Vol 13 No 4, April 2004
1009-1963/2004/13(04)/0454-05

Chinese Physics

© 2004 Chin. Phys. Soc.
and IOP Publishing Ltd

Chaotic time series prediction using least squares support vector machines*

Ye Mei-Ying(叶美盈)[a)†] and Wang Xiao-Dong(汪晓东)[b)]

[a)] *College of Mathematics and Physics, Zhejiang Normal University, Jinhua* 321004, *China*

[b)] *College of Information Science and Engineering, Zhejiang Normal University, Jinhua* 321004, *China*

(Received 10 July 2003; revised manuscript received 1 August 2003)

We propose a new technique of using the least squares support vector machines (LS-SVMs) for making one-step and multi-step prediction of chaotic time series. The LS-SVM achieves higher generalization performance than traditional neural networks and provides an accurate chaotic time series prediction. Unlike neural networks' training that requires nonlinear optimization with the danger of getting stuck into local minima, training LS-SVM is equivalent to solving a set of linear equations. Thus it has fast convergence. The simulation results show that LS-SVM has much better potential in the field of chaotic time series prediction.

Keywords: chaotic time series, time series prediction, support vector machines
PACC: 0545

1. Introduction

Time series prediction is a very important practical problem with a diverse range of applications from economic and business planning, inventory and production control, weather forecasting, signal processing and control.[1] The problem of predicting the values of one variable in a future point in time is a time series prediction problem. The chaotic time series prediction can be looked upon as one of its varieties. Since chaotic systems depend sensitively on initial conditions, the trajectories starting at two similar state vectors will diverge exponentially, which leads to the apparent unpredictability of chaotic systems. However, a great deal of research effort has shown that reliable short-term prediction can still be achieved.

Most traditional time series models, such as autoregressive, moving averaging and autoregressive moving averaging, are of linear type, but there are also many nonlinear time series prediction problems in practical situations. The artificial neural network (ANN), as a powerful tool of nonlinear processing, has already been used in nonlinear time series modelling and chaotic control in recent years.[2−14] In the area of chaotic time series forecasting, the most popular neural network model is the feed-forward neural network, in which the back propagation (BP) algorithm is usually used, due to its simple architecture yet powerful problem-solving ability. However, the feed-forward neural network based on BP algorithm suffers from drawbacks like the multiple local minima problem, the choice of the number of hidden units and the danger of over fitting.[15]

In this paper, we explore an additional possibility, namely the use of least squares support vector machines (LS-SVMs)[16−18] for predicting chaotic time series. LS-SVM is established based on the structural risk minimization principle rather than minimizing the empirical error commonly implemented in the neural networks. LS-SVM achieves higher generalization performance than traditional neural network in solving these machine learning problems. Another key property is that unlike neural networks' training that requires nonlinear optimization with the danger of getting stuck into local minima, training LS-SVM is equivalent to solving a set of linear equations. Consequently, the solution of LS-SVM is always unique and globally optimal. In order to demonstrate the predictive capabilities of LS-SVM, experiments were carried out on the Makey–Glass equation.[19] In all cases, the LS-SVM was able to find an appropriate representation to internally encode useful information.

*Project supported by the Zhejiang Provincial Natural Science Foundation, China (Grant No 602145).
†E-mail: ymy@mail.zjnu.net.cn

2. Problem description

A time series is an array of values belonging to subsequent samples usually coming from a chaotic dynamic system's output. It is assumed that neither the state of the dynamic system is measurable nor the equation describing its state is known. If the dynamic system is deterministic, we can try to predict the time series by reconstructing the state space. The object of our work is building an estimate function for the system's transfer function only using its output. Many conventional regression techniques can be used to solve problems of estimating function. In this investigation, we concentrate on the LS-SVM, a novel regression technique developed by Suykens *et al.*[16−18]

Mark a chaotic time series as $\{x_t\}$, its mathematical model can be described by

$$x_{t+p} = f(x_t, x_{t-\tau}, \ldots, x_{t-(m-1)\tau}). \tag{1}$$

Here $p = 1$ means one-step prediction, and $p > 1$ means multi-step prediction; τ is the time delay. In this paper, both one-step prediction and multi-step prediction of $\tau = 6$ are analysed. The chaotic time series prediction can be stated as a numerical problem: split the time series $\{x_t\}$ into windows $x_t, \ldots, x_{t-(m-1)}$ of size m, then find a good estimate for the function $f : R^m \to R$ such that

$$x_{t+p} = f(x_t, x_{t-1}, \ldots, x_{t-(m-1)}). \tag{2}$$

We try applying LS-SVM to estimate the unknown function $f(\cdot)$ in Eq.(2).

3. LS-SVM predictor

LS-SVM is a new technique for regression. The LS-SVM predictor is trained using a set of time series historic values as inputs and a single output as the target value. The LS-SVM method of performing time series prediction can be called the sliding window technique as the historic values input slides over the full training set.

In the following, we briefly introduce LS-SVM, which can be used for chaotic time series prediction.

Consider a given training set of N data points $\{x_k, y_k\}_{k=1}^N$ with input data $x_k \in R^n$ and output $y_k \in R$. In feature space SVM models take the form

$$y(x) = w^{\mathrm{T}}\varphi(x) + b, \tag{3}$$

where the nonlinear mapping $\varphi(\cdot)$ maps the input data into a higher dimensional feature space. Note that the dimension of w is not specified (it can be infinite dimensional). In LS-SVM for function estimation the following optimization problem is formulated:

$$\min J(w, e) = \frac{1}{2} w^{\mathrm{T}} w + \gamma \frac{1}{2} \sum_{k=1}^{N} e_k^2, \tag{4}$$

subjecting to the equality constraints

$$y(x) = w^{\mathrm{T}}\varphi(x_k) + b + e_k, k = 1, \ldots, N. \tag{5}$$

Important differences from standard SVM[20,21] are the equality constraints and the squared error term, which greatly simplifies the problem.

The solution is obtained after constructing the Lagrangian

$$\begin{aligned} &L(w, b, e, \alpha) \\ =&J(w, e) - \sum_{k=1}^{N} \alpha_k \{w^{\mathrm{T}}\varphi(x_k) + b + e_k - y_k\} \end{aligned} \tag{6}$$

with Lagrange multipliers α_k. The conditions for optimality are

$$\begin{cases} \dfrac{\partial L}{\partial w} = 0 \to w = \displaystyle\sum_{k=1}^{N} \alpha_k \varphi(x_k), \\ \dfrac{\partial L}{\partial b} = 0 \to \displaystyle\sum_{k=1}^{N} \alpha_k = 0, \\ \dfrac{\partial L}{\partial e_k} = 0 \to \alpha_k = \gamma e_k, \\ \dfrac{\partial L}{\partial \alpha_k} = 0 \to w^{\mathrm{T}}\varphi(x_k) + b + e_k - y_k = 0, \end{cases} \tag{7}$$

for $k = 1, \ldots, N$. After elimination of e_k and w the solution is given by the following set of linear equations:

$$\begin{bmatrix} 0 & \mathbf{1}^{\mathrm{T}} \\ \mathbf{1} & \varphi(x_k)^{\mathrm{T}}\varphi(x_l) + \gamma^{-1} I \end{bmatrix} \begin{bmatrix} b \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ y \end{bmatrix}, \tag{8}$$

where $y = [y_1; \ldots; y_N]$, $\mathbf{1} = [1; \ldots; 1]$, $\alpha = [\alpha_1; \ldots; \alpha_N]$ and the Mercer condition

$$\Psi(x_k, x_l) = \varphi(x_k)^{\mathrm{T}}\varphi(x_l), \quad k, l = 1, \ldots, N \tag{9}$$

has been applied. This finally leads to the following LS-SVM model for function estimation:

$$y(x) = \sum_{k=1}^{N} \alpha_k \Psi(x, x_k) + b, \tag{10}$$

where α_k, b are the solution to the linear system, $\Psi(\cdot, \cdot)$ represents the high-dimensional feature spaces that is nonlinearly mapped from the input space

x. The LS-SVM approximates the function using Eq.(10).

The choice of the kernel function $\Psi(\cdot,\cdot)$ has several possibilities. In this work, the radial basis function (RBF) is used as the kernel function of the LS-SVM because RBF kernels tend to give good performance under general smoothness assumptions. Consequently, they are especially useful if no additional knowledge of the data is available. For RBF kernels one has

$$\Psi(x_k, x_l) = \exp(-\|x - x_k\|_2^2/\sigma^2), \qquad (11)$$

where σ is a positive real constant. Note that in the case of RBF kernels, one has only two additional tuning parameters σ in Eq.(11) and γ in Eq.(4), which is less than for standard SVM.

4. Simulation and results

In this section, we present one example showing the effectiveness of using LS-SVM for chaotic time series forecasting. In this paper, we use data sets generated by the Mackey–Glass differential-delay equation. All the data sets are partitioned into two parts according to the time sequence. The first part is used for training, the second part for testing.

We generate a time series by numerically integrating the Mackey–Glass time-delay differential equation

$$\frac{\mathrm{d}x(t)}{\mathrm{d}t} = -bx(t) + \frac{ax(t-\Delta)}{1 + x^{10}(t-\Delta)} \qquad (12)$$

with parameters $a = 0.2$, $b = 0.1$, $\Delta = 17$ and initial conditions x_0=1.2 and x_t=0 for $t < 0$. Equation (12) was originally introduced as a model of blood cell regulation.[19] The time series data were obtained by applying the conventional fourth-order Runge–Kutta algorithm to numerically solve Eq.(12). This time series is chaotic, and so there is no clearly defined period. The series will neither converge nor diverge, and the trajectory is highly sensitive to initial conditions. The prediction of future values of this series is a benchmark problem.

From the Mackey–Glass time series $\{x_t\}$, we obtained training and test sets. Following the conventional settings for predicting the Mackey–Glass time series, we set $m = 4$ and a sampling rate τ=6 $(x_t, x_{t-\tau}, \ldots, x_{t-(m-1)\tau})$, where t=118 to t=1117. The input data for LS-SVM is a four-dimensional vector and the output data is a one-dimensional vector. The first 400 pairs were used for training, the remaining 700 were used for testing the model of prediction.

The problem can be formulated as: given values $x_t, x_{t-1}, \ldots, x_{t-(m-1)}$; determine x_{t+p}, where m and p are fixed positive integers and t is the series index. The selections ranged from one-step prediction through four-step ahead prediction and finally eight-step ahead prediction (i.e. p=1, p=4, and p=8 respectively). All these simulations used four previous values (i.e. $m = 4$).

The results are presented in terms of the accuracy of the one-step-ahead prediction measured on the root mean squared error metric (RMSE)

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(x_t - \hat{x}_t)^2}, \qquad (13)$$

where x_t is the true output and $\hat{x}_t$ is the prediction output.

The simulations for a four-input single-output problem for one-step or multi-step prediction can be implemented on the LS-SVM. In our simulations, the tuning parameters σ in Eq.(11) and γ in Eq.(4) are $\sigma = 600$, $\gamma = 8000$, respectively.

To show clearly the predictive results of the LS-SVM for making one-step and multi-step prediction of chaotic series, only one is plotted for each five points compared with the actual values in the following numerical simulation results (Figs.1(a), 2(a) and 3(a)).

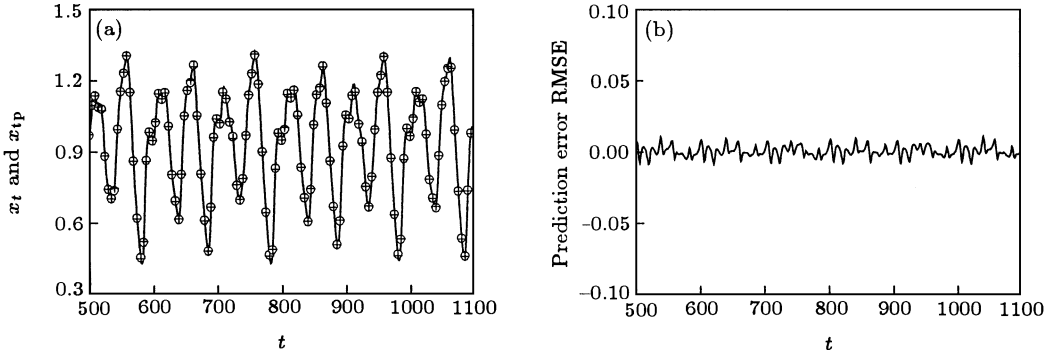


Fig.1. (a) Prediction results of the Mackey–Glass series by the LS-SVM with p=1 and RMSE=0.0038 (prediction values x_{tp}—○, true values x_t—+; (b) Prediction error between predicted and actual series by the LS-SVM with p=1 and RMSE=0.0038.

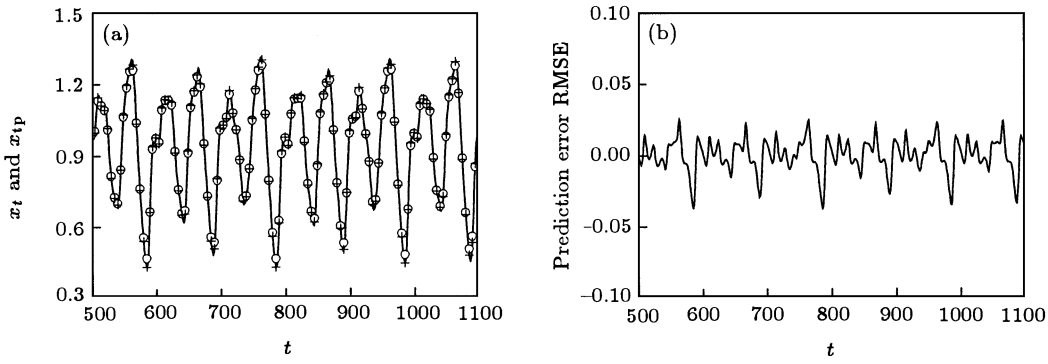


Fig.2. (a) Prediction results of the Mackey–Glass series by the LS-SVM with p=4 and RMSE=0.0113 (prediction values x_{tp}—○, true values x_t—+; (b) Prediction error between predicted and actual series by the LS-SVM with p=4 and RMSE=0.0113.

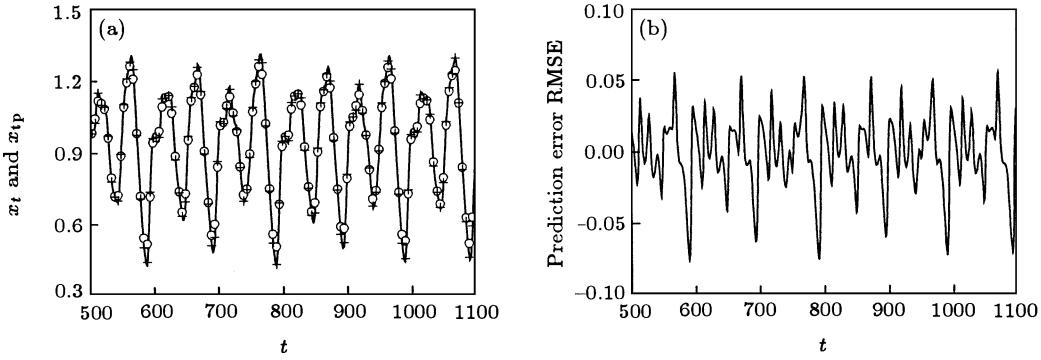


Fig.3. (a) Prediction results of the Mackey–Glass series by the LS-SVM with p=8 and RMSE=0.0245 (prediction values x_{tp}—○, true values x_t—+; (b) Prediction error between predicted and actual series by the LS-SVM with p=8 and RMSE=0.0245.

The accuracy of one-step prediction is illustrated in Fig.1(a), which shows that the predicted and actual points of the series are largely indistinguishable (note this graph was obtained for one-step prediction using four previous values). The difference between the predicted and actual results, in terms of the prediction error, is illustrated in Fig.1(b). Figures 2 and 3 illustrate the prediction results of the Mackey–Glass series and the prediction error between predicted and actual series by the LS-SVM for p=4 and p=8.

The proposed LS-SVM was readily implemented in Matlab. To compare the performance of ANN and LS-SVM, the ANNs using two different training algorithms, the gradient descent (GD) and Levenberg–Marquardt (LM) algorithms, were also implemented. We trained the ANN with 4–10–1 (four input units, 10 nonlinear hidden units, and one linear output unit) architecture. The ANN was trained to be a predictor of the Mackey–Glass time series. The training was stopped after 1000 epochs. We computed the root mean squared error of multi-step predictions using ANN and LS-SVM, respectively. The results are listed in Table 1. In general, the gradient descent (GD) algorithms are too slow for practical problems. The LM algorithms use numerical optimization techniques for ANN training. On function approximation problems, for ANN that contain up to a few hundred weights, the LM algorithms [22] will have fast convergence and can obtain lower root mean squared error than any of the other algorithms. Based on these results in Table 1, we see that the LS-SVM has the fastest convergence and provides an accurate chaotic time series prediction.

Table 1. Performance of LS-SVM and ANN for Mackey–Glass series prediction.

Predicting models	training time /s	RMSE (4 step prediction)	RMSE (8 step prediction)	RMSE (12 step prediction)
LS-SVM	1.57	0.0113	0.0245	0.0496
ANN (LM Algorithms)	28.11	0.0134	0.0525	0.0875
ANN (GD Algorithms)	13.29	0.4043	0.6306	0.8464

5. Conclusions

We have presented a new technique of using the LS-SVM for making one-step and multi-step accurate prediction of chaotic time series. From the above simulation results, one can see that this method has the following advantages. Firstly, the LS-SVM achieves higher generalization performance than traditional neural networks and provides an accurate chaotic time series prediction. Secondly, unlike neural networks' training that requires nonlinear optimization with the danger of getting stuck into local minima, training LS-SVM is equivalent to solving a set of linear equations. Thus it has fast convergence. The simulation results show that LS-SVM has much better potential in the field of chaotic time series prediction.

It should be pointed out that, although the processes are focused on the Mackey–Glass differential-delay equation, we believe that the proposed method can be used to many other complex chaotic time series prediction problems.

References

[1] Maguire L P *et al* 1998 *Inform. Sci.* **112** 125
[2] Yuan J and Xiao X C 1997 *Acta Phys. Sin.* **46** 2095 (in Chinese)
[3] Guan X P *et al* 2001 *Acta Phys. Sin.* **50** 2112 (in Chinese)
[4] Thiesing F M and Vornberger O 1997 *IEEE* (*ed*) *Proc. ICNN'97*, 2125
[5] Zhang S *et al* 2003 *Chin. Phys.* **12** 594
[6] Gençay R and Liu T 1997 *Physica* D **108** 119
[7] Gençay R 1994 *Phys. Lett.* A **187** 397
[8] Liu D *et al* 2003 *Acta Phys. Sin.* **52** 531 (in Chinese)
[9] Tan W *et al* 2002 *Acta Phys. Sin.* **51** 2463 (in Chinese)
[10] Tan W *et al* 2003 *Acta Phys. Sin.* **52** 795 (in Chinese)
[11] Ren H P and Liu D 2002 *Acta Phys. Sin.* **51** 982 (in Chinese)
[12] He G G and Cao Z D 2001 *Acta Phys. Sin.* **50** 2103 (in Chinese)
[13] Guan X P and Fan Z P 2001 *Acta Phys. Sin.* **50** 1670 (in Chinese)
[14] Zhang J S 2000 *Chin. Phys.* **9** 408
[15] Bishop C M 1995 *Neural Networks for Pattern Recognition* (Oxford: Clarendon)
[16] Suykens J A K and Vandewalle J. 1999 *Neural Process. Lett.* **9** 293
[17] Suykens J A K 2000 *Neural Network World* **10** 29
[18] Suykens J A K *et al* 2002 *Neurocomputing* **48** 85
[19] Mackey M C and Glass L 1977 *Science* **197** 287
[20] Vapnik V N 2000 *The Nature of Statistical Learning Theory* (New York: Springer)
[21] Vapnik V N 1998 *Statistical Learning Theory* (New York: Wiley)
[22] Hagan M T and Menhaj M 1994 *IEEE Trans. NN* **6** 989