Neurocomputing 128 (2014) 50–58

Contents lists available at ScienceDirect

Neurocomputing

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

An incremental extreme learning machine for online sequential learning problems

Lu Guo [a,b], Jing-hua Hao [a,b], Min Liu [a,b,*]

[a] Department of Automation, Tsinghua University, Beijing 100084, China
[b] Tsinghua National Laboratory for Information Science and Technology, Beijing 100084, China

ARTICLE INFO

Article history:
Received 6 September 2012
Received in revised form
22 February 2013
Accepted 4 March 2013
Available online 5 November 2013

Keywords:
Incremental learning algorithm
Extreme learning machine (ELM)
Incremental ELM (IELM)
Online sequential ELM (OS-ELM)
Fixed size LSSVM (FS-LSSVM)

ABSTRACT

A fast and outstanding incremental learning algorithm is required to meet the demand of online applications where data comes one by one or chunk by chunk to avoid retraining and save precious time. Although many interesting research results have been achieved, there are still a lot of difficulties in real applications because of their unsatisfying generalization performance or intensive computation cost. This paper presents an Incremental Extreme Learning Machine (IELM) which is developed based on Extreme Learning Machine (ELM), a unified framework of LS-SVM and PSVM presented by Hang et al. (2011) in [15]. Under different application demand and different computational cost and efficiency, three different alternative solutions of IELM are achieved. Detailed comparisons of the IELM algorithm with other incremental algorithms are achieved by simulation on benchmark problems and real critical dimension (CD) prediction problem in lithography of actual semiconductor production line. The results show that kernel based IELM solution performs best while least square IELM solution is the fastest of the three alterative solutions when the number of training data is huge. All the results show that the presented IELM algorithms have better performance than other incremental algorithms such as online sequential ELM (OS-ELM) presented by Liang et al. (2006) [8] and fixed size LSSVM presented by Espinoza et al. (2006) [11].

1. Introduction

Over the past few decades, batch learning algorithms have been discussed and investigated thoroughly. Plenty of interesting research work has been developed, such as Back-propagation Neutral Network (BP-NN) [1], Support Vector Machine (SVM) [2], Least Square Support Vector Machine (LSSVM) [3], Proximal Support Vector Machine (PSVM) [4], Extreme Support Vector Machine (ESVM) [5], Extreme Learning Machine (ELM) [6], etc. Although many interesting research results have been attained, there is still a lot of difficulties in real applications because of the typical features for the datum attained from an actual environment. The typical feature for the datum attained from the actual environment is that the training data often arrives one by one or trunk by trunk. Retraining of all the data with the batch learning algorithm when one new data comes is very time consuming and cannot meet the time demand of actual applications. Intensive computation cost and strict time requirements severely restrict these batch learning algorithms from real applications.

Originated from the batch learning methods that have been developed, many online sequential algorithms have been presented to meet the actual application demand. In [7], a generalized growing and pruning RBF (GGAP-RBF) is presented. The authors introduce the significance of the neurons and develop an online sequential algorithm by generalized growing and pruning methods. In [8], an online sequential extreme learning machine (OS-ELM) is introduced which is much faster and produces better generalization performance compared with other sequential learning algorithms, such as GGAP-RBF. Based on the batch learning algorithm of SVM, an incremental support vector machine is presented in [9]. The basic idea is that the new SVM is built based on the new arrival data and the trained support vectors. The method is called a SV-incremental algorithm in [10]. Fixed-size LSSVM (FS-LSSVM) is developed in [11] for large scale regression problems. Based on quadratic Renyi entropy criteria, an active support vector selection method is developed. Compared with LSSVM, FS-LSSVM needs much less support vectors and produces better performance. In [12], an incremental learning algorithm for ESVM (IESVM) is developed by He et al., and the parallel version of IESVM (PIESVM) is also presented based on the powerful parallel programming framework of MapReduce. The presented PIESVM is much more efficient than ESVM, while the solutions obtained are exactly the same as that by ESVM.

GGAP-RBF and OS-ELM may be over-fitting because of the modeling basis of Empirical Risk Minimization principle. The SV-incremental algorithm and FS-LSSVM are constructed on the basis of Structural Risk Minimization principle, thus avoiding over-fitting and having been very popular in recent years. However in the SV-incremental

* Corresponding author at: Department of Automation, Tsinghua University, Beijing 100084, China.
E-mail address: lium@tsinghua.edu.cn (M. Liu).

0925-2312/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.neucom.2013.03.055

algorithm, the previous support vectors may have only a little influence on the new SVM [10] and the performance cannot be satisfied. Also, the computation cost of FS-LSSVM is very high because of the eigenvectors and eigenvalues computation. The PIESVM algorithm can efficiently solve the large-scale problems and the online problems at the same time and has shown great prospects in real applications. A new online sequential learning algorithm based on an enhanced extreme learning machine has been presented recently in [13] by using left or right pseudo-inverse. The algorithm performs much better than other sequential algorithms and is very promising.

To join the ELM and SVM together is an important method to overcome the over-fitting problem of ELM. The first important contribution in joining the ELM and SVM together is presented in [5] by Liu et al. The presented ESVM algorithm is much faster than nonlinear SVM algorithms and obtains better generalization performance than ELM. A similar work is also done on standard SVM by Frnay and Verleysen in [14]. It has been shown that better performance can be achieved by simply replacing the SVM kernels with random ELM kernels in SVMs [5,14].

The general algorithm is proposed by Huang et al. in [15]. A constrained optimization based Extreme Learning Machine as a unified learning framework for LS-SVM, PSVM and other regularization algorithms is developed. Huang et al. also proved in the paper that LS-SVM and PSVM can only obtain suboptimal solutions comparing with ELM both in theoretical and simulation results. Different from other kernel based algorithms, ELM provides a unified platform with a special mapping referred to as ELM feature mapping, which is not relevant with the target value and can be constituted by almost all the nonlinear piecewise continuous functions. The ELM feature mapping can be regarded as a good dimensionality reduction which has a fixed dimension size while the dimension of input data increases. The kernel based ELM is also given in [15] when the feature mapping is unknown.

In this paper, based on the batch-learning idea of ELM [15], an Incremental Extreme Learning Machine (IELM) is presented to meet the demand of actual applications where data comes one by one or chunk by chunk. Under different computational cost and efficiency, three different alterative solutions of IELM are achieved. The three different alterative solutions are referred to as Minimum Norm Incremental Extreme Learning Machine (MN-IELM), Least Square Incremental Extreme Learning Machine (LS-IELM) and Kernel Based Incremental Extreme Learning Machine (KB-IELM). All three different alterative solutions of IELM are proposed for online industrial applications to avoid retraining on all the data when new data comes, thus improving the training speed and efficiency.

However, the computation cost will be sharply different for the three proposed algorithms under different application conditions. It is suggested that different algorithms should be selected under different applications. The same suggestions are also presented in [15]. For the case where the number of training data is not huge and the dimensionality of ELM feature space is very large, the MN-IELM is preferred for application to reduce computation cost. For the case where the number of training data is very huge, for example, the number is much larger than the feature space dimensionality, LS-IELM is a better option for application to improve the computation efficiency. For the case where the feature mapping is unknown and MN-IELM and LS-IELM cannot be used any more, KB-IELM should be selected for applications.

The rest of the paper is organized as follows. Section 2 gives the brief review of the unified framework of ELM and the developed three alterative batch learning algorithms. Section 3 presents our three incremental ELM algorithms. Simulations are carried out and results are analyzed in Section 4. Conclusions are drawn in Section 5.

2. Brief review of the unified framework of ELM

Huang et al. propose a unified framework of ELM by introducing a constrained optimization problem, thus bridges the gap between ELM and SVMs. Given the input training dataset $X=\{x_i\}_{i=1}^N$ and the corresponding output training dataset $Y=\{y_i\}_{i=1}^N$ where N is the total number of training data, and given the ELM mapping as $h(x_i)$ and the output weight as W, Huang et al. propose the multi-classifier with multi-outputs constrained optimization based ELM by formulating as

$$\text{Minimize}: L_{P_{ELM}}=\tfrac{1}{2}\|W\|^2+\nu\tfrac{1}{2}\|\xi\|^2 \tag{1}$$

Subject to:

$$HW=Y-\xi \tag{2}$$

where the training error and the target class value of all the sample points are denoted as $\xi=[\xi_1{}',\xi_2{}',\dots,\xi_N{}']^T$ and $Y=[y_1,\ y_2,\dots,y_N]^T$ respectively. $\xi_i{}'=[\varepsilon_{i1},\varepsilon_{i2},\dots,\varepsilon_{im}]$ is the training error of the m output nodes with respect to the input sample point x_i. $y_i=[t_{i1},t_{i2},\dots,t_{im}]$ is the target value only if the corresponding position p is one and the other positions are zero, where p is the class of the training sample x_i. $W=[w_1,w_2,\dots,w_m]_{L\times m}^T$ is the weight vector of the link between the hidden layer and the output node. $H=[h(x_1),h(x_2),\dots,h(x_N)]_{N\times L}^T$ is the mapping matrix for all the input sample points.

To train ELM is equivalent to solve the optimization problem. The Lagrange relaxation function can be given as follows:

$$L_{D_{ELM}}=\tfrac{1}{2}\|W\|^2+\nu\tfrac{1}{2}\|\xi\|^2-S^T(HW-Y+\xi) \tag{3}$$

where the Lagrange multipliers vector is $S=[s_1,s_2,\dots,s_N]^T$ and $s_i=[l_{i1},l_{i2},\dots,l_{im}]^T$. The optimized solution of $L_{P_{ELM}}$ is the saddle point of $L_{D_{ELM}}$ according to KKT optimality conditions. Using KKT optimality conditions, we get the following equations:

$$\frac{\partial L_{D_{ELM}}}{\partial W}=W-H^TS=0 \tag{4}$$

$$\frac{\partial L_{D_{ELM}}}{\partial \xi}=\nu\xi-S=0 \tag{5}$$

$$HW-Y+\xi=0 \tag{6}$$

2.1. Minimum norm incremental extreme learning machine

If we substitute (4) and (5) into (6), we can get

$$HH^TS-Y+\frac{1}{\nu}S=0$$

thus

$$S=\left(\frac{1}{\nu}+HH^T\right)^{-1}Y \tag{7}$$

So the output weight vector can be attained by substituting (7) into (4) as

$$W=H^T\left(\frac{1}{\nu}+HH^T\right)^{-1}Y \tag{8}$$

2.2. Least square incremental extreme learning machine

The above equation can also be simplified by substituting (5) and (6) into (4) as

$$W-\nu H^T(-HW+Y)=0 \tag{9}$$