Neurocomputing 174 (2016) 72–84

Contents lists available at ScienceDirect

Neurocomputing

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

Online sequential reduced kernel extreme learning machine

Wan-Yu Deng [b,a], Yew-Soon Ong [a,*], Puay Siew Tan [c], Qing-Hua Zheng [d]

[a] School of Computer Engineering, Nanyang Technological University, Singapore 639798, Singapore
[b] School of Computer, Xi'an University of Posts & Telecommunications, 710021, China
[c] Singapore Institute of Manufacturing Technology, Singapore
[d] Department of Computer Science and Technology, Xi'an Jiaotong University, 710049, China

ARTICLE INFO

Article history:
Received 15 October 2014
Received in revised form
8 June 2015
Accepted 9 June 2015
Available online 10 August 2015

Keywords:
Extreme learning machine
Support vector machine
Online sequential learning
Big data

ABSTRACT

In this paper, we present an Online Sequential Reduced Kernel Extreme Learning Machine (OS-RKELM). In OS-RKELM, only a small part of the instances in the original training samples is employed for training the kernel neurons, while the output weights are attained analytically. Similar to the Online Sequential Extreme Learning Machine (OS-ELM), OS-RKELM learns data samples in a chunk-by-chunk or one-by-one mode and does not require an archival of the data sample once it has been learned. OS-RKELM also contains few control parameters, thus avoiding the need for cumbersome fine-tuning of the algorithm. OS-RKELM supports a widespread types of kernels as hidden neurons and is capable of addressing the singular problem that arises when the initial training samples are smaller than the neuron size. A comprehensive performance evaluation of the OS-RKELM against other state-of-the-art sequential learning algorithms, including OS-ELM, Large-scale Active Support Vector Machine (LASVM) and Budgeted Stochastic Gradient Descent Support Vector Machine (BSGD) using popular time series, regression and classification benchmarks have been conducted. Experimental results obtained indicate that the proposed OS-RKELM showcases improved prediction accuracy and efficiency over the OS-ELM, LASVM and BSGD in many cases.

1. Introduction

For the last few decades, single hidden layer feedforward neural networks (SLFNs) and kernel machines such as support vector machines (SVMs) have been the core research topics of interests in the computational intelligence and machine learning communities [1–7]. In many of the real world applications that employs SLFNs in their work, the training methodology employed has mostly been based on a batch learning mode. When a new instance is available, batch learning typically combines the new member with previously received instances to perform a complete re-training of the learning model. Such a process is often very time consuming, especially when the data is Big, and can be extremely inefficient if new data continues to come in, rapidly and indefinitely (see e.g. [8]). Today, as we move into the era of Big Data, the big capacity of the data that now arises, also far exceeds the computational memory capacity of a modern system [3,4]. Hence, some researchers are turning to online sequential learning as an alternative over batch learning, since the former does not require a re-training and can cope naturally with the constraints of computational resources even on Big data.

Over the years, sequential learning has gained increasing attention and popularity in neural networks research. The Stochastic Gradient descent back-propagation (SGBP) [9] is among the most commonly studied variant of the popular back-propagation neural networks. In SGBP neural networks, the model parameters are tuned iteratively based on the first-order information given the training samples. It however suffers from the effect of slow convergence. To elevate the deficiency in speed, second-order information was considered in SGBP [10,5].

Many effective kernel-based online sequential algorithms have also been proposed [11,12,9] to date. With linear kernels, many efficient online learning algorithms have been suggested [4,13–20]. For instance, Pegasos [20] introduced an improved SGD in which every gradient descent step is accompanied with a projection step to show that the training rate can be significantly improved on linear SVM. On the other hand, for non-linear kernel machines, Kivinen et al. studied the gradient-based non-linear kernel SVM in [21], where weights of the learning model are found using first order gradient information. Afterwards, to improve learning efficiency, Bordes et al. [22] introduced second-order information [10,5] into the Stochastic Gradient Descent (SGD) when training the nonlinear SVM. Incremental and Decremental Support Vector Machine (IDSVM) [11] is another category of online sequential kernel-based algorithm that maintains the optimal SVM solutions by incrementally updating the Karush–Kuhn–Tucker (KKT) conditions. As the approach focuses on providing a

* Corresponding author.
E-mail address: asysong@ntu.edu.sg (Y.-S. Ong).

http://dx.doi.org/10.1016/j.neucom.2015.06.087
0925-2312/© 2015 Elsevier B.V. All rights reserved.

guarantee on optimality, the high computational cost involved thus makes it less attractive or even infeasible for large-scale learning. LASVM [3], on the other hand, trades off optimality with scalability via sequential minimal optimization to update the model incrementally. However, a potential unlimited growth of support vectors can limit its application on large-scale problems. To address the issue, TVM [23] imposes an upper bound on the support vectors, leading to a constant update space and time complexity. Nevertheless, TVM was designed only for solving binary classification problems. Last but not least, the Budgeted Stochastic Gradient Descent based SVM (BSGD) [24] is an alternative online algorithm that is designed for working with both binary and multi-class classification problems, but not on regression tasks.

In spite of the extensive works on the topic, current sequential learning algorithms (kernel and non-kernel based) come with several drawbacks. Some of the key factors are summarized as follows:

(1) SGBP, IDSVM, BSGD and LASVM have many control parameters that need to be appropriately defined or fine-tuned, if robust accuracy is desirable.
(2) IDSVM, BSGD, LASVM and TVM cannot work for binary classification, multi-class classification and regression as a unified learning model.
(3) IDSVM, BSGD and LASVM were designed to work with one data sample each time when updating the model. Thus they do not cope with chunk-by-chunk data update elegantly.

In this paper, a novel online sequential learning method which is referred to here as the Online Sequential Reduced Kernel Extreme Learning Machine (OS-RKELM) is introduced. The proposed OS-RKELM has the following core contributions:

(1) OS-RKELM is a fast online kernel-based algorithm that can process data in a one-by-one or chunk-by-chunk mode, and can be used to perform binary classification, multi-class classification as well as regression tasks seamlessly.
(2) We prove that OS-RKELM possesses universal learning properties. Moreover, the solution of OS-RKELM is also accurate and every incremental computation can be rolled back conversely; this implies that OS-RKELM could not only perform incremental learning tasks but also suitable for decremental learning tasks where one (or more) sample(s) becomes irrelevant and need to be forgotten or unlearned from the model.
(3) OS-RKELM does not impose a constraint on the size of the initial training dataset, since it overcomes the singular problem of OS-ELM via the regularized parameter. At the same time, OS-RKELM is shown to attain better generalization performances [25] than OS-ELM and many other online kernel-based algorithms. It also has a faster learning rate and contains fewer control parameters.

The proposed OS-RKELM is compared with several other state-of-the-art sequential learning algorithms such as OS-ELM, LASVM and BSGD. Experimental results for regression, classification, and time-series problems indicate that OS-RKELM produces improved generalization performances at fast learning rate over the current state-of-the-arts. In the context of regression, the benchmarks considered include three real-world datasets from the UCI machine learning repository [26], namely (1) Abalone (age prediction of Abalone); (2) auto-MPG (fuel consumption prediction of cars); (3) California housing (median house prices estimation in the California area). With respect to classification tasks, study is conducted on six benchmarks also available in the UCI repository [26]. These include the datasets of (1) image segment, (2) satellite image, (3) DNA, (4) Waveform, (5) USPS and (6) adult. Last but not least, in the time-series domain, the Mackey–Glass chaotic benchmark dataset [27] is considered.

The paper is organized as follows: Section 2 gives a review of the classical Extreme Learning Machine (ELM) [28–30], Kernel Extreme Learning Machine (KELM) and Reduced Kernel Extreme Learning Machine (RKELM) [25].[1] Section 3 presents the OS-RKELM algorithm, while Section 4 demonstrates the relationships between OS-RKELM and OS-ELM. Section 5 highlights the differences to other kernel-based sequential learning algorithms including the LASVM and BSGD. Performance evaluations of OS-RKELM are then presented in Section 6 using a diversity of benchmark problems in the fields of regression, classification, and time-series prediction. A brief conclusion of the present study is then given in Section 7.

2. Review of ELM, KELM and RKELM

This section briefly reviews the batch learning RKELM described by Deng et al. in [25]. To provide the necessary background of RKELM, a brief description of the classical ELM and KELM are first presented in what follows.

2.1. ELM

The classical ELM [28–30] was proposed for the SLFNs, where the hidden layer can be any piecewise continuous computational functions including sigmoid, RBF, trigonometric, threshold, fuzzy inference systems, fully complex, high-order, ridge polynomial, and wavelet [31,32]. In ELM, only the number of the hidden neurons needs to be predefined, while the parameters of the hidden neurons (for example, the centers and impact factors of the RBF nodes or the biases and input weights of additive nodes) need not be fine-tuned, but are instead randomly assigned. The prediction of ELM is then given by

$$f_L(\mathbf{x}) = \sum_{i=1}^{L} \beta_i h_i(\mathbf{x}) = \mathbf{h}(\mathbf{x})\boldsymbol{\beta} \tag{1}$$

where $\boldsymbol{\beta} = [\beta_i, \ldots, \beta_L]^T$ is the vector of the output weights connecting the hidden layer and the output layer, and $\mathbf{h}(\mathbf{x}) = [h_1(\mathbf{x}), \ldots, h_L(\mathbf{x})]$ is the output of the hidden layer with respect to the sample $\mathbf{x}$. According to Bartlett theory [33], for neural networks with a small training error, the smaller the norms of the weights, the more likely is the generalization performance of the networks. In this spirit, ELM minimizes the training error in tandem with the norm of the output weights [34,35]:

$$\text{Minimize}: \|\mathbf{H}\boldsymbol{\beta} - \mathbf{T}\|^2 \quad \text{and} \quad \|\boldsymbol{\beta}\| \tag{2}$$

where $\mathbf{H}$ denotes the output matrix of hidden-layer

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}(\mathbf{x}_1) \\ \vdots \\ \mathbf{h}(\mathbf{x}_N) \end{bmatrix} = \begin{bmatrix} h_1(\mathbf{x}_1) \cdots h_L(\mathbf{x}_1) \\ \vdots \\ h_1(\mathbf{x}_N) \cdots h_L(\mathbf{x}_N) \end{bmatrix} \tag{3}$$

and

$$\mathbf{T} = \begin{bmatrix} \mathbf{t}_1^T \\ \vdots \\ \mathbf{t}_N^T \end{bmatrix} \tag{4}$$

The minimal norm least square method was used in the original implementation of the classical ELM [35]

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

If standard optimization method is used, the constrained-optimization-based ELM [34] can be written as

$$\text{Minimize}: \quad L = \tfrac{1}{2}\|\boldsymbol{\beta}\|^2 + C\tfrac{1}{2}\|\boldsymbol{\xi}\|^2$$

[1] A preliminary study on the Reduced KELM was presented.