An Incremental Learning Method for L1- Regularized Kernel Machine in WSN

Ji Xin-rong[1,3, a], Hou Cui-qin[1,b], Hou Yi-bin[1,c*] and Li Da[2,d]

[1]Embedded Computing Institute, Beijing University of Technology, Beijing, China

[2]Beijing Engineering Research Center for IOT Software and Systems, Beijing, China

[3]School of Information & Electrical Engineering, Hebei University of Engineering, Handan, China

[a]jixinrong@emails.bjut.edu.cn, [b]houcuiqin@bjut.edu.cn,
[c]yhou@bjut.edu.cn,[d]lida1204@bjut.edu.cn

Keywords: Kernel Machine; Incremental Learning Method; L1 Regularized; Wireless Sensor Network (WSN)

Abstract. Due to the limited energy, memory space and processing ability on wireless sensor nodes, the batch learning method will be infeasible for larger number of samples or sequence samples. This paper focuses on the incremental learning method for kernel machine by involving L1 regularized, a novel incremental learning algorithm for L1 regularized Kernel Minimum Squared Error machine (L1-KMSE-Increm) is proposed and evaluated on both synthetic and real data sets. The simulation results reveal that L1-KMSE-Increm algorithm can obtain almost the same prediction accuracy as that of corresponding batch learning method, and significantly outperforms the competitor on the sparse ratio of model and the running time.

Introduction

Wireless Sensor Network (WSN), as a new technology of information collecting and processing, has been widely used in many fields. Classification and regression problems are the most important and basic tasks in many applications of WSN, so many machine learning methods are becoming popular in WSN [1-3]. However, WSN is a resource-constrained network, where each node has very limited energy, memory space and processing ability. When there are large amount of training samples or training samples are available in sequence, the batch learning method performed on sensor node will be infeasible. Alternatively, the incremental learning method of classifiers or regression machines is initiated and researched for Support Vector Machine (SVM) to deal with large-scale datasets [6-11]. Kernel method or kernel machine, short for machine learning method based on kernel function, attracts widely attention and research because of the successful application of SVM and statistical learning theory, and has become the mainstream method in machine learning. Kernel method has incomparable advantages in solving non-linear problems, and has been successfully applied to many practical applications [4,5]. However, kernel method relies on all training samples to predict the result of new sample. Therefore, with limited energy, memory space and processing ability on sensor nodes, the research on incremental learning method for kernel machine, has great significance to study and apply kernel machine in WSN.

Most similar to the research presented here are for SVM [6-11]. However, the sparse characteristic of SVM is determined by Hinge loss function, the support vectors will gradually increase with the sequentially executing of incremental learning method on training samples, so it will rapidly increase the computational cost and the need for larger memory space on subsequent training samples. As an extension to Minimum Squared Error, Kernel Minimum Squared Error (KMSE) is developed for solving non-linear problems, and has been proven to have excellent performance and general applicability [12]. L1 regularized, widely used to solve optimization problems by incorporating a L1 penalty, can find a simplistic model for training samples, such as Lasso and Compressive Sensing [13]. To overcome the drawback of existing incremental learning methods for kernel machines, this paper constructs the optimization problem for kernel Minimum squared error machine by involving L1 regularized, its goal is to obtain more sparse models to reduce the computational cost and the need for larger memory space.

This paper proposes a novel incremental learning algorithm for L1 regularized kernel Minimum squared error machine (L1-KMSE-Increm), which can obtain almost the same prediction accuracy as that of corresponding batch learning algorithms, and significantly outperforms the competitor on the sparse characteristic and the computational cost.

The remainder of this paper is organized as follows: Section 2 briefly reviews the supervised learning model for Minimum Squared Error estimation base on regularized kernel methods. Section3 presents the detail solving, derivation and procedure of L1-KMSE-Increm. Section 4 evaluates, through extensive simulations, the performance of our proposal on synthetic and real data sets. Finally, this paper is concluded in Section 5.

Preliminaries

Consider the supervised learning model: $DS = \{(x_i, y_i) \mid x_i \in R^d\}_{i=1}^n$ is an independent and identically distributed training sample set with $(x_i, y_i) \in c \times y \ \forall i \in \{1,...,n\}$, c is known as the feature or input space, y is known as the label or output space. The minimum squared error estimation problem is to use training samples to infer a decision function mapping inputs to outputs that minimizes the expected squared error. Regularized kernel methods offer one approach to nonparametric estimation. The convex optimization problem for regularized kernel minimum squared error estimation can be written as (1) [7]:

$$\min_{\substack{f \in H_K \\ f}} \sum_{i=1}^{n} (f(x_i) - y_i)^2 + l\,(\|f\|_{H_K}) \tag{1}$$

where H_K denotes the Reproducing Kernel Hilbert Space induced by a positive semi-definite kernel $K(\cdot,\cdot) := c \times c \rightarrow R$, $\|\cdot\|_{H_K}$ denotes the norm associated with H_K , and the kernel K is a parameter, chosen as a similarity measure between inputs. $f(\cdot) \in H_K$ is defined as the solution of (1), $l > 0$ is a regularized parameter which specifies the tradeoff between minimization of the empirical risk function and the smoothness or simplicity. The explicit form of $f(\cdot)$ is given by the celebrated “Representer Theorem” which plays a central role in solving practical problems of statistical estimation.

Representer Theorem [7]: Let $f(\cdot)$ be the minimizer of (1). Then, there exists $a_n \in R^n$ such that

$$f^*(x) = \sum_{i=1}^{n} a_i K(x_i, x) \tag{2}$$

where n is the number of training examples, $K(x_i, x)$ denotes the similarity measure between training sample x_i and new sample x, and $a_i \in R \ \forall i \in \{1,...,n\}$ is the weight of $K(x_i, x)$. It can be shown from (2) that the prediction of x depends on all samples used in model training.

For ease of description, a definition is introduced.

Definition (Spare rate of model): The ratio of the number of training samples which correspond to the nonzero components of weight vector to that of all samples used in model training.

Incremental learning method for L1 regularized kernel machine

This paper focuses on binary classification problems and three key points of incremental learning method are considered: first, with increasing number of samples in model training, the accuracy and generalization ability of model will be greatly improved; second, based on the latest model, the performance of model will be further optimized by new training samples; and, third, after a stable model is obtained, the new training samples will not affect the performance of model, namely, the predictions of new samples by two successive latest models are exactly the same. Inspired by the above facts, the incremental learning optimization problem for L1 regularized kernel minimum square error machine is constructed and formalized as (3):

$$\begin{aligned} &\min_{\substack{f_n, f_o \in H_K \\ f}} \frac{1}{m}\sum_{i=1}^{m}(y_i - f_n(x_i))^2 + l\,\|f_n\|_1 \\ &\quad s.t.\quad f_n(x_i) = f_o(x_i),\quad i = 1,\cdots,m \end{aligned} \tag{3}$$

where x_i is the i-th feature vector, $y_i \in \{-1,1\}$ is the label of x_i, and m is the number of training samples. $f_n(\cdot)$ is the solution of (3), and $f_o(\cdot)$ is the latest model. $l\,\|f_n\|_1$ is $\mathbf{l}_1$ regularized term , $l > 0$ is a scalar regularized parameter that is usually chosen by cross-validation.

The Augmented Lagrangian Method of Multiplier is used for the optimization problem (3), and the Augmented Lagrangian function [14] is constructed and shown as (4):

$$L(f_n, p_i) = \frac{1}{2}\sum_{i=1}^{m}(y_i - f_n(x_i))^2 + l\,\|f_n\|_1 + \sum_{i=1}^{m}\left(p_i(f_n(x_i) - f_o(x_i)) + \frac{c}{2}\|f_n(x_i) - f_o(x_i)\|_2^2\right) \tag{4}$$

where p_i is the dual variable or Lagrange multiplier, c is a positive scalar parameter. The iterative formulas for problem (4) are derived and shown as (5) - (6):

$$f_n^{k+1}(x) = \arg\min \frac{1}{2}\sum_{i=1}^{m}(y_i - f_n(x_i))^2 + l\,\|f_n\|_1 + \sum_{i=1}^{m}\left(p_i^k(f_n(x_i) - f_o(x_i)) + \frac{c}{2}\|f_n(x_i) - f_o(x_i)\|_2^2\right) \tag{5}$$

$$p_i^{k+1} = p_i^k + c\left(f_n^{k+1}(x_i) - f_o(x_i)\right) \tag{6}$$

However, the problem (5) is still a non-constrained convex optimization problem. For convenience, this problem is reformulated in matrix form, shown as (7):

$$\min \frac{1}{2}(\boldsymbol{Y} - \boldsymbol{K}a)^T(\boldsymbol{Y} - \boldsymbol{K}a) + p^T(\boldsymbol{K}a - f_o(\boldsymbol{x})) + \frac{c}{2}\|\boldsymbol{K}a - f_o(\boldsymbol{x})\|_2^2 + l\,\|a\|_1 \tag{7}$$

where $\boldsymbol{Y} \in R^m$ is the label vector, $\boldsymbol{K} \in R^{m\times(m+1)}$ is the augmented kernel matrix obtained by $K(x_1, x_2)$, $\forall x_1, x_2 \in DS$, m is the number of training samples, and $\boldsymbol{\alpha} \in R^{m+1}$ is the weight vector. To solve problem (7), Alternating Direction Method of Multiplies (ADMM) [13] is used. In ADMM form, the problem (7) can be reformulated as (8), and the resulting iterations as (9), (10) and (11):

$$\begin{aligned} &\min \frac{1}{2}(\boldsymbol{Y} - \boldsymbol{K}a)^T(\boldsymbol{Y} - \boldsymbol{K}a) + p^T(\boldsymbol{K}a - f_o(\boldsymbol{x})) + \frac{c}{2}\|\boldsymbol{K}a - f_o(\boldsymbol{x})\|_2^2 + l\,\|\boldsymbol{z}\|_1 \\ &\qquad s.t.\quad \boldsymbol{\alpha} - \boldsymbol{z} = 0 \end{aligned} \tag{8}$$

$$\boldsymbol{\alpha}_j^{k+1} = [(1+c)\boldsymbol{K}^T\boldsymbol{K} + r\boldsymbol{I}]^{-1}[\boldsymbol{K}^T(\boldsymbol{Y} + cf_o(\boldsymbol{x}) - p^{kT}) + r(\boldsymbol{z}^k + \boldsymbol{u}^k)] \tag{9}$$

$$\boldsymbol{z}^{k+1} := S_{l/r}(\boldsymbol{\alpha}^{k+1} + \boldsymbol{u}^k) \tag{10}$$

$$\boldsymbol{u}^{k+1} := \boldsymbol{u}^{k} + \boldsymbol{\alpha}^{k+1} - \boldsymbol{z}^{k+1} \tag{11}$$

In (9), $\boldsymbol{I}$ is an identity matrix, and $(1+c)\boldsymbol{K}^T\boldsymbol{K} + r\,\boldsymbol{I}$ is always invertible, since $r > 0$. S in (10) is the soft thresholding operator, and is defined as (12):

$$S_q(\mathrm{a}) = \begin{cases} \mathrm{a} - q & \mathrm{a} > q \\ 0 & |\mathrm{a}| \le q \\ \mathrm{a} + q & \mathrm{a} < -q \end{cases} \tag{12}$$

A sparse weigh vector $\boldsymbol{\alpha}$ can be obtained by iteratively executing in order as (9), (10), (11), (6) on each group training samples, l is the number of nonzero components of weight vector $\boldsymbol{\alpha}$. The model obtained on k -th group training samples can be expressed as (13):

$$f^k(x) = \sum_{i=1}^{l} a_i k(x, x_i) \tag{13}$$

Based on the above solving and derivation of incremental learning method for L1 regularized kernel minimum squared error classification problem, an incremental learning algorithm for L1 regularized kernel minimum squared error machine (L1-KMSE-Increm) is proposed, its detailed procedure is illustrated as *Algorithm1.*

Algorithm1: L1-KMSE-Increm

Input: Initialize the number of each class training samples m, iterations $k = 0$, $f^k = 0$, and set kernel parameter s and regularized parameter l .

Output: the sparse model $f^*(x) = \sum_{i=1}^{l} a_i k(x_i, x)$.

Repeat:

$k = k + 1$;

Step1: Select randomly m samples from each class of training sets and initialize p a zero vector.

Step2: Incorporate the samples of $f^{k-1}(x) = \sum_{i=1}^{l} a_i k(x, x_i)$ and predict the output of current samples.

Step3: Obtain the model $f^k(x) = \sum_{i=1}^{l} a_i k(x, x_i)$ by using (8) -(11).

Step4: If no new samples are available, stop and go to Output; else $k = k + 1$ return to Repeat.

Numerical simulations

To verify the proposed L1-KMSE-Increm algorithm, simulation experiments have been conducted on synthetic and real datasets, and compared with the well-known SVM batch learning algorithm (SVM-batch), the L1 regularized KMSE batch learning algorithm (L1-KMSE-batch), and SVM incremental learning algorithm (SVM-Increm) [6] in terms of prediction accuracy, sparse rate of model, and running time. The Gaussian Kernel $k(x, y) = \exp(-\|x - y\|^2 / 2s^2)$ with s being a parameter controlling the width of the Kernel is chosen for our simulation. SMO method is used to train SVM to accelerate its speed.

Synthetic dataset. Synthetic dataset is composed of labeled training samples from two different equiprobable nonlinear separable classes C_1 and C_2. Class C_1 contains samples from a two dimensional Gaussian distribution with covariance matrix $\Sigma = [0.6, 0; 0, 0.4]$, and mean vector $mu_1 = [0, 0]^T$. Class C_2 is a mixture of Gaussian distributions with mixing parameters $p_1 = 0.3$ and $p_2 = 0.7$, mean vectors $mu_2 = [-2,-2]^T$ and $mu_3 = [2,2]^T$, and equal covariance matrix Σ. To illustrate the effect of training set size on the performance of different algorithms, 6 training sets with different size are generated, and testing samples with the same number as that of training samples in each training set are generated. Each training set is divided into 20 groups, and each group has the same number of samples of per class. The optimal values of parameters for four different algorithms on Synthetic dataset are chosen by cross-validation. The optimal values $s = 8.0$ and $C = 2.0$ are used in SVM-batch and SVM-Increm, and $s = 0.2$ and $l = 1.2$ are employed in L1-KMSE-batch and L1-KMSE-Increm. Based on the above experimental setup, 30 Monte Carlo runs are performed. The simulation results of SVM-batch, L1-KMSE-batch, L1-KMSE-Increm and SVM-Increm in terms of prediction accuracy, spares rate of model and running time, are shown in Fig.1, Fig.2 and Fig.3, respectively.

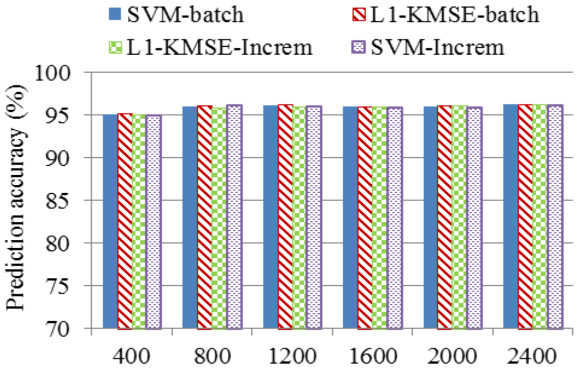

Fig.1. Comparison of prediction accuracy of four different algorithms on Synthetic dataset.

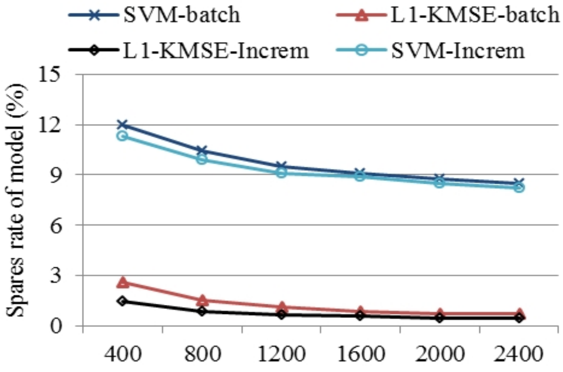

Fig.2. Comparison of spares rate of models of four different algorithms on Synthetic dataset.

As shown in Figure 1, SVM-batch, L1-KMSE-batch, L1-KMSE-Increm and SVM-Increm obtain approximately the same prediction accuracy on different groups of training dataset. The prediction accuracy of L1-KMSE-Increm on training sets composed of 400 and 800 samples is slightly lower than that of SVM-batch and L1-KMSE-batch, and the prediction accuracy of L1-KMSE-Increm on others training sets are perfectly consistent with that of SVM-batch and L1-KMSE-batch. It is shown that there is no distinct difference in terms of prediction accuracy obtained by SVM-batch, L1-KMSE-batch and L1-KMSE-Increm, and the number of training samples has a little influence on the prediction accuracy, the more training samples the better the prediction accuracy is.

Fig.2 shows the spares rate of models obtained by SVM-batch, L1-KMSE-batch, L1-KMSE-Increm and SVM-Increm. The spares rate of model of L1-KMSE-batch and L1-KMSE-Increm is significantly better than that of SVM-batch and SVM-Increm, and the spares rate of model of L1-KMSE-Increm slightly outperforms that of L1-KMSE-batch. It is shown that L1-KMSE-Increm can achieve extremely sparse model which can greatly reduce the computational cost on new training samples.

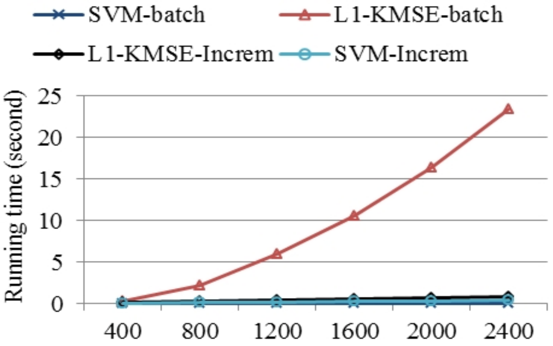

Fig.3. Comparison of running time of four different algorithms on Synthetic dataset.

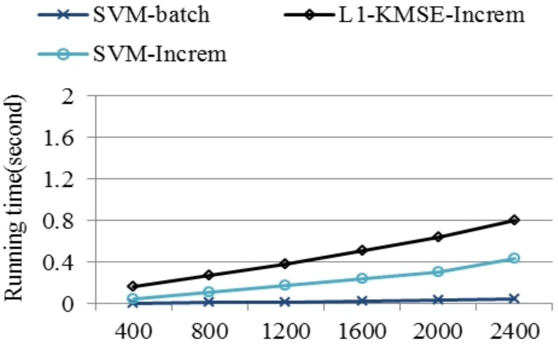

Fig.4. A larger version of the running time for L1-KMSE-Increm, SVM-batch and SVM-Increm.

As shown in Fig.3, with the increase of the number of training samples, the running time of L1-KMSE-batch increases sharply, but that of L1-KMSE-Increm, SVM-batch and SVM-Increm is grows slowly. Fig.4 shows the comparison of running time of L1-KMSE-Increm, SVM-batch and SVM-Increm in Fig.3, the running time of L1-KMSE-Increm slightly more than that of SVM-Increm and SVM-batch. It is shown that the running time of L1-KMSE-Increm is far better than that of L1-KMSE-batch, that is, L1-KMSE-Increm can obtain very high predictive accuracy with relatively much less computational cost, but L1-KMSE-Increm is slightly worse than SVM-Increm and SVM-batch in terms of running time on Synthetic dataset.

UCI dataset. Waveform Database Generator from the UCI repository (http://archive.ics.uci.edu/ml/) is used in our experiment. It consists of 21 attributes, 5000 instances and 3 forms. The binary classification problem of form 0 and form 1 is considered, 1000 samples per form are randomly selected as training samples, and the rest of per form as test samples. In this experiment, training samples are randomly divided into 50 groups, and each group has the same number of samples of per form. Similarly, the optimal values of parameters for different algorithms on UCI dataset are chosen by cross-validation. The optimal values $s = 0.70$ and $C = 4.0$ are employed in SVM-batch and SVM-Increm, and $s = 1.8$ and $C = 0.5$ are used in L1-KMSE-batch and L1-KMSE-Increm. Based on above experimental setup, 30 Monte Carlo runs are performed on UCI dataset. The simulation results of SVM-batch, L1-KMSE-batch, L1-KMSE-Increm and SVM-Increm in terms of prediction accuracy, spares rate of model and running time, are shown in Table 1.

	Prediction accuracy (%)	Spares rate of model (%)	Running time (sec.)
SVM-batch	91.1584	21.53	0.1224
L1-KMSE-batch	91.7472	1.22	9.8587
L1-KMSE-Increm	91.0924	0.34	0.7285
SVM-Increm	90.9926	19.38	1.2031

Table 1. Comparisons of four different algorithms on UCI dataset

As shown in Table 1, the prediction accuracy of L1-KMSE-Increm is extremely approximate to that of L1-KMSE-batch and SVM-batch, the spares rate of model of L1-KMSE-Increm is significantly superior to that of SVM-batch and SVM-Increm, and more importantly, the running time of L1-KMSE-Increm is significantly less than that of L1-KMSE-batch, and is slightly less than that of SVM-Increm. Although SVM-batch takes much less running time than L1-KMSE-Increm, it requires all of the training samples and a very immense memory space. It is shown that L1-KMSE-Increm can achieve good performance in terms of prediction accuracy, spares rate of model, and running time.

Conclusions

We have proposed an Incremental learning algorithm L1-KMSE-Increm for L1 regularized Kernel Minimum Squared Error machine and proved its validity by synthetic and real data sets. L1-KMSE-Increm can obtain almost the same prediction accuracy as batch training algorithms and extremely sparse model. More importantly, it can significantly reduce the training time of model. Because of the remarkable advantage in terms of running time and spare model, L1-KMSE-Increm is considered as a practicable method to train kernel machines in applications limited by processing power and memory space.

Acknowledgments

This work was financially supported by the National Natural Science Foundation of China under Grant No. 61203377.

References

[1] FX Lv, JC Zhang, XK Guo, et al.,The Acoustic Target in Battlefield Intelligent Classification and Identification with Multi-features in WSN, Science Technology and Engineering.13(35) (2013) 10713-10721(In Chinese).

[2] Taghvaeeyan S, Rajamani R, Portable Roadside Sensors for Vehicle Counting, Classification, and Speed Measurement, IEEE Transactions on Intelligent Transportation Systems., 15(1) (2014) 73-83.

[3] Xinrong Ji, Cuiqin Hou, Yibin Hou, Research on the Distributed Training Method for Linear SVM in WSN, Joural of electronics & information technology.37(3) (2015)708-714(In Chinese).

[4] Scholkopf B., Smola A, Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond, MIT Press, 2002.

[5] Shawe-Taylor J., Cristianini N, An Introduction to Support Vector Machines and Other Kernel-based Learning Methods,Cambridge University Press, 2000.

[6] Syed,N., Liu,H., and Sung,K., Incremental learning with support vector machines, International Joint Conference on Articial Intelligence (IJCAI), Stockholm, Sweden. (1999).

[7] Stefan Rüping, Incremental Learning with Support Vector Machines, IEEE International Conference on Data Mining (ICDM).26(5) (2001)641-652.

[8] Pavel Laskov, Christian Gehl, Stefan Krüger and Klaus-Robert Müller, Incremental Support Vector Learning: Analysis, Implementation and Applications, Journal of Machine Learning Research. 7(2006)1909-1936.

[9] Jyrki Kivinen, Alexander J.Smola, and Robert C.Williamson, Online Learning with Kernels, IEEE Tractions on signal processing. 100(10) (2010)1-12.

[10] Guoqi Li, Guangshe Zhao, Feng Yang, Towards the online Learning with Kernels in Classification and Regression, Evolving Systems. 5(1) (2014)11-19.

[11] Paul Honeine, Analyzing sparse dictionaries for online learning with kernels, IEEE Transactions on Signal Processing. (2015)1-25.

[12] Jianhua Xu, Xuegong Zhang, Yanda Li, Kernel MSE algorithm: a unified framework for KFD, LS-SVM and KRR, International Joint Conference on Neural Networks. 2(2001)1486-1491(In Chinese).

[13] S Boyd, N Parikh, E Chu, Distributed Optimization and Statistical Learning Via the Alternating Direction Method of Multipliers, Foundations and Trends in Machine Learning. 3(1) (2011) 1-122.

[14] D P Bertsekas, J N Tsitsiklis, Parallel and Distributed Computation: Numerical Methods, Athena Scientific, 1997.