A Parameter Choosing Method of SVR for Time Series Prediction

Shukuan Lin, Shaomin Zhang, Jianzhong Qiao, Hualei Liu, Ge Yu
College of Information Science and Engineering, Northeastern University, Shenyang 110004, Liaoning, China
linshukuan@ise.neu.edu.cn

Abstract

It is important to choose good parameters in Support Vector Regression (SVR) modeling. Choosing different parameters will influence the accuracy of SVR models. This paper proposes a parameter choosing method of SVR models for time series prediction. In the light of data features of time series, the paper improves the traditional Cross-Validation method, and combines the improved Cross-Validation with ε-weighed SVR in order to get good parameters of models. The experiments show that the method is effective for time series prediction.

Keywords: Parameter choosing, SVR, time series prediction, improved Cross-Validation, ε-weighed.

1. Introduction

Support Vector Machine(SVM) is a new learning method, proposed by Vapnik according to Statistics Learning Theory[1]. It follows the rule of Structural Risk Minimization (SRM) with the characteristics of structure-simple, global optimization, good generalization, and has become new research hotspot in recent years. It was used to solve problems of Pattern Recognition at first. With the introduction of ε-non-sensitive loss function, its use has extended to regression function estimation, nonlinear system discrimination, prediction, and so on, showing better learning performance.

For time series prediction, Support Vector Regression (SVR) has better generalization performance, compared with other learning methods such as Neural Network (NN), which are based on Empirical Risk Minimization. However, when we model and predict time series by SVR, whether the predicted values are accurate or not greatly depends on the parameter choosing of SVR models. Different parameters chosen, the output of model will have obvious difference. Therefore, the parameter choosing of SVR is very important to its predicting result. During practical modeling, there are little effective and applied methods to choose parameters of SVR at present. Some methods which have stronger theoretical basis are not applied because of their complexity. So, the methods based on experience or Cross-Validation are commonly adopted. The parameters chosen by experience are usually coarse and easily get into local extremum. The traditional Cross-Validation method is not very effective for time series prediction[2]. The paper proposes a practical method to choose the parameters of SVR for time series prediction, which combines ε-weighed SVR with improved Cross-Validation and gets better effect.

The reminder of the paper is organized as below: Section 2 introduces related work. Section 3 discusses the parameters of SVR models and their influences on models. Section 4 introduces the traditional Cross-Validation method. The features of time series and its prediction are analyzed in section 5. Section 6 presents our parameter choosing method of SVR models for time series prediction. Section 7 gives the experimental results and analysis. Conclusions are drawn in section 8.

2. Related work

Ref. [3] proposed an iterative algorithm with fixed step length to optimize the kernel parameter of SVM. Ref. [4] optimized the parameters of SVM by minimizing the upper bound of Radius-Margin. Ref. [5] set SVM in the Evidence Framework, choosing its optimal parameters based on the second-class reasoning and the third-class reasoning respectively. All these methods optimize parameters of SVM for pattern classification, not optimizing parameters of SVR for time series prediction. Moreover, their optimizing processes are very complicated. Therefore, it is difficult to apply them into practical applications.

Ref. [6] proposed a grads-descended algorithm to solve the problem of multi-parameter optimization of

978-0-7695-3398-8/08 $25.00 © 2008 IEEE
DOI 10.1109/ICYCS.2008.393

SVR. The algorithm needs to compute the partial derivatives of objective function on the optimized parameters, and specifies step length and direction in the next iteration according to the size and direction of the grads. The method can get the optimal parameters in theory. But in reality, it is not practical because of its complexity. Besides, it cannot get the optimal parameters in the case some partial derivatives on some parameters do not exist.

Some scholars chose parameters of SVM (or SVR) by some optimizing algorithms such as Genetic Algorithm. It needs to define fitness function related to predicting errors, which makes every crossover and aberrance have to set up models repeatedly.

Leave-One-Out method introduced by Ref. [7] is a simple and commonly used method for parameter choosing of SVR. In N samples, it uses N-1 samples as training data, testing the left one sample, and then changes the sample to be tested continually until the procedure is repeated N times. Sum up the number k of misclassified samples. The group of parameters making k/N minimized is the one to be chosen. The method is fussy because it only leaves one sample to be tested and needs to repeat N times. The Cross-Validation is an extended method of Leave-One-Out. Different from Leave-One-Out, it uses a group of samples as the ones to be tested but not only one sample.

The paper improves the traditional Cross-Validation method and combines it with ε-weighed SVR to choose parameters of SVR in the light of the characteristics of time series.

3. The parameters of SVR and their influences

Set the sample space of time series is

$$(\mathrm{x}_1, y_1), (\mathrm{x}_2, y_2), ..., (\mathrm{x}_n, y_n) \in R^m \times R \tag{1}$$

Where x_i is m-dimension vector. SVR imports a function φ to map original data into a new feature space. Therefore, a linear learning machine can be used to learn a nonlinear problem. In this case, the regression prediction function can be expressed as:

$$f(x) = w^T \varphi(x_i) + b \tag{2}$$

Where, w and b can be got by solving the following optimizing problem:

$$\begin{aligned} \min \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}(\xi_i + \xi_i^*) \\ \text{subject to} \quad & (\langle w^T \phi(x_i)\rangle + b) - y_i \le \varepsilon + \xi_i \\ & (\langle w^T \phi(x_i)\rangle + b) - y_i \le \varepsilon + \xi_i \\ & \xi_i, \xi_i^* \ge 0 \quad (i = 1, 2, \cdots, n) \end{aligned} \tag{3}$$

The detail information about SVR can be seen in Ref. [1]. Here, only the contents related to parameters to be optimized are presented. In the objective function (3), the first item makes the function plainer and can improve generalization. The second item is experiential error. The parameter C is called punishing coefficient, which expresses the punishment for experiential error and plays the role of tradeoff between believing risk and experiential error. When C is a small value, the punishment for experiential error is small, which will not get smaller fitting and predicting errors; When C is large, the punishment for experiential error will be over large, which will lead to over-learning and low generalization[8]. So, in order to get accurate predicting results and enhance generalization, the value of parameter C should be in specific range.

In formula (3), ε is the non-sensitive loss, which is a positive constant. If the difference between the predictive value $f(x_i)$ and the real value y_i is less than ε, it is ignored. This is to say, the error is zero. If the difference is more than ε, the error is $|f(x_i) - y_i| - \varepsilon$. The non-sensitive loss ε expresses the expectation for the error between the predictive value and real value. The less ε is, the less demanded error is, and the higher prediction precision is.

By introducing Lagrange function and kernel function, the problem in formula (3) can be transformed into the following quadratic function maximizing problem.

$$\begin{aligned} \max \quad & \sum_{i=1}^{l}(\alpha_i^* - \alpha_i) y_i - \varepsilon \sum_{i=1}^{l}(\alpha_i^* + \alpha_i) \\ & -\frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j) K(x_i, x_j) \\ \text{subject to} \quad & 0 \le \alpha_i, \alpha_i^* \le C \quad (i = 1, 2, \cdots, n) \\ & \sum_{i=1}^{l}(\alpha_i^* - \alpha_i) = 0 \end{aligned} \tag{4}$$

Therefore, the final prediction function will be

$$f(x, \alpha_i, \alpha_i^*) = \sum_{i=1}^{l}(\alpha_i - \alpha_i^*) K(x_i, x) + b \tag{5}$$

In formula (4) and (5), $K(x_i, x_j)$ is kernel function. Here, we only discuss the RBF kernel function, as shown in formula (6).

$$K(x, y) = \exp\left\{-\frac{|x - y|^2}{\sigma^2}\right\} \tag{6}$$

Here, σ is kernel parameter *gamma*. A large number of experiments show that the value of *gamma* influences the performance of SVR models seriously. The larger *gamma* is, the smaller structural risk is, and the more slippery the function curve is, but the bigger experiential risk is. The smaller *gamma* is, the smaller experiential risk is, but the bigger structural risk is.

Over-small *gamma* value will lead to over-fitting. Therefore, parameter *gamma* should be chosen correctly.

In a word, when SVR models are set up, the parameters to be chosen include punishing coefficient C, non-sensitive loss ε and kernel parameter *gamma*. They are important to the accuracy of SVR models.

4. The traditional Cross-Validation

The traditional Cross-Validation divides sample data into n sections with same length. Consecutive n-1 sections serve as training samples to test the left section, which is called a tested section. The tested section changes from the first section to the last section. So, after n times of training and testing, each sample will be tested once, which makes the accurate rate steady and avoids over-fitting. In LIBSVM, a kind of shuffling method was used[9], which throws the order of all samples into confusion and resets them. It divides the reset data into n sections and carries through Cross-Validation.

Cross-Validation is usually combined with Grid-Search. Firstly, the ranges of parameter C, ε and *gamma* must be specified (for example, $C=2^{-1}, 2^{0},\ldots, 2^{12}$; $\varepsilon=2^{-12}, 2^{-11},\ldots, 2^{-1}$; $gamma=2^{-8}, 2^{-7},\ldots, 2^{0}$). Secondly, Cross-Validation is carried out for every group of parameters (C, ε, *gamma*). As shown in Fig.1, samples are divided into 5 sections with same size (The section with "test" will serve as the tested one). The section 5 is tested based on section 1 to 4. The section 1 is tested based on section 2 to 5. The section 2 is tested based on section 3, 4, 5, 1. The section 3 is tested based on section 4, 5, 1, 2. The section 4 is tested based on section 5, 1, 2, 3. And then the average error of tests for 5 times is computed for each group of parameters(C, ε, *gamma*). Finally, the group of parameters which make the average error least will be chosen as the parameters of SVR. So, the objective function of parameter choosing can be expressed as:

$$\min_{C,\varepsilon,gamma} \frac{1}{n}\sum_{i=1}^{n} E_i \qquad (7)$$

Where, n is the number of sections. E_i is the test error of the ith section.

1	2	3	4	5 test
1 test	2	3	4	5
1	2 test	3	4	5
1	2	3 test	4	5
1	2	3	4 test	5

Fig.1 The traditional Cross-Validation

The traditional Cross-Validation fits for the problems without time characteristic, such as classification, function fitting, probability density estimation. It doesn't consider the features of time series. So, for time series prediction, the predictive results with the parameters chosen by traditional Cross-Validation will not be satisfying. The experiments in section 7 prove this point.

5. The features of time series and its prediction

In this paper, we improve the traditional Cross-Validation in the light of the data features of time series. Here, we summarize the features of time series and its prediction as follows:

(1) Time series prediction has direction. Earlier data will influence later data. On the contrary, later data will not influence earlier data. This is to say, we can only use earlier data to predict later data. It is meaningless to predict earlier data via later data. Therefore, we mustn't choose an earlier section as tested samples in Cross-Validation.

(2) In time series, the predicted value is influenced by some historical data before it. But their influences are not balanceable. The influences of historical data nearer to predicted point are larger, and those of historical data farther from predicted point are smaller. Based on this point, the paper takes the different influences of historical data into account when choosing parameters of SVR by improved Cross-Validation.

6. The parameter choosing method of SVR models for time series prediction

In the light of the features of time series and its prediction, the paper improves the traditional Cross-Validation and combines it with ε-weighed SVR in order to get parameters of SVR models with high predicting precision, which is embodied at the following two aspects:

(1) Improve the traditional Cross-Validation to ensure the direction character and to learn adequately on numbered samples.

(2) In the light of the features of time series and its prediction, improve the traditional SVR learning process to ensure the predicting precision for time series. Here, we make the non-sensitive loss ε adjustable along with time series, which is fixed in traditional SVR.

6.1 The improved Cross-Validation for time series

In traditional Cross-Validation, the event happening later will be used to test the event happening before. Because of the direction character of time series prediction, this is illogical. For example, in Fig. 1, it is unsuitable to predict section 1 based on section 2 to 5 for time series. The experimental results in section 7 also prove this point. The improved Cross-Validation proposed by the paper only uses former sections to predict latter section in order to ensure the direction character of time series prediction. In this way, it seems that only section 5 is to be predicted based on section 1 to 4. In the improved Cross-Validation, numbered samples are learned adequately in order to get better model parameters and enhance predicting precision. So, besides the section 5 to be predicted, the section 4 will be predicted based on section 1 to 3, section 3 predicted based on section 1 to 2, and section 2 predicted based on section 1, as shown in Fig.2. The group of parameters, which make the average error least in 4 times of predictions, will be the parameters to be chosen. In this way, each sample (except those in section 1) is predicted once on the basis of ensuring the direction character of time series, which enhances prediction precision and its stability.

1	2	3	4	5 predict
1	2	3	4 predict	
1	2	3 predict		
1	2 predict			

Fig. 2 The improved Cross-Validation

6.2 The ε-weighed method for time series prediction

In traditional SVR, non-sensitive loss ε is fixed. For time series, the paper proposes a ε-weighed method when parameters of SVR models are chosen. In time series, the value at a predicted point is influenced by some historical data before it. However, their influences are not same. The influences of those farther from the predicted point are smaller, but the influences of those nearer to the predicted point are bigger. Thus, the data nearer to the predicted point should be paid more attention to. It can be seen from formula (3) that the data nearer to the predicted point have bigger slack variance ξ, and the corresponding ε should be smaller. Whereas, the non-sensitive loss ε corresponding with the data farther from the predicted point should be larger. Thereby, in the objective function of formula (4), non-sensitive loss ε should be adjusted along with time series, which will accord with the features of time series and improve its prediction accuracy.

According to the above discussion, the quadratic optimization problem in formula (4) can be transformed into the following form,

$$\begin{aligned} \max \quad & \sum_{i=1}^{l}(\alpha_i^* - \alpha_i)y_i - \varepsilon_i\sum_{i=1}^{l}(\alpha_i^* + \alpha_i) \\ & -\frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)K(x_i, x_j) \qquad (8) \\ \text{subject to} \quad & 0 \le \alpha_i, \alpha_i^* \le C(i = 1,2,\cdots,n) \\ & \sum_{i=1}^{l}(\alpha_i^* - \alpha_i) = 0 \end{aligned}$$

The paper consults the way for Jordan Neural Network and EWMA model to process historical data. The parameter ε_i (i=1, 2, …, n) corresponding with sample i is adjusted as follows,

$$\varepsilon_i = (1-d)^i \varepsilon_0 \quad (i = 1,2,\cdots n) \qquad (9)$$

Where, ε_0 is the parameter value chosen initially. d is adjusting factor(between 0 and 1). The sample x_1 corresponding with ε_1 is the farthest datum from predicted point, and the sample x_n corresponding with ε_n is the nearest datum to the predicted point.

7. Experimental results and analysis

The paper combines the two methods introduced in section 6, improving traditional Cross-Validation, at the same time, weighing to the parameter ε along with time series. In order to verify validity of the proposed method, the paper designs 5 groups of experiment schemes:

(1) **Scheme 1** is the traditional Cross-Validation and Grid-Search method, including 2 experiments, which correspond to with-shuffle and without-shuffle respectively.

(2) **Scheme 2** predicts section 5 based on section 1 to 4, including 2 experiments, which correspond to ε-weighed and ε-not-weighed respectively. The two experiments both embody the direction feature of time series prediction, showing the effect of ε-weighed SVR.

(3) **Scheme 3** increases one time of training and predicting on the basis of scheme 2, which predicts section 4 based on section 1 to 3, including ε-weighed and ε-not-weighed likewise. Which group of parameters is to be chosen lies on the average predicting error of section 4 and section 5.

(4) **Scheme 4** increases one time of training and predicting on the basis of scheme 3, which predicts section 3 based on section 1 to 2, including ε-weighed and ε-not-weighed likewise. Which group of parameters is to be chosen lies on the average predicting error of section 3 to 5.

(5) **Scheme 5** increases one time of training and

predicting on the basis of scheme 4, which predicts section 2 based on section 1, including ε-weighed and ε-not-weighed likewise. Which group of parameters is to be chosen lies on the average predicting error of section 2 to 5.

It is obvious that scheme 2 to 5 all ensure the direction feature of time series. And samples can be learned more and more adequately from scheme 2 to scheme 5. The comparison of experimental results among these schemes can verify the validity of the method proposed in section 6.1. Inside each scheme, two experiments corresponding with ε-weighed and ε-not-weighed can verify the validity of the method proposed in section 6.2. Synthetically comparing among and inside different schemes can verify the validity of the parameter choosing method proposed by the paper.

The 5 schemes are based on the dataset of *sinc* function[10], which is a typical time series. *sinc* function is defined as:

$$f(t) = sinc(t + a) + b \tag{10}$$

Here, set a=-4, b=0.5 without losing universality. We get 400 data points equably in the section $t \in [0,399]$, constructing 300 training samples with step length 7, and predicting the latter 60 and 93 data. In Grid-Search, set the ranges of the parameters C, ε, *gamma* are $C=2^{-1}$, $2^{0},\ldots, 2^{12}$; $\varepsilon=2^{-12}$, $2^{-11},\ldots, 2^{-1}$; $gamma=2^{-8}$, $2^{-7},\ldots, 2^{0}$. RBF kernel function is used in the improved Cross-Validation as below:

$$\exp(-gamma * |u - v| \wedge 2) \tag{11}$$

Mean Absolute Percentage Error (MAPE) is used to evaluate the performance of the method, namely,

$$\mathrm{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{t_i - t_i^*}{t_i}\right| \tag{12}$$

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

The results of the 5 experiment schemes can be seen in Table 1. Where, the values of C, ε, *gamma* respectively denote the ones of parameters chosen by each scheme. MAPE60 and MAPE93 are predicting errors of 60 and 93 predicted data. The comparison of predicting precisions among various schemes is shown in Fig.3.

It can be seen from the experimental results: the traditional Cross-Validation (whether shuffle is used or not) in scheme 1 is not suitable to time series prediction because of the direction feature of time series. Especially, Cross-Validation with shuffle brings more error because it completely throws data order of time series into confusion and doesn't consider this sort of order during parameter choosing. Although the Cross-Validation without shuffle brings less error, the value of parameter C is over big (is 256 times as big as that in scheme 5), which will lead to severe punishment to empirical error and weaken generalization of models, thereby losing the superiority of SVR.

It can be seen obviously that the predicting precisions from scheme 2 to 5 enhance gradually.

The prediction result of scheme 2 is not so good as that of scheme 3 to 5 because it only predicts section 5 based on section 1 to 4 to specify parameters of model. The training samples are not adequately used to learn, which leads to deficient training and more error.

There is the same problem as scheme 2 in scheme 3 and 4 compared with scheme 5. The using of samples is not quite sufficient. However, more information is learned during training with increase of the number of predicted sections. So, predicting precisions are enhanced to varying extent in scheme 3 and scheme 4.

Scheme 5 learns most information from samples. Its prediction result is the best, and punishing coefficient C is not augmented, whose maximal value is only 8. Thus, the over-learning phenomenon, which comes forth in scheme 1 (without-shuffle), is avoided.

It can be seen from Table 1 and Fig.3 that in all the 4 experiments with ε-weighed, 3 experiments are effective. Only the ε-weighed in scheme 2 has no ideal result. This is because parameter choosing only relies on prediction for one section (section 5), and samples are not utilized adequately. The first experiment in scheme 5 not only learns from samples adequately, but also ε is weighed on the basis of keeping predicting direction. So, it gains highest predicting precision with good generalization performance, which verifies the validity of the parameter choosing method of SVR for time series prediction.

8. Conclusions

The paper proposes a parameter choosing method of SVR models for time series prediction, which improves traditional Cross-Validation, and non-sensitive loss ε in SVR is weighed during parameter choosing in order to get good parameters of SVR models for time series prediction. It fully considers the data features of time series and mines the information included in numbered samples at the premise of keeping generalization performance. It solves the problem that traditional Cross-Validation can't be well applied in time series prediction and gets good effect.

Table 1 Results of five groups of experimental schemes

		C	*gamma*	*ε*	MAPE60	MAPE93
Scheme 1	With-shuffle	64	0.125	0.03125	0.010338	0.01047
	Without-shuffle	2048	0.0078125	0.000244141	0.000559	0.000561
Scheme 2	*ε*-weighed	0.5	0.00390625	0.0625	0.035033	0.035892
	ε-not-weighed	2	0.25	0.015625	0.020572	0.020648
Scheme 3	*ε*-weighed	0.5	0.00390625	0.0078125	0.011637	0.011135
	ε-not-weighed	2	0.00390625	0.03125	0.014871	0.015499
Scheme 4	*ε*-weighed	0.5	0.015625	0.000976563	0.006581	0.006171
	ε-not-weighed	1	0.0078125	0.001953125	0.006948	0.006693
Scheme 5	*ε*-weighed	8	0.015625	0.00390625	0.001189	0.001144
	ε-not-weighed	1	0.03125	0.00390625	0.002341	0.002238

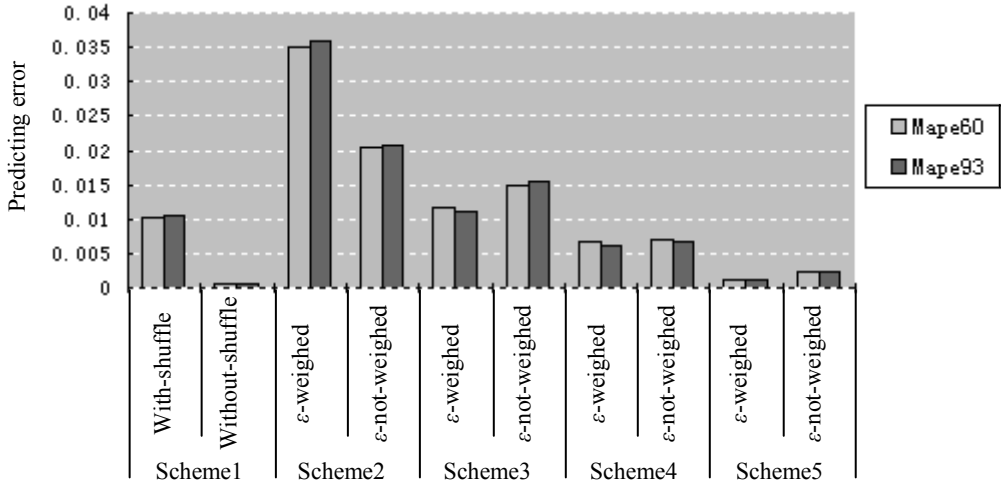

Fig. 3 Comparison of predicting errors among five groups of experimental schemes

References

[1] Vapnic V N. The Nature of Statistical Learning Theory[M]. New York: Springer-Verlag, 1995.

[2] Chang M. W., Lin C. J. Analysis of Switching Dynamics with Competing Support Vector Machines[J], IEEE Transactions on Neural Networks, 2004, 15:720-727.

[3] Zhang Z. S., Li L. J. He Z. J. Research on Parameter Optimization of Fault Classifier Based on Support Vector Machine[J], Journal of Xi'an Jiaotong University, 2003, 37(11): 1101-1109.

[4] Keerthi S. Efficient Tuning of SVM Hyperparameters Using Radius/Margin Bound and Iterative Algorithms[J], IEEE Transactions on Neural Networks, 2001, 13(5): 1225-1229.

[5] Kwok J. T. Y. The Evidence Framework Applied to Support Vector Machines[J], IEEE Transaction on Neural Networks, 2000, 11(5): 1162-1173.

[6] Chapelle O., Vapnik V. N., Bousquet O., et al. Choosing Multiple Parameter for Support Vector Machines[J]. Machine Learning, 2002, 46(1): 131~159.

[7] Keerthi S., ong C., Lee M. Two Efficient Methods for Computing Leave-one-out Error in SVM Algorithms[R], Dept of Mechanical Engineering, National University of Singapore, 2000.

[8] Du S. X., Wu T. J. Support Vector Machines for Regression[J], Journal of System Simulation, 2003, 15(11): 1580-1585.

[9] Chang C. C., Lin C. J. LIBSVM: A Library for Support Vector Machines[EB/OL], 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[10] Li L. J., Zhang Z. S., He Z. J. Research on Condition Trend Prediction of Mechanical Equipment Based on Support Vector Machine[J], Journal of Xi'an Jiaotong University, 2004, 38(3): 230-238.