



The memory degradation based online sequential extreme learning machine

Quan-Yi Zou^a, Xiao-Jun Wang^{a,*}, Chang-Jun Zhou^{a,*}, Qiang Zhang^{a,b,*}

^a Key Laboratory of Advanced Design and Intelligent Computing, Ministry of Education, Dalian University, Dalian 116622, China

^b School of Computer Science & Technology, Dalian University of Technology, Dalian 116622, China

ARTICLE INFO

Article history:

Received 17 May 2017

Revised 22 September 2017

Accepted 11 November 2017

Available online 24 November 2017

Communicated by Yudong Zhang

Keywords:

Online learning

Extreme learning machine

Memory factor

Similarity

ABSTRACT

In online learning, the contribution of old samples to a model decreases as time passes, and old samples gradually become invalid. Although the Online Sequential Extreme Learning Machine (OS-ELM) can avoid the repetitive training of old samples, invalid samples are still used, which goes against improving the accuracy of an OS-ELM model. The Online Sequence Extreme Learning Machine with Forgetting Mechanism (FOS-ELM) timely discards invalid samples, but it does not consider the differences among valid samples and then has the limitation on boosting the accuracy and generalization. To solve this issue, the Memory Degradation Based OS-ELM (MDOS-ELM) is proposed in this paper. The MDOS-ELM adjusts the weights of the old and new samples in real time by a self-adaptive memory factor, and simultaneously discards invalid samples. The self-adaptive memory factor is determined by two elements. One is the similarity between the new and old samples, and the other is the prediction errors of the current training samples on the previous model. The performance of the proposed MDOS-ELM is validated on both regression and classification datasets which include an artificial dataset and twenty-two real-world dataset. The results demonstrate that the MDOS-ELM model outperforms the OS-ELM and the FOS-ELM models on the accuracy and generalization.

© 2017 Elsevier B.V. All rights reserved.

1. Introduction

The Single-hidden Layer Feed-forward Neural-network (SLFN) with a nonlinear activation function is a strong learning algorithm, which can approximate any nonlinear continuous function [1]. Since it was proposed, the SLFN has been widely concerned in both theory and practice [2–5]. Unfortunately, it is required to iteratively adjust the parameters for optimizing the SLFN model, which makes the SLFN lack of the fast learning ability.

To address this problem, based on the Moore-Penrose generalized inverse theory, the Extreme Learning Machine (ELM) is proposed by Huang et al. [6–8]. Compared with the traditional SLFN algorithm, the ELM only needs to set reasonable numbers of the hidden layer nodes and randomly assign parameters of the hidden layer and then the output weights of the ELM are obtained by the least squares method. Only one step is carried out in the whole

learning process of the ELM. Namely, the ELM does not iteratively update all the parameters of the SLFN, which makes it have the advantage of fast learning.

However, the ELM has the following disadvantages. First, the ELM only considers the empirical risk minimization and easily causes the over-fitting issue. To solve this issue, the regularized ELM (RELM) [9] was proposed by Deng et al., which was based on both the structural risk and the empirical risk minimizations, but the RELM has the shortcoming which cannot cope with non-Gaussian noises [10]. Second, the ELM cannot handle the curse of dimensionality well. To handle this disadvantage, kernel functions were introduced into the ELM, which could map samples into higher dimensional feature space [11–13], but these algorithms have the shortcoming which is hard to select an optimal kernel function for a specific application. Third, it is hard to set a suitable number of nodes in hidden layer. If the number is too large, the computation of the ELM model is time-consuming, and inversely, and the ELM model is not accurate enough. To address this problem, literature [14–18] proposed strategy that gradually added or reduced the number of nodes in the hidden layer to optimize the structure of an ELM model, but the operation pruned or increased is time consuming. Fourthly, the ELM has the limitation in drawing deeper relationship among the data from too complicated

* Corresponding authors.

E-mail addresses: [wxjessicaxj0903@126.com](mailto:wqxjessicaxj0903@126.com) (X.-J. Wang), zhouchangjun@dlu.edu.cn (C.-J. Zhou), zhangq26@126.com (Q. Zhang).

practical problems. To deal with this disadvantage, the ELM was usually modified by combining with the deep learning algorithms [19], and then the multilayer ELM [20] and the deep ELM [21] were proposed. Additionally, many revisions of the ELM were presented and applied in various fields, such as the wind forecast [22,23], the image processing [24,25], the face recognition [26], the solar radiation prediction [27], the medical diagnosis [28,29], pathological brain detection [30–32], and the time series predictions [33].

The traditional ELM and most of its revisions are more suitable for offline learning. Prior to offline learning, a certain number of samples are accumulated, which are used to train a model and then the model is applied to obtain the prediction of a new sample. However, in many real applications, such as the price forecasting [34], the weather forecasting [35], and the robotic online controllers [36], the samples are updated in real time. If the ELM is directly used for the online learning, when the chunk of new samples is acquired, the ELM takes the old and new samples together as a new sample chunk. Then the new sample chunk is re-learned by the ELM. As the accumulation of samples and the repeated learning of the old samples, the learning speed of the ELM will gradually decline, and even worse, the ELM may be unable to make the online learning continuous.

In order to effectively avoid the repeated learning of old samples, the Online Sequential ELM (OS-ELM) is proposed by Liang et al. [37], which gradually adds the new samples for learning chunk by chunk. By using the output weights of the previous OS-ELM model and the new samples, the OS-ELM updates the output weights and improves the speed of online learning. On the basis of the OS-ELM, the fuzzy OS-ELM based on fuzzy inference system is proposed by Rong et al. [38], and the OS-ELM with kernels is proposed by Scardapane et al. [39]. The OS-ELM and its improved algorithms randomly assign initial parameters of the hidden layer, which brings the instability of the models. To deal with the issue, the ensemble OS-ELMs is proposed by Lan et al. [40], which is a combination of several independent OS-ELM sub-models. Although the EOS-ELMs algorithm has strong stability, it is not prominent to improve the accuracy and the generalization. The reason is that it ignores the timeliness of samples.

The so-called timeliness of samples is that as time elapses old samples will be gradually invalid in online learning. Applying the invalid samples in online learning usually leads to the declining accuracy and generalization of the EOS-ELMs model. Aiming at this issue, the OS-ELM with Forgetting Mechanism (FOS-ELM) is proposed by Zhao et al. [41], in which the invalid samples are timely discarded by the forgetting mechanism. However, the differences among valid samples are ignored by the FOS-ELM. In an online learning process, the importance of the valid samples to a model is related with their age [42]. New samples are usually more important than old ones, which should be given greater weights, vice versa.

In order to fully utilize the differences among valid samples and to further improve the accuracy and generalization of the FOS-ELM model, the Memory Degradation based OS-ELM (MDOS-ELM) is proposed in this paper. On the one hand, the self-adaptive memory factor is introduced in the MDOS-ELM, which can dynamically adjust the weights of the old and new valid samples in the online learning processes. The self-adaptive memory factor is determined by two elements. One is the similarity between the input variables of the new and old samples, and the other is the predicting errors of the current training samples on the previous model. The former is from the point of view of the input variables, and the latter is from the point of view of output variable. There are many methods for evaluating the similarity of samples, such as Euclidean distance, Cosine, Pearson-Correlation coefficient [43] and Robust-Huber Similarity Measure (RHSM) [44]. The first three are common methods

and they are too sensitive to outliers. With Huber function [45], the last one is more robustness to outliers.

On the other hand, the mechanism of timely discarding invalid samples in the FOS-ELM is referred to by the MDOS-ELM. The performance of the MDOS-ELM is verified on nine regression datasets and fourteen classification datasets. The accuracy and generalization of the MDOS-ELM is compared with those of the OS-ELM and the FOS-ELM.

The remainder of this paper is organized as follows. Section 2 introduces the ELM, the OS-ELM and the FOS-ELM algorithms. In Section 3, the MDOS-ELM algorithm is proposed and the self-adaptive memory factor is discussed. In Section 4, nine regression datasets and fourteen classification datasets are adopted in the experiments to verify the performance of the proposed MDOS-ELM. The performance of the MDOS-ELM is compared with that of the OS-ELM and the FOS-ELM. In Section 5, the conclusion of this paper is summarized.

2. Background

2.1. The ELM algorithm

The learning set is $S = \{(\mathbf{x}_i, t_i) | \mathbf{x}_i \in \mathbb{R}^n, t_i \in \mathbb{R}, i = 1, 2, \dots, N\}$, where t is the output variable and $\mathbf{x} = [x_1, x_2, \dots, x_n]$ is the input vector. N is the number of samples and n is the dimension of the input variables. A unified SLFN model with L nodes in the hidden layer can be expressed as follows [5],

$$f_L(\mathbf{x}) = \sum_{l=1}^L \beta_l G(\mathbf{a}_l, b_l, \mathbf{x}), \mathbf{x} \in \mathbb{R}^n, \tag{1}$$

where β_l is the output weight of the l th node of the hidden layer. $G(\cdot)$ is the activation function of the hidden layer node, and it can be 'RBF', 'Sigmoid', 'Sine', or 'hradlim' etc. \mathbf{a} and b are parameters of $G(\cdot)$.

The output of the SLFN with L hidden layer nodes is expected to approximate the real output with zero errors, and then

$$t_i = \sum_{l=1}^L \beta_l G(\mathbf{a}_l, b_l, \mathbf{x}_i), i = 1, 2, \dots, N.$$

The above N equations can be written in the matrix form below

$$\mathbf{T} = \mathbf{H}\boldsymbol{\beta},$$

where $\mathbf{T} = [t_1, t_2, \dots, t_N]^T$ is the expected output matrix, and $\boldsymbol{\beta} = [\beta_1, \beta_2, \dots, \beta_L]^T$ is the output weight. \mathbf{H} is the output matrix of the hidden layer, and

$$\mathbf{H} = \begin{bmatrix} G(\mathbf{a}_1, b_1, \mathbf{x}_1) & \cdots & G(\mathbf{a}_L, b_L, \mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ G(\mathbf{a}_1, b_1, \mathbf{x}_N) & \cdots & G(\mathbf{a}_L, b_L, \mathbf{x}_N) \end{bmatrix}_{L \times N}.$$

$\hat{\boldsymbol{\beta}}$ denotes the estimated value of $\boldsymbol{\beta}$, which is obtained by the least-square method. The relationship between $\hat{\boldsymbol{\beta}}$ and \mathbf{T} is as follows,

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

where \mathbf{H}^\dagger is the Moore–Penrose generalized inverse of \mathbf{H} , and

$$\mathbf{H}^\dagger = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T.$$

2.2. The OS-ELM algorithm

The OS-ELM is an online version of the ELM. It can gradually obtain new sample chunk (with fixed or chunk sizes). The OS-ELM algorithm is shown in Algorithm 1.

متن کامل مقاله

دریافت فوری ←

ISIArticles

مرجع مقالات تخصصی ایران

- ✓ امکان دانلود نسخه تمام متن مقالات انگلیسی
- ✓ امکان دانلود نسخه ترجمه شده مقالات
- ✓ پذیرش سفارش ترجمه تخصصی
- ✓ امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
- ✓ امکان دانلود رایگان ۲ صفحه اول هر مقاله
- ✓ امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
- ✓ دانلود فوری مقاله پس از پرداخت آنلاین
- ✓ پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات