\beta), \tag{2.3}$$

where μ is a small positive number called the regularizing factor, and J is a differentiable, strictly convex and coercive function with respect to β. In this case, the minimization of (2.3) has a unique solution. Particularly, if we take $J(\beta) = \|\beta\|_2^2$, then the regularization model becomes

$$\Phi_\mu(\beta) = \|\mathbf{H}\beta - \mathbf{T}\|_2^2 + \mu\|\beta\|_2^2. \tag{2.4}$$

If μ is given such that $(\mathbf{H}^\intercal\mathbf{H} + \mu\mathbf{I})$ is invertible, then the minimizer of (2.4) can be written as

$$\beta^\star = (\mathbf{H}^\intercal\mathbf{H} + \mu\mathbf{I})^{-1}\mathbf{H}^\intercal\mathbf{T}, \tag{2.5}$$

where $\mathbf{I}$ denotes the identity matrix.

The regularizing factor is fixed beforehand, and should be taken properly so that $\mathbf{H}^\intercal\mathbf{H} + \mu\mathbf{I}$ continues to be invertible for all random assignments. Furthermore, the computation of the inverse matrix in (2.5) will become very difficult and impractical for large scale datasets. In such a background, it is necessary to develop more effective schemes for problem solving. Based on the iterative method proposed in [8] which is further explored in [22], we first establish the following supportive result (its proof is detailed in the Appendix).