Proceedings of the IEEE
International Conference on Automation and Logistics
Qingdao, China September 2008

Nonlinear Time Series Fault Prediction Online Based on Incremental Learning LS-SVM

Zhanxin Zhou
Department of Control Science and Engineering
Tongji University
Shanghai, China
zhouzhanxin@mail.tongji.edu.cn

Yongqi Chen
Department of Control Science and Engineering
Tongji University
Shanghai, China
School of Science and Technology
Ningbo University
Ningbo, China
lingfen7781@163.com

***Abstract* - For nonlinear time series fault prediction online, an incremental learning least squares support machine (LS-SVM) is presented to replace LS-SVM which is as a kind of regression method with good generalization ability and trained offline in batch way. The incremented learning LS-SVM fully utilizes historical training results and reduces memory and computation time, which guarantee to predict time series online. Two simulations results show that the incremental learning LS-SVM has good performance for predicting nonlinear series fault prediction online.**

***Index Terms* - Least squares support vector machine (LS-SVM), incremental learning, prediction, time series.**

I. Introduction

The unknown system model fault prediction has become a very important research branch. Although the method and theories on linear system fault prediction have been well developed, there is not a standard method for nonlinear time series system fault prediction now. For this reason, the study on nonlinear time series system fault prediction has attracted much attention.

At present, researchers mainly concentrate on neural networks [1], [2], such as BP network, RBF neural network etc. However, the neural model based on the design of networks is too complicated to be described precisely. Gradient descent and back propagation are always used to adjust the weights of neural networks. Slow convergence and local minimum are main drawbacks of these algorithms. All these disadvantages limit neural networks in the application of nonlinear time series prediction online.

Recently, LS-SVM is developed for function estimation by Suykens [3], [4]. LS-SVM has emerged as an alternative approach to neural networks. This is due to the fact that the LS-SVM is established based on the structural risk minimization principle rather than the empirical error commonly implemented in the neural networks [5], [6]. For this reason, LS-SVM can achieve higher generalization performance, well fast convergence and higher precision than neural networks. But almost all of nonlinear series system identification by SVM is off-line [7], because LS-SVM model is trained offline in batch way. There is a big obstacle for using SVM to predict fault of nonlinear time series online.

In this paper, the nonlinear time series fault prediction online based on incremental learning LS-SVM has been presented. Incremental learning has the similar idea as sequential SVM [8]. But the application is different. The training of LS-SVM can be implemented in a way of incremental learning avoiding computing large-scale matrix inverse, but maintaining the precision when training and testing data [9], [10]. Because this method fully utilizes the historical training results and reduces memory and computation time. It can predict nonlinear time series fault prediction online. Simulation results show the effective and efficiencies of this method.

This paper is organized as the following. The normal LS-SVM regression is reviewed briefly in section 2. Incremental learning LS-SVM for nonlinear time series prediction online is discussed in details in section 3. In section4, there are two simulation examples. One of them is a nonlinear time series fault prediction online, simulations result is analyzed. Section 5 makes a conclusion.

II. Least Squares Support Vector Machine Regression

LS-SVM uses a hypothesis space of linear function in a high dimensional feature space by using the kernel theory. This algorithm is trained by optimization theory.

Given training dataset

$$x_i, y_i \qquad i = 1 \cdots N, x_i \in R^n, y_i \in R^c \ .$$

LS-SVM supposes the unknown function as the following:

$$y = w^T \varphi(x) + b \tag{1}$$

where x, y are input variable and output variable, $\varphi(x)$ is a nonlinear function which maps the feature space of input into a higher dimension feature space and can be reached by the kernel strategy. w is a coefficient determining the margin of support vectors and b is a bias term. The coefficients (w, b) are determined by minimizing the following regularized risk function and using the equality constraints.

This work was supported in part by the National Science & Technology Pillar Program in the Eleventh Five-year Plan Period (2007BAF10B00).

978-1-4244-2503-7/08/$20.00

$$\begin{cases} \min J(w,e)=\frac{1}{2}w^T w+\frac{1}{2}c\sum_{i=1}^{N}e_i^2 \quad (c>0) \\ y_i=w^T\varphi(x_i)+b+e_i \quad i=1\cdots N \end{cases} \tag{2}$$

where e_i is the nonnegative error variable and is used to construct a soft margin hyper plane. In (2), the first term measures the inverse of the margin distance. In order to obtain the minimum structural risk, the first term should be minimized. c is the regularization parameter determining the fitting error minimization and smoothness. This optimization problem including the constraints can be solved by using the Lagrange function as following:

$$L(w,b,e,\alpha)=\frac{1}{2}w^T w+\frac{1}{2}c\sum_{i=1}^{N}e_i^2-\sum_{i=1}^{N}\alpha_i\{w^T\varphi(x_i)+b+e_i-y_i\} \tag{3}$$

where α_i is the Lagrange multiplier set. Through computing the partial derivatives of $L(w,b,e,a)$, the optimal condition of (3) can be obtained as following:

$$\begin{cases} \frac{\partial L}{\partial w}=0\Rightarrow w=\sum_{i=1}^{N}\alpha_i\varphi(x_i) \\ \frac{\partial L}{\partial b}=0\Rightarrow\sum_{i=1}^{N}\alpha_i=0 \\ \frac{\partial L}{\partial \alpha_i}=0\Rightarrow w^T\varphi(x_i)+b+e_i-y_i=0 \\ \frac{\partial L}{\partial e_i}=0\Rightarrow a_i=ce_i \end{cases} \tag{4}$$

These equality constraints can be transformed as following:

$$\begin{bmatrix} 0 & e1^T \\ e1 & Q+c^{-1}I \end{bmatrix}\begin{bmatrix} b \\ a \end{bmatrix}=\begin{bmatrix} 0 \\ y \end{bmatrix} \tag{5}$$

where $y=[y_1\cdots y_N]^T$, $a=[a_1\cdots a_N]^T$, $e1=[1\cdots 1]^T$,

$$Q_k=\begin{bmatrix} \varphi(x_1)^T\varphi(x_1) & \cdots & \varphi(x_1)^T\varphi(x_N) \\ \vdots & \ddots & \vdots \\ \varphi(x_N)^T\varphi(x_1) & \cdots & \varphi(x_N)^T\varphi(x_N) \end{bmatrix}.$$

$$U=Q+c^{-1}I \tag{6}$$

For this time, the LS-SVM model has been deduced. But there is no information about $\varphi(x)$ and (6) can't be calculated. Fortunately, LS-SVM has an important attribute that people can assume as following:

$$K(x_i,x_j)=\varphi(x_i)^T\varphi(x_j) \quad i=1\cdots N, j=1\cdots N \tag{7}$$

Without knowing $\varphi(x)$, $k(x_i,x_j)$ is the kernel function. There are various kernels used in the LS-SVM. Different kernel functions present different mappings from the input space to the feature space. As a result, LS-SVM model changes with the different kernel function. Among widely available kernels, especially, Radial Basis Function kernel is used in many areas because of its robustness. RBF-function is presented as

$$K(x,x_i)=\exp\{-\frac{\|x-x_i\|^2}{2\sigma^2}\}.$$

Through (5)-(7), parameters a,b can be calculated. Finally, the LS-SVM regression model can be expressed as the following:

$$y(x)=\sum_{i=1}^{N}a_iK(x,x_i)+b \tag{8}$$

III. Incremental Learning LS-SVM For Nonlinear Time Series Prediction Online

Assume nonlinear time series $\{x_1,x_2,\cdots x_n\}$, in order to predict the value of x_{k+p}, $\{x_k,x_{k+1}\cdots x_{k+p-1}\}$ must be the input data of LS-SVM. p is the embedding dimension and $k\in 1,\cdots n-p$. So the initial training samples can be obtained as following:

$$X=\begin{bmatrix} x_1 & x_2 & \cdots & x_p \\ x_2 & x_3 & \cdots & x_{p+1} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n-p} & x_{n-p+1} & \cdots & x_{n-1} \end{bmatrix}, Y=\begin{bmatrix} x_{p+1} \\ x_{p+2} \\ \vdots \\ x_n \end{bmatrix} \tag{9}$$

At the initial moment,

$$Q_{n-p}(i,j)=\varphi(X_i)^T\varphi(Y_j)$$
$$=K(X_i,Y_j) \quad i,j=1,2\cdots n-p$$

$$\partial_{n-p}=(a_1,a_2,\cdots a_{n-p})^T, b_{n-p}=b \tag{10}$$

Equation (5) can be rewritten as following:

$$\begin{bmatrix} 0 & e1^T \\ e1 & Q_{n-p}+c^{-1}I \end{bmatrix}\begin{bmatrix} b_{n-p} \\ a_{n-p} \end{bmatrix}=\begin{bmatrix} 0 \\ Y \end{bmatrix} \tag{11}$$

where

$$U_{n-p}=Q_{n-p}+c^{-1}I$$
$$=\begin{bmatrix} K(X_1,X_1)+1/c & \cdots & K(X_{n-p},X_1) \\ \vdots & \ddots & \vdots \\ K(X_1,X_{n-p}) & \cdots & K(X_{n-p},X_{n-p})+1/c \end{bmatrix} \tag{12}$$

Assume $M_{n-p}=U_{n-p}{}^{-1}$, from (11), it can be deduced:

$$\begin{cases} b_{n-p}=\frac{e1^T M_{n-p}Y}{e1^T M_{n-p}e1} \\ \partial_{n-p}=M_{n-p}(Y-\frac{e1e1^T M_{n-p}Y}{e1^T M_{n-p}e1}) \end{cases} \tag{13}$$

Then, the LS-SVM regression model for nonlinear time series prediction can be expressed as following:

$$y(x\_test)=\sum_{i=1}^{n-p}a_iK(X_i,x\_test)+b \tag{14}$$

If x_test is equal to $\{x_{n-p+1},x_{n-p+2}\cdots x_n\}$, $\hat{x}_{n+1}$ can be obtained as the prediction value of x_{n+1}. Because LS-SVM regression mode is applied for time series prediction online, new data points are obtained continuously. To track the dynamics of the nonlinear time series system, the new data points should be included into training data. For this reason, the x_{n+1} can be obtained and forms the new training sample $\{x_{n-p+2}, x_{n-p+3} \cdots x_{n+1}\}$. Then LS-SVM starts with $n-p+1$ training data points to predict the value of x_{n+2} and U_{n-p+1} can be presented as following:

$$U_{n-p+1}=\begin{bmatrix} K(X_1,X_1)+1/c & \cdots & K(X_{n-p},X_1) & K(X_{n-p+1},X_1) \\ \vdots & \ddots & \vdots & \vdots \\ K(X_1,X_{n-p}) & \cdots & K(X_{n-p},X_{n-p})+1/c & K(X_{n-p+1},X_{n-p}) \\ K(X_1,X_{n-p+1}) & \cdots & K(X_{n-p},X_{n-p+1}) & K(X_{n-p+1},X_{x-p+1})+1/c \end{bmatrix}$$

$$=\begin{bmatrix} U_{n-p} & V_{n-p+1}^T \\ V_{n-p+1} & d_{n-p+1} \end{bmatrix}$$

$$d_{n-p+1} = K(X_{n-p+1}, X_{x-p+1}) + 1/c$$

$$V_{n-p+1} = [K(X_1, X_{n-p+1}), \cdots K(X_{n-p}, X_{n-p+1})]^T \tag{15}$$

In order to compute

$$\begin{cases} \partial_{n-p+1}=(a_1,a_2,\cdots a_{n-p+1})^T \\ b_{n-p+1} \end{cases}$$

U_{n-p+1}^{-1} must be computed. But large-scale matrix inverse needs much time and memory. Time series prediction online system needs LS-SVM to track dynamical system quickly. For this reason, the training of LS-SVM can be placed in a way of incremental chunk avoiding computing large-scale matrix inverse but maintaining the precision.

So the $M_{n-p+1} = U_{n-p+1}^{-1}$ should be deduced as following:

$$\begin{aligned} M_{n-p+1} &= U_{n-p+1}^{-1} \\ &= \begin{bmatrix} U_{n-p} & V_{n-p+1}^T \\ V_{n-p+1} & d_{n-p+1} \end{bmatrix}^{-1} \\ &= \begin{bmatrix} M_{n-p} & 0 \\ 0 & 0 \end{bmatrix} + F_{n-p+1}F_{n-p+1}^T W_{n-p+1} \end{aligned} \tag{16}$$

where $F_{n-p+1} = [v_{n-p+1}^T M_{n-p}, -1]^T$,

$$W_{n-p+1} = \frac{1}{d_{n-p+1} - V_{n-p+1}^T M_{n-p} V_{n-p+1}}.$$

From (16), matrix M_{n-p+1} can be deduced directly without calculation of U_{n-p+1}^{-1}. Then b_{n-p+1} and ∂_{n-p+1} can be calculated through (13). This iteration process will save much time and memory. For this reason, incremental LS-SVM can predict nonlinear time series online. The whole process of nonlinear time series prediction online with incremental LS-SVM model is presented as following.

Step 1: Initialization of model including parameter n, p and samples X, Y;

Step2: Computing the U_{n-p} and $M_{n-p} = U_{n-p}^{-1}$ by (12);

Step3: Computing the b_{n-p} and ∂_{n-p} by (13);

Step4: Predicting $\hat{x}_{n+1}$ by (14);

Step5: $k = n - p + 2$;

Step6: Adding $\{x_k, x_{k+1} \cdots x_{k+p-1}\}$ into training samples;

Step7: Computing the U_{k-p} and M_{k-p} by (16), predicting $\hat{x}_{k+1}$;

Step8: k=k+1, go to step6.

IV. EXAMPLES

In this section, two examples are presented. One is prediction of the chaotic Mackey-Glass time series, the other one is nonlinear Henon time series fault prediction.

A. *Prediction of the Chaotic Mackey-Glass Time-Series*

The time series used in the simulation is generated by the chaotic Mackey-Glass delay equation:

$$\frac{dx}{dt} = -b * x(t) + \frac{a * x(t-\tau)}{x + x^{10}(t-\tau)} \tag{17}$$

In this simulation, parameter τ is set 17 and $b = 0.1$, $a = 0.2$. The goal of this simulation is to use known values of the time series to predict the future value x_{k+p}. So $\{x_k, x_{k+1} \cdots x_{k+p-1}\}$ must be presented to predict it. In the simulation, the parameter p is set 8. For this reason, eight point's values in the series are used to predict the value of the next time point.

The Root Mean Square Error (RMSE) is used to evaluate the performance of incremental learning LS-SVM prediction for the example. The RMSE of the validation set is computed as following:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (y(i) - \hat{y}(i))^2} \tag{18}$$

where N denotes the total number of test data samples. In the validation set, y represents the output of the original nonlinear time series system, while $\hat{y}$ represents the estimation output of the incremental learning LS-SVM model.

In this simulation, 16 points of the time series from x(500) to x(515) are used to construct initial training samples, and the next points of the series from x(516) to x(1000) are used as test samples. Then, RBF is selected as the kernel function of LS-SVM. Parameters $\sigma^2 = 1.5$ and $c = 50$. Fig. 1 depicts the results of Mackey-Glass time series prediction online using the incremental learning LS-SVM.

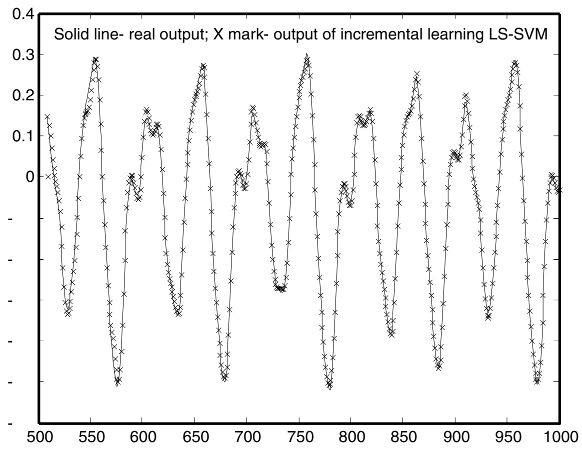

Fig. 1 The real Mackey-Glass output and the prediction results using incremental learning LS-SVM

Table I demonstrates comparison between incremental learning LS-SVM and LS-SVM in some important parameters such as RMSE, average time to predict one point of time series (Time-pred-point).

TABLE I
COMPARISON BETWEEN INCREMENTAL LEARNING LS-SVM AND LS-SVM

method	(c,σ^2)	RMSE	Time-pred-point
LS-SVM	(50,1.5)	0.0135	1.25s
Incremental LS-SVM	(50,1.5)	0.0147	0.0857s

From Fig.1 and Table I, perfect prediction effect is achieved by Incremental LS-SVM. Especially, incremental LS-SVM only uses 0.0857s to predict one point. For this reason, Incremental LS-SVM can track the dynamics time series system online.

B. Nonlinear time series fault prediction online

The time series used in the simulation is generated by the Henon delay equation:

$$x(n+1)=1-1.4x(n)^2+0.3x(n) \tag{19}$$

Fault function is $0.5\exp(-(n-300)^2/100)$. In the simulation, the first 250 time series data are selected to train incremental learning LS-SVM model. The next 50 data is used to test the prediction. Embedding dimension p is set 8. RMSE is used to evaluate the performance of incremental learning LS-SVM prediction. The normal Henon time series data set is constructed firstly. Meanwhile the Henon time series added fault function is predicted online by incremental learning LS-SVM. Fault prediction can be implemented by calculating the difference between the normal time series data set and the predicting series. Fig.2 depicts the results of nonlinear time series fault prediction.

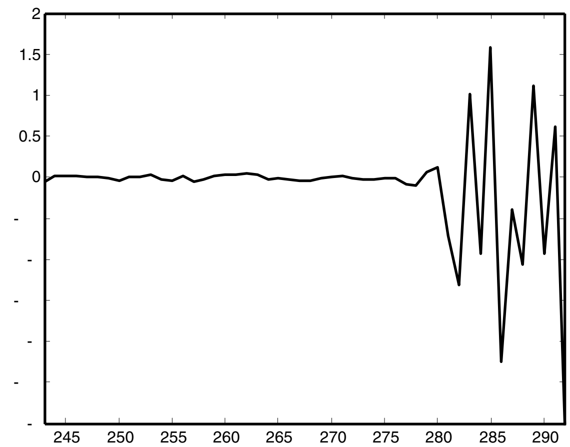

Fig.2 the online fault prediction results using incremental learning LS-SVM

From Fig.2, fault is already predicted by incremental learning LS-SVM, when n is equal to 280, perfect prediction ability is obtained by incremental learning LS-SVM.

V. CONCLUSIONS

In this paper, incremental learning LS-SVM is presented to solve the nonlinear time series prediction online. Two different time series models are used as the evaluation of the prediction power of the incremental learning LS-SVM. Simulation indicates that this method is effective. The incremental learning LS-SVM can be used for time series prediction online because the training of LS-SVM is in a way of incremental chunk avoiding computing large-scale matrix inverse but maintaining the precision.

VI. REFERENCES

[1] Schroder D, Hintz C, Rau M, "Intelligent Modeling, Observation,and Control for Nonlinear Systems Mechatronics", IEEE Transactions on ASME, vol.6, No.2, pp.122-131, 2001.

[2] Norgaard M, Ravn O, Poulsen N K, "NNSYSID and NNCTRL Tools for System Identification and Control with Neural Networks", Computing & Control Engineering Journal, vol.12, No.1, pp.29-36,2001.

[3] J.A.K. Suykens, J. Vandewalle, "Least Squares Support Vector Machine Classifiers", Neural Processing Letters, vol. 9, no. 3, pp.293-300, 1999.

[4] J.A.K. Suykens, J. De Brabanter, L. Lukas, J. Vandewalle, "Weighted Least Squares Support Vector Machines: robustness and sparse approximation", Neurocomputing, 48, pp. 85-105, 2002.

[5] V. Vapnik, *The Nature of Statistical Learning Theory*, Springer, New York, 1995.

[6] V. Vapnik, *Statistical Learning Theory*, Wiley, New York, 1998.

[7] M.Y. Ye and X.D. Wang, "Chaotic Time Series Prediction Using Least Squares Support Vector Machines", Chinese Physics, vol. 13 pp. 454-458, Apr. 2004.

[8] Platt, J.C., "Fast Training of Support Vector Machines Using Sequential Minimal Optimization. Advances in Kernel Methods-Support Vector Learning", The MIT Press, Cambridge, Massachusetts, 1999, pp.185-208.

[9] Zhili Zhang, Hanggeng Guo, "Web Prediction Using Online Support Vector Machine", Proceedings of the17th IEEE international Conference on Tools with Artificial Intelligence.