Expert Systems with Applications 37 (2010) 4261–4265

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Time series forecasting by a seasonal support vector regression model

Ping-Feng Pai [a,*], Kuo-Ping Lin [b], Chi-Shen Lin [c], Ping-Teng Chang [c]

[a] Department of Information Management, National Chi Nan University, University Rd., Puli, Nantou 545, Taiwan
[b] Department of Information Management, Lunghwa University of Science and Technology, Taoyuan 333, Taiwan
[c] Department of Industrial Engineering and Enterprise Information, Box 985, Tunghai University, Taichung 407, Taiwan

ARTICLE INFO

Keywords:
Seasonal time series
Forecast
Support vector regression
Seasonal autoregressive integrated moving average

ABSTRACT

The support vector regression (SVR) model is a novel forecasting approach and has been successfully used to solve time series problems. However, the applications of SVR models in a seasonal time series forecasting has not been widely investigated. This study aims at developing a seasonal support vector regression (SSVR) model to forecast seasonal time series data. Seasonal factors and trends are utilized in the SSVR model to perform forecasts. Furthermore, hybrid genetic algorithms and tabu search (GA/TS) algorithms are applied in order to select three parameters of SSVR models. In this study, two other forecasting models, autoregressive integrated moving average (SARIMA) and SVR are employed for forecasting the same data sets. Empirical results indicate that the SSVR outperforms both SVR and SARIMA models in terms of forecasting accuracy. Thus, the SSVR model is an effective method for seasonal time series forecasting.

1. Introduction

The seasonal time series is a sequence of seasonal data points recorded sequentially in time. Over the past several decades, many works have been devoted to develop and improve seasonal time series forecasting models. The SARIMA model (Box & Jenkins, 1976) is one of the most popular approaches in seasonal time series forecasting. The SARIMA model has been successfully utilized in many fields of forecasting, such as a soil dryness index (Li, Campbell, Haswell, Sneeuwjagt, & Venables, 2003), predicting tourism demand (Goh & Law, 2002; Huang & Min, 2002) and municipal solid waste management (Navarro-Esbrı́, Diamadopoulos, & Ginestar, 2002). However, the seasonal time series is a complex and nonlinear problem. The artificial neural network (ANN) model is an alternative in forecasting seasonal data pattern (Zhang & Qi, 2005). Some literature (Nam & Schaefer, 1995; Pai & Hong, 2005; Tang & Fishwick, 1993; Williams, 1997) indicted that ANN can obtain desirable results in seasonal and trend forecasting. With the introduction of Vapnik's ε-insensitive loss function, SVR (Vapnik, Golowich, & Smola, 1996) has been extended so as to solve forecasting problems and has provided many exciting results. In recent years, SVR schemes have been extended to cope with forecasting problems, and have provided many promising results in customer demand (Levis & Papageorgiou, 2005), finance (Huang, Nakamori, & Wang, 2005; Kim, 2003; Tay & Cao, 2002) intermittent demand (Hua & Zhang, 2006), tourism demand (Pai & Hong, 2005), air quality (Lu & Wang, 2005), wind speed (Mohandes, Halawani, Rehman, & Hussain, 2004), plant control systems (Xi, Poo, & Chou, 2007), rainfall (Hong & Pai, 2007), prices for the electricity market (Gaoa, Bompard, Napoli, & Cheng, 2007) and flood control (Yu, Chen, & Chang, 2006). However, applications of the SVR models in seasonal time series data have not been widely studied. Therefore, this study attempts to develop a SSVR model for exploiting the unique strength of the decomposition techniques and for the SVR model in the seasonal time series forecasting problems. The rest of this paper is organized as follows. The SSVR model is introduced in Section 2. In Section 3, two numerical examples are utilized to demonstrate performances of different forecasting models. Finally, conclusions are made in Section 4.

2. Forecasting models

2.1. Support vector regression

The support vector regression technique is based on the structured risk minimization principle. Rather than finding empirical errors, SVR aim to minimize an upper bound of the generalization error. A regression function is generated by applying a set of high dimensional linear functions. Hence, given a set of data $(x_i, A_i)_{i=1}^{N}$ (where x_i is the input vector; A_i is the actual value, and N is the total number of data patterns), the regression function is expressed as follows:

$$G = w\phi(x_i) + b \tag{1}$$

* Corresponding author. Tel.: +886 49 2915205.
E-mail addresses: paipf@ncnu.edu.tw (P.-F. Pai), kplin@mail.lhu.edu.tw (K.-P. Lin), d933309@thu.edu.tw (C.-S. Lin), ptchang@ie.thu.edu.tw (P.-T. Chang).

0957-4174/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2009.11.076

where $\phi(x_i)$ denotes the feature of the inputs, and w and b are coefficients. The coefficients (w_i and b) are calculated by minimizing the regularized risk function shown as Eq. (2).

$$P(G) = C\frac{1}{N}\sum_{i=1}^{N} L_\varepsilon(A_i, G_i) + \frac{1}{2}\|w\|^2 \tag{2}$$

where

$$L_\varepsilon(A_i, G_i) = \begin{cases} 0 & \text{if } |A_i - G_i| \leqslant \varepsilon \\ |A_i - G_i| - \varepsilon & \text{otherwise} \end{cases} \tag{3}$$

where C and ε are user-defined parameters. The parameter ε is the difference between actual values and values calculated from the regression function. This difference can be treated as a tube around the regression function. The points outside the tube are viewed as training errors. In Eq. (2), $L_\varepsilon(A_i,G_i)$ is called an ε-insensitive loss function.

The loss equals zero if the forecasted value is within the ε-tube. Furthermore, the second item of Eq. (2), $\frac{1}{2}\|w\|^2$, is used to estimate the flatness of a function. Thus, C is a parameter determining the trade-off between the empirical risk and the model flatness. Two positive slack variables (ξ_i and ξ_i^*), representing the distance from actual values to the corresponding boundary values of the ε-tube, are then introduced. These two slack variables equal zero when the data points fall within the ε-tube. Eq. (2) is then reformulated into the following constrained form:

$$\begin{aligned} \text{Min}: \quad & \frac{1}{2}\|w\|^2 + C\left(\sum_{i=1}^{N}(\xi_i + \xi_i^*)\right) \\ \text{subjective to} \quad & w\phi(x_i) + b - A_i \leqslant \varepsilon + \xi_i^*, \quad i = 1,2,\ldots,N \\ & A_i - w\phi(x_i) - b \leqslant \varepsilon + \xi_i, \quad i = 1,2,\ldots,N \\ & \xi_i, \xi_i^* \geqslant 0, \quad i = 1,2,\ldots,N \end{aligned} \tag{4}$$

This constrained optimization problem is solved by the following primal Lagrangian form:

$$\begin{aligned} \text{Min} \quad & \frac{1}{2}\|w\|^2 + C\left(\sum_{i=1}^{N}(\xi_i + \xi_i^*)\right) \\ & - \sum_{i=1}^{N}\beta_i[w\phi(x_i) + b - A_i + \varepsilon + \xi_i] \\ & - \sum_{i=1}^{N}\beta_i^*[A_i - w\phi(x_i) - b + \varepsilon + \xi_i^*] - \sum_{i=1}^{N}(\alpha_i\xi_i + \alpha_i^*\xi_i^*) \end{aligned} \tag{5}$$

Eq. (5) is minimized with respect to primal variables w, b, ξ, and ξ^*, and is maximized with regard to non-negative Lagrangian multipliers α_i, α_i^*, β_i, and β_i^*. Finally, Karush–Kuhn–Tucker conditions are applied to Eq. (4), and the dual Lagrangian form is given by Eq. (6).

$$\begin{aligned} \text{Max} \quad & \sum_{i=1}^{N}A_i(\beta_i - \beta_i^*) - \varepsilon\sum_{i=1}^{N}(\beta_i + \beta_i^*) \\ & -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}(\beta_i - \beta_i^*)(\beta_j - \beta_j^*)K(x_i, x_j) \\ \text{subjective to} \quad & \sum_{i=1}^{N}(\beta_i - \beta_i^*) = 0 \\ & 0 \leqslant \beta_i \leqslant C, \quad i = 1,2,\ldots,N \\ & 0 \leqslant \beta_i^* \leqslant C, \quad i = 1,2,\ldots,N \end{aligned} \tag{6}$$

The Lagrange multipliers in Eq. (6) satisfy the equality $\beta_i * \beta_i^* = 0$. The Lagrange multipliers, β_i and β_i^*, are determined, and an optimal weight vector of the regression hyperplane is represented by Eq. (7).

$$w^* = \sum_{i=1}^{N}(\beta_i - \beta_i^*)K(x, x_i) \tag{7}$$

Thus, the regression function is expressed by:

$$G(x, \beta, \beta^*) = \sum_{i=1}^{N}(\beta_i - \beta_i^*)K(x, x_i) + b \tag{8}$$

$K(x_i,x_j)$ is a Kernel function whose value equals the inner product of two vectors, x_i and x_j, in the feature space $\phi(x_i)$ and $\phi(x_j)$, meaning that $K(x_i,x_j) = \phi(x_i)^* \phi(x_j)$. Any function that satisfies Mercer's condition (Mercer, 1909) can serve as the Kernel function. This study employs the Gaussian function.

2.2. GA/TS algorithms for parameter selection of SVR models

In this study, the GA/TS (Glover, Kelly, & Laguna, 1995) algorithm with binary coding was employed to determine parameters of SSVR models. The procedure of GA/TS is illustrated as follows.

Step 1 (*Initialization*): Establish randomly an initial population of chromosomes. Three parameters, σ, C and ε, are expressed in a binary format; and represented by a chromosome.

Step 2 (*Evaluating fitness*): Evaluate the fitness of each chromosome. In this study, the negative MAPE was used as the fitness function and expressed as follows:

$$Fitness = -MAPE(\%) = -\frac{100}{M}\sum_{t=1}^{M}\left|\frac{g_t - z_t}{g_t}\right| \tag{9}$$

where M is the number of forecasting periods; g_t is the actual value at period t; and z_t is the forecasting value at period t.

Step 3 (*Selection*): Based on the fitness functions, chromosomes with higher fitness values are more likely to yield offspring in the next generation. The roulette wheel selection principle (Holland, 1975) is applied to select chromosomes for reproduction.

Step 4 (*Crossover* and mutation): Create new offspring by performing crossover and mutation operations. Mutations were performed randomly by converting a "1" bit into a "0" bit or a "0" bit into a "1" bit. In this study, the single-point-crossover principle was employed. Segments of paired chromosomes between two determined breakpoints are exchanged. The probabilities of crossover and mutation were set at 0.5 and 0.1, respectively.

Step 5 (*Perform* TS on each chromosome): Evaluating neighbor chromosome and adjusting the tabu list. The tabu list size is 20 in this study. The chromosome with the smallest MAPE value and not having been recorded in the tabu list was placed in the tabu list. In this study, a first-in-first-out policy is performed for operating the tabu list. If the best neighbor chromosome is the same as one of the chromosome in the tabu list, then it generates the next set of neighbor chromosomes and calculates the fitness value of chromosome. The next set of neighbor chromosome is generated from the best neighbor chromosome in the current iteration.

Step 6 (*Current* chromosome selection by TS): If the best neighbor chromosome is better then the current chromosome, then the current chromosome is replaced by the best neighbor chromosome. Otherwise, keep the current chromosome.

Step 7 (*Next generation*): Form a population for the next generation. The size of the population was set to 50.

Step 8 (*Stop criterion*). If the number of epochs equals a given scale, then the best chromosomes are presented as a solution; otherwise go back to Step 2. The number of epochs was set to 1000.