



# Journal of Applied Sciences

ISSN 1812-5654

**science**  
alert

**ANSI***net*  
an open access publisher  
<http://ansinet.com>

## A Comparison of Time Series Forecasting using Support Vector Machine and Artificial Neural Network Model

<sup>1</sup>R. Samsudin, <sup>2</sup>A. Shabri and <sup>1</sup>P. Saad

<sup>1</sup>Department of Software Engineering, Faculty of Computer Science and Information System, Universiti Teknologi Malaysia, 81310, Skudai, Malaysia

<sup>2</sup>Department of Mathematic, Faculty of Science, Universiti Teknologi Malaysia, 81310, Skudai, Malaysia

---

**Abstract:** Time series prediction is an important problem in many applications in natural science, engineering and economics. The objective of this study is to examine the flexibility of Support Vector Machine (SVM) in time series forecasting by comparing it with a multi-layer back-propagation (BP) neural network. Five well-known time series data sets are used in this study to demonstrate the effectiveness of the forecasting model. These data are utilized to forecast through an application aimed to handle real life time series. The grid search technique using 10-fold cross validation is used to determine the best value of SVM parameters in the forecasting process. The experiment shows that SVM outperforms the BP neural network based on the criteria of Mean Absolute Error (MAE). It also indicates that SVM provides a promising technique in time series forecasting techniques.

**Key words:** Time series, neural networks, support vector machine, forecasting, back propagation

---

### INTRODUCTION

Time series analysis and forecasting is an active research area over the last five decades. Thus, various kinds of forecasting models are developed and researchers have relied on statistical techniques to predict time series data. The accuracy of time series forecasting is fundamental to many decisions processes and hence the research for improving the effectiveness of forecasting models has never been stopped (Zhang, 2003). Successful time series prediction is a major goal in many areas of application from economic and business planning, inventory and production control, weather forecasting, signal processing and control. In the past, conventional statistical methods are employed to forecast time series data. However, the data time series are often full of nonlinearity and irregularity.

To address this, numerous artificial techniques, such as Artificial Neural Network (ANN) are proposed to improve the prediction results. In the last decade, Artificial Neural Network (ANN) are being used more frequently in the analysis of time series forecasting, pattern classification and pattern recognition capabilities (Haoffi *et al.*, 2007; Sharda, 1994; Wu *et al.*, 2005; Zou *et al.*, 2007). The ANN provides an attractive

alternative tool for both forecasting researchers and has shown their nonlinear modeling capability in data time series forecasting.

Recently, the Support Vector Machine (SVM) method, which was first suggested by Vapnik (1995), has recently been used in a range of applications such as in data mining, classification, regression and time series forecasting (Cao and Tay, 2001; Flake and Lawrence, 2002; Zhao *et al.*, 2006). The ability of SVM to solve nonlinear regression estimation problems makes SVM successful in time series forecasting. It has become a hot topic of intensive study due to its successful application in classification and regression tasks.

The main purpose of this study was to investigate the applicability and capability of the SVM methods for modeling of time-series forecasting. To verify the application of this approach, the benchmarked data are used in this study. The well known data are five well-known data sets that always handled in real life time series application. There are the chemical process concentration, the IBM common stock closing prices, the chemical process temperature, the Wolf's sunspot data and the international airline passengers. The prediction result by SVM method is compared with those by ANN.

---

**Corresponding Author:** Ruhaidah Samsudin, Department of Software Engineering, Faculty of Computer Science and Information System, Universiti Teknologi Malaysia, 81310, Skudai, Malaysia Tel: 607-5532228 Fax: 607-5565044

**MATERIALS AND METHODS**

**Support vector machines:** The basic idea of SVM for function approximation is mapping the data  $x$  into a high-dimensional feature space by a nonlinear mapping and then performing a linear regression in the feature space. Consider a given training set of  $n$  data points  $\{x_i, y_i\}_{i=1}^n$  with input data  $x_i \in \mathbb{R}^p$ ,  $p$  is the total number of data patterns) and output  $y_i \in \mathbb{R}$ . The SVM approximate the function in the following form:

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

where,  $\phi(x)$  represents the high dimensional feature spaces, which is nonlinearly mapped from the input space  $x$ . The coefficient are  $w$  and  $b$  are estimated by minimizing the regularized function:

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

$$L_\epsilon(d_i, y_i) = \begin{cases} |d_i - y_i| - \epsilon, & |d_i - y_i| \geq \epsilon \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

To obtain the estimation of  $w$  and  $b$ , Eq. 2 is transformed to the primal function given by Eq. 4 by introducing the positive slack variables  $\xi$  and  $\xi^*$  as follows:

$$\text{Minimize } R(w, \xi^*) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*) \tag{4}$$

$$\text{subject to } \begin{cases} d_i - w\phi(x_i) - b \leq \epsilon + \xi_i \\ w\phi(x_i) + b - y_i \leq \epsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \end{cases}$$

The first term  $(1/2) \|w\|^2$  is the weights vector norm,  $d_i$  the desired value and  $C$  is referred as regularized constant determining the tradeoff between the empirical error and the regularized term.  $\epsilon$  is called the tube size of SVM and it is equivalent to the approximation accuracy placed on the training data points. Here, the slack variables  $\xi$  and  $\xi^*$  are introduced. By introducing Lagrange multipliers and exploiting the optimality constraints, the decision function given by Eq. 1 has the following explicit form:

$$y(x) = \sum_{i=1}^n (a_i - a_i^*) K(x, x_i) + b \tag{5}$$

In Eq. 5,  $a_i$  and  $a_i^*$  are the so-called Lagrange multipliers. They satisfy the equalities  $a_i \times a_i^* = 0$ ,  $a_i \geq 0$  and

$a_i^* \geq 0$  where,  $i = 1, 2, \dots, n$  and are obtained by maximizing the dual function of Eq. 4 which has the following form:

$$R(a_i, a_i^*) = \sum_{i=1}^n d_i (a_i - a_i^*) - \epsilon \sum_{i=1}^n (a_i + a_i^*) - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (a_i - a_i^*) (a_j - a_j^*) K(x_i, x_j) \tag{6}$$

with the constraints

$$\sum_{i=1}^n a_i = \sum_{i=1}^n a_i^*$$

$$0 \leq a_i \leq C, i = 1, 2, \dots, n$$

$$0 \leq a_i^* \leq C, i = 1, 2, \dots, n$$

$K(x_i, x_j)$  is defined as the kernel function. The value of the kernel is equal to the inner product of two vectors  $X_i$  and  $X_j$  in the feature space  $\phi(x_i)$  and  $\phi(x_j)$ , that is,  $K(x_i, x_j) = \phi(x_i) \times \phi(x_j)$ . The typical examples of the kernel function are as follows:

Linear:  $K(x_i, x_j) = x_i^T x_j$

Sigmoid:  $K(x_i, x_j) = \tanh(\gamma x_i^T x_j + r)$

Polynomial:  $K(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \gamma > 0$

Radial basis function (RBF):  $K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2), \gamma > 0$

Here,  $\gamma$ ,  $r$  and  $d$  are kernel parameters. The kernel parameters should be carefully chosen as it implicitly defines the structure of the high dimensional feature space  $\phi(x)$  and thus controls the complexity of the final solution. The SVM architecture is shown in Fig. 1.

**The artificial neural network:** An Artificial Neural Network (ANN) is a computer system that simulates the learning process of human brain. The greatest advantage of a neural network is its ability to model complex nonlinear in the data series. The model usually consists of three layers: the first layer is the input layer where the data are introduced to the network, the second layer is the hidden layer where data are processed and the last layer is the output layer where the results of given input are produced. The structure of a feed-forward ANN is shown in Fig. 2.

The output of the ANN assuming a linear output neuron  $j$ , a single hidden layer with  $h$  sigmoid hidden nodes and an input variable ( $x_i$ ) is given by:

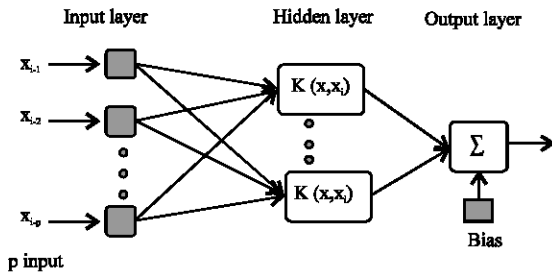


Fig. 1: SVM Architecture

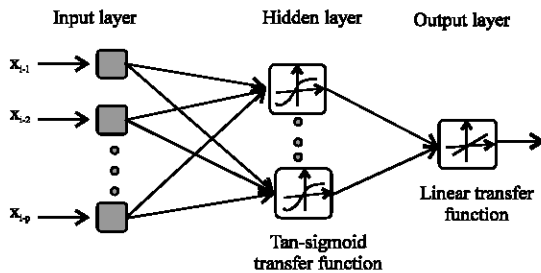


Fig. 2: Architecture of three layers feed-forward back-propagation ANN

$$x_i = g(\sum_{j=1}^h w_j f(s_j) + b_k) \quad (7)$$

where,  $g(\cdot)$  is the linear transfer function of the output neuron  $k$  and  $b_k$  is its bias,  $w_j$  is the connection weights between hidden layers and output units,  $f(\cdot)$  is the transfer function of the hidden layer. The transfer functions can take several forms and the most widely used transfer functions are:

$$\text{Log-sigmoid: } f(s_i) = \text{logsig}(s_i) = \frac{1}{1 + \exp(-s_i)}$$

$$\text{Linear: } f(s_i) = \text{purelin}(s_i) = s_i$$

$$\text{Hyperbolic tangent sigmoid: } f(s_i) = \text{tansig}(s_i) = \frac{2}{1 + \exp(-2s_i)} - 1$$

where:

$$s_i = \sum_{i=1}^n w_i x_i$$

is the input signal referred to as the weighted sum of incoming information.

Training a network is an essential factor for the success of the neural networks. Among the several learning algorithms available, back-propagation (BP) has been the most popular and most widely implemented learning algorithm for all neural network paradigms (Zou *et al.*, 2007). The procedure of the BP is repeated by

adjusting the weights of the connection in the network so as to minimize the error. The gradient descent method is utilized to calculate the weight of the network and adjust the weight of interconnection to minimize the sum-squared error, SSE, of the network which is given by:

$$SSE = \frac{1}{2} \sum_k (y_k - \hat{y}_k)^2 \quad (8)$$

where,  $y_k$  and  $\hat{y}_k$  are the true and predicted output vector of the  $k$ th output node.

For a univariate time series forecasting problem, the inputs of the network are the past lagged observations ( $x_{t-1}, x_{t-2}, \dots, x_{t-p}$ ) and the output is the predicted value ( $x_t$ ) (Zhang *et al.*, 2001). Hence the ANN of Eq. 2 can be written as:

$$x_t = g(x_{t-1}, x_{t-2}, \dots, x_{t-p}, w) + \varepsilon_t \quad (9)$$

where,  $w$  is a vector of all parameters and  $g(\cdot)$  is a function determined by the network structure and connection weights. A major advantage of neural networks is their ability to provide flexible nonlinear mapping between inputs and outputs. They can capture the nonlinear characteristics of time series well.

## RESULTS

**Data sets:** We examined the five well-known data sets in our experiment. These data sets are utilized to forecast through an application aimed to handle real life time series. These data are the chemical process concentration reading, the IBM common stock closing prices, the chemical process temperature reading, the Wolf's sunspot data and the international airline passengers are used in this study to demonstrate the effectiveness of the SVM model. These time series data come from different areas and have different statistical characteristics. They have been widely studied in the statistical as well as the ANN (Hamzaçebi *et al.*, 2009) and ARIMA literature (Brockwell and Davis, 2002). All series were classified into four main categories that encompass the majority of the time series types, namely seasonal and trended, seasonality, trended and nonlinear. Information regarding the series distributed among the training and testing sets are shown in Table 1. The training set is used for model selection and parameter optimization, being the test set used to compare the proposed approach with other models. However, currently there is no suggested systematic way to determine ratio of splitting for training and testing of the data set. Most studies in the literatures use convenient ratio of splitting

for training and testing such as 70:30%, 80:20%, or 90:10%, etc. (Zou *et al.*, 2007). In this study, we select the ratio 85:15%.

**Chemical process concentration reading for every 2 h:** A very interesting problem is the prediction of data that are not generated from a mathematical expression and thus governed by a particular determinism. For this purpose, we used a natural phenomenon, the chemical process concentration reading for every 2 h has a total of 197 data (Box *et al.*, 1994). The series was collected to show the effect on the output of uncontrolled and unmeasured disturbances. The first 167 data pairs of the series were used as training data, while the remaining 30 (15% of the data) were used to test the model identified. A time series plot of chemical process concentration reading for every 2 h appears in Fig. 3. The series shown in Fig. 3, exhibits certain regularities and suggests the absence of a homogeneous non-stationary pattern

**IBM common stock closing prices:** This time series is a real series of the daily data from May 17, 1961 to November 2, 1962. Figure 4 shows the daily closing prices of the IBM common stock during the study period. As shown in Fig. 4, the IBM share prices show a break in the last third of the series and no obvious trend or seasonality. This series has been analyzed by Hamzaçebi *et al.* (2009).

Table 1: The series data that are used to compare forecast methods

Series	Data	Size	Training set	Test set	Period
A	Chemical process concentration	197	167	30	2 h
B	IBM common stock closing prices	369	319	50	Daily
C	Chemical process viscosity	310	265	45	1 h
D	Wolf's sunspots No.	100	85	15	1 year
E	International airline passengers	144	120	24	1 month

**Chemical process viscosity reading for every hour:** This series has 310 data and also has been analyzed by Hamzaçebi *et al.* (2009). Figure 5 shows this time series and referred to as the 'uncontrolled' series (Box *et al.*, 1994).

**Wolf's sunspots numbers:** The sunspot data contains the annual number of sunspots from 1770 to 1869, giving a total of 100 observations. The study of sunspot activity has practical importance to geophysicists, environment scientists and climatologists. The plot of this time series (Fig. 6) suggests that there is a cyclical pattern with the mean cycle of about 11 years. This series also has been analyzed by Hamzaçebi *et al.* (2009).

**International airline passengers:** The airline passenger data set contains the monthly totals of international airline passengers from January 1949 to December 1960, has 144 data. As shown in Fig. 7, the series is affected by increasing trend and seasonal changes. The series is an example of time series data with a clear trend and multiplicative seasonality. The data has long served as a benchmark and has been well studied in statistical literature (Ghiassi *et al.*, 2005; Shabri, 2001).

**Performance criteria:** The performances of the each model for both the training data and forecasting data are evaluated according to the Root-Mean-Square Error (RMSE), Mean Absolute Error (MAE) and correlation coefficient (R) which are widely used for evaluating results of time series forecasting. The RMSE, MAE and R are defined as:

$$RMSE = \sqrt{\frac{1}{N} \sum_{t=1}^N (y_t - \hat{y}_t)^2} \tag{10}$$

$$MAE = \frac{1}{N} \sum_{t=1}^N |y_t - \hat{y}_t| \tag{11}$$

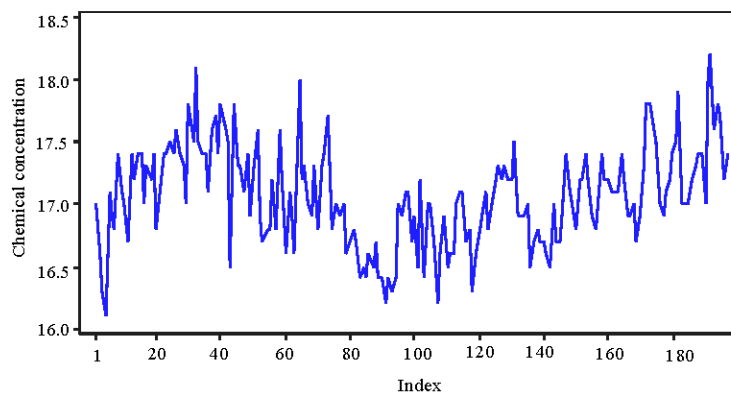


Fig. 3: Chemical process concentration reading series: every 2 h

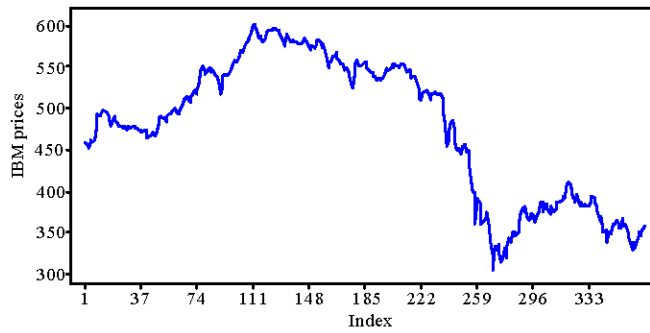


Fig. 4: IBM common stock closing prices data series (May 17 1961-Nov. 1962)

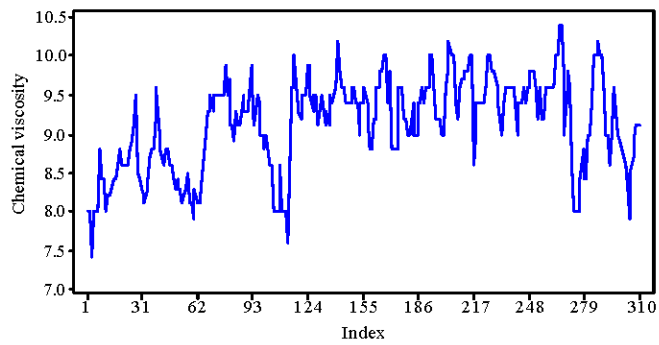


Fig. 5: Chemical process viscosity reading: every hour

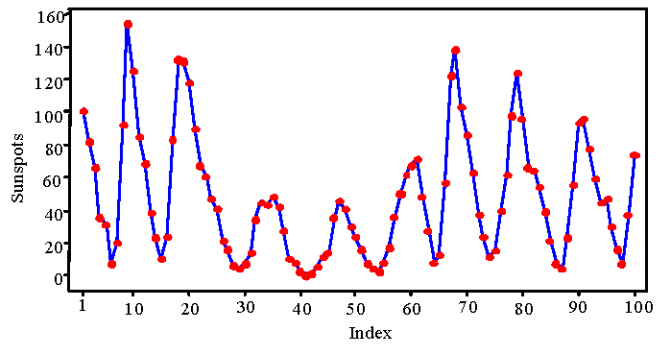


Fig. 6: Wolfer Sunspots Numbers data series (1770-1869)

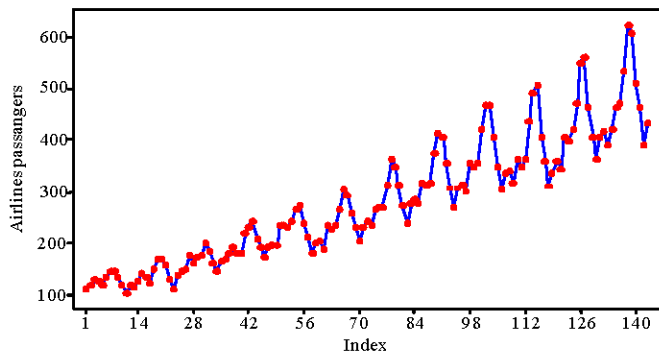


Fig. 7: International airline passengers series (1949-1960)

$$R = \frac{\frac{1}{n} \sum_{i=1}^n (y_i - \bar{y})(o_i - \bar{o})}{\sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \bar{y})^2} \sqrt{\frac{1}{n} \sum_{i=1}^n (o_i - \bar{o})^2}} \quad (12)$$

where  $O_i$  and  $y_i$  are the observed and forecasted values at data point  $i$ , respectively,  $\bar{o}$  is the mean of the observed values and  $N$  is the number of data points. The criterions to judge for the best model are relatively small of RMSE and MAE in the modeling and forecasting. Correlation coefficient measures how well the flows predicted correlate with the flows observed. Clearly, the  $R$  value close to unity indicates a satisfactory result, while a low value or close to zero implies an inadequate result.

### FITTING NEURAL NETWORK MODELS TO THE DATA

One of the most important steps in developing a satisfactory forecasting model such as ANN and SVM models is the selection of the input variables. In this study, the input structures having various input variables are determined by setting the input layer nodes equal to the number of the lagged variables from river flow data,  $(x_{t-1}, x_{t-2}, \dots, x_{t-p})$  where,  $p$  is time delay. The time delay is 2, 4, 6, 8, 10 and 12. The input structures of the models for forecasting time series is shown in Table 2.

In this study, a typical three-layer feed-forward ANN model is constructed for time series forecasting. The training and testing data were normalized in the range zero to one. From the input layer to the hidden layer, the hyperbolic tangent sigmoid transfer function has been commonly used in time series was applied. From the hidden layer to the output layer, a linear function was employed as the transfer function because the linear function is known to be robust for a continuous output variable.

The network was trained for 5000 epochs using the conjugate gradient descent back-propagation algorithm with a learning rate of 0.001 and a momentum coefficient of 0.9. The six models (M1-M6) having various input structures are trained and tested by ANN models and the optimal number of neuron in the hidden layer was identified using several practical guidelines. These includes using  $I/2$  (Kang, 1991),  $2I$  (Wong, 1991) and  $2I+1$  (Lippmann, 1987), where,  $I$  is the number of input.

The networks that yielded the best results for the forecasting set were selected as the best ANN for the corresponding series. The best ANN structures and the best results for the training and forecasting are shown in Table 3.

In this study, a trial and error method is performed to optimize the number of neurons in the hidden layer. Table 3 shows the performance of ANN varying with the

Table 2: The input structure of the models for forecasting of time series data

Model	Input structure
M1	$x_t = f(x_{t-1}, x_{t-2})$
M2	$x_t = f(x_{t-1}, x_{t-2}, x_{t-3}, x_{t-4})$
M3	$x_t = f(x_{t-1}, x_{t-2}, x_{t-3}, x_{t-4}, x_{t-5}, x_{t-6})$
M4	$x_t = f(x_{t-1}, x_{t-2}, x_{t-3}, x_{t-4}, x_{t-5}, x_{t-6}, x_{t-7}, x_{t-8})$
M5	$x_t = f(x_{t-1}, x_{t-2}, x_{t-3}, x_{t-4}, x_{t-5}, x_{t-6}, x_{t-7}, x_{t-8}, x_{t-9}, x_{t-10})$
M6	$x_t = f(x_{t-1}, x_{t-2}, x_{t-3}, x_{t-4}, x_{t-5}, x_{t-6}, x_{t-7}, x_{t-8}, x_{t-9}, x_{t-10}, x_{t-11}, x_{t-12})$

Table 3: Comparison of neural network model

Series	Neural networks structure			Training		Training	
	Input	Hidden	Output	RMSE	MAE	RMSE	MAE
A	2	5	1	0.0175	0.0137	0.0211	0.0168
B	6	13	1	0.0158	0.0118	0.0146	0.0112
C	2	4	1	0.0275	0.0200	0.0368	0.0293
D	4	4	1	0.0773	0.0580	0.0711	0.0571
E	12	6	1	0.0337	0.0270	0.0693	0.0558

number of neurons in the hidden layer. The best neural networks structure found consists of three layers: 2-5-1 for series A, 6-13-1 for series B, 2-4-1 for series C, 4-4-1 for series D and 12-6-1 for series E.

**Fitting SVM models to the data:** There is no theory that can used to guide the selection the optimal number of input nodes of the LSSVM model. In the training and testing of SVM model, the same input structures of the data set (M1-M6) are used. Model selection and parameter search play a crucial role in the performance of the SVM. It is well known that SVM generalization performance (estimation accuracy) depends on a good setting of hyper-parameters  $C$ ,  $\epsilon$  and the kernel parameters.

This study focuses on the use of the RBF kernel for its good performance and advantages in time series forecasting problem proved in past research (Ding *et al.*, 2008; Eslamian *et al.*, 2008; Wang *et al.*, 2009). The RBF kernel nonlinearly maps samples into a higher dimensional space can handle nonlinear problem, influences the complexity of model selection and has less numerical difficulties.

One of most popular techniques of evaluating a set of parameter values is the use of cross validation. In cross validation, the training set is randomly divided into a number of  $V$  folds ( $S_1, S_2, \dots, S_V$ ). The same type of SVM analysis is then successively applied to the observations belonging to the  $V-1$  folds. The results of the analyses are applied to sample  $V$  (the sample or fold that was not used to fit the SVM model; i.e., this is the testing sample) to compute the error usually defined as the sum-of-squared. The average accuracy of these  $V$  times is used to yield a single measure model error of the stability of the respective model, i.e., the validity of the model for predicting unseen data.

For implementation of the SVM it is necessary to determine approximate values of optimal hyper parameters

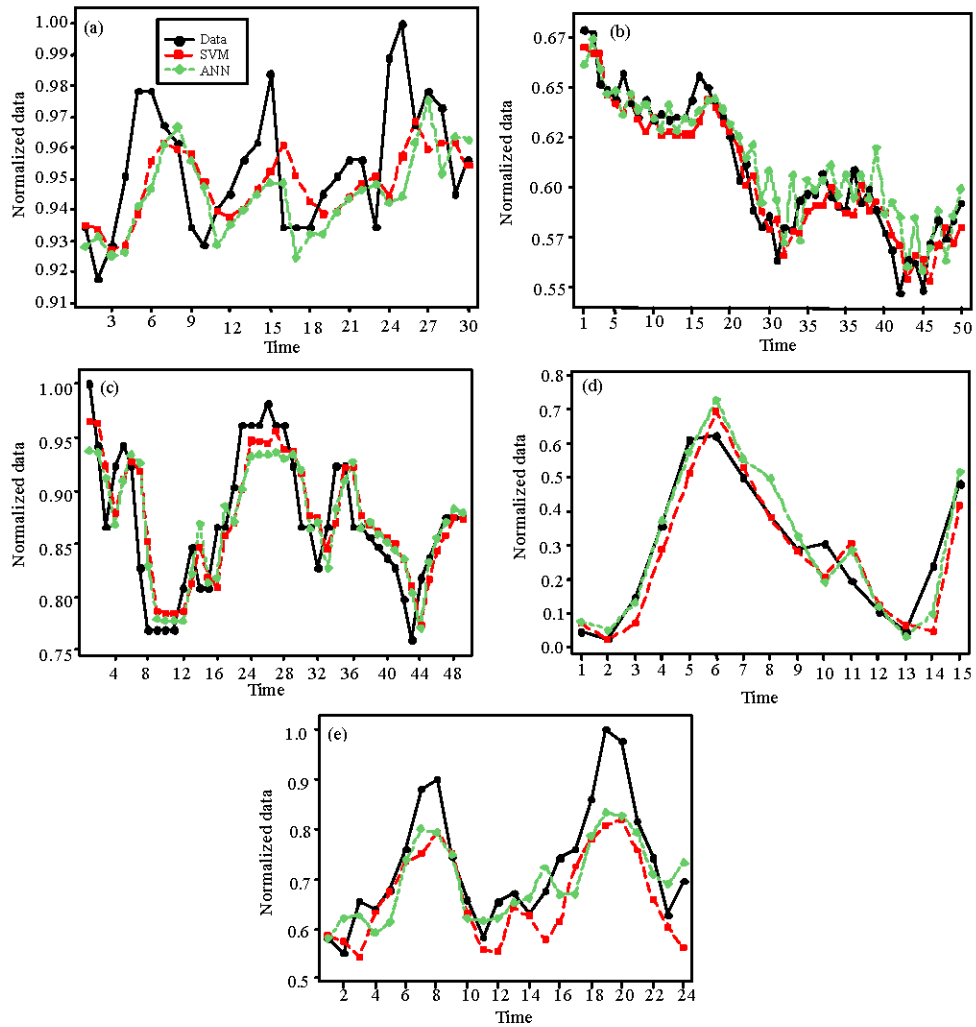


Fig. 8: (a-e) Forecasting performances of fitted models of series A-E

$C$ ,  $\epsilon$  and  $\gamma$ . Therefore, it needs some trial and error procedure. For each problem, we estimate the generalized accuracy using different kernel hyper parameters  $C$ ,  $\epsilon$  and  $\gamma$ . The ranges of search was set to  $[1, 10]$  at increment of 1.0 for  $C$  and  $[0.1, 0.5]$  at increment of 0.1 for  $\epsilon$  with  $\gamma$  being fixed to be 0.5. Therefore, for each problem we try  $10 \times 5 = 50$  combinations. For each parameters  $C$ ,  $\epsilon$  and  $\gamma$  in the search space, conduct 10-fold cross validation on the training set. The optimal values of  $C$ ,  $\epsilon$  and  $\gamma$  are selected using 10-fold cross-validation repeated ten times to increase the reliability of the results. For the final predictions used in the competition, we used all the available data to train the models with the parameters producing minimum validation errors. The parameters value that yields the minimum generalization error is, then, chosen. The best input structure, the

suitable parameters and the best result for the different time series data sets in training and forecasting phase are shown in Table 4.

The experimental result shows that the prediction performance of SVM is sensitive to the parameters  $C$ ,  $\epsilon$  and  $\gamma$  and the number of inputs structure. In Table 4, the results of SVM show the best testing and forecasting performances when  $C$  is 10 and  $\epsilon$  is 0.1 on the most data set.

For comparison purpose, the training and the forecast performance of SVM were compared with the ANN model. Figure 8a-e shows the performances of the ANN and SVM models during the forecasting periods for 5 data series. Based on the Fig. 8, the ANN and SVM forecast are closely to actual values. It show that the both approaches work well for the data set used.



**Table 4: The optimal svm parameters for different time series**

Series	Input structure	SVM parameters			Training		Forecasting	
		C	$\epsilon$	$\gamma$	RMSE	MAE	RMSE	MAE
A	4	10	0.4	0.25	0.0164	0.0125	0.0198	0.0157
B	4	10	0.1	0.25	0.0129	0.0095	0.0118	0.0099
C	6	3	0.1	0.17	0.0266	0.0195	0.0359	0.0289
D	2	10	0.1	0.50	0.0868	0.0642	0.0777	0.0590
E	10	10	0.1	0.10	0.0334	0.0293	0.0866	0.0666

**Table 5: Comparative performance indices of the ANN and SVM models for five series data**

Series	Training				Forecasting			
	ANN	SVM	ANN	SVM	ANN	SVM	ANN	SVM
A	0.0175	0.0164	0.0137	0.0125	0.0211	0.0198	0.0125	0.0157
B	0.0158	0.0129	0.0118	0.0095	0.0146	0.0118	0.0095	0.0099
C	0.0275	0.0266	0.0200	0.0195	0.0368	0.0359	0.0195	0.0289
D	0.0773	0.0868	0.0580	0.0642	0.0711	0.0777	0.0642	0.0590
E	0.0337	0.0334	0.0270	0.0293	0.0693	0.0866	0.0293	0.0666

Table 5 shows the comparison of training/forecasting precision among the two approaches based on RMSE and MAE statistical measures. Empirical results on the five data set using two different models clearly reveal the efficiency of the SVM model. In terms of RMSE and MAE values, for series A, B and C, SVM performed better than the ANN model in training and forecasting phase. For series D, ANN has the lowest value of RMSE and MAE in the training phase. For forecasting data, the best value of RMSE was found for ANN model and the lowest RMSE were observed for SVM model. For series E in training phase, the lowest RMSE was calculated for SVM model. For forecasting data, the best value of RMSE and MAE were found for ANN model.

**CONCLUSION**

The application of SVM in various time series data forecasting is studied in this paper and the effect of the value of parameters C,  $\epsilon$  and  $\gamma$  in the SVM is investigated. The accuracy of the forecasts, in term of MAE has been compared with that of the same number of ANN model which are trained and applied to the same data set used by the SVM. These time series come from different location and have different statistical characteristics. These data utilized to forecast through an application aimed to handle real life time series. From the experimental results comparing the performance of two models, it indicates that SVM significantly outperform ANN in term of RMSE and MAE for series A, B and C, where series A and C represent homogeneous non stationary while series B shows no obvious trend or seasonality. For the data series has a cyclical pattern (for series D) or has a trend and multiplicative seasonality (for series E), ANN model obtained the best RMSE and MAE.

The application of SVM for time series forecasting is relatively new. The initial results shown in this research have clearly demonstrated the potential of this approach in predicting time series data. Future works can be centered on investigating the performance of SVM with different kernel functions and optimal hyper parameters of SVM forecasting model, which has the potential to improve the accuracy of the forecast. The research results demonstrate that SVM method is a promising alternative approach for the prediction of the time series data.

**ACKNOWLEDGMENTS**

This study has been funded by the Ministry of Science, Technology and Innovation (MOSTI), Malaysia with Grant Vote No. 79346. We also wish to thank Muda Agricultural Development Authority (MADA), Ministry of Agricultural and Agro Based Industry, Malaysia for providing necessary data in this study.

**REFERENCES**

Box, G.E.P., G.M. Jenkins and G.C. Reinsel, 1994. Time Series Analysis: Forecasting and Control. 3rd Edn., Prentice Hall, New Jersey, ISBN-13: 978-0130607744.

Brockwell, P.J. and R.A. Davis, 2002. Introduction to Time Series and Forecasting. 8th Edn., Springer, Berlin, ISBN: 978-0-387-95351-9, pp: 469.

Cao, L.J. and E.H. Tay, 2001. Support vector with adaptive parameters in financial time series forecasting. IEEE Trans. Neural Network, 14: 1506-1518.

Ding, Y., X. Song and Y. Zen, 2008. Forecasting financial condition of Chinese listed companies based on support vector machine. Expert Syst. Appl., 34: 3081-3089.

Eslamian, S.S., S.A. Gohari, M. Biabanaki and R. Malekian, 2008. Estimation of monthly pan evaporation using artificial neural networks and support vector machines. J. Applied Sci., 8: 3497-3502.

Flake, G.W. and S. Lawrence, 2002. Efficient SVM regression training with SMO. Machine Learning, 46: 271-290.

Ghiassi, M., H. Saidane and D.K. Zimbra, 2005. A dynamic artificial neural network model for forecasting time series events. Int. J. Forecasting, 21: 341-362.

Hamzaçebi, C., D. Akay and F. Kutay, 2009. Comparison of direct and iterative artificial neural network forecast approaches in multi-periodic time series forecasting. Expert Syst. Appl. Int. J., 36: 3839-3844.

Haoffi, Z., X. Guoping, Y. Fagting and Y. Han, 2007. A neural network model based on the multi-stage optimization approach for short-term food price forecasting in China. Expert Syst. Appl., 33: 347-356.

- Kang, S., 1991. An investigation of the use of feedforward neural network for forecasting. Ph.D. Thesis, Kent State University.
- Lippmann, R.P., 1987. An introduction to computing with neural nets. *IEEE Trans. ASSP Magazine*, 4: 4-22.
- Shabri, A.B., 2001. Comparison of time series forecasting methods using neural networks and box-jenkins model. *Matematika*, 17: 24-32.
- Sharda, R., 1994. Neural networks for the MS/OR analyst: An application bibliography. *Interfaces*, 24: 116-130.
- Vapnik, V.N., 1995. *The Nature of Statistical Learning Theory*. 1st Edn., Springer-Verlag, New York, ISBN: 0-387-94559-8.
- Wang, W.C., K.W. Chau, C.T. Chen and L. Qiu, 2009. A comparison of performance of several artificial intelligence methods for forecasting monthly discharge time series. *J. Hydrol.*, 374: 294-306.
- Wong, F.S., 1991. Time series forecasting using backpropagation neural networks. *Neurocomputing*, 2: 147-159.
- Wu, S. Jr., J. Han, S. Annambhotla and S. Bryant, 2005. Artificial neural networks for forecasting watershed runoff and stream flows. *J. Hydrologic Eng.*, 5: 216-222.
- Zhang, G.P., 2003. Time series forecasting using a hybrid arima and neural network model. *Neurocomputing*, 50: 159-175.
- Zhang, G.P., G.E. Patuwo and M.Y. Hu, 2001. A simulation study of artificial neural networks for nonlinear time-series forecasting. *Comput. Operat. Res.*, 28: 381-396.
- Zhao, C.Y., H.X. Zhang, M.C. Liu, Z.D. Hu and B.T. Fan, 2006. Application of support vector machine (SVM) for prediction toxic activity of different data sets. *Toxicology*, 217: 105-119.
- Zou, H.F., G.P. Xia, F.T. Yang and H.Y. Wang, 2007. An investigation and comparison of artificial neural network and time series models for chinese food grain price forecasting. *Neurocomputing*, 70: 2913-2923.