Fuzzy Support Vector Machines Regression for Business Forecasting: An Application

Yukun Bao[12], Rui Zhang[1], and Sven F. Crone[2]

[1] Department of Management Science & Information System, School of Management, Huazhong University of Science and Technology, Wuhan 430074, China
yukunbao@mail.hust.edu.cn
[2] Department of Management Science, Management School, Lancaster University, Lancaster LA1 4YX, United Kingdom

Abstract. This study proposes a novel method for business forecasting based on fuzzy support vector machines regression (FSVMR). By an application on sales forecasting, details of proposed method are presented including data preprocessing, kernel selection, parameters tuning and so on. The experimental result shows the method's validity.

1 Introduction

Business forecasting has consistently been a critical organizational capability for both strategic and tactical business planning [1]. Time series forecasting methods such as exponential smoothing have been widely used in practice, but it always doesn't work when the market fluctuates frequently and at random [2]. Research on novel business forecasting techniques have evoked researchers from various disciplines such as computational intelligence.

Recently, support vector machines (SVMs) have been extended to solve nonlinear regression estimation problems and they have been shown to exhibit excellent performance in time series forecasting [3, 4, 5].

One of the key issues encountered in training support vector is the data preprocessing. Some raw data points corrupted by noises are less meaningful and they make different senses to later training process. But standard SVMs algorithm lacks this ability. To solve this problem, Fuzzy support vector machines regression(FSVMR) apply a fuzzy membership to each input points so that different input points can make different contributions to the learning of decision surface and can enhances the SVM in reducing the effect of outliers and noises in data points. Details on the principal and application of FSVMR can be found in ref. [6, 7, 8]

2 Experimental Setting and Algorithms

2.1 Data Sets

We selected 5 goods with 430 daily sales data from a manufacturing firm's management information system. We used the former 400 data points as training

L. Wang et al. (Eds.): FSKD 2006, LNAI 4223, pp. 1313–1317, 2006.

Table 1. Details of Data Sets

Goods	Mean	SD	Min	Max	Train	Test
A	6.95	3.58	0	14.7	400	30
B	171	81.24	23	285	400	30
C	79.17	28.31	12	167	400	30
D	7.52	5.24	1	13.6	400	30
E	18.19	10.11	12.33	41.56	400	30

data sets and the rest 30 data points as testing data.More details of the data sets are listed in Table 1.

2.2 Embedding Dimension

Given a time-series $\{x_1, x_2, \ldots, x_n\}$ generated by a dynamical system. We assume that $\{x_{t+\tau}\}(\tau \geq 1)$ is a projection of dynamics operation in a high-dimensional state space [3]. In order to make prediction, we must reconstructing the input time series data into state space. That is to say if $\{x_t\}$ is the goal value of prediction, the previous values $\{x_{t-(d-1)\tau}, x_{t-(d-2)\tau}, \ldots, x_{t-\tau}\}$ should be the corrected state vector. We call d the embedding dimension or the sliding window, τ the prediction period. In this experiment, we only analyze forecasting $\{x_{t+1}\}$, thus, $\tau = 1$. After transformation, we get the samples in matrix form:

$$\mathbf{X} = \begin{pmatrix} x_1 & x_{1+\tau} & \cdots x_{1+(d-1)\tau} \\ x_2 & x_{2+\tau} & \cdots x_{2+(d-1)\tau} \\ \vdots & \vdots & \ddots \quad \vdots \\ x_{n-(d-1)\tau} & x_{n-(d-2)\tau} & \cdots \quad x_{n-\tau} \end{pmatrix} \quad \mathbf{Y} = \begin{pmatrix} x_{1+d\tau} \\ x_{2+d\tau} \\ \vdots \\ x_n \end{pmatrix} \tag{1}$$

The value of embedding dimension of a time series data set affects prediction performance. In following experiments, embedding dimension is fixed at 5 balancing error and training cost.

2.3 Defining Fuzzy Membership

It is easy to choose the appropriate fuzzy membership. First, we choose $\sigma > 0$ as the lower bound of fuzzy membership. Second, we make fuzzy membership s_i be a function of time t_i

$$s_i = f(t_i) \tag{2}$$

We suppose the last point x_n be the most important and choose $x_n = f(t_n) = 1$, and the first point x_1 be the most least important and choose $s_1 = f(t_1) = \sigma$. If we want to let fuzzy membership be a linear function of the time, we can select

$$s_i = f(t_i) = \alpha t_i + b = \frac{1-\sigma}{t_n - t_1} t_i + \frac{t_n \sigma - t_1}{t_n - t_1} \tag{3}$$

If we want to make fuzzy membership be a quadric function of the time, we can select

$$s_i = f(t_i) = \alpha(t_i - b)^2 + c = (1-\sigma)\left(\frac{t_i - t_1}{t_n - t_1}\right)^2 + \sigma \tag{4}$$

2.4 Performance Criteria

The prediction performance is evaluated using the normalized mean squared error (NMSE). NMSE is the measures of the deviation between the actual and predicted values. The smaller the values of NMSE, the closer are the predicted time series values to the actual values. The NMSE of the test set is calculated as follows:

$$\text{NMSE} = \tfrac{1}{\delta^2 n} \sum_{i=1}^{n} (y_i - \hat{y_i})^2 , \tag{5}$$

$$\delta^2 = \tfrac{1}{n-1} \sum_{i=1}^{n} (y_i - \overline{y})^2 , \tag{6}$$

where n represents the total number of data points in the test set. $\hat{y_i}$ represents the predicted value. $\overline{y}$ denotes the mean of the actual output values.

2.5 Kernel Function Selection and Parameters Tuning

The literatures [9, 10] show that RBF kernel usually get better results than others and use it as the default kernel in predicting time series data. In our experiment,We use general RBF as the kernel function. Comparative results of Goods A between different kernels are shown in Table 2.

There are two parameters while using RBF kernels: C and γ. We use a grid-search on C and γ using cross-validation. We found that trying exponentially growing sequences of C and γ is a practical method to identify good parameters.

Table 2. Results of forecasting with different kernels on Goods A. $\varepsilon = 0.1$.

Kernels	Parameter	NMSE Training	NMSE Testing	Time
Poly	$C = 8, d = 1$	0.012	0.401	0.211
RBF	$C = 8, \gamma = 0.25$	0.008	0.317	0.102
Sigmoid	$C = 8, \gamma = 0.0625$	0.021	0.474	0.176

3 Experimental Results

Table 3 shows the averaged NMSE values of EMA, standard SVM compared with FSVMR. Figure 1 illustrates the predicted and actual values of Goods A in testing. By computing of standard deviation, FSVMR's accuracy is 6.6% and 14.3% higher than standard SVM and EMA respectively.

Table 3. Averaged Forecasting Results for All 5 Goods

Methods	EMA	Standard SVMs	FSVMR
NMSE	0.3610	0.3313	0.3095

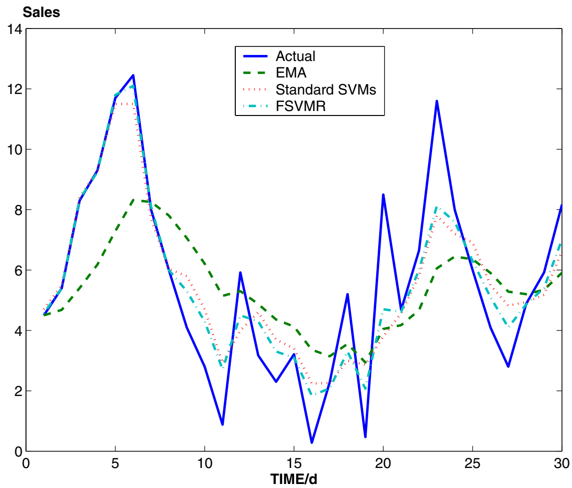

Fig. 1. Forecasting results comparison for Goods A

Acknowledgements

This research is granted by NSFC Project No.70401015 and Hubei Provincial Key Social Science Research Center of Information Management.

References

1. Fildes R, Hastings R. The organization and improvement of market forecasting. Journal of Operation Research Society, 45(1994) 1-16.
2. Lawrence M, O'Connor M. Sales forecasting updates: how good are they in practice? International Journal of Forecasting,16(2000) 369-382.
3. Müller K.R., Smola A.J., Rätsch G., Schölkopf B., Kohlmorgen J., and Vapnik V. Prediction time series with support vector machies. Lecture Notes in Computer Science Vol.1327,1997,999-1004.
4. Francis, E.H.T. and Cao L. Application of support vector machines in financial time series forecasting. Omega: International Journal of Management Science, 29(2001) 309-317.
5. Bao YK, Lu YS and Zhang JL. Forecasting stock price by SVMs regression. Lecture Notes in Artifical Intelligence, Vol.3192,2004:295-303.
6. Chun-Fu Lin, Sheng-De Wang, Fuzzy Support Vector Machines, IEEE Trans on Neural Networks. 13(2002)464-471.

7. Dug H. H, Changha H.. Support vector fuzzy regression machines.Fuzzy Sets and Systems, 138 (2003) 271-281.
8. Bao YK, Zou H and Liu ZT. Forecasting intermittent demand by fuzzy support vector machines. Lecture Notes in Artifical Intelligence, Vol.4031,2006:1080-1089.
9. T. B. Trafalis and H. Ince. Support vector machine for regression and applications to financial forecasting. In Proceedings of the IEEE-INNS-ENNS International Joint Conference, 2000:348-353.
10. Kim K.J. Financial time series forecasting using support vector machines. Neurocomputing, 55(2003) 307-319.