http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 5 (2): 353-357, 2006
ISSN 1812-5638

Time Series Prediction Based on Support Vector Regression

Shukuan Lin, Guoren Wang, Shaomin Zhang and Jingyin Li
College of Information and Engineering, Northeastern University, Shenyang, 110004, China

Abstract: This study introduces Support Vector Regression (SVR) model and predicts on time series based on SVR, proposing several new approaches which improve traditional SVR in order to enhance prediction accuracy. In terms of the feature of time series, the approaches give different weights to different history data at different times and make the punishing coefficient C and non-sensitive loss ε in optimization objective function of SVR adjustable along with different sample data. The experimental results show that the proposed approaches improve the prediction precision and testify the validity of these approaches.

Key words: Support vector regression, time series prediction, punishing coefficient C, non-sensitive loss ε, quadratic programming optimization

INTRODUCTION

SVM learning method proposed by Vapnik is based on statistic learning theory. It has attracted much attention in academic community in recent years and is becoming a new research hotspot in the machine learning field after Neural Network (NN). SVM is based on structural risk minimization principle, which makes the method have better generalization performance. The other important feature owing to the Mercer condition of kernel[1] makes corresponding optimization problem a protruding one. Therefore there is no local least spot. This ensures the received solutions must be global optimal ones. These features make SVM a powerful tool solving classification and regression problem of SVR.

In order to improve generalization performance and avoid local optimal solution, the study adopts SVR to predict on time series. Furthermore, the study improves the traditional SVR and proposes four approaches to improve the prediction accuracy.

In order to enhance the prediction precision on time series, the feature of time series must be analyzed. In time series, the value at predicted point is influenced by history data at its former times, but the influences of these history data are not even. The nearer to the predicted point the history data is, the bigger its impact on prediction result is. On the contrary, the farther from the predicted point the data is, the smaller its impact on prediction result is. So if the different influences of history data at different times are considered, the prediction precision can be improved. Based on the above thought, this paper proposes four approaches to improve the prediction accuracy, which consider the different influences of different history data. That is, (1) attenuating historical sample data gradually along with the distance to predicted point. (2) Making punishing coefficient C adjustable which is a constant in traditional SVR. The farther from the predicted point the sample data X_i is, the smaller its corresponding C_i is and visa versa. (3) Making non-sensitive loss ε adjustable which is a constant in traditional SVR, too. It is different from punishing coefficient C that the farther from the predicted point the sample data X_i is, the bigger its corresponding ε_i is and visa versa. (4) Combining the above approaches together. The experimental results show that all the four approaches above improve the prediction precision comparing with the traditional SVR and that the proposed approach (4) is the best and obtains very satisfactory prediction result.

SUPPORT VECTOR MACHINE

Support Vector Machine is based on least structure risk principle. It transforms non-linear problems into linear ones in multi-dimensional space by using kernel functions. In order to learn a non-linear relation using a linear learner, a non-linear mapping must be needed to map original data to a new feature space. Then the linear learner is used to classify or regress in the new feature space. So the final classifying or regression function can be a form as follows[1]:

$$f(x) = \sum_{i=1}^{l} w_i \phi_i(x) + b \quad (1)$$

Corresponding Author: Shukuan Lin, College of Information and Engineering, Northeastern University, Shenyang, 110004, China

Where, ø: X→F is a mapping from original input space to a certain new feature space, l is the number of samples. This means it should be divided into two steps to build a non-linear learner: firstly, the data is transformed into a new feature space F, then a linear learner is used to classify or regress in this new feature space. By expressing the linear learner in multi-dimensional space as dual form, formula (1) can be denoted into the form of inter product of testing data and training data[1], that is:

$$f(x)=\sum_{i=1}^{l}\alpha_i y_i\ \phi(x_i)\ \phi(x)\ +b \tag{2}$$

Let the kernel function K(x, z) = <ø(x).ø (z)>. Then the linear learner can be got by computing the kernel function for l times:

$$f(x)=\sum_{i=1}^{l}\alpha_i y_i K(x_i\ \ x)+b \tag{3}$$

Importing appropriate kernel functions not only keeps away from the problem of building mapping from original space to new feature space, but also transforms non-linear problems in original space into linear problems in new feature space. The parameters in formula (3) can be got by transforming the problem into a protruding quadratic programming optimization problem.

SUPPORT VECTOR REGRESSION MODEL

SVM considers empirical risk and believing range at the same time[2]. So the solution of the problem can be gained by minimizing the following objective function:

$$Q(w)=\tfrac{1}{2}\|w\|^2+C\ \ R_{emp}(f) \tag{4}$$

where, the first term makes the function flatter in order to improve generalization performance. The second term is empirical error. The constant C is called punishing coefficient and expresses the punishing extent to the samples with error. It can also realize the tradeoff between believing range and empirical risk[3,4]. Common loss functions $R_{emp}(f)$ include[5] quadratic function, Huber function, Laplace function, ε-non-sensitive loss function and so on. The ε-non-sensitive loss function is widely applied because of its excellent quality. ε-non-sensitive loss function $L^{\varepsilon}(x,y,f)$ is defined as below:

$$\begin{aligned}L^{\varepsilon}(x,y,f)&=|y-f(x)|_{\varepsilon}\\&=\max(0,|y-f(x)|-\varepsilon)\end{aligned} \tag{5}$$

Where, f is a real-value function on domain X, x ∈ X and y ∈ R.

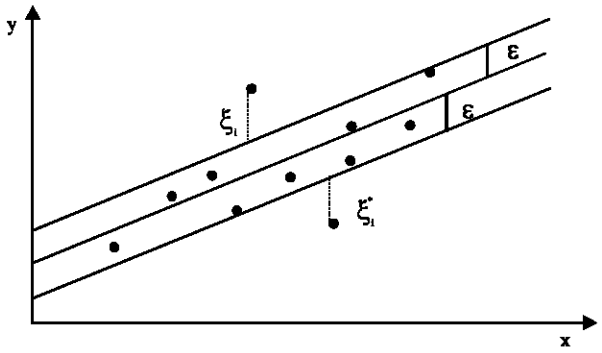

Fig. 1: Non-sensitive band of linear regression problem

Obviously, when $|y-f(x_i)|=|y-(w.x_i)-b|=\varepsilon$ (i=1, 2, …, l), the sample falls into the band between f(x)+ε and f(x)-ε (Fig.1), the error is zero. Because the above condition cannot be satisfied fully, slack genes ξ_i and ξ_i^* are imported. So if a linear ε-non-sensitive loss function is adopted, the optimization objective function is turned into[1] :

$$\begin{aligned}&\text{minimize } \tfrac{1}{2}\|w\|^2+C\sum_{i=1}^{l}(\xi_{si}+\xi_{si}^*)\\&\text{subject to } (w\ \ x_i\ +b)-y_i\le\varepsilon+\xi_{si}\\&y_i-(w\ \ x_i\ +b)\le\varepsilon+\xi_{si}^*\\&\xi_{si},\xi_{si}^*\ge 0\qquad (i=1,2,\cdots,l)\end{aligned} \tag{6}$$

Where, slack variance ξ_{si} is for exceeding ε above objective value; ξ_{si}^* is for exceeding ε below objective value.

This is a protruding quadratic programming optimization problem. So the following Lagrange function is introduced:

$$\begin{aligned}L(w,b,\xi,\xi^*,\alpha,\alpha^*,\gamma,\gamma^*)&=\frac{1}{2}\|w\|^2\\&+C\sum_{i=1}^{l}(\xi_{si}+\xi_{si}^*)-\sum_{i=1}^{l}\alpha_i[\xi_{si}+\varepsilon-y_i+f(x_i)]\\&-\sum_{i=1}^{l}\alpha_i^*\ \xi_{si}^*+\varepsilon+y_i-f(x_i)\ -\sum_{i=1}^{l}(\xi_{si}\gamma_i+\xi_{si}^*\gamma_i^*)\end{aligned} \tag{7}$$

where α, α^*, γ, γ^* is Lagrange multiplier.
Formula (7) can be transformed into dual form, so the Lagrange multiplier α, α^* can be obtained by maximizing the following quadratic function:

$$\begin{aligned}&\text{maximize }\sum_{i=1}^{l}(\alpha_i^*-\alpha_i)y_i-\varepsilon\sum_{i=1}^{l}(\alpha_i^*+\alpha_i)-\\&\frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i^*-\alpha_i)(\alpha_j^*-\alpha_j)\ x_i\ \ x_j\end{aligned}$$

$$\text{subject to } 0 \le \alpha_i, \alpha_i^* \le C \quad (i = 1,2,\cdots,l)$$

$$\sum_{i=1}^{l} (\alpha_i^* - \alpha_i) = 0 \tag{8}$$

By importing kernel function $K(x_i,x_j)$ to substitute the inter product $\langle x_i, x_j \rangle \mu$, the above problem is transformed into the following form:

$$\text{maximize} \sum_{i=1}^{l} (a_i^* - a_i) y_i - e \sum_{i=1}^{l} (a_i^* + a_i) - \frac{1}{2} \sum_{i,j=1}^{l} (a_i^* - a_i)(a_j^* - a_j) K(x_i \times x_j)$$

$$\text{subject to } 0 \le \alpha_i, \alpha_i^* \le C \quad (i = 1,2,\cdots,l) \tag{9}$$

The corresponding prediction function is changed into:

$$f(x,\alpha_i,\alpha_i^*) = \sum_{i=1}^{l} (\alpha_i - \alpha_i^*) K(x_i,x) + b \tag{10}$$

Where, α_i, α_i^* can be received by solving the quadratic optimization problem of expression (9). Variance b can be gained from the restrained condition.

Summing up, a SVR problem can be transformed into a protrude quadratic programming optimization problem by importing Lagrange function of objective function and its dual form and obtain prediction function by solving the optimization solution.

TIME SERIES PREDICTION BASED ON SVR

Time series prediction is to predict value y(t+k) of future time t+k (k>0) on the basis of real history data of time series {y(t), y(t-1), …, y(t-m+1)} and corresponding variances which influence the time series[6], that is, to find the relationship between future value y(t+k) and history data {y(t), y(t-1), …, y(t-m+1)}, variance $\{x_1(t), x_2(t), \ldots, x_n(t)\}$. When k=1, it is called one-step prediction. When k>1, it is called direct many-step prediction. Parameter m is called steplength. Now consider one-step prediction. When training regression model, the following sample-pair can be formed by inputting vector {y(t), y(t-1), …, y(t-m+1), $x_1(t)$, $x_2(t)$, …, $x_n(t)$} corresponding to outputting value y(t+1), inputting vector {y(t+1), y(t), …, y(t-m+2), $x_1(t+1)$, $x_2(t+1)$, …, $x_n(t+1)$} corresponding to outputting value y(t+2) and so on.

In sample data, the importance of history data at different times to prediction result is different. The effect of farther data from predicted point on prediction result is weaker than that of nearer data. So this paper proposes giving different weights to history data at different times. The weight of farther data is smaller, but that of nearer data is bigger. The weights will be attenuated exponentially. Here, in order to predict value y(t+1) at time t, input vector will be changed into the following form:

$$y(t), (1-d)y(t-1), (1-d)^2 y(t-2), \cdots, (1-d)^{m-1} y(t-m+1), x_1(t), x_2(t), \cdots, x_n(t) \tag{11}$$

where, d is attenuating coefficient.

In traditional SVR, punishing coefficient C, which is constant, expresses the punishing degree to empirical error. Because of the feature of time series, the nearest sample to predicted point is the most important, the farthest sample is the least important[7]. So nearer samples to predicted point should be paid more attention to (including nearer errors). So this paper improves the objective function of traditional SVR and makes punishing coefficient C adjustable. For farther sample data X_i, corresponding value C_i is smaller. On the contrary, for nearer sample data X_j (j>i), corresponding value C_j is bigger. Value C is attenuated exponentially (attenuating coefficient is d) along with the distance to the predicted point.

Similarly, non-sensitive loss ε is fixed in traditional SVR, too. This paper also proposes adjustable non-sensitive loss ε_i (i=1,2,…,l) on the basis of the feature of time series. Nearer sample data to predicted point needs smaller ε and farther sample data needs bigger ε because a smaller ε means a bigger slack gene ξ and means that nearer sample data is paid more attention to according to (6) and vice versa. Value ε is increased exponentially (increasing coefficient is s) along with the distance to predicted point.

Here, the protruding quadratic programming optimization problem (9) is transformed into the following form:

$$\text{maximize} \sum_{i=1}^{l} (\alpha_i^* - \alpha_i) y_i - \sum_{i=1}^{l} \varepsilon_i (\alpha_i^* + \alpha_i) - \frac{1}{2} \sum_{i,j=1}^{l} (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j) K(x_i \; x_j)$$

$$\text{subject to } 0 \le \alpha_i, \alpha_i^* \le C_i \quad (i = 1,2,\cdots,l)$$

$$\sum_{i=1}^{l} (\alpha_i^* - \alpha_i) = 0 \tag{12}$$

Lagrange multiplier α and α^* can be gained by solving the quadratic optimization problem of expression (12), thereby finding regression function (10).

EXPERIMENTAL RESULTS AND DISCUSSION

This section verifies the above proposed approaches with emulate data provided by function sinc[8] which is defined as follows:

$$f(t) = \sin c(t + a) + b \tag{13}$$

Let a=-4, b=0.5 without losing universality. Make 81 points in the interval t ∈ [0, 8]. The former 66 points are regarded as training sample data. The latter 15 points are regarded as testing data. This is a typical time series problem[5]. Here, SVR method is used to predict on this time series. By experimental contrast, it is found that higher prediction accuracy is achieved when punishing coefficient C=10000, non-sensitive loss ε=0.001, regression steplength m=5 meanwhile adopting polynomial kernel function as below (parameter p=3):

$$K(x, y) = (x \ \ y + 1)^p \tag{14}$$

In order to evaluate prediction accuracy, the following formula is used to compute the prediction error[9]:

$$M_{APE} = \frac{1}{n} \sum_{i=1}^{n} \left| \frac{t_i - t_i^*}{t_i} \right| \tag{15}$$

Where, n is the number of predicted points, t_i is the real value of a certain point i. t_i^* is its predicted value.

The prediction error based on the traditional SVR M=0.0724 (Table 1) with the above parameters. The prediction result is shown in Fig. 2.

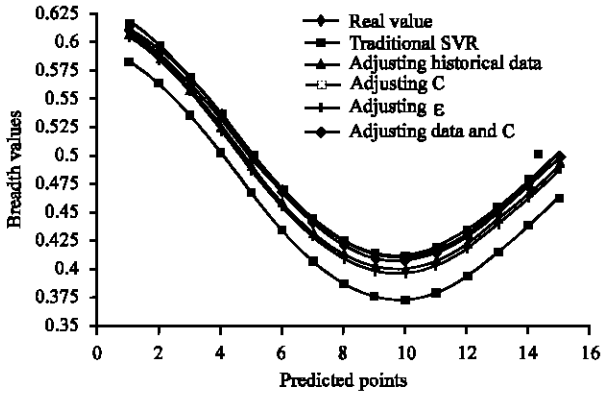

Fig. 2: Comparison of prediction results of proposed approaches and traditional SVR on the latter 15 points

Table 1: Comparison of average errors among traditional SVR and several proposed approaches

Traditional SVR	Attenuating historical data	Adjusting C	Adjusting ε	Attenuating historical data and adjusting C
0.0724	0.0205	0.0057	0.0250	0.0048

In correspondence with the traditional SVR above, the following four experiments verify the validity of the proposed approaches of this study, respectively.

(1) Historical sample data is attenuated gradually along with the distance to predicted point.

Keep punishing coefficient C, non-sensitive loss ε and kernel function unchanged. A sample pair $\{(t_i, t_{i-1}, t_{i-2}, t_{i-m+1}), t_{i+1}\}$ is changed into the following form:

$$\left\{\left(t_i, (1-d)t_{i-1}, (1-d)^2 t_{i-2}, \cdots, (1-d)^{m-1} t_{i-m+1}\right), t_{i+1}\right\} \tag{16}$$

Where, d is attenuating coefficient. It is found that better result is obtained when d=0.835.

The prediction error based on this approach M = 0.0205 which is much smaller than the traditional SVR (Table 1). The prediction result is shown in Fig. 2. From this experiment, it is found that attenuating coefficient d is bigger, which explains that only recent several history data has stronger effect on the predicted value.

(2) Let non-sensitive loss ε adjustable

Keep punishing coefficient C, kernel function and sample data unchanged. Let the initial value of non-sensitive loss ε_0=0.001. The non-sensitive loss ε_i corresponding with sample X_i is strengthened along with the distance from X_i to predicted point. In the course of the experiment, it is found that better result is obtained when the increasing coefficient s = 0.100.

Its prediction error M=0.0250 which is smaller than traditional SVR (Table 1). The prediction result is shown in Fig. 2.

(3) Let punishing coefficient C adjustable

Keeping non-sensitive loss ε, kernel function and sample data unchanged. Let the initial value of punishing coefficient C_0=10000. The punishing coefficient C_i corresponding with sample X_i is attenuated along with the distance from X_i to predicted point. By experiment, it is found that better result is obtained when the attenuating coefficient d=0.205. Its prediction error M=0.0057 which is the best in the three approaches (Table 1). The prediction result can be seen in Fig. 2.

(4) The three approaches above are combined together.

Altogether there are four kinds of combinations which are the combination of (1) and (2), that of (1) and (3), that of (2) and (3), that of (1), (2) and (3). By experimental contrast, it is found that the combination of (1) and (3), whose error M=0.0048 (Table 1), is the best in all the four approaches. This explains that it can obtain satisfactory result of time series prediction to attenuate historical data and punishing coefficient C together along with the distance to predicted point. The prediction result can be seen in Fig. 2.

CONCLUSIONS

There are many ways to predict time series, among of which SVR is adopted widely because of its better generalization performance and its higher accuracy. This study predicts time series basing on SVR and improves traditional SVR. In terms of the feature of time series, the paper proposes giving different weights to history data at different times and solving the objective function of quadratic programming optimization with adjustable punishing coefficient C and non-sensitive loss ε. The experiments show that the proposed approaches enhance prediction accuracy, proving the validity of these methods, which provide references on time series prediction. Of course, some parameters such as the initial values of punishing coefficient C and non-sensitive loss ε, steplength m, attenuating coefficient d, increasing coefficient s and kernel function may be changed according to reality. In the course of experiments, it is found that the initial values of punishing coefficient C and non-sensitive loss ε are sensitive to prediction results. Here, the optimal values of C and ε are produced by experimental contrast in this study. How to choose more reasonable initial values of punishing coefficient C and non-sensitive loss ε is a very important aspect in future research.

ACKNOWLEDGMENTS

This research was supported by the National Natural Science Foundation of China(Grant No. 60273039 and 60473074), the Foundation for University Key Teacher and the Teaching and Research Award Program for Outstanding Young Teachers in High Education Institution of Chinese Ministry of Education and Natural Science Foundation of LiaoNing Province and Shenyang City (Grant No. 20042015 and 1041036-1-06-07).

REFERENCES

1. Nello, C. and S.T. John, 2004. An introduction to support vector machines and other kernel-based learning methods. Li G.Z., *et al.* Beijing: Publishing House of Electronics Industry.
2. Zhang, X.G., 2000. Introduction to statistical learning theory and support vector machines. Acta Automatica Sinica, 26: 32-42.
3. Vapnik, V.N., 1995. Nature of Statistical Learning Theory. New York: Springer-Verlag.
4. Vapnic, V.N., 1998. Statistical Learning Theory. New York: Wiley.
5. Li, L.J., Z.S. Zhang and Z.J. He, 2004. Research on condition trend prediction of mechanical equipment based on support vector machine. J. XiAn Jiao Tong University, 38: 230-238.
6. Wang, Y.T., 1999. Research and application of neural network in expert system oriented complex industrial process. Shenyang: Information and Engineering College of Northeastern University.
7. Lin, C.F. and S.D. Wang, 2002. Fuzzy support vector machines. IEEE Trans. Neural Networks, 13: 464-471.
8. Chuang, C.C. and S.F. Su, 2002. Robust support vector regression networks for function approximation with outliers. IEEE Trans. Neural Networks, 13: 1322-1330.
9. Chen, B.J., M.W. Chang and C.J. Lin, 2001. Load forecasting using support vector machine: A study on EUNITE competition 2001.