

## Forecasting ozone concentration levels using Box-Jenkins ARIMA modelling and artificial neural networks: a comparative study

<sup>1</sup>Norhashidah Awang, <sup>2</sup>Ng Kar Yong and <sup>3</sup>Soo Yin Hoeng

School of Mathematical Sciences, Universiti Sains Malaysia  
11800 Penang, Malaysia

e-mail: <sup>1</sup>shidah@usm.my, <sup>2</sup>karyong92@yahoo.com, <sup>3</sup>rachaelsyh@gmail.com

**Abstract** An accurate forecasting of tropospheric ozone ( $O_3$ ) concentration is beneficial for strategic planning of air quality. In this study, various forecasting techniques are used to forecast the daily maximum  $O_3$  concentration levels at a monitoring station in the Klang Valley, Malaysia. The Box-Jenkins autoregressive integrated moving-average (ARIMA) approach and three types of neural network models, namely, back-propagation neural network, Elman recurrent neural network and radial basis function neural network are considered. The daily maximum data, spanning from 1 January 2011 to 7 August 2011, was obtained from the Department of Environment, Malaysia. The performance of the four methods in forecasting future values of ozone concentrations is evaluated based on three criteria, which are root mean square error (RMSE), mean absolute error (MAE) and mean absolute percentage error (MAPE). The findings show that the Box-Jenkins approach outperformed the artificial neural network methods.

**Keywords** Autoregressive integrated moving-average; neural network; ozone; forecasting.

**AMS mathematics subject classification** 62M10

### 1 Introduction

Ozone ( $O_3$ ) is a colourless and odourless gas having high reactivity.  $O_3$  exists in both upper atmosphere (stratosphere) and lower atmosphere (troposphere). The prediction and forecasting of  $O_3$  is challenging as it is not released directly from a source, but resulting from complicated chemical reaction in air.  $O_3$  at different atmosphere shows contrasting effects, in which stratospheric  $O_3$  is helpful, while tropospheric  $O_3$  is harmful [1]. Stratospheric  $O_3$  is formed naturally through the interaction of sunlight and oxygen ( $O_2$ ). It is important to human and other living organisms on the earth as it shields the dangerous ultraviolet radiation from sunlight [1]. Meanwhile, tropospheric  $O_3$ , which is our focus in this study, is caused by the chemical reaction between the nitrogen oxides ( $NO_x$ ) and the volatile organic compounds (VOCs) in the existence of sunlight. The VOCs are mainly originated from motor vehicles and industry. Hence, urban areas normally record higher  $O_3$  levels.  $O_3$  is detrimental to not only human's respiratory systems, but also plants. Other factors affecting the  $O_3$  concentration in an area include wind speed, humidity and temperature. High concentration of  $O_3$  is always associated with dry, warm and breezy weather [1]. Rural and sub-urban areas have also been impacted due to the wind effect transporting  $O_3$  pollution from the sources of  $O_3$  precursors [2].

Before the 70s, environmental problems in Malaysia gained little heed. The frequent haze incidences in Southeast Asia since 1983 have challenged the environmental surveil-

lance system in Malaysia as well as raising consciousness of the environment. From July to October 1997, smog from bonfire seriously influenced the air quality in Malaysia. During normal periods, automotive emissions contributed to a large percentage of the total emissions in the city. As a result, the government inaugurated the Air Pollution Index (API) in 1989 to act as an indicator of air quality [3]. The Department of Environment (DOE) is responsible for the monitoring of air quality in Malaysia via a system of 52 continuous monitoring stations, which are placed strategically in different areas (urban, sub-urban and industrial) all around Malaysia [2]. Based on the Malaysian Ambient Air Quality Guidelines, the stipulated level for  $O_3$  is 0.10 and 0.06 parts per million (ppm) for the averaging time of an hour and eight hours respectively.

To have an enhanced system of air quality control and public forewarning for  $O_3$  levels, there is a need for developing an accurate modelling approach to predict the future concentration of  $O_3$  [4]. To better understand the characteristics of certain pollutants over time, analysis of time series, which is a set of ordered data with constant spaced time intervals, is among the best methodologies. One of the popular time series methods is the Box-Jenkins autoregressive integrated moving average (ARIMA). This method has been widely used in various studies to explain the relationship of specific variable with its historical data. An alternative method that is commonly used for modelling time series is the artificial neural network (ANN), a sophisticated mathematical methodology which is inspired by the behaviour of human brain and nervous system's neurons. The most familiar ANN methods used are the back-propagation neural network (BPNN), Elman recurrent neural network (ERNN) and radial basis function neural network (RBFNN). BPNN is a multi-layered perceptrons (MLPs) in which the signals flow forward in uni-direction with the algorithm of back propagation of errors. In contrast to BPNN, ERNN is a type of recurrent neural network in which the signals can flow forward and backward with an augmented context layer to store the lagged hidden values. This property enables ERNN to learn the time series data more vigorously. RBFNN is another popular feed-forward neural network which uses radial basis activation function and it is developed to solve the primary pitfall of BPNN of easily dropping into the local minima during the training [5].

Many studies have been conducted to compare the capability of the Box-Jenkins ARIMA approach, the neural networks and other techniques in forecasting future values of time series. Yi and Prybutok [4] compared a feed-forward neural network model with the Box-Jenkins ARIMA models and the regression method for forecasting the daily maximum  $O_3$  level in the Dallas-Fort Worth area in the United States of America. The study included secondary variables such as temperature, wind speed, wind direction, and other air pollutant measurements. Based on the root mean square error (RMSE) as the performance indicator, the neural network showed its superiority over the other two models. In another research, Ettouney *et al.* [6] developed an innovative modelling technique which utilized two feed-forward neural networks (FFNNs) to predict the  $O_3$  level in Kuwait. Several precursors of  $O_3$  and other meteorological parameters were taken into consideration. Principal Component Analysis (PCA) was applied to minimize the dimensionality of the multiple variables. The two similar neural networks functioned as two regressors in series. The original data was fed into the first FFNN for training and the residual dataset, calculated from the predicted values from the first FFNN, was used to train the second FFNN. The outputs and the original series were then compared for the two FFNNs and the single FFNN to obtain the regression coefficient (R) and mean square error (MSE), and the results showed

that double FFNNs was better. However, this study suggested that shorter-period modelling was better due to the noise present in the data. In Malaysia, Ul-Saufie *et al.* [7] compared multiple linear regression (MLR) model with ANN in estimating PM<sub>10</sub> concentration in Seberang Jaya, Pulau Pinang, based on gaseous and meteorological parameters. The hourly observations for PM<sub>10</sub>, from January 2004 to December 2007, were chosen and averaged to obtain 1430 daily observations to predict PM<sub>10</sub> level. The performance of the models was evaluated based on prediction accuracy (PA), coefficient of determination (R<sup>2</sup>), index of agreement (IA), normalised absolute error (NAE) and RMSE. The findings proved that ANN performed better than MLR.

In Peninsular Malaysia, the Klang Valley area is one of the areas that recorded high levels of O<sub>3</sub> concentrations. This area is densely populated due to rapid industrialisation and rising economy. Motivated by previous research, this study aims to model the O<sub>3</sub> concentration in an area of Klang Valley by using the Box-Jenkins ARIMA approach and three types of artificial neural network methods which are BPNN, ERNN and RBFNN. The performance of all the methods in forecasting O<sub>3</sub> concentrations is compared based on some evaluation indices.

## 2 Methods

In this study, the O<sub>3</sub> time series is fitted with the Box-Jenkins ARIMA modeling approach and three types of ANN methods and then the performance of these methods in forecasting the future values of the series is compared. The above-mentioned approaches are introduced as follows.

### 2.1 ARIMA modelling

The Box-Jenkins ARIMA modelling approach is one of the most popular techniques to analyze time series. If we denote  $\{X_t\}$ ,  $t = 1, 2, \dots, n$  as a stationary ARMA process with zero mean, then the ARMA( $p, q$ ) model is defined as

$$(1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p) X_t = (1 + \theta_1 B + \theta_2 B^2 + \dots + \theta_q B^q) Z_t$$

where  $\phi_i$ ,  $i = 1, \dots, p$  and  $\theta_j$ ,  $j = 1, \dots, q$  are  $p$  AR parameters and  $q$  MA parameters, respectively. Here,  $\{Z_t\}$  is white noise with mean 0 and variance  $\sigma^2$ , and  $B$  is the backward shift operator with  $B^k X_t = X_{t-k}$  where  $k$  is the lag number. Special cases of ARMA model are AR( $p$ ) model and MA( $q$ ) model, which are expressed by

$$(1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p) X_t = Z_t$$

and

$$X_t = (1 + \theta_1 B + \theta_2 B^2 + \dots + \theta_q B^q) Z_t,$$

respectively. If the series is not stationary and the autocovariance function is slowly decreasing, then the series can be transformed to generate a new series that exhibits stationarity and have a fast decreasing autocovariance function. This can be achieved by differencing the series. By doing this, the series  $\{X_t\}$  is assumed to have an ARIMA ( $p, d, q$ ) process if  $Y_t = (1 - B)^d X_t$ , where  $d$  is a nonnegative integer, is a causal ARMA ( $p, q$ ) process. ARIMA models are useful for representing data with trends.

The Box-Jenkins approach involves four steps: model identification, estimation of the parameters, diagnostic checking and forecasting. In our analysis, the procedure is implemented using the Interactive Time Series Modelling (ITSM) software [8]. In model identification, the stationarity and the existence of seasonality or trend of the series are determined. This is done by plotting the sample autocorrelation function (ACF) and the partial autocorrelation function (PACF) up to lag 40. If the functions are rapidly decreasing, then we assume the series is stationary; otherwise differencing will be performed until the series have rapidly decreasing ACF and PACF. Besides observing the ACF and PACF plots, the stationarity of the series is tested by the Phillips-Perron test which is performed using Eviews software. Then, the orders,  $p$  and  $q$ , of the ARMA model are identified by using ACF and PACF of the stationary series.

In the next step, the parameters of the possible ARMA model are estimated. This is done by some preliminary estimation procedures, followed by the maximum likelihood method. There might be some possible ARMA models suggested by the ACF and PACF plots. Therefore, a common selection criterion, Akaike Information Corrected Criterion (AICC) is used for selecting the best model, where the best one is that with the lowest AICC.

In the third step, diagnostic checking is performed on the residuals,  $\{Z_t\}$ . The residuals are supposed to be white noise which is independent and identically distributed. If the assumptions are not satisfied, another model should be considered for the series. There are several tests available in ITSM to test for randomness of the residual: the Ljung-Box test, the McLeod-Li test, turning points test, different sign test and rank test. The normality assumption is tested by Jarque-Bera test. In the final step, the best model will be used to forecast future values of the series if the randomness of the residuals is satisfactory.

## 2.2 Artificial neural networks (ANN)

ANN is a model equipped with the capability to extract nonlinear information from a time series. The gradient descent algorithm used to train the MLPs is well-known as back-error propagation algorithm [9]. In ANN, the input neurons correspond to the synapses in the biological neurons, and the weights correspond to the strength of the signals. First, the input data is sent into the input layer. The weighted sums of the inputs are then fed forward to the hidden layer with an activation function. Then, the weighted hidden sums (activation value) are processed toward output layer with another mathematical function to compute the final output where different weights produce different final output. It has been shown that through learning or training (weight adjustment procedure), ANN is able to approximate any (nonlinear) functions if suitable weight parameters are selected. The weight adjusting procedure is known as learning or training. Diverse applications of ANNs cover forecasting, pattern recognition and classification [9]. One of the advantages of ANN is the fast computation resulted from the parallel architecture of ANN model in which all neurons function concurrently [10]. In addition, ANN could infer the undercover nonlinear relationship in time series. Besides, ANN model does not impose any underlying assumptions on the distribution of data collected [5].

In our analysis, the modelling of ANN is implemented using the software Matlab Neural Network Toolbox. The function `newff()` is used to realize the training of BPNN. There are various training algorithms that can be used to train the network. Three different activation functions can be selected from the Matlab function, namely logistic sigmoid (`logsig`),

hyperbolic tangent sigmoid (`tansig`) and linear (`purelin`) transfer functions. The function `newelm()` in Matlab is used to train the ERNN while the function of `newrb()` is used for the training of RBF network. This function realizes the RBFNN with supervised OLS algorithm [5]. The default activation function in Matlab, which is the Gaussian function, is used in the hidden layer, while the linear function (default function) is used in the output layer.

In neural network modelling, the series is partitioned into three non-overlapping subsets, which are training, validation and testing sets. The training set is used for modifying the weights of the network, and the validation set is used to determine the stopping of the training process. Finally, the testing set is used to verify the network's performance. The need of validation set is to avoid overfitting, which is a situation where the training starts to model the noise in the training data. In the training stage, the training and the validation errors decrease along with the iterations. Just as overfitting occurs, the validation error increases whereas the training error remains its downtrend [9]. The number of successive iterations that the MSE in the validation set fails to decrease is called the number of validation failures. To overcome overfitting and to improve the generalization capability of BPNN and ERNN, several stopping criteria are used in the Matlab Neural Network Toolbox. Normally, the training stops when the gradient reaches the minimum gradient, whose default depends on the type of learning algorithm, and (or) when the number of validation failures reaches the maximum number of failures (6 times in default). When the number of validation failures reaches the maximum number of iterations, the training stops and the weights of the network at which the MSE of the validation set is minimal are returned and the structure is fixed. The fixed structure is then used to predict the  $O_3$  concentration in the testing set. The procedure is repeated for all the predetermined values of  $k$  (1 to 5), where  $k$  is lag number and as defined in Section 2.1. The best model, based on the smallest value of testing error, is used to forecast the  $O_3$  concentration levels for the next seven days iteratively.

On the other hand, RBFNN training iteration using Matlab is different from that of BPNN and ERNN. Initially, the RBF network is simulated linearly (without hidden layer). Then, the hidden radial basis neurons are added iteratively with corresponding weights, and the output-layer weights are adjusted accordingly whenever a hidden neuron is added in order to minimize the error. These procedures continue until the error (MSE) falls below the desired minimum error. The minimum error acts as the stopping criterion as in BPNN and ERNN to avoid the overfitting of the data. Finally, the best model chosen is the model that provides the lowest testing error.

### 3 Results and discussion

In current study, the data of the daily maximum  $O_3$  concentration levels which was collected at SMK Permaisuri, Cheras, from 1 January 2011 to 7 August 2011 are used. The data are obtained from the Department of Environment (DOE), Malaysia. This monitoring station is located in the Klang Valley area and it is classified by DOE in urban category. The  $O_3$  level recorded from this station is among the highest in Malaysia in 2011. The range for the  $O_3$  level for this time period of study is from 0.002 to 0.150 ppm with mean 0.071 ppm and variance 0.000831. The time series plot of the data is shown in Figure 1. The ACF and PACF for the series drawn up to lag 40 are displayed in Figure 2. From the figures, it is observed that the series does not clearly show any seasonal pattern.

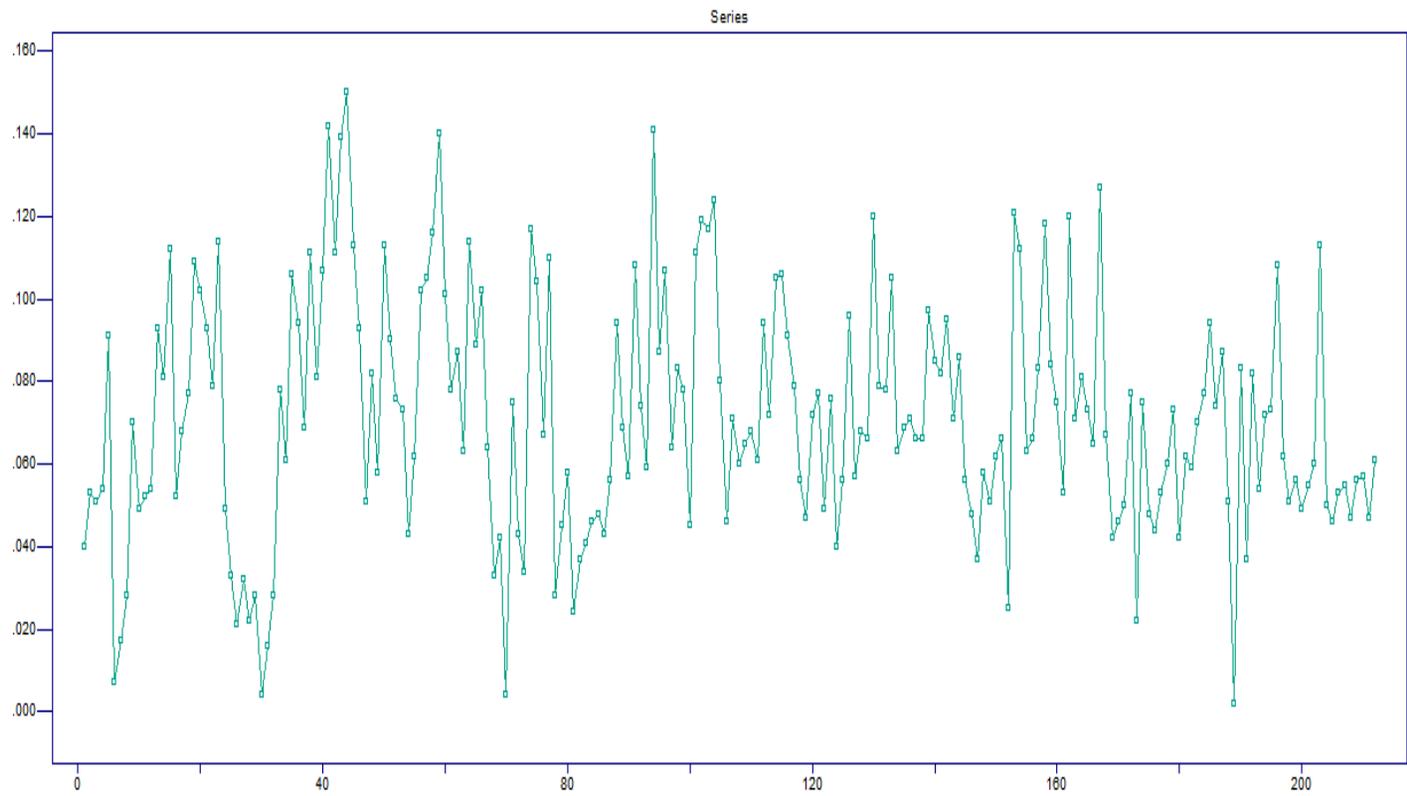


Figure 1: Time series plot of O<sub>3</sub> concentration levels

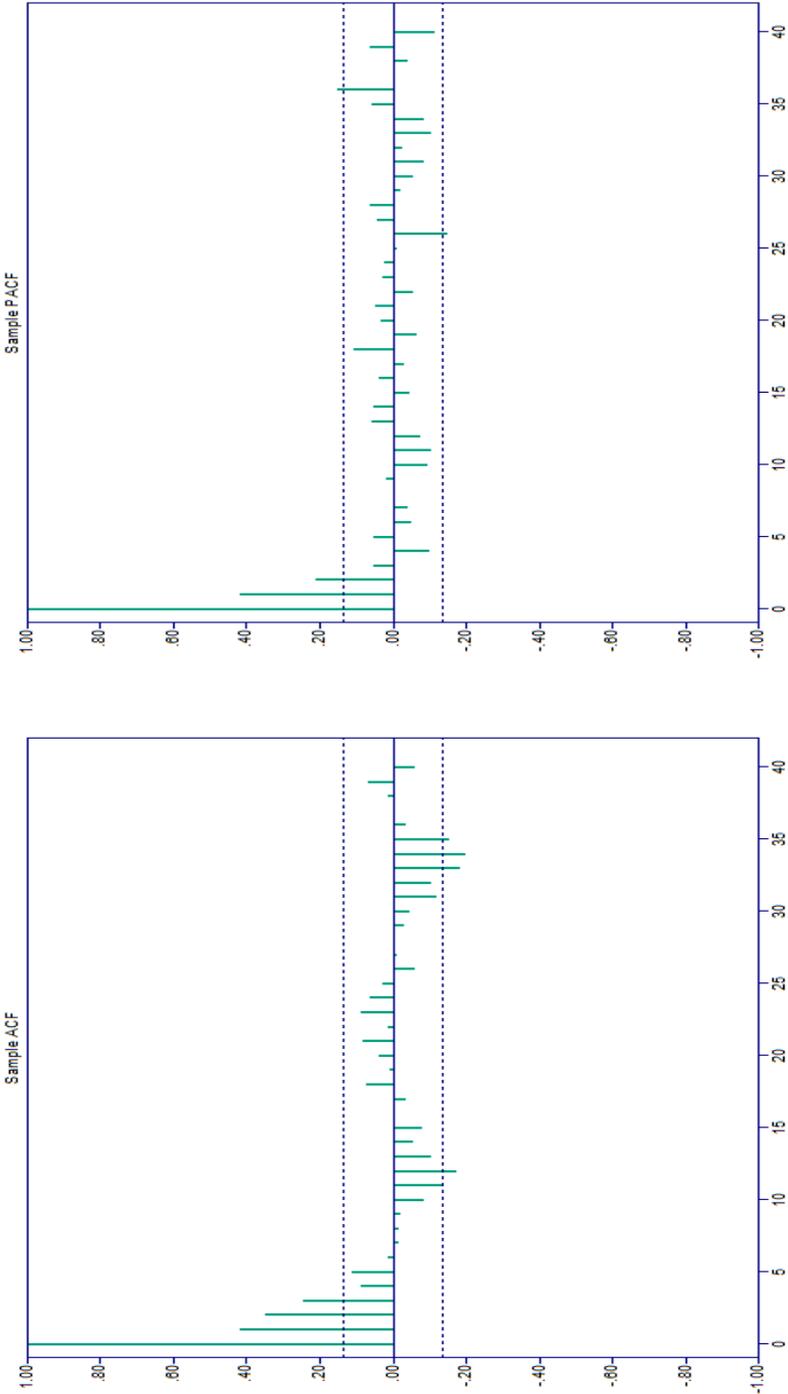


Figure 2: Sample ACF and PACF of the O<sub>3</sub> series

Since it is known that ARIMA model is not effective in doing long-term forecast, short-term forecast period was set. Hence, data from 1 January to 31 July 2011 are used for ARIMA modelling and the remaining data from 1 August to 7 August 2011 are kept for forecasting evaluation. Thus, we have 212 observations for modelling and 7 observations for forecasting evaluation. The procedure is implemented using ITSM software. The Phillips-Perron test and the Jarque-Bera test are performed to check for stationarity and normality assumptions, respectively. It can be concluded from the results that the series is stationary and normally distributed. After satisfying the stationarity of the series, the ARMA model is fitted to the data with various combinations of  $p$  and  $q$ , where we fix  $p, q = 0, 1, 2, 3, 4$  and  $5$ . The best model is chosen based on the minimum value of AICC. The results show that the best model is AR(2) with AICC =  $-947.156$ . Residual analyses conclude that the residual is white noise and normally distributed (all p-values  $>0.05$ ). Furthermore, all the coefficients are statistically significant at 5% level of significance. Therefore, this model is used to forecast  $O_3$  levels from 1 August to 7 August 2011. Having AR(2) as the best model, we can infer that the current value of the concentration is associated with the concentration levels of the previous two days.

For the neural network modelling, the first 212 observations are again used for modelling. To determine the size of the input matrix or the number of input neurons, ACF and PACF for the original series (Figure 2) and the order of ARIMA model are referred to. Since the best fitted model that we obtained was AR(2), this suggests that the third value may be associated with the previous two consecutive values. Therefore, it is possible to generate an input matrix of  $2 \times 210$  (similarly, the number of input neurons is two), and its corresponding output (target) matrix of  $1 \times 210$ . However, in practice the number of input neurons is tested from one to five neurons for all the three neural networks. The output neuron is one, standing for the next day forecast of  $O_3$  concentration. A single hidden layer is selected as many experimental results have shown that one hidden layer is adequate to map any nonlinear functions [11]. Suppose  $X_t$  represents the  $O_3$  concentration at time  $t$ , the input and its corresponding target matrices of the data are written as follows:

$$\text{Input matrix} = \begin{pmatrix} X_1 & X_2 & \dots & X_{212-k} \\ X_2 & X_3 & & X_{213-k} \\ \vdots & & \ddots & \vdots \\ X_k & X_{k+1} & \dots & X_{211} \end{pmatrix}$$

and the corresponding target matrix =  $[X_{k+1}, X_{k+2}, \dots, X_{212}]$  where  $k = 1, 2, 3, 4, 5$  is the number of lags, indicating the current value depends on  $k$  previous values. From these  $(212-k)$  columns, the data is divided into training, validation and testing sets in the ratio of 7:2:1. The training proceeds after setting the parameters into the specific Matlab functions of neural network.

For BPNN and ERNN, learning rates, training algorithms and the number of hidden nodes influence the generalization capability of the neural networks [11]. Few numbers of hidden neurons may cause underfitting while excessive hidden neurons may cause over fitting [12]. To date, there is no theory developed for selecting the number of hidden-layer neurons. Trial and error is used to fix the neural network structure. Therefore, in our analysis for BPNN, the learning rate constants ranged from 0.01 to 0.05 are examined, with the increment of 0.01. The number of hidden neurons are selected empirically from 2 to 50,

where 2 to 20 with the increment of 2, 20 to 30 with the increment of 5, and lastly 30 to 50 with the increment of 10. The Levenberg-Marquardt training algorithm is selected. This combining algorithm of steepest descent method and the Gauss-Newton's method hence possesses the stability of the steepest descent algorithm and the speed advantage of the Gauss-Newton's algorithm [5]. The activation function used in both hidden and output layers is the tansig transfer function as it gives lowest value of RMSE. The best result is obtained when the number of hidden neuron is 18, learning rate is 0.02 and  $k = 3$ .

The parameters of ERNN are similar to that of BPNN. The learning rate constants are tested from 0.01 to 0.05 with an increment of 0.01, and the numbers of hidden neurons are tried from 2 to 50, where 2 to 30 with an increment of 2 and 30 to 50 with an increment of 5. The Levenberg-Marquardt algorithm is applied in this study. The hidden layer employs the tansig function while the output layer uses the purelin function as these two functions give lowest value of RMSE after some experiments with various combinations of parameters. The best result is obtained when the learning rate constant is set to 0.03, the number of hidden neuron is 22 and  $k = 2$ .

For RBFNN, different performance goal (MSE) of the training can be set accordingly to obtain the lowest testing error. Different spread constants are examined, namely from 0.1 to 2.0 with an increment of 0.1. Different numbers of input neurons and spread constants will affect the number of hidden neurons required to attain a specific error. Having the same desired error, the number of hidden neurons needed increases as the spread constant increases. The best result is obtained when the spread constant is 0.3, the number of hidden neuron is 190 and  $k = 2$ . The minimum error is set by trial and error from 0 to  $6.5 \times 10^{-4}$ .

The performance of the four methods in forecasting future values of  $O_3$  concentrations is compared and evaluated based on three metrics, which are root mean square error (RMSE), mean absolute error (MAE) and mean absolute percentage error (MAPE). These three metrics are defined by:

$$\begin{aligned} \text{RMSE} &= \sqrt{\frac{1}{T} \sum_{t=1}^T (O_t - P_t)^2} \\ \text{MAE} &= \frac{1}{T} \sum_{t=1}^T |O_t - P_t| \\ \text{MAPE} &= \frac{1}{T} \sum_{t=1}^T \frac{|O_t - P_t|}{O_t} \end{aligned}$$

where  $O_t$  and  $P_t$  are the observed and forecast values at time  $t$ , respectively. Table 1 displays the observed and forecast values (up to three decimal places) of the four methods together with the values of RMSE, MAE and MAPE. The plot of observed and forecast values is shown in Figure 3. It can be seen from Table 1 that the ARIMA model outperforms the neural networks. Among the neural networks, BPNN gives the best performance. The worst method is ERNN as it has the largest values of RMSE, MAE and MAPE. From Figure 3, it seems that all the methods fail to capture the sharp drops of  $O_3$  levels on 4<sup>th</sup> and 7<sup>th</sup> August 2011. Furthermore, all the methods overestimate the  $O_3$  levels for all the 7 days of forecast.

Table 1: The observed and forecast values of O<sub>3</sub> concentration level from the four techniques

Date	Observed	AR (2)	BPNN	ERNN	RBFNN
1 August 2011	0.061	0.063	0.070	0.064	0.064
2 August 2011	0.060	0.066	0.077	0.076	0.075
3 August 2011	0.057	0.068	0.078	0.078	0.075
4 August 2011	0.041	0.069	0.077	0.083	0.081
5 August 2011	0.067	0.070	0.076	0.081	0.080
6 August 2011	0.058	0.070	0.075	0.083	0.083
7 August 2011	0.036	0.070	0.075	0.082	0.082
RMSE		<b>0.018</b>	0.024	0.028	0.027
MAE		<b>0.014</b>	0.021	0.024	0.023
MAPE		<b>0.312</b>	0.458	0.517	0.499

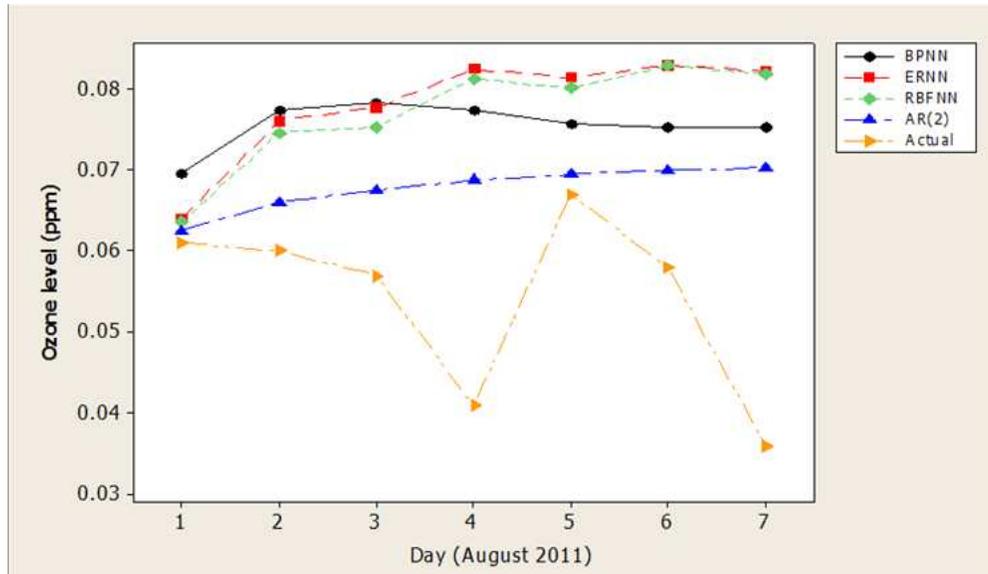


Figure 3: Plot of the observed and forecast values of O<sub>3</sub> concentrations level from the four techniques

## 4 Conclusion

An accurate forecasting of tropospheric ozone ( $O_3$ ) concentration is beneficial for the strategy planning and control of air quality. In previous studies, several researchers introduced different approaches to forecast  $O_3$  concentration and other air pollutants. Motivated from previous research, the Box-Jenkins autoregressive integrated moving-average (ARIMA) modelling approach and the three artificial neural network (ANN) models mentioned above are used to model and forecast the time series of  $O_3$  daily maximum concentration from a monitoring station at SMK Cheras, Selangor. This station is located in the Klang Valley and the  $O_3$  level recorded here is among the highest in Malaysia. The performance of the four methods in forecasting future values of  $O_3$  concentrations is evaluated based on three criteria, which are root mean square error (RMSE), mean absolute error (MAE), and mean absolute percentage error (MAPE). The findings show that the AR(2) model outperforms the artificial neural network models and the worst model is ERNN. The findings also indicate that all the models overestimate the levels for the entire period of forecasting and fail to capture sharp drops of  $O_3$  levels on 4<sup>th</sup> and 7<sup>th</sup> August.

Theoretically, ANN is capable of capturing the nonlinear correlation in the time series. Furthermore, any desired accuracy of the function approximation can be achieved. Conversely, ARIMA model can only extract the linear relationship in the time series. However, the ARIMA model can easily be interpreted, while the black-box feature of ANN makes the interpretation of the behaviour of particular variable to be difficult. Moreover, reproducible outputs are unable to obtain due to the random assignment of weights and biases to the BPNN and ERNN in Matlab's functions. From our findings, it is shown that the descending order of the forecast performance of the three models is BPNN, RBFNN, and ERNN.

There are several limitations in our study. Firstly, only a seven-month period of  $O_3$  concentration data is available in this study. The comparatively short period of data may affect the forecasting accuracy of the methods. Secondly, other forecasting methods can be used besides the four different methods used in this study. Furthermore, similar research could be conducted on the data from other stations to confirm the performance of these models. Related factors that will cause different effects on model performance can also be included.

Tropospheric  $O_3$  seriously endangers human health and environment. The effective management of the control and public warning strategies for  $O_3$  concentration can be efficiently implemented by the accurate forecast of  $O_3$  concentration. Hence, further research on more accurate forecasting techniques should be carried out and these techniques should be implemented in real application.

## Acknowledgement

The authors would like to thank the Department of Environment, Malaysia for providing the data. This research is funded by Universiti Sains Malaysia research grant (ref no: 304/PMATHS/6312094).

## References

- [1] Rajab, J. M., Matjafri, M. Z. and Lim, H. S. Combining multiple regression and principal component analysis for accurate predictions for column  $O_3$  in Peninsular Malaysia. *Atmospheric Environment*. 2013. 71: 36–43.
- [2] Department of Environment. Malaysia Environmental Quality Report. Ministry of Natural Resources and Environment Malaysia. 2012.
- [3] Afroz, R., Hassan, M. N. and Ibrahim, N. A. Review of air pollution and health impacts in Malaysia. *Environmental Research*. 2003. 92: 71–77.
- [4] Yi, J. and Prybutok, V. R. A neural network model forecasting for prediction of daily maximum  $O_3$  concentration in an industrialized urban area. *Environmental Pollution*. 1996. 92: 349–357.
- [5] Zhang, X., Liu, Y., Yang, M., Zhang, T., Young, A. A. and Li, X. Comparative study of four time series methods in forecasting typhoid fever incidence in China. *PLoS ONE*. 2013. 8(5): 1–11.
- [6] Ettouney, R. S., Mjalli, F. S., Zaki, J. G., El-Ridai, M. A. and Ettouney, H. M. Forecasting of  $O_3$  pollution using artificial neural networks. *Management of Environmental Quality: An International Journal*. 2009. 20: 668–683.
- [7] Ul-Saufie, A. Z., Yahya, A. S., Ramli, N. A. and Abdul Hamid, H. Comparison between multiple linear regression and feed forward back propagation neural network models for predicting  $PM_{10}$  concentration level based on gaseous and meteorological parameters. *International Journal of Applied Science and Technology*. 2011. 1: 42–49.
- [8] Brockwell, P. J and Davis, R. A. *Introduction to Time Series and Forecasting*, Second Edition. New York: Springer. 2002.
- [9] Dayhoff, J. E. and DeLeo, J. M. Artificial neural networks. In *Proceedings of the 2001 Conference on Prognostic Factors and Staging in Cancer Management: Contributions of Artificial Neural Networks and Other Statistical Methods*. Cancer. 2001. 91: 1615–1635.
- [10] Rao, M. A. and Srinivas, J. *Neural networks: Algorithms and Applications*. Oxford: Alpha Science International. 2003.
- [11] Kii, Ö. and Uncuolu, E. Comparison of three back-propagation training algorithms for two case studies. *Indian Journal of Engineering & Material Sciences*. 2005. 12: 434–442.
- [12] Sheela, K. G. and Deepa, S. N. Review on methods to fix number of hidden neurons in neural networks. *Mathematical Problems in Engineering*. 2013. 2013: 1–11.