Applied Energy 218 (2018) 22–35

Contents lists available at ScienceDirect

Applied Energy

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

A robust extreme learning machine for modeling a small-scale turbojet engine

Yong-Ping Zhao*, Qian-Kun Hu, Jian-Guo Xu, Bing Li, Gong Huang, Ying-Ting Pan

College of Energy and Power Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China

HIGHLIGHTS

- A robust regularized extreme learning machine is proposed.
- The robustness of the proposed algorithm is proved theoretically.
- The proposed algorithm is applied to model a small-scale turbojet engine.
- The precise power control for unmanned aerial vehicles is realized.

ARTICLE INFO

Keywords:
Extreme learning machine
Small-scale turbojet engine
System modeling
Machine learning

ABSTRACT

In this paper, a robust extreme learning machine is proposed. In comparison with the original extreme learning machine and the regularized extreme learning machine, this robust algorithm minimizes both the mean and variance of modeling errors in the objective function to overcome the bias-variance dilemma. As a result, its generalization performance and robustness are enhanced, and these merits are further proved theoretically. In addition, this proposed algorithm can keep the same computational efficiency as the original extreme learning machine and the regularized extreme learning machine. Then, several benchmark data sets are used to test the effectiveness and soundness of the proposed algorithm. Finally, it is employed to model a real small-scale turbojet engine. This engine is fit well. Especially, on the idle phase, where the signal-to-noise ratio is low and it is very hard to model, the proposed algorithm performs well and its robustness is sufficiently showcased. All in all, the proposed algorithm provides a candidate technique for modeling real systems.

1. Introduction

Single hidden layer feedforward networks (SLFNs) have been extensively and intensively investigated in the last few decades [1,2], and a large number of algorithms about them have been presented. In SLFNs, one vital problem is how to determine the weights in the network. To address this problem, the gradient descent methods are commonly used to optimize the network weights, such as error back-propagation algorithm [3], Levenberg-Marquardt algorithm [4,5], and neuron by neuron algorithm [6]. However, those algorithms usually suffer from the risk of slow convergence or/and local minimum. Recently, a promising and high-efficiency training tool for SLFN was proposed, i.e., extreme learning machine (ELM) [7,8]. ELM has two remarkable features [9]: (1) Its input weights and the biases of the hidden nodes are randomly generated, and the output weights between the hidden layer and output layer are analytically calculated by solving a linear system. Such improvement greatly alleviates the computational burden of weight tuning caused by the widely used gradient descent methods and thus guarantees the fast learning speed of ELM. (2) ELM aims to minimize both training errors and the norm of output weights, which leads to ELM to generalize well on the unseen testing data. Generally speaking, ELM spends much less time on training and tends to achieve much better generalization performance. Further, ELM theories and philosophy show that some earlier learning theories such as ridge regression theory [10], Bartlett's neural network generalization performance theory [11] and SVM's maximal margin [12–14] are actually consistent in machine learning [15,16]. Due to its powerful and excellent capabilities, ELM is widely adopted in the classification and regression problems.

In the implementation of ELM, it is found that the generalization performance of ELM is not sensitive to the number of the hidden nodes (#HN) and good performance can be reached as long as #HN is large enough [15]. However, too many random hidden nodes may deteriorate the condition number of the hidden layer output matrix, thus

* Corresponding author.
E-mail address: y.p.zhao@nuaa.edu.cn (Y.-P. Zhao).

https://doi.org/10.1016/j.apenergy.2018.02.175
Received 22 November 2017; Received in revised form 13 February 2018; Accepted 28 February 2018
0306-2619/

impairing the generalization performance of ELM. In an effort to improve its performance, the Tikhonov's method [17] was used to regularize the original ELM, that is, a regularization term was added to the objective function to control the model complexity, thus yielding the regularized ELM (RELM) [18]. From the viewpoint of numerical analysis, RELM can boost the performance with the trick of improving the condition number of the hidden layer output matrix [19]. The original ELM is a special case of RELM when the regularization parameter equals infinity.

In both ELM and RELM, the least squares loss function is adopted. As thus, they are sensitive to outliers and non-Gaussian noise, which may result in an unsatisfactory modeling performance in an environment with noise [20]. And, in many real-life applications, practical data usually contain the presence of different distributions of noise because of sampling errors, modeling errors, measurement errors, and operation errors. Hence, their performance may not be so satisfactory when ELM and RELM are used to model these practical systems. To reduce the influence of outliers and noise, some actions of boosting the robustness of ELM or RELM can be taken. Loosely speaking, two strategies can be taken: (1) The weight approach: each training point is assigned with a different weight according to some principle. Generally, a smaller error training point is endowed with a large weight, and a larger error training one with a small weight. For example, in [18,21], this principle is followed, and the weight is determined with the modeling error. In this family of algorithms, the mathematical model usually needs to be solved many times. Thus, they need more training time. (2) The alternative loss function: the least squares loss function is replaced with a robust one. For instance, the ℓ_2-norm loss function is replaced with the ℓ_0-norm or ℓ_1-norm loss function [22]. However, this will lead to the result that the optimization on the mathematical model becomes difficult, that is, there are no sophisticated tools to solve the optimization problem. All in all, although the two strategies aforementioned can mitigate the influence of outliers and noise in some way, they will bring about other disadvantages. In this paper, we propose a novel robust machine learning algorithm, which not only overcomes those drawbacks but also enhances the generalization performance of RELM and boosts its robustness. Our main contributions include:

(1) A novel robust RELM (RRELM) is proposed, which is an extension of RELM. In other words, RELM is a special case of RRELM. As for this point, it can be proved. In RRELM, its objective function minimizes both mean and variance of modeling errors. In doing so, on one hand, the smaller error training points are endowed with large weights, and the larger error training ones with small weights. Thus, the robustness is boosted and the generalization performance is improved. On the other hand, the property of the ℓ_2-norm loss function is unchanged, which signifies the same high-efficiency computation as that in RELM.
(2) Several benchmark data sets are utilized to test the effectiveness and soundness of the proposed RRELM. When RRELM is employed to model a small-scale turbojet engine, we will find that RRELM can obtain better generalization performance than RELM and ELM. Especially in those scenarios of small thrust (the idle), where the signal-to-noise ratio (SNR) is small and it is more difficult to model this engine system, it is found that RRELM obviously dominates RELM and ELM with respect to the generalization performance and robustness.

The rest of this paper is organized as follows. Sections 2 and 3 briefly introduce ELM, and RELM, respectively. In Section 4, the proposed RRELM is elaborated on. Firstly, its motivation is introduced, and then its dual and primal solutions are given. Finally, its performance is analyzed. In Section 5, several benchmark data sets are used to test the effectiveness of the proposed RRELM, and some viewpoints are experimentally confirmed. In Section 6, ELM, RELM, and RRELM are employed to model a small-scale turbojet engine, and their modeling results are discussed. In the last section, conclusions follow.

Table 1
Specifications of benchmark data sets.

Data sets	#Training	#Testing	#Inputs	#Outputs
Boston housing	304	202	13	1
Energy efficiency	461	307	8	2
Concrete	603	402	8	1
Sml2010	2482	1655	16	2
Parkinsons	3525	2350	18	2

Notes: #Training represents the number of training points, #Testing represents the number of testing points, #Inputs represents the number of input attributes, #Outputs represents the number of output targets.

2. ELM

An unknown nonlinear system can be described as

$$\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x}) \tag{1}$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ denote the input and output of the system, respectively, and $\boldsymbol{f}(\cdot)$ represents this unknown nonlinear system. This unknown system can be approximated with the ELM model:

$$\hat{\boldsymbol{f}}(\boldsymbol{x}) = \sum_{i=1}^{\#\mathrm{HN}} \theta_i h(\boldsymbol{a}_i,b_i,\boldsymbol{x}),\ \boldsymbol{x},\boldsymbol{a}_i \in \mathscr{R}^n,\ b_i \in \mathscr{R} \tag{2}$$

where $h(\cdot)$ is the activation function, $\boldsymbol{a}_i$ and b_i are the randomly generated parameters of hidden nodes, θ_i is the weight vector connecting the ith hidden node to the output nodes. Considering a set of training data $\mathscr{D} = \{(\boldsymbol{x}_i,\boldsymbol{y}_i)\}_{i=1}^N$, where $\boldsymbol{x}_i \in \mathscr{R}^{n\times 1}$ is the ith input vector, $\boldsymbol{y}_i \in \mathscr{R}^{1\times m}$ is the target vector, ELM lets the network outputs equal the targets with zero errors, so the following equation is obtained [7,8]:

$$\boldsymbol{H\Theta} = \boldsymbol{Y} \tag{3}$$

where

$$\boldsymbol{H} = \begin{bmatrix} h(\boldsymbol{a}_1,b_1,\boldsymbol{x}_1) & \cdots & h(\boldsymbol{a}_{\#\mathrm{HN}},b_{\#\mathrm{HN}},\boldsymbol{x}_1) \\ \vdots & \ddots & \vdots \\ h(\boldsymbol{a}_1,b_1,\boldsymbol{x}_N) & \cdots & h(\boldsymbol{a}_{\#\mathrm{HN}},b_{\#\mathrm{HN}},\boldsymbol{x}_N) \end{bmatrix} \tag{4}$$

is the so-called hidden nodes output matrix, $\boldsymbol{\Theta} = [\theta_1^\top,\ldots,\theta_{\#\mathrm{HN}}^\top]^\top$, $\boldsymbol{Y} = [\boldsymbol{y}_1^\top,\cdots\boldsymbol{y}_N^\top]^\top$. Getting the solution of (3) in the least square sense is equivalent to solving the following optimal problem

$$\min_{\boldsymbol{\Theta}} \left\{\mathscr{J}_{\mathrm{ELM}} = \frac{1}{2}\|\boldsymbol{E}\|_F^2\right\} \quad \text{s.t.} \quad \boldsymbol{Y} = \boldsymbol{H\Theta} + \boldsymbol{E} \tag{5}$$

where $\|\cdot\|_F$ represents the Frobenius norm, $\boldsymbol{E} = [\boldsymbol{e}_1^\top,\ldots,\boldsymbol{e}_N^\top]^\top$ is the modeling error matrix. The minimal norm least square solution of (5) is

$$\boldsymbol{\Theta}^{\mathrm{ELM}} = \boldsymbol{H}^\dagger \boldsymbol{Y} \tag{6}$$

where $\boldsymbol{H}^\dagger$ is the Moore-Penrose generalized inverse of matrix $\boldsymbol{H}$. When $\boldsymbol{H}^\top\boldsymbol{H}$ is nonsingular, $\boldsymbol{H}^\dagger = (\boldsymbol{H}^\top\boldsymbol{H})^{-1}\boldsymbol{H}^\top$, or when $\boldsymbol{H}\boldsymbol{H}^\top$ is nonsingular, $\boldsymbol{H}^\dagger = \boldsymbol{H}^\top(\boldsymbol{H}\boldsymbol{H}^\top)^{-1}$. The prediction for a new input $\boldsymbol{x}$ with the ELM model is

$$\hat{\boldsymbol{f}}_{\mathrm{ELM}}(\boldsymbol{x};\mathscr{D}) = \sum_{i=1}^{\#\mathrm{HN}} \theta_i^{\mathrm{ELM}} h(\boldsymbol{a}_i,b_i,\boldsymbol{x}) \tag{7}$$

3. RELM

The mathematical model of RELM [18] is described as:

$$\min_{\boldsymbol{\Theta}} \left\{\mathscr{J}_{\mathrm{RELM}} = \frac{1}{2}\|\boldsymbol{\Theta}\|_F^2 + \frac{C}{2}\|\boldsymbol{E}\|_F^2\right\} \quad \text{s.t.} \quad \boldsymbol{Y} = \boldsymbol{H\Theta} + \boldsymbol{E} \tag{8}$$

where $C \in \mathscr{R}^+$ is the regularization parameter. The optimal solution to