Chapter 10
Support Vector Regression with Kernel Mahalanobis Measure for Financial Forecast

James N.K. Liu and Yan-xing Hu

Abstract. For time series forecasting which have data sets coming from an unstable and nonlinear system such as the stock market. Support Vector Regression (SVR) appears to be an efficient tool which has been widely used in recent years. It is also reported to have a higher accuracy and generalization ability than other traditional methods. The SVR method deals with the nonlinear problem by mapping the input feature space into a high dimensional space so that it becomes a linear problem. Kernel function is one of the crucial components in SVR algorithm as it is used to calculate the inner product between vectors in the mapped high dimensional space. The kernel function of Radial Basis Function (RBF), which is based on the Euclidean distance, is the most commonly used kernel function in SVR. However, the SVR algorithm may neglect the effect of correlation among the features when processing the training data in time series forecasting problems due to the limitation of Euclidean distance. In this chapter, a Mabalanobis distance RBF kernel is introduced. It is well known that when we need to calculate similarity between two vectors (samples), the use of Mahalanobis distance can take into account the correlation among the features. Thus, the SVR with Mahalanobis distance kernel function may follow the behavior of the data sets better so that it can give more accurate result. From the comparative investigation, we find that in some circumstances, the Mabalanobis distance RBF kernel based SVR can outperform the Euclidean distance based SVR.

1 Introduction

In the past ten years, Support Vector Regression (SVR) has been widely applied to deal with time series forecasting problems in different domains; especially for those

James N.K. Liu · Yan-xing Hu
Department of Computing,
The Hong Kong Polytechnic University, Hong Kong
e-mail: {csnkliu,csyhu}@comp.polyu.edu.hk

W. Pedrycz & S.-M. Chen (Eds.): Time Series Analysis, Model. & Applications, ISRL 47, pp. 215–227.
DOI: 10.1007/978-3-642-33439-9_10

domains characterized as some complex, nonlinear and unstable discipline. For example, it has been reported that the SVR performed well in financial forecasting [1-3] and also in temperature forecasting [4,5]. When applied to these problems, empirical results showed many advantages with the SVR algorithm; for example, higher generalization ability and the capability of avoiding overfitting problems. Due to its well-founded statistical learning theory, the SVR showed a better performances than other methods such as, Artificial Neural Networks (ANN) or Auto-Regressive and Moving Average (ARMA) Model in some previous works[6,7].

In 1995, Vapnik first introduced the Support Vector Machine (SVM) [8]. The SVM was originally applied to deal with classification problems and soon extended to regression problems [9,10]. Compared with other estimation models such as ANN and ARMA, SVR substitutes the traditional Empirical Risk Minimization (ERM) principle with Structure Risk Minimization (SRM) to address the overfitting problem and able to offer higher generalization ability. Accordingly, as we discussed above, when applied in a complex, nonlinear and unstable system, the SVR can demonstrate better performance than other methods.

The kernel function is one of the crucial components in the SVM algorithm. By using the kernel function, we can map all the samples into high dimensional feature space so that the nonlinear problem can be solved as the linear problem. The choice of kernel function may affect the performance of SVM algorithm. However, till now, there is still no guideline how to determine which kernel function can provide the best performance of SVM. Among all the kernel functions, the Radial Basis Function (RBF) kernel $K(x_i,x_j) = \exp(\frac{-\left\|x_i-x_j\right\|^2}{2\gamma^2})$ is most frequently used because the number of parameters of RBF kernel function is less than in other kernel functions. Moreover, previous experiments also showed that in most conditions, RBF kernel function could provide better performance than other kernel functions[11]. Many previous experiments that used SVR to deal with time series forecasting problems also considered RBF kernel function.

In this chapter, we introduce a new Mahalanobis distance based RBF kernel function. We focus our investigation on the comparison of SVRs with the two (Mahalanobis distance based and Euclidean distance based) different kinds of RBF kernel functions and their performance on financial time series forecasting problems. It is well known that compared with Euclidean distance, Mahalanobis distance takes into account the correlations among attributes (features) of the data set. Based on Euclidean distance, SVM algorithm neglects the correlations among attributes (features) of the training samples. Some previous researches have noticed this limitation of SVM, and subsequently introduced the Mahalanobis distance into SVM to take into consideration of the effect of correlations among attributes (features) in the training process. Some encouraging results have been obtained when Mahalanobis distance based SVM is applied to deal with classification problems [12,13]. Particularly, Wang and Yeung analysed some conventionally used kernel functions in SVM and employed Mahalanobis distance to modify the kernel function so as to improve the classification accuracy [14]. Nevertheless till now, there is little investigation on the effect of Mahalanobis distance based SVM on regression problems.

Since that in the calculation process of SVR, the Euclidean distance is applied to express the distances among the sample points, the SVR algorithm will inevitably accept the limitation of Euclidean distance: the correlations among the input variables are neglected. However, for a typical time series forecasting problem, we have features selected according to a certain time interval, such as stock index of $(t-1)$ day, stock index $(t-2)$ day, ... , stock index $(t-n)$ day. Obviously, these features are not independent from each other. When using classical SVR to deal with such types of time series forecasting problems, the correlation influence will be neglected. Therefore, it would be beneficial to improve the performance of SVR if we could take into account correlation in the training process. Therefore in this chapter, a Mahalanobis distance based RBF kernel function will be used in the SVR to deal with the financial time series forecasting problems; and we will investigate the performance of the proposed kernel function through a series of experiments.

The chapter is organized as follows: in Section 2 we will briefly introduce some background, and analysis of the Mahalanobis distance based RBF kernel and the Euclidean distance based RBF kernel in Section 3. Section 4 discusses the experiments along with some comparative analysis. The last section gives the conclusion and future work.

2 Background Knowledge

2.1 Support Vector Regression

Based on the structural risk minimization (SRM) principle, SVM method seeks to minimize an upper bound of generalization error instead of the empirical error as in other neural networks. Additionally, SVM models generate the regression function by applying a set of high-dimensional linear functions. The SVR function is formulated as follows:

$$y = w\phi(x) + b \tag{1}$$

where $\phi(x)$ is called the feature, which is nonlinear and mapped from the input space $\Re^n$. y is the target output value we want to estimate. The coefficients w and b are estimated by minimizing:

$$R = \frac{1}{2}\|w\|^2 + \frac{1}{n}C\sum_{i=1}^{n} L_\varepsilon(d_i, y_i) \tag{2}$$

where:

$$L_\varepsilon(d,y) = \begin{cases} |d-y| - \varepsilon, \ |d-y| \geq \varepsilon \\ 0, \text{otherwise} \end{cases} \tag{3}$$

Eq. (2) is the risk function consisting of the empirical error and a regularization term that is derived from the SRM principle. The term $\frac{1}{n}\sum\limits_{i=1}^{n} L_\varepsilon(d_i, y_i)$ in Eq. (2) is the empirical error (risk) measured by the ε-insensitive loss function (ε-insensitive tube) given by Eq. (3); in the meanwhile, the term $\frac{1}{2}\|w\|^2$ is the regularization term.

The constant $C > 0$ is taken as the regularized constant that determines the trade-off between the empirical error (risk) and the regularization term. Increasing the value of C will add importance to the empirical risk in the risk function. ε is called the tube size of the loss function and it is equivalent to the accuracy approximation placed on the training data points. Both C and ε are user-prescribed parameters.

Then the slack variables ζ and ζ^* which represent the distance from the actual values to the corresponding boundary values of ε-insensitive tube are introduced. With these slack variables, Eq. (3) can be transformed to the following constraint based optimization:

Minimize:

$$R(w,\zeta,\zeta^*) = \frac{1}{2}ww^T + C(\sum_{i=1}^{n}(\zeta+\zeta^*)) \tag{4}$$

Subject to:

$$\begin{aligned} &w\phi(x_i) + b_i - d_i \leq \varepsilon + \zeta_i^* \\ &d_i - w\phi(x_i) - b_i \leq \varepsilon + \zeta_i \\ &\zeta_i, \zeta_i^* \geq 0, i = 1,2,\cdots,n \end{aligned} \tag{5}$$

Finally, by introducing the Lagrangian multipliers and maximizing the dual function of Eq. (4), it can be changed to the following form:

$$\begin{aligned} R(\alpha_i - \alpha_i^*) = &\sum_{i=1}^{n} d_i(\alpha_i - \alpha_i^*) - \varepsilon\sum_{i=1}^{n}(\alpha_i - \alpha_i^*) \\ &-\tfrac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}(\alpha_i - \alpha_i^*)\times(\alpha_j - \alpha_j^*)(\Phi(x_i)\cdot\Phi(x_k)) \end{aligned} \tag{6}$$

with the constraints:

$$\sum_{j=1}^{n}(\alpha_i - \alpha_i^*) = 0, 0 \leq \alpha_i \leq C, 0 \leq \alpha_i^* \leq C, i = 1,2,\cdots,n \tag{7}$$

In Eq. (7), α_i and α_i^* are called Lagrangian multipliers which satisfy $\alpha_i \times \alpha_i^* = 0$, the general form of the regression estimation function can be written as:

$$f(x,\alpha_i,\alpha_i^*) = \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)K(x,x_i) + b \tag{8}$$

In this equation, $K(x_i \cdot x)$ is called the kernel function. It is a symmetric function $K(x_i \cdot x) = (\Phi(x_i)\cdot\Phi(x))$ satisfying Mercer's conditions. When the given problem is a nonlinear problem in the primal space, we may map the sample points into a high-dimensional feature space where the linear problem can be performed. Linear, Polynomial, Radial Basis Function (RBF) and sigmoid are four main kernel functions in use. As we discussed above, in most of the time series forecasting problems, the SVR employs RBF kernel function to estimate the nonlinear behavior of the forecasting data set because RBF kernels tend to give good performance under general smoothness assumptions.

2.2 Euclidean Distance Measure verse Mahalanobis Distance Measure

It is well known that the Euclidean distance is the most widely used measure to define the distance between two points in Euclidean space. In Euclidean space, for any two points $x_i = (x_{i1}, x_{i2}, \ldots, x_{in})$ and $x_j = \left(x_{j1}, x_{j2}, \ldots, x_{jn}\right)$, the Euclidean distance between these two points can be calculated as:

$$d_E(X_i, X_j) = \sqrt{\sum_{k=1}^{n} \left|x_{ik} - x_{jk}\right|^2} \tag{9}$$

Although the Euclidean distance is widely used, it also has an obvious limitation. As discussed earlier, different features of the samples are considered as equal in the calculation of Euclidean distance; also, the correlations among the features are neglected.

One of the methods to address the limitation of Euclidean distance is to use the Mahalanobis distance [15]. Let X be a $l \times n$ input matrix containing l random observations $x_i \in \Re^n$, $i = 1, \ldots, l$. The Mahalanobis distance d_M between any two samples x_i and x_j can be calculated as follows:

$$d_M(x_i, x_j) = \sqrt{(x_i - x_j)^T \sum{}^{-1} (x_i - x_j)} \tag{10}$$

$\sum$ is the covariance matrix which can be calculated as:

$$\sum = \frac{1}{l} \sum_{k=1}^{l} (xk - \mu) \cdot (xk - \mu)^T \tag{11}$$

where μ is a mean vector of all samples.

Originally, the Mahalanobis distance can be defined as a dissimilarity measure between two random vectors of the same distribution with covariance matrix $\sum$.

From the definition of Mahalanobis distance we can see that the Mahalanobis distance is based on correlations between variables where different samples that can be identified and analyzed. It differs from Euclidean distance based on the correlations of the data set and is scale-invariant. Then again, if the covariance matrix is the identity matrix, the Mahalanobis distance will be equal to the Euclidean distance.

Considering that samples are locally correlated, a local distance measure incorporating the samples' correlation might be a better choice as a distance measure. Mahalanobis distance can take into account the covariance among the variables in calculating distances. Accordingly, in some circumstances, it may be a more suitable measure to calculate the distance and evaluate the similarity between two points[13].

3 Mahalanobis Distance RBF Kernel Based SVR

3.1 The Analysis of Kernel Functions in SVR

In SVR, to enable the nonlinear problem to be estimated by a linear function as shown in Eq. (8), we have to map the original input feature space into a high-dimensional feature space. Note that the mapping is $\Phi(x)$ as given in Eq. (6), we have to calculate the inner products of every two vectors in the transformed high-dimensional feature space. Thus, the curse of dimensionality[16] will emerge.

To deal with this problem, we can obtain a kernel function that meets this requirement $K(x_i \cdot x_j) = (\Phi(x_i) \cdot \Phi(x_j))$ and calculate the $K(x_i, x_j)$ instead of calculating the inner products of the vectors in the transformed high-dimensional feature space. In fact, every kernel function meeting the Mercer's Theorem can be used in SVM algorithm[9]. Usually, Linear, Polynomial, Radial Basis Function (RBF) and Sigmoid are the four main kernel functions in use.

Table 1 shows the form of the four kernel functions.

Table 1 The main kernel functions used in SVM

Linear	$K(x_i, x_j) = x_i \cdot x_j$
Polynomial	$K(x_i, x_j) = (c + x_i \cdot x_j)^d$
Radial Basis Function (RBF)	$K(x_i, x_j) = \exp(\frac{-\lVert x_i - x_j \rVert^2}{2\gamma^2})$
Sigmoid	$K(x_i, x_j) = \tanh(c(x_i \cdot x_j) + \theta)$

The RBF kernel function is the most commonly used among the four kernel functions in real applications. From what we notice of the linear kernel function, the Polynomial kernel function and the Sigmoid kernel function are all based on the inner products of the vectors. In other words, these kernel functions can be considered as functions with the variable $(x_i \cdot x_j)$. Unlike other three kernel functions, the RBF kernel function is based on the Euclidean distance between two points in the feature space: having examined the format of RBF kernel function we can observe that the variable of the RBF kernel function can be considered as the Euclidean distance between two points denoted as $\lVert x_i - x_j \rVert$ [14].

In fact, the RBF kernel function is a measure of the similarity between two vectors in the Euclidean feature spaces. If x_i and x_j are very close in Euclidean distance ($\lVert x_i - x_j \rVert \approx 0$), the value of the RBF kernel function will tend to be 1, conversely, if x_i and x_j are quite far apart in Euclidean distance ($\lVert x_i - x_j \rVert >> 0$), the value of the RBF kernel function will tend to be 0.

3.2 Substituting Euclidean Distance with Mahalanobis Distance in RBF

The Euclidean distance has the limitation that it neglects the correlations among the features. From the above discussion, we can see that the RBF kernel function can be considered as a Euclidean distance variable-based function. Accordingly, the RBF kernel function inherits the limitation of Euclidean distance.

One possible method of addressing this limitation is the use of Mahalanobis distance instead. We substitute the Euclidean distance with Mahalanobis distance as the variable in RBF kernel function. The Mahalanobis distance based RBF kernel function is:

$$K_M(x_i,x_j) = \exp\left(\frac{-((x_i - x_j)^T \Sigma^{-1}(x_i - x_j))}{2\gamma^2}\right) \tag{12}$$

Σ is the covariance matrix which can be calculated as:

$$\Sigma = \frac{1}{l}\sum_{k=1}^{l}(xk - \mu)\cdot(xk - \mu)^T \tag{13}$$

where μ is a mean vector of all samples. The format of linear estimate function in Eq.(8) can now be transformed to:

$$f(x) = \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)K_M(x,x_i) + b \tag{14}$$

By introducing the Mahalanobis distance into the RBF kernel function, we can measure the similarity between two vectors with the Mahalanobis distance rather than Euclidean distance. The Mahalanobis distance based kernel function can take into account the correlations among attributes of the samples in the SVM training processing. When SVR with the proposed kernel function is applied to deal with time series forecasting problems, it should be beneficial to the performance improvement of the SVR forecasting result.

4 Experimental Results and Analysis

In this chapter, our investigation mainly focuses on the performance of Mahalanobis distance RBF kernel function in SVR and its performance in financial time series forecasting. To evaluate the performance of Mahalanobis distance BRF kernel function based SVR in time series forecasting, a series of experiments are conducted. 15 financial data sets about time series forecasting problem are applied in our experiment. Three forecasting methods are used in our experiment for comparison: we use the Mahalanobis distance RBF based SVR, Euclidean distance RBF based SVR and the BP neural network to estimate the target values and analyze the result with comparison.

4.1 Data Collection

As discussed in the above, the SVR is reported to be very suitable in dealing with complex, unstable and nonlinear forecasting problems such as problems in financial forecasting domain. In our experiment, 15 financial datasets from the real world are collected to evaluate the performance of the SVR. We have chosen 6 stocks from China A share market in Shanghai and 7 stocks from China A share market in Shenzhen. We aim to forecast the close prices of the 13 stocks. In addition, the Stock Indexes of the two markets: Shanghai composite index and Shenzhen composite index are also used as data sets in our experiment. This data covers the period from the 15th, September 2006 to the 31st, December, 2009. Thus, each of the data sets contains more than 750 samples.

Table 2 The features for the stock price forecasting in China A share markets

1	Today's lowest price
2	Today's highest price
3	The lowest price of the last trading day
4	The highest price of the last trading day
5	The moving average lowest price of the last 5 trading days
6	The moving average highest price of the last 5 trading days
7	Today's open price
8	The highest price of the last trading day
9	The moving average highest price of the last 5 trading days
10	Today's turnover
11	The turnover of the last trading day
12	The moving average turnover of the last 5 trading days
13	Today's volume
14	The volume of the last trading day
15	The moving average volume of the last 5 trading days

Table 3 The features for the Shanghai/Shenzhen composite index forecasting

1	Today's daily open index
2	The open index of the last trading day
3	The open index of the $(t-2)$ trading day
4	The open index of the $(t-3)$ trading day
5	The open index of the $(t-4)$ trading day
6	The open index of the $(t-5)$ trading day
7	The close index of the last trading day
8	The close index of the $(t-2)$ trading day
9	The close index of the $(t-3)$ trading day
10	The close index of the $(t-4)$ trading day
11	The close index of the $(t-5)$ trading day

For different data types, we select different features for constructing the regression models. Table 2, shows features for the stock price forecasting in China A share market in Shanghai and Shenzhen; Table 3, shows the input features for the forecasting of Shanghai composite index and Shenzhen composite index.

4.2 Data Pre-processing

4.2.1 Shift Windows

In this experiment; to test the learning capability of the algorithms and to follow as well as forecast the trend of the stock price movement, a shift window was designed. For each of the data set, there were 30 samples in one window, approximately 5% of the total samples, and the first 25 of these samples were used as training data and the last 5 samples as testing data. We then shifted forward this window by the shift step of 5 days. For example, the first shift window contains 30 trading days of data from 15th, September, 2006 to 9th, November, 2006, in this shift window, the first 25 samples, which began on 15th, September and finished on 2nd, November, are used as training set, and the data of the following five days, from 3rd, November to 9th, November, are used as testing data. We predict the stock price of these five days, and compare that with the actual price of these five days. Then we shift the window forward and the training set started from 21st, September till 9th, November. The actual stock prices of these 25 samples are used as training set to predict the following 5 day's stock price. Analogically, we can predict all the stock prices of our set by shifting the windows. In every window, the ratio of training samples and testing samples is 5:1.

4.2.2 Normalization of Data

When we use Euclidean distance RBF kernel function-based SVR to do the prediction, the data set should be normalized to avoid features that may contain a greater numeric value range from dominating the features; that have smaller numeric ranges in the process of training and regression. In this experiment, the formula we applied to normalize the data is:

$$v' = \frac{v - \min_\alpha}{\max_\alpha - \min_\alpha}, \tag{15}$$

where v' is the normalized value and v is the original value. After the process, all the values of the features were normalized within the range of [0, 1].

4.3 Evaluation Criteria

The prediction performance can be evaluated by the following statistical metrics[17]:

Normalized Mean Squared Error (NMSE) measures the deviation between the actual values and the predicted values. The smaller the values are, the closer the predicted values to the actual values. The formula of NMSE is:

$$\mathrm{NMSE} = 1/(\delta^2 n) \sum_{i=1}^{n} (a_i - p_i)^2 \tag{16}$$

where

$$\delta^2 = 1/\left(\mathrm{n}-1\right) \sum_{i=1}^{n} (a_i - p_i)^2 \tag{17}$$

Directional symmetry (DS) indicates the correctness of the predicted direction of predicted value in terms of percentages. The formula of DS is:

$$\mathrm{DS} = (100/n) \times \sum_{i=1}^{n} d_i \tag{18}$$

where

$$d_i = \begin{cases} 1,(a_i - a_{i-1})(p_i - p_{i-1}) \geq 0 \\ 0, \text{otherwise} \end{cases} \tag{19}$$

4.4 Experimental Results and Discussion

Table 4, shows the results of the experiment. The stocks which have a stock number starting with ”6” are from China A share market in Shenzhen, the stocks which have a stock number starting with ”0” are from China A share market in Shanghai. The columns denoted as MRBFSVR present the results of Mahalanobis distance RBF based SVR, the columns denoted as ERBFSVR present the results of Euclidean distance RBF based SVR, and BPNN is short for BP neural network. Fig. 1 and Fig. 2 give the comparison results of the three methods.

The results in Fig. 1 and Fig. 2 show that both of the two SVRs, the Mahalanobis distance RBF based SVR and Euclidean distance RBF based SVR; outperform the BP neural network with respect to the criteria of NMSE and DS in most of the 15 data sets. Obviously, from these results, we can observe that the SVM regression method is more suitable for time series forecasting problems in financial forecasting than the BP neural network algorithm.

From Table 4, we cannot conclude that the Mahalanobis distance RBF based SVR is definitely a better algorithm than the Euclidean distance RBF based SVR. We can observe that the criteria of NMSE, the Mahalanobis distance RBF based SVR reduces to a lower NMSE value in 8 of the 15 data sets than the Euclidean distance RBF based SVR; although, for the criteria of DS, the Mahalanobis distance RBF based SVR receives a higher DS value in 10 of the 15 data sets than the Euclidean distance RBF based SVR. These results are not enough to support the conclusion that the Mahalanobis distance RBF based SVR is superior to the Euclidean distance RBF based SVR when applied to time series forecasting.

Table 4 The NMSE and DS values of the three algorithms for the 15 data sets

	NMSE MRBFSVR	NMSE ERBFSVR	NMSE ERBFSVR	DS MRBFSVR	DS ERBFSVR	DS ERBFSVR
Stock600111	1.452	2.031	3.251	0.723	0.721	0.692
stock600839	1.234	1.252	2.187	0.832	0.751	0.747
stock600644	2.122	2.574	3.145	0.765	0.862	0.691
stock600688	3.217	3.202	4.012	0.658	0.723	0.735
Stock601318	1.231	1.439	2.809	0.852	0.635	0.821
Stock600031	3.381	3.226	4.515	0.721	0.696	0.734
Stock000858	1.535	2.024	2.991	0.890	0.791	0.695
Stock000014	1.213	1.201	2.715	0.724	0.713	0.627
Stock000024	2.642	2.412	2.499	0.635	0.731	0.522
Stock000002	1.924	2.213	2.758	0.592	0.591	0.670
Stock000063	1.983	71.327	2.301	0.832	0.751	0.753
Stock000100	2.910	1.045	1.703	0.901	0.912	0.715
Stock000527	1.523	1.213	1.609	0.749	0.812	0.842
Shanghai index	1.237	2.341	2.764	0.831	0.826	0.731
Shenzhen index	1.101	1.923	2.113	0.877	0.841	0.687
Avarage	1.913	1.961	2.758	0.772	0.757	0.710

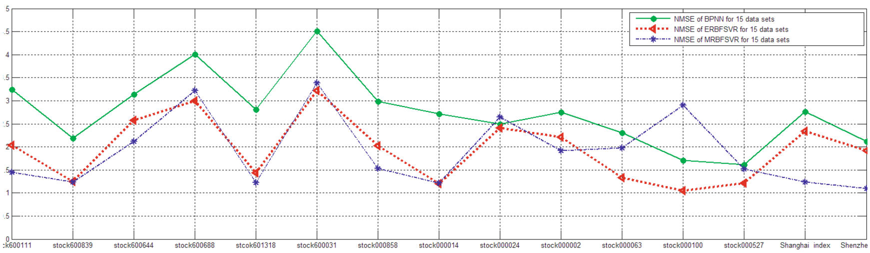

Fig. 1 The comparison of NMSE value of the three algorithms

This phenomenon can be explained by the limitation of Mahalanobis distance. Compared with Euclidean distance, the Mahalanobis distance takes into account the effect of correlation among the features of the training samples; but it also has the limitation that the Mabalanobis distance may enlarge the effect of correlation among the features. Such an enlargement might have generated some negative effect for some data sets.

However, from Table 4, we can see that for 6 of the 15 data sets, the Mahalanobis distance RBF based SVR outperforms the Euclidean distance RBF based SVR in both of the criteria of NMSE and criteria of DS; and for 12 of the 15 data sets, the Mahalanobis distance RBF based SVR gives a better performance based on at least one criterion. With only 3 data sets, the Mahalanobis distance RBF based SVR obtains the worst performance in both of the criteria of NMSE and criteria of DS.

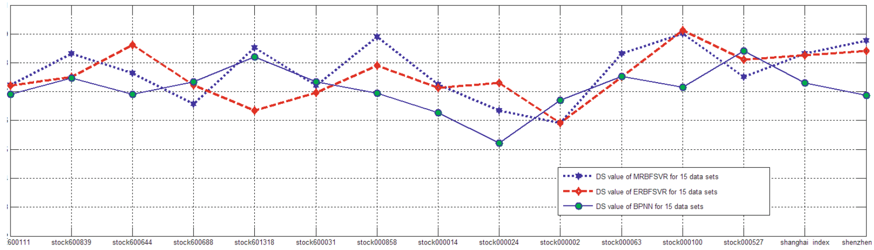

Fig. 2 The comparison of DS value of the three algorithms

What is more, for the average of the results of 15 data sets, the Mahalanobis distance RBF based SVR shows better performance for both of the two criteria of NMSE and criteria of DS. Hence, we can conclude that on the whole the Mahalanobis distance RBF based SVR performs better than the Euclidean distance RBF based SVR. But for a certain new data set, we cannot determine which one can achieve a better performance if there is no prior knowledge.

In summary, we find that under certain circumstances when applied to time series forecasting problems; the Mabalanobis distance RBF kernel can be a better choice than the traditional Euclidean distance RBF kernel function for SVR. In consideration of the correlation among the features, the use of Mabalanobis distance RBF kernel appears to be beneficial to the improvement of the forecasting result.

5 Conclusion and Future Work

SVR is an efficient tool for time series forecasting problems when the data sets stem from an unstable and nonlinear system. However, based upon the Euclidean distance, the SVR neglects the effect of correlation among the features when processing the training data in time series forecasting problems. Since the Mabalanobis distance can address the limitation of Euclidean distance, a Mabalanobis distance RBF kernel is introduced in this chapter. From the comparison investigation, we find that in some circumstances, the Mabalanobis distance RBF kernel based SVR can outperform the Euclidean distance based SVR. Consequently, when people use the SVR to deal with the time series forecasting problems, the proposed Mabalanobis distance RBF kernel based SVR is worthy for consideration.

One of the limitations of our work is that we have not yet provided the means for determining which RBF kernel may attain better results for a certain data set. Some of our current work reveals that there could be some influence due to the selection and distribution of the features in the data set. This will be our further work in future.

References

1. Kim, K.: Financial time series forecasting using support vector machines. Neurocomputing 55(1-2), 307–319 (2003)
2. Khemchandani, R., Chandra, S.: Regularized least squares fuzzy support vector regression for financial time series forecasting. Exp. Sys. with Appl. 36(1), 132–138 (2009)
3. Huang, W., Nakamori, Y., Wang, S.: Forecasting stock market movement direction with support vector machine. Comput Operat. Res. 32, 2513–2522 (2005)
4. Radhika, Y., Shashi, M.: Atmospheric temperature prediction using support vector machines. Int. J. Comput. Theor. and Eng. 1(1), 1793–8201 (2009)
5. Liu, B., Su, H., Huang, W., Chu, J.: Temperature prediction control based on least squares support vector machines. J. Contr. Theor. and Appl. 2(4), 365–370 (2004)
6. Bhasin, M., Raghava, G.: Prediction of CTL epitopes using QM, SVM and ANN techniques. Vaccine 22(23-24), 3195–3204 (2004)
7. Hossain, A., Zaman, F., Nasser, M., Islam, M.: Comparison of GARCH, Neural Network and Support vector machine in financial time series prediction. Pat. Rec. and Mach. Intel., 597–602 (2009)
8. Vapnik, V.N.: The Nature of Statistical Learning Theory. Springer, New York (1995)
9. Smola, A.J., Schlkopf, B.: A tutorial on support vector regression. Stat. and Comput. 14(3), 199–222 (2004)
10. Vapnik, V., Golowich, S.E., Smola, A.J.: Support vector method for function approximation, regression estimation, and signal processing. In: Michael, C., Michael, I., Thomas, P. (eds.) Advances in Neural Information Processing Systems 9, pp. 281–287. MIT Press (1996)
11. Cherkassky, V., Ma, Y.: Practical selection of SVM parameters and noise estimation for SVM regression. Neur. Netw. 17(1), 113–126 (2004)
12. Tsang, I.W., Kwok, J.T.: Learning the Kernel in Mahalanobis One-Class Support Vector Machines. In: The 2006 IEEE International Joint Conference on Neural Network Proceedings, Vancouver, BC, pp. 1169–1175 (2006)
13. Zhang, Y., Xie, F., Huang, D., Ji, M.: Support vector classifier based on fuzzy c-means and Mahalanobis distance. J. of Intel. Inf. Sys. 35, 333–345 (2009)
14. Wang, D., Yeung, D.S., Tsang, E.C.C.: Weighted mahalanobis distance kernels for support vector machines. IEEE Trans. on Neural Networks 18, 1453–1462 (2007)
15. Mahalanobis, P.C.: On the generalized distance in statistics. In: Proceedings National Institute of Science, India, pp. 49–55 (1936)
16. Bellman, R., Kalaba, R.: On adaptive control processes. IEEE Trans. on Auto. Contr. 4(2), 1–9 (1959)
17. Makridakis, S., Wheelwright, S.C., Hyndman, R.J.: Forecasting methods and applications. Wiley-India (2008)