Journal of Applied Sciences

ISSN 1812-5654

science alert

ANSInet
an open access publisher
http://ansinet.com

Journal of Applied Sciences 8 (13): 2428-2434, 2008
ISSN 1812-5654

A Backpropagation Method for Forecasting Electricity Load Demand

Zuhaimy Ismail and Faridatul Azna Jamaluddin
Department of Mathematics, Faculty of Science, Universiti Teknologi Malaysia,
Skudai, 81310 Johor, Malaysia

Abstract: This study presents the implementation of back propagation neural network method to improve forecasting of electricity load demand where the demand is highly dependent on various independent variables such as the weather, temperature, holidays, days of the week or even strikes. The implementation of this method requires mathematical software, data preparation and the calculation of degree of freedom, which is necessary for the neural networks architecture. We also consider the use of various combinations of activation functions in input layer to hidden layer and hidden layer to output layer and using analysis of variance and multiple comparison using Duncan's tests to analyze the neural network's performance. Two modifications to the backpropagation methods were developed to improve error with selected activation functions and a new improved error using mean square error. The data used are the daily electricity load demand for Malaysian from 2006 to 2007. The forecast accuracy based on the error statistics of forecast between the models for a month ahead is presented and behaviour of data is also observed.

Key words: Forecasting, backpropagation, artificial neural network, electricity load demand, forecast accuracy

INTRODUCTION

The areas Artificial Neural Networks (ANN) have seen an explosion of interest over the last decades and are being applied across an extraordinary range of problem domains. A number of ANN model have been successfully implemented in solving complex problems (Khamis *et al.*, 2006; Lee, 2008). ANN is a causality model that is widely used to solve complex problems. The problem becomes more complex in a situation where information is incomplete. Back propagation networks in ANN may provide us with some solutions. Backpropagation is the most wide used algorithm for training multilayer feedforward neural network (Rumelhart and McClellana, 1986). Backpropagation consists of three layers namely the input layer, at least one intermediate hidden layer and the output layer. In backpropagation network the connection weights is in a one-way direction. Typically, units are connected in a feed-forward manner with the input units fully connected to the hidden layer while the hidden units are fully connected to the output layer.

A backpropagation network consists of many elements or neurons that are connected by communications channels or connectors. These connectors carry numeric data arranged by a variety of means and organized into layers. The network can perform a particular function when certain values are assigned to the connections or weights between elements. To describe a system, there is no assumed structure of the model, instead the network are adjusted or trained so that a particular input leads to a specific target output (Haykin, 1999; Bansel *et al.*, 1993). The mathematical model of a backpropagation network comprises a set of simple functions linked together by weights. The network consists of a set of input x, output units y and hidden units z, which link the inputs to outputs (Fig. 1). The hidden units extract useful information from inputs and use them to predict the output. The type of neural network shown here is known as the multilayer perceptron (Haykin, 1999).

A network with an input vector of elements x_l (l = 1, 2,..,N_i) is transmitted through a connection that is multiplied by weight, w_{ji}, to give the hidden unit z_i (j = 1, 2, 3, …, N_k):

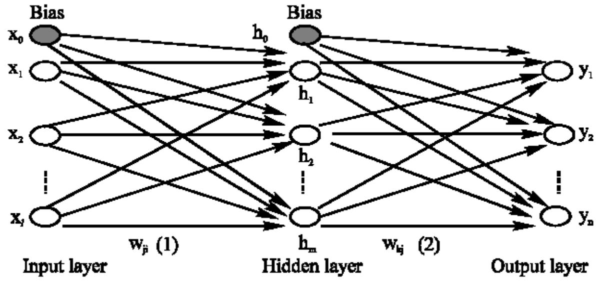

Fig. 1: Feed-forward neural network

Corresponding Author: Zuhaimy Ismail, Department of Mathematics, Faculty of Science, Universiti Teknologi Malaysia, Skudai, 81310 Johor, Malaysia Tel: +60197133940

$$z_j = \sum_{l=1}^{N_i} w_{jl} x_l + w_{j0} \qquad (1)$$

where, N_k is the number of hidden units and N $_i$is the number of input units. The hidden units consist of the weighted input and a bias (w_{j0}). A bias is simply a weight with constant input of 1 that serves as a constant added to the weight. These inputs are passed through a layer of activation function f which produces:

$$h_j = f\left[\sum_{l=1}^{N_i} w_{jl} x_l + w_{j0}\right] \qquad (2)$$

The activation functions are designed to accommodate the nonlinearity in the input-output relationships. A common activation function is the sigmoid or hyperbolic tangent:

$$f(z) = \tan h\ (z) = 1 - \frac{2}{[1 + \exp(2z)]} \qquad (3)$$

The outputs from hidden units pass another layer of filters:

$$v_k = \sum_{j=1}^{N_h} w_{kj} h_j + w_{k0} = \sum_{j=1}^{N_h} w_{kj} f\left[\sum_{l=1}^{N_i} w_{jl} x_l + w_{j0}\right] + w_{k0} \qquad (4)$$

and feed into another activation function F to produce output y (k = 1, 2, 3, …, N_o)

$$y_k = F(v_k) = F\left[\sum_{j=1}^{N_h} w_{kj} f\left(\sum_{l=1}^{N_i} w_{jl} x_l + w_{j0}\right) + w_{k0}\right] \qquad (5)$$

The weights are adjustable parameters of the network and are determined from a set of data through the process of training (Lai, 1998; Zhang *et al*., 2001). The training of a network is accomplished using an optimization procedure (such as nonlinear least squares). The objective is to minimize the Sum of Squares of the Error (SSE) between the measured and predicted output. There are no assumptions about functional form, or about the distributions of the variables and errors of the model, ANN model is more flexible than the standard statistical technique (Fareway and Chatfield, 1998). It allows for nonlinear relationship and complex classificatory equations. The users do not need to specify as much details about the functional form before estimating the classification equation. Instead, it lets the data determine the appropriate functional form Limsombunchai *et al.* (2004) and Herrera *et al.* (2007).

In accordance to standard analytical practice, the sample size was divided on a random basis into two sets, namely the training set, which contain 80 and 20% of the total sample, respectively. To evaluate the modeling accuracy the correlation coefficient, r and Mean Squares Error (MSE) were calculated. The model with a higher r and lower MSE was considered to be a relatively superior model.

MATERIALS AND METHODS

In Malaysia, the short term forecast of electricity demand is being generated by senior engineer at Operation and System Planning Division from the Tenaga National Berhad, the national power electricity company in order to arrive at hourly, daily and weekly forecast, various factors need to be taken into account (Ismail and Mahpol, 2007). Some of the factors are the daily temperature, weather, days of the week, legal and religious holidays. Seasonal effects and human behaviour whether they will take a day off preceding and following the holidays as to take advantage of a long break (Nima, 2001).

This study proposed the modification of the backpropagation algorithm to predict the Malaysian daily electricity load demand. Backpropagation is the most widely used techniques in the ANN literature and the easiest to understand. Its learning and update procedure is intuitively appealing because it is based on a relatively simple concept, if the network gives the wrong answer, the weights are corrected so that the error is lessened and as a result, future responses of the network are more likely to be corrected (Klein and Rossin, 1999a; Mandal *et al*., 2007). When the network is given an input, the updating of activation value propagates forward from the input layer of processing units through each internal layer, to the output layer of processing units (Haykin, 1999). The output units then provide the network's response. When the network corrects its internal parameters, the correction mechanism starts with the output units and back propagates backward through each internal layer to the input layer. Backpropagation network is a gradient descent method and its objective is to minimize the mean squared error between the target values and the network outputs (Manda *et al*., 2007). Thus the Mean Square Error (MSE) function is defined as:

$$E = \frac{1}{2}\sum_k (t_{kj} - o_{kj})^2 \qquad (6)$$

Where:

t_{kj} = Target value from output node (k) to hidden node (j)

o_{kj} = Network value from output node (k) to hidden node (j)

The proposed error function for standard backpropagation (mm) is defined implicitly as:

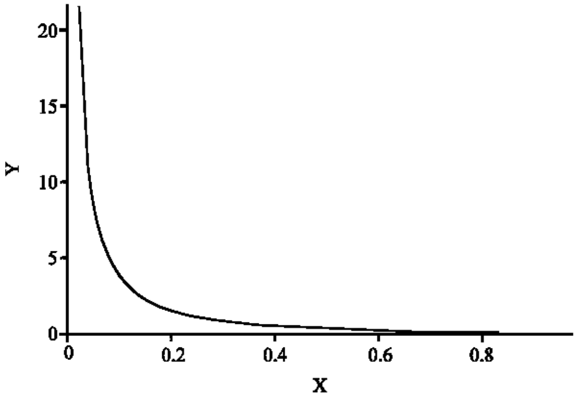

Fig. 2: Proposed error function of backpropagation model with activation function of $1/(1+e^{-2x})$

$$mm = \sum_K \rho_k$$

$$\rho_k = \frac{E_k^2}{2a_k(1-a_k^2)}$$

where, $E_k = t_k - a_k$
E_k = Error at output unit k
t_k = Target value of output unit k
a_k = An activation of unit k

A proposed backpropagation error function is shown in Fig. 2, with its errors reducing rapidly compared to MSE (Fig. 3), thus the proposed error will give less iterations for convergence.

The updating weight of standard backpropagation model is proportional to the negative gradient of weight changes as shown:

$$\Delta W_k \propto -\frac{\partial E}{\partial W_k} \tag{7}$$

Thus, in this case, the above relation in Eq. 1 can be rewritten as:

$$\frac{\partial \rho_k}{\partial w_k} = \frac{\partial \rho}{\partial Net_k} . \frac{\partial Net_k}{\partial w_k} \tag{8}$$

It is known that, $Net_k = \sum_j w_{kj} o_j + \theta_k$ where θ_k is a bias term, thus by taking partial derivative and substituting it into Eq. 8 gives:

$$\frac{\partial \rho}{\partial w_k} = \frac{\partial \rho}{\partial Net_k} . \ o_j \tag{9}$$

Assume tha t $\frac{\partial \rho}{\partial Net_k} = \delta_k$. By using chain rule, this gives:

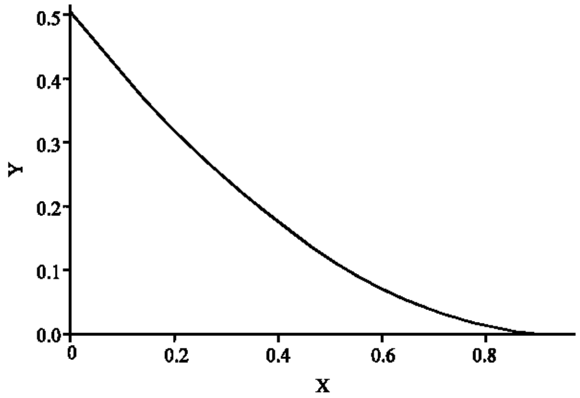

Fig. 3: Mean square error of backpropagation model with activation function of $1/(1+e^{-x})$

$$\frac{\partial \rho_k}{\partial Net_k} = \frac{\partial \rho_k}{\partial a_k} . \frac{\partial a_k}{\partial Net_k} \tag{10}$$

By taking partial derivatives of our first activation function, sigmoid function $f(Net_k) = 1/(1+e^{-2Net_k})$ and simplifying it by substituting in terms of a_k, gives:

$$a_k = f(net_k) = \frac{1}{1+e^{-2net_k}}$$

$$f'(net_k) = \frac{2e^{-2net_k}}{(1+e^{-2net_k})^2}$$

Given $a_k = \frac{1}{1+e^{-2net_k}}$, we have $e^{-2net_k} = \frac{1-a_k}{a_k}$
Derivative of f (net_k) in terms of a_k is given by:

$$f'(Net_k) = \frac{\partial a_k}{\partial Net_k} = 2a_k(1-a_k) \tag{11}$$

It is known that $\rho_k = \frac{E^2{}_k}{2a_k(1-a^2{}_k)}$. Thus taking the partial derivatives with respect to a_k gives:

$$\frac{\partial \rho_k}{\partial a_k} = \frac{-\left[4a_k(t_k-a_k)(1-a^2{}_k)+2(t_k-a_k)^2(1-3a_k^2)\right]}{4a^2{}_k(1-a_k^2)} \tag{12}$$

In simplified form, Eq. 12 becomes

$$\frac{\partial \rho_k}{\partial a_k} = -\left(\frac{E+\rho\left(1-3a_k^2\right)}{a_k\left(1-a_k^2\right)}\right) \tag{13}$$

By substituting Eq. 11 and 13 into Eq. 10 we have the improved error signal of backpropagation for the output layer as:

$$\frac{\partial \rho_k}{\partial Net_k} = \frac{2\left(E+\rho\left(1-3a_k^2\right)\right)}{1+a_k} = \delta_k \tag{14}$$

and an error signal for modified backpropagation of the hidden layer which is the same as standard backpropagation:

$$\delta_j = \sum \delta_k w_j f'(a_j) \qquad (15)$$

ACTIVATION FUNCTION

Activation function: Inverse tangent f(x) = $\tan^{-1}(x)$
If the arctangent activation function (Fig. 4) is used instead of sigmoid function then we have another improved error function:

From Eq. 11, we assume $\frac{\partial \rho}{\partial Net_k} = \delta_k$. By using chain rule, this gives $\frac{\partial \rho_k}{\partial Net_k} = \frac{\partial \rho_k}{\partial a_k} . \frac{\partial a_k}{\partial Net_k}$. We know that, a_k = f(Net_k). For the proposed method, instead of using sigmoid function we use the arctangent function, $\tan^{-1}$ (x). By taking partial derivatives of a_k and simplify it by substituting in terms of a_k, gives:

$$f(x) = \tan^{-1}(x) \qquad -\infty < x < \infty \qquad (16)$$

$$f'(x) = \frac{1}{1+x^2} \qquad (17)$$

$$a_k = f(Net_k) = \tan^{-1}(Net_k),\ Net_k = \tan a_k \qquad (18)$$

From the Eq. 16-18, we have

$$\frac{\delta a_k}{\delta net_k} = f'(net) = \frac{1}{1+\tan^2 a_k} \quad ,-\pi/2 < a_k < \pi/2 \qquad (19)$$
$$= \frac{1}{sek^2 a_k} = kos^2 a_k$$

Improved error signal of backpropagation using inverse tangent is obtained as:

$$\frac{\partial \rho_k}{\partial Net_k} = \frac{\partial \rho_k}{\partial a_k} . \frac{\partial a_k}{\partial Net_k}$$
$$= -\left(\frac{E + \rho\left(1 - 3a_k^2\right)}{a_k\left(1 - a_k^2\right)} \right) kos^2 a_k = \delta_k \qquad (20)$$

The derivatives of sigmoid and inverse tangent are shown in Fig. 5. The inverse tangent has higher derivatives value at the saturation region, when output is approaching ±1 or 0 for sigmoid function.

Activation function: Hyperbolic tangent f(x) = Tan h(x)
Another function to be used is hyperbolic tangent (Fig. 6, 7) function,

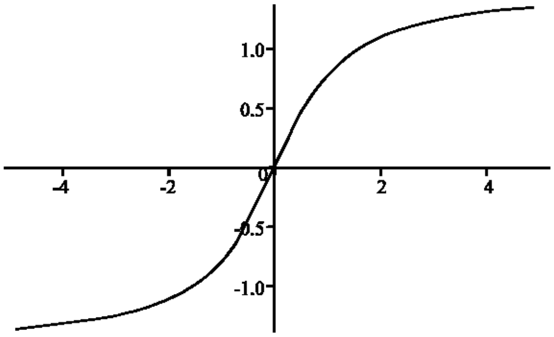

Fig. 4: Activation function of inverse tangent f(x) = $\tan^{-1}(x)$

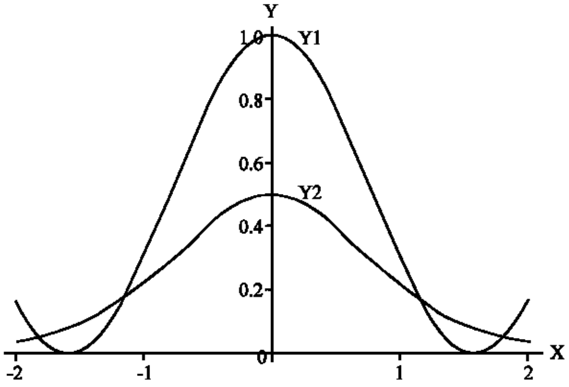

Fig. 5: Derivative of arctangent $y_1 = kos^2 a_k$ and sigmoid, $y_2 = 2a_k(1-a_k)$

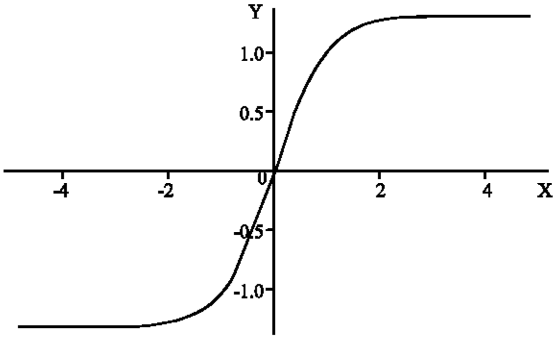

Fig. 6: Activation function of tan h f(x) = tan h (x)

$$f(x) = \tanh(x) \qquad -\infty < x < \infty$$

$$a_k = f(net_k) = \tanh(net_k)$$

$$\frac{\delta a_k}{\delta net_k} = f'(net_k) = (1 - a_k^2)$$

Kalman Error (Kalman and Kwasny, 1997) was defined as:

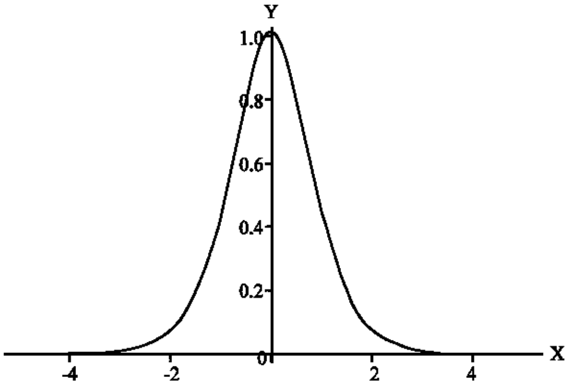

Fig. 7: First derivative of tanh f(x) = tan h (x)

$$g_k = \frac{e_k^{\ 2}}{(1-a_k^{\ 2})} = \frac{(t_k - a_k)^2}{(1-a_k^{\ 2})}$$

$$\frac{\partial g_k}{\partial w_k} = \frac{\partial g_k}{\partial net_k} . \frac{\partial net_k}{\partial w_k} = \frac{\partial g_k}{\partial net_k} .o_j$$

$$\frac{\partial g_k}{\partial Net_k} = -\frac{\partial g_k}{\partial a_k} . \frac{\partial a_k}{\partial Net_k}$$

$$\frac{\delta g_k}{\delta a_k} = -\left(\frac{2(t_k - a_k)(-1)\left(1-a_k^2\right) - (t_k - a_k)(-2a_k)}{(1-a^2{}_k)^2}\right)$$

$$= -\left(\frac{-2e_k + 2a_k g_k}{1-a_k^{\ 2}}\right)$$

$$\frac{\delta g_k}{\delta net_k} = -\left(\frac{-2e_k + 2a_k g_k}{1-a_k^{\ 2}}\right)(1-a_k^{\ 2})$$

$$= 2(e_k - a_k g_k) = \delta_k$$

Error signal at the hidden layer is $\delta_j = \sum \delta_k w_{kj} f'(net_j)$

MODIFICATION OF IMPROVED ERROR WITH MSE

$$\rho_k = \frac{E_k^2}{2a_k(1-a_k^2)} \text{ where } E = \frac{1}{2}\sum_k (t_{kj} - o_{kj})^2, \text{ Log(E)} = \log(\frac{1}{2}\sum_k (t_{kj} - o_{kj})^2)$$

Activation function: Sigmoid $f(x) = 1/(1+e^{-2x})$

We modify the error function further by substituting the error term $E_k = t_k - a_k$ with mean squared error, in the improved error function in Eq. 6 $\rho_k = \frac{E_k^2}{2a_k(1-a_k^2)}$.

Where:

$$E = \frac{1}{2}\sum_k (t_{kj} - a_{kj})^2$$

E_k = Error at output unit k
t_k = Target value of output unit k
a_k = An activation of unit k

$$\frac{\partial \rho_k}{\partial Net_k} = \frac{-\left[\frac{4}{n^2}\sum (t_k - a_k)^3 + 2\rho_k(1-3a_k^2)\right]}{(1+a_k)} = \delta_k$$

If n = 2, then the error signal becomes:

$$\frac{\partial \rho_k}{\partial Net_k} = \frac{-\left[\sum (t_k - a_k)^3 + 2\rho_k(1-3a_k^2)\right]}{(1+a_k)} = \delta_k$$

Activation function: Inverse tangent, f(x) = tan^{-1} (x)

$$\frac{\partial \rho_k}{\partial Net_k} = \frac{-\left[\frac{1}{2}\sum (t_k - a_k)^3 + \rho_k(1-3a_k^2)\right]}{a_k(1-a_k^2)}(kos^2 a_k)$$

If n = 2, then the error signal becomes:

$$\frac{\partial \rho_k}{\partial Net_k} = \frac{-\left[\frac{1}{2}\sum (t_k - a_k)^3 + \rho_k(1-3a_k^2)\right]}{a_k(1-a_k^2)}(kos^2 a_k) = \delta_k$$

Activation function: Tangent hyperbolic f(x) = tan h (x)

$$\frac{\partial \rho_k}{\partial Net_k} = \frac{\partial \rho_k}{\partial a_k} . \frac{\partial a_k}{\partial Net_k}$$

$$= -\left(\frac{\frac{4}{n^2}\sum (t_k - a_k)^3 + 2\rho_k(1-3a_k^2)}{a_k\left(1-a_k^2\right)}\right)(1-a_k)(1+a_k)$$

If n = 2, then the error signal becomes:

$$= -\left(\frac{\sum (t_k - a_k)^3 + 2\rho_k(1-3a_k^2)}{a_k}\right) = \delta_k$$

RESULTS AND DISCUSSION

The variables used in this study is to train and test the backpropagation networks including the peak load demand, mean temperature, maximum temperature and minimum temperature for Peninsular Malaysia, previous day load, previous 7 days load (same day of the previous week) and the previous 6 days load. The input neurons were selected based on a correlation analysis which shows the existence of a significant correlation between peak load, lag loads and temperature. Thus there are seven input signals to the neural network with one hidden layer and one output layer which produces the single output – the system peak load. In this study, various combinations of networks were trained to get the optimum value. For comparison, we selected networks with 10 and 20 hidden neurons. The performance of proposed error function was compared to the mean squared error from standard backpropagation. Both algorithm were measured using MAPE (mean absolute percentage error), given as:

Table 1: Comparison of convergence speed between sigmoid and arctangent (hidden = 20)

		Improved error (sigmoid)		Improved error (arctangent)	
η	α	Time (sec)	Iteration	Time (sec)	Iteration
0.01	0.9	3	566	153	27850
	0.8	5	1091	221	40138
	0.7	9	2060	137	45367
0.02	0.9	1	315	199	36470
	0.8	3	607	333	60705
0.03	0.7	3	706	566	102737

Table 2: Comparison of convergence speed between sigmoid and arctangent (hidden = 10)

		Improved error (sigmoid)		Improved error (arctangent)	
η	α	Time (sec)	Iteration	Time (sec)	Iteration
0.01	0.9	2	509	67	21924
	0.8	3	912	99	32865
	0.7	4	1329	137	45367
0.02	0.9	1	341	104	34598
	0.8	2	558	98	32654
0.03	0.7	2	631	270	89093

Table 3: Comparison of percentage error between sigmoid and arctangent (hidden = 20)

		Improved error (sigmoid)		Improved error (arctangent)	
η	α	Learning	Testing	Learning	Testing
0.01	0.9	5.088694	6.203020	3.245404	5.743697
	0.8	5.133071	5.700943	3.249369	5.337469
	0.7	5.186969	5.555489	3.192278	6.454494
0.02	0.9	5.169751	5.880153	3.510042	7.518763
	0.8	5.108866	5.789114	3.379688	6.083081
0.03	0.7	5.137113	5.665796	3.525647	6.454494

$$\text{MAPE} = \frac{1}{N}\sum_{i=1}^{N}\frac{|p_i - a_i|}{a_i}$$

where, p_i is the predicted value and a_i is the actual value, N is the number of patterns used for training and testing.

Table 1 and 2 show the convergence rates of improved error used by backpropagation algorithm for the sigmoid function and the arctangent function. From the chosen learning rate (η) and momentum (α), it is clear that improved error with sigmoid function converges faster compared to improved error with arctangent function. This is true for all chosen learning rate (η) and momentum (α) and for both 20 and 10 hidden nodes.

Table 3 shows the forecasting accuracy of daily maximum load using hidden nodes of 20 and Table 4 shows the forecast accuracy using 10 hidden nodes. From these results, we find that the inverse tangent function has smaller learning error compared to sigmoid function for all chosen learning rate (η) and momentum (α). When both networks were used for testing for two months ahead, the percentage errors of both functions was comparable in the range of 5-8%.

Table 4: Comparison of percentage error between sigmoid and arctangent: (hidden = 10)

		Improved error (sigmoid)		Improved error (arctangent)	
η	α	Learning	Testing	Learning	Testing
0.01	0.9	5.111438	5.795093	3.235687	5.701169
	0.8	5.115418	6.021092	3.250164	5.640709
	0.7	5.120176	5.823463	3.214065	5.741585
0.02	0.9	5.154532	5.667570	3.218197	5.783885
	0.8	5.163044	5.446131	3.221758	5.569385
0.03	0.7	5.133071	5.700943	3.247321	5.252509

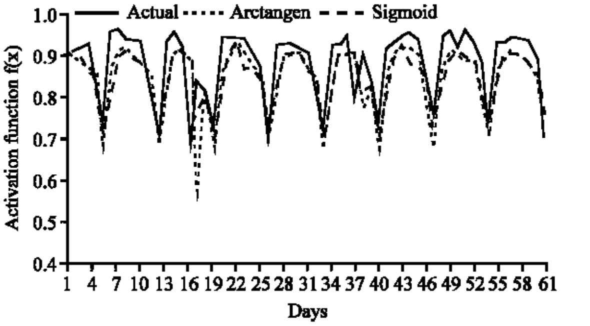

Fig. 8: Forecasted Load for arctangent and sigmoid functions (hidden = 20, Eta = 0.01, alpha = 0.9)

Figure 8 shows the forecasted load for months of March and April 2006. From the graph, both activation functions show that there is not much difference in terms of forecast accuracy. However, but when forecast value were plotted, we can see that the sigmoid function does follows the actual pattern better compare to arctangent function, even though the accuracies of both functions are not quite satisfactory for forecasting purposes.

CONCLUSION

This study introduces improved error signals of backpropagation using two activation functions, the sigmoid and the arctangent, in forecasting daily maximum electricity load. The improved error has been tested on daily maximum load data for year 2006 until 2007 for Peninsular Malaysia. The result shows that convergence rate of improved error with sigmoid function is much faster compared to the improved error with arctangent function. The arctangent function should theoretically yield faster convergence speed as it has higher derivative values compare to the sigmoid function. The results are contradictory to the expectation. In terms of forecast accuracy both function shows similar results, with sigmoid function following the actual pattern better. From both results, it seems that sigmoid function with improved error has better convergence rate and better forecast ability but needs further refinement in terms of forecast accuracy to be used as forecasting tools.

ACKNOWLEDGMENTS

The research is supported by IRPA Grant Vote No. 03-01-06-SF0177 and the Department of Mathematics, Faculty of Science UTM. The authors would like to thank lecturers and friends for many helpful ideas and discussions. We also wish to thank TNB for providing necessary data in this study.

REFERENCES

Bansel, A., R. Kauffman and R. Weitz, 1993. Comparing the modeling performance of regression and neuralnetworks as data quality varies: A business value approach. J. Manage. Inform. Syst., 10: 11-33.

Faraway, J. and C. Chatfield, 1998. Time series forecasting with neural networks: A case study. J.R. Stat. Soc.: Ser. C Applied Statistics, 47: 231-250.

Haykin, S., 1999. Neural Networks: A Comprehensive Foundation. 2nd Edn. Prentice Hall, New Jersey .

Herrera, L.J., H. Pomares, I. Rojas, A. Guillén, A. Prieto and O. Valenzuela, 2007. Recursive prediction for long term time series forecasting using advanced models. Neurocomputing, 70: 2870-2880.

Ismail, Z. and K.A. Mahpol, 2007. Neuro-fuzzy forecasting for peak load electricity demand. Prosiding Seminar Kebangsaan Aplikasi Sains dan Matematik, 28-29 Nov., Malaysia, pp: 371-382.

Kalman, B.L. and S.C. Kwasny, 1997. High performance training of feedforward and simple recurrent networks. Neurocomputing, 14: 63-83.

Khamis, A., Z. Ismail, K. Haron and A.T. Mohammed, 2006. Neural network model for oil palm yield modeling. J. Applied Sci., 6: 391-399.

Kim, H. and C. Singh, 2005. Power system probabilistic security assessment using bayes classifier. Elect. Power Syst. Res., 74: 157-165.

Lai, L.L., 1998. Intelligent System Applications in Power Engineering: Evolutionary Programming And Neural Networks. 1st Edn. John Wiley and Sons, West Sussex, England.

Lee, T.L., 2008. Backpropagation neural network for prediction of the short storm surge in Taichung harbor, Taiwan. Eng. Applic. Artificial Intell., 21: 63-72.

Limsombunchai, V., C. Gan and M. Lee, 2004. House price prediction: Hedonic price model vs. artificial neural network. Am. J. Applied Sci., 1: 193-201.

Mandal, D.S., K. Pal and P. Saha, 2007. Model of electricity discharge machining process using back propagation neural network and multi-objective optimization using non-dominating sorting genetic algorithm-II. J. Mater. Process. Technol., 186: 154-162.

Nima, A., 2001. Short-term hourly load forecasting using time series modeling with peak load estimation capability. IEEE Trans. Power Syst., 16: 498-505.

Rumelhart, D.E. and J.L. McClelland, 1986. The PDP Research Group, Parallel Distributed Processing. 1st Edn. MIT Press, USA .

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.