SCIENCE & TECHNOLOGY

PERTANIKA
JOURNALS

External Constraints of Neural Cognition for CIMB Stock Closing Price Prediction

Chang Sim Vui[1], Chin Kim On[1]*, Gan Kim Soon[1], Rayner Alfred[1] and Patricia Anthony[2]

[1]*Faculty of Informatics and Computing, Universiti Malaysia Sabah, 88400 Kota Kinabalu, Sabah, Malaysia*
[2]*Department of Applied Computing, Faculty of Environment, Society and Design, Lincoln University, Christchurch, New Zealand*

ABSTRACT

This paper investigates the accuracy of Feedforward Neural Network (FFNN) with different external parameters in predicting the closing price of a particular stock. Specifically, the feedforward neural network was trained using Levenberg-Marquardt backpropagation algorithm to forecast the CIMB stock's closing price in the Kuala Lumpur Stock exchange (KLSE). The results indicate that the use of external parameters can improve the accuracy of the stock's closing price.

Keywords: Artificial Neural Network (ANNS), Backpropagation, Feedforward Neural Network (FFNN), Levenberg-Marquardt algorithm, Macroeconomic parameters, Stock market forecasting

INTRODUCTION

Stock market plays a vital role in today's economy. The performance of the stock market reflects the country's economy. If the stock market is good, the index value

of the stock market will rise and this is a good indicator that the country's economy is growing. For investors, the goal is to gain profit in which they buy the stock at a low price and sell it off at a higher price. However, the volatile, nonlinear and uncertain nature of stock market trading makes it more complicated for investors as they are not able to predict the stock's price and as such, it is difficult to decide when to buy and sell the stock that can guarantee a profit. Thus, many theories have been developed to uncover the great fortune in the stock market. Among the earlier theories for the stock market forecasting are efficient market hypothesis (EMH) and random walk theory. The EMH

states that the market is efficient, and thus, all the information is fully reflected in the price. This causes the stock price to be unpredictable because the information which can influence the stock price has already affected the stock price once it's available (Malkiel & Fama, 1970). There are three types of EMH: are weak, semi-strong and strong. The different types of the EMH reflect the different level of information available to predict the stock price, from the stock closing price to all the information that can influence the stock price. Whereas in the random walk theory, the stock market price fluctuates randomly and there is no dependency and relationship in the fluctuation of prices (Malkiel, 1999). As a result, the stock market price is hard to predict since there is no trend in the price fluctuation.

Many techniques have been developed to challenge the traditional prediction theories such as fundamental analysis and technical analysis. Fundamental analysis is a stock price technique which is based on economic variables and the company's profile such as annual report, balance sheet, cash flow statement and income statement (Graham, Dodd, & Cottle, 1934; Thomsett, 1998). Data used in fundamental analysis are long term value that does not reflect immediate changes on a daily basis making it unsuitable for short term predictions. Technical analysis is a numerical time series approach to predict stock markets based on historical data by using charts as the primary tool (Pring, 2002). Technical indicators are used such as autoregressive, moving average, and RSI to predict the stock market's movement. However, technical analysis method is highly subjective to the judgment of the analyst.

Traditional time series forecasting techniques such as moving average (MA), autoregressive integrated moving average (ARIMA) and Box-Jenkins are also used to predict the stock market's closing price (Hibon & Makridakis, 1997; Rao & Miller, 1971). However, these linear forecasting methods are unable to capture the non-linearity characteristic of the stock market. Researchers have come up with several parametric nonlinear models such as autoregressive conditional heteroskedasticity and general autoregressive heteroskedasticity to predict the nonlinearity in the function. However, these nonlinear statistic techniques require a priori knowledge regarding the nonlinear model to estimate or predict the function correctly. With the advancement of computation capability, researches have experimented with machine learning techniques to solve the prediction problem.

Machine learning is a computational approach that uses experience to learn and improve the system performance or the prediction's result (Mitchell, 1997). There are many types of machine learning algorithms such as decision tree learning, data mining algorithms, support vector machines and artificial neural network. Among these algorithms, artificial neural network has been widely used in solving prediction tasks related to robotics (Kim, Teo, & Saudi, 2008), computer games (Tong, On, Teo, & Kiring, 2011; Tan, Teo, & Chin, 2013), speech recognition (On et al., 2006a), image processing (On et al., 2006b), to name a few.

Artificial neural network (ANN) is a bio-inspired algorithm that mimics the mechanism of the human brain (Haykin, 2004). The basic building block for ANN is an artificial neuron, also known as a simple processing unit, where each neuron will perform simple computation in parallel. An ANN is formed from several components which are organised in layers. These components are input neurons, output neurons, hidden neurons, activation function, bias and weight (Engelbrecht, 2007). The input neuron receives signal from the external sources. The hidden neuron performs the computation to obtain the net input by multiplying the connection

weight with the input signal that are connected to it plus a bias value. This net input is then passed to a transfer function to produce an output signal. The output neuron performs the same computation as the hidden neuron and produces the output value. The ANN can be organised in different topologies referred to as ANN architectures. There are many different types of ANN architectures such as feedforward neural network (FFNN) and recurrent neural network (RNN).

In Pérez-Rodríguez et al., (2005), the Spanish Ibex-35 stock index returns were predicted using daily returns. Data was collected from 30 December 1989 to February 2000 which amounted to 2520 observations. A linear AR model, the ESTAR and LSTAR smooth transition autoregressive models and three ANN models - Multilayer perceptron, JCN and Elman networks - were used in the experiments. The results showed that MLP with one-step ahead prediction performed better than linear model. In Abraham et al., (2001), ANN was used to predict one day ahead of the Nasdaq-100 index. In this work, historical data Nasdaq-100 index, six of the companies listed in this index which included Microsoft, Yahoo, Cisco, Sun Microsystems and Oracle were also used together to forecast the index value. The ANN was trained using scaled conjugate gradient algorithm. A 24-month data set from 22 March 1999 to 20 March 2001 were used in the experiment. The ANN prediction showed promising result it achieved a RMSE value of 0.021 - 0.034 for the predicted stocks. Tilakaratne, Morris, Mammadov, & Hurst (2007) predicted the one day ahead of the Australian All Ordinary Index based on the current day returns and the US S&P 500 Index, the UK FTSE 100 Index, French CAC 40 Index and German DAX Index; this was based on data obtained from 2nd July 1997 to 30th December 2005. The FFNN and probabilistic neural network were used in this investigation. The results showed that FFNN performed better in this study than probabilistic neural network. Schierholt & Dagli (1996) predicted the one day ahead of the Standard and Poor's (S&P) 500 Index by using S&P 500 historical data and the currency exchange rates of Japanese Yen, British Pound and German Mark. Multilayer perceptron and probabilistic neural network were used to predict the S&P 500 index based on the previous 5 days' data. The data set used in the experiments is from February 1994 to September 1995. In this study, probabilistic neural network performed slightly better than MLP.

In Sutheebanjard and Premchaiswadi (2010), the back propagation neural network is used to forecast the stock exchange of Thailand (SET) index. Additionally, the Dow Jones index, Straits Times index, Nikkei Index, Hang Seng index, domestic minimum loan rate and the domestic gold price were used as ANN input to predict the SET index. A data set of 124 trading days from 2 July 2004 to 30 December 2004 were used in this study. These studies had used external parameters such indexes and currency exchange rates to predict the closing price or index value while Dong, Fataliyev and Wang (2013), Kumar and Murugan (2014), Jabin, (2014) and Dhiraj, Gaurav, Sachin and Trupti (2015) used either FFNN or external parameters for stock prediction.

In this paper, FFNN was used to forecast the CIMB stock closing price. The CIMB stock was selected due to its price fluctuation. The CIMB Group, a financial institution in Malaysia, had grown to become one of the most powerful banking powerhouses in ASEAN, and is listed in Kuala Lumpur Stock Exchange. Most of the studies used past closing price to predict the future closing price. In this study, besides the CIMB's closing price, external parameters such as KLCI index, interest rate and currency exchange rates were used to improve the accuracy of

prediction. The influence of these external parameters on the accuracy of the stock's prediction is investigated in this work. Experiments were carried out and the results were analysed to evaluate the effect of the external parameters.

The remainder of the paper is structured in the following manner. Section 2 reviews important studies related to this topic. Section 3 explains the experimental setup and the configuration of FFNN while the simulation prototypes are described in Section 4. Section 5 discusses the results while Section 6 concludes the paper.

MATERIALS AND METHODS

This section describes the parameter configuration used to conduct the experiment. There are many configurable parameters in ANN such as its architecture, number of inputs and hidden neurons, transfer function and learning algorithm. The setting of the ANN is configured to predict one day ahead of CIMB stock closing price.

As an initial step, data relating to CIMB stock information (open price, close price, highest price, low price and volume trade) and the market reflected information such as interest rate, KLCI and currency exchange rates (USD, EUR and SGD) were used as input for the neural network. The inputs for the experiments are open price, closing price, highest price, lowest price, volume trade, KLCI, interest rate, USD, EUR and SGD. The first five parameters are often used in technical analysis. KLCI is an index value that is affected by the stocks in the Kuala Lumpur stock exchange which reflects health of the stock market. The interest rate reflects the economic growth and monetary policy. Meanwhile, USD, EUR and SGD were selected as these international currencies are traded the most in Malaysia. Based on these parameters, data from the previous five consecutive days was used to predict the CIMB stock closing price on the sixth day. Data for this research included stock information and KLCI index retrieved from Yahoo Finance, while the interest rate was obtained from the central bank of Malaysia (Bank Negara Malaysia). The currency exchange rates are obtained from Oanda forex. These data sets were retrieved between January 2000 and Jun 2015.

Before the data was fed into the neural network, they were first pre-processed which entailed finding the missing value, outlier and normalisation. Data was then tallied according to the fixed aforementioned timeline. If there is a missing value, this missing value is then derived by averaging the previous and next day's values. The outlier of each parameter data set was determined using interquartile range. The outlier is the value that is less than $Q1 - (1.5 * IQR)$ and value greater than $Q3 + (1.5*IQR)$ where Q1 is the first quartile, Q3 is the third quartile and IQR is the interquartile range. Finally, each of the parameter data set is normalised and denormalised using the following equation where x is the data value, x_{min} the smallest data value, x_{max} the largest data value while x_n is the normalised value:

$$x_n = \frac{x - x_{min}}{x_{max} - x_{min}}$$

$$x = x_n(x_{max} - x_{min}) + x_{min}$$

The output of the neural network will be the prediction of the 6th day stock closing price based on the previous 5 days' data. After the data is pre-processed, the input data is arranged into a vector called x while the target value is arranged into a vector called y. The data is arranged into time series of 4000 data points (x, y) for computation. This amount of data is sufficient for the training, validation and testing sets with data points divided into 70%, 15% and 15% respectively (Sivanandam & Paulraj, 2009).

The number of epoch used in this research is 1000. However, early termination is possible to avoid the ANN memorising the pattern instead of learning during the training process. The feedforward neural network (FFNN), one of the widely used neural network architectures, is used in this study to forecast the CIMB stock closing price. As its name implies, FFNN pattern connection is directed in one way which is feed forward. A two-layer neural network architecture is used in this experiment. In other words, the ANN used in this research comprises one input layer, one hidden layer and one output layer. It has been proven that a hidden layer with enough hidden neurons can approximate any continuous function (Cybenko, 1989; Hornik, Stinchcombe, & White, 1989). The number of hidden neurons in the hidden layer is selected based on a rule of thumb which is the number of input neuron plus the number of output neuron divided by two. Several researches have been using this particular rule of thumb to decide on the number of hidden neurons to be selected (Shahidehpour, Yamin, & Li, 2002; Sharma & Sharma, 2009; Berg, Engel, & Forrest, 1998).

The training algorithm used in this study is Levenberg-Marquardt algorithm. Levenberg-Marquardt algorithm (LMA) was first introduced by Kenneth Levenberg (1994) and later extended by Donald Marquardt (1963). It is also known as the damped least squares (DLS) method. This algorithm helps in solving non-linear square problems by finding the minimum point of a function. Levenberg-Marquardt algorithm is also used to train backpropagation neural network (Hagan & Menhaj, 1994). The Levenberg-Marquardt is an algorithm that interpolates between the methods of gradient descent and Gauss-Newton algorithm.

The hyperbolic tangent is used as the activation function which fulfils the requirement for backpropagation algorithm (Vogl, Mangis, Rigler, Zink, & Alkon, 1988). Mean square error (MSE) is used to evaluate the performance of the ANN in predicting the closing price of the stock. The MSE is calculated based on the formula below:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2$$

where ŷ is the output of ANN, y is the target value of the input-output data pair and n is total input-output data pairs. Table 1 is a summary of the experimental setting for this study.

Table 1
Experiment parameters

Parameter	Value
Input Value	Closing price, open price, highest price, lowest price, volume trade, KLCI, interest rates, currency exchange rates (USD, EUR, and SGD)
No. of Historical Day	5
Output Value	6th day closing price
ANN Architecture	2-layer FFNN
No. Hidden Neurons	(input neurons + output neurons) / 2
Epoch	1000
Activation Function	tansig
Training Data Set	70%
Testing Data Set	15%
Validation Data Set	15%
Training Algorithm	Levenberg-Marquardt Algorithm
Evaluation Function	MSE
No. of Run Per Experiment	10

Simulation

In order to facilitate the experiment, a simulation prototype was developed. The prototype, built using MATLAB, allows for flexible configuration of the parameters to be tested. The simulation prototype allows for flexibility of parameter configuration in the experimental setup and reduces the effort for rewriting codes for different experiments, thereby, making it desirable for future experiments. In the interface, it allows the user to set various parameters from input parameter to the neural network setting and also the directory location of the result to be saved. The interface consists of 4 main panels: data panel, network panel, training method panel and result panel.
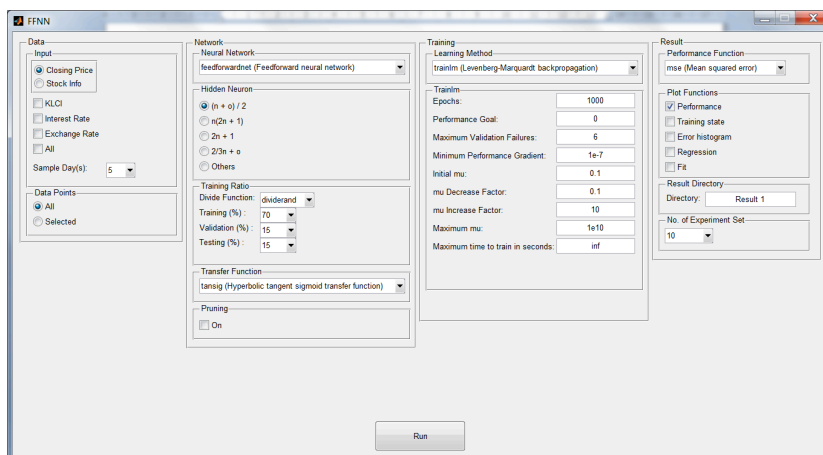

Figure 1. The simulation prototype

The data panel contains all the input parameters selection that can be chosen in the experiment. These parameters include the closing price, the stock information (open price, highest price, lowest price, and volume trade), KLCI index, the interest rate and foreign currency exchange rates (USD, EUR and SGD). There is also an option for the user to choose all or some of the data points. The network panel consists of the parameter configuration for FFNN which are the number of neurons to be used, the partition ratio of the training, validation and testing data set and the transfer function. The training panel specifies the learning algorithm (in this case is the Levenberg-Marquardt algorithm) and configuration setting for the algorithm which consists of the training epoch, performance goal, maximum validation failures, minimum performance gradient, the initial mu value, the mu decrease factor, the mu increase factor, the maximum mu value and the maximum time to train in seconds. The last panel is the result panel which allows the user to choose the performance functions, the different plot function, the directory to store the result and the number of experiments to be run in each test.

RESULTS

A series of experiments were carried out in this experiment to investigate the influence of the external parameters in predicting accurately the stock's closing price. The result shows that each of the external parameters does improve the accuracy prediction by achieving a lower MSE value. Combining all the parameters together improves prediction accuracy. The result is shown in Table 2.

Table 2
Experiment MSE result

Input	Best MSE Result	Mean MSE Result
Closing price	0.03724	0.03743
Closing price, open price, highest price, lowest price, volume trade	0.03478	0.03628
Closing price, KLCI	0.03679	0.03702
Closing price, interest rate	0.03547	0.03700
Closing price, exchange rates (USD, EUR, SGD)	0.02396	0.03432
All parameters above	0.02198	0.03149

Based on Table 2, it can be seen that each of the external parameter can generate better MSE result than the closing price alone. The MSE value of taking all the external parameters as input generated the best MSE value in both individual and the mean result of the 10 different runs. Thus, it can be concluded that each of the tested external parameters improves the prediction accuracy and by combining all these parameters, the MSE can be reduced by 40%. Figure 2 shows the prediction accuracy of the FFNN that used all the parameters. The blue circle is the actual closing price and the red asterisk is the predicted output. It can be seen that the predicted output (red line) is very close to the blue line which means the prediction is very close to the actual closing price.
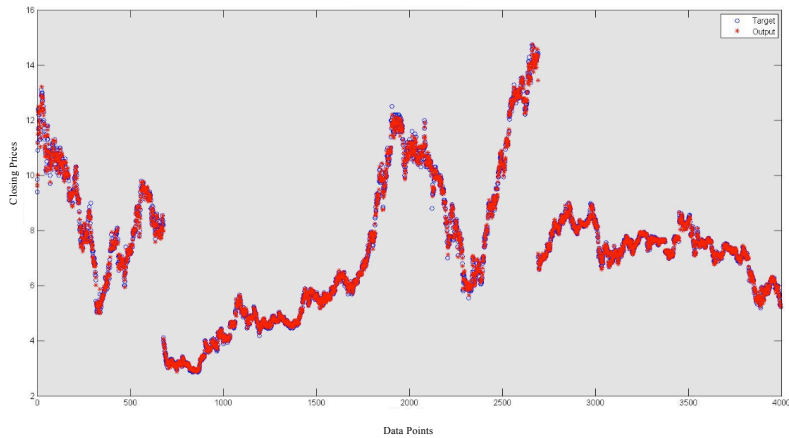
Figure 2. The prediction accuracy of FFNN that used all external parameters

CONCLUSION

This paper describes the application of FFNN in predicting the CIMB stock closing price. The results showed that FFNN performed well in the prediction task by achieving more than 90% prediction accuracy. Additionally, different input parameters were used in tuning the FFNN. The external parameters tested in this study seemed to have influence on the stock's closing price prediction by generating a better MSE value compared with just using the closing price as an input to FFNN. Combining all these influential parameters as input to FFNN is able to reduce 40% in MSE value compared with just closing price alone. In the future, more external parameters and technical indicators such as moving average and RSI can be considered to improve the prediction accuracy. Besides that, different ANN architectures can be used to investigate their performance on predicting the stock closing price.

REFERENCE

Abraham, A., Nath, B., & Mahanti, P. K. (2001). Hybrid intelligent systems for stock market analysis. In *Computational science-ICCS 2001* (pp. 337-345). Springer Berlin Heidelberg.

Berg, E. P., Engel, B. A., & Forrest, J. C. (1998). Pork carcass composition derived from a neural network model of electromagnetic scans. *Journal of Animal Science, 76*(1), 18-22.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems (MCSS), 2*(4), 303-314.

Dhiraj, M., Gaurav, C., Sachin, C., & Trupti, B., (2015). Stock value prediction system. *International Journal on Recent and Innovation Trends in Computing and Communication (IJRITCC), 3*(4), 2217 - 2219

Dong, G., Fataliyev, K., & Wang, L. (2013, December). One-step and multi-step ahead stock prediction using backpropagation neural networks. *9th International Conference on Information, Communications and Signal Processing (ICICS)* (pp. 1-5). IEEE.

Engelbrecht, A. P. (2007). *Computational intelligence: An introduction.* John Wiley & Sons.

Graham, B., Dodd, D. L. F., & Cottle, S. (1934). *Security analysis* (p. 54). New York: McGraw-Hill.

Hagan, M. T., & Menhaj, M. B. (1994). Training feedforward networks with the Marquardt algorithm. *IEEE transactions on Neural Networks, 5*(6), 989-993.

Haykin, S., & Network, N. (2004). A comprehensive foundation. *Neural Networks, 2*(2004).

Hibon, M., & Makridakis, S. (1997). ARMA models and the Box–Jenkins methodology.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks, 2*(5), 359-366.

Jabin, S. (2014). Stock market prediction using feed-forward artificial neural network. *Int. J. Comput. Appl.(IJCA), 99*(9).

Kim On, C., Teo, J., & Saudi, A. (2008, June). Multi-objective artificial evolution of RF-localization behavior and neural structures in mobile robots. *IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)* (pp. 350-356). IEEE.

Kumar, D. A., & Murugan, S. (2014, February). Performance analysis of MLPFF neural network back propagation training algorithms for time series data. *World Congress on Computing and Communication Technologies* (pp. 114-119). IEEE.

Levenberg, K. (1944). A method for the solution of certain problems in least-squares. *Quart. Appl. Math., 2,* 164-168.

Malkiel, B. G. (1999). A *random walk down Wall Street: including a life-cycle guide to personal investing.* WW Norton & Company.

Malkiel, B. G., & Fama, E. F. (1970). Efficient capital markets: A review of theory and empirical work. *The Journal of Finance, 25*(2), 383-417.

Marquardt, D. W. (1963). An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial and Applied Mathematics, 11*(2), 431-441.

Mitchell, T. M. (1997). Machine learning. WCB.

On, C. K., Pandiyan, P. M., Yaacob, S., & Saudi, A. (2006a). Formants analysis with zero crossing extraction in automatic speech recognition using neural network. *Engineering e-Transaction, 1*(2), 33-39.

On, C. K., Pandiyan, P. M., Yaacob, S., & Saudi, A. (2006b, June). Mel-frequency cepstral coefficient analysis in speech recognition. *International Conference on Computing and Informatics, ICOCI* (pp. 1-5). IEEE.

Pring, M. J. (2002). *Technical analysis explained: The successful investor's guide to spotting investment trends and turning points.* McGraw-Hill Professional.

Pérez-Rodríguez, J. V., Torra, S., & Andrada-Félix, J. (2005). STAR and ANN models: Forecasting performance on the Spanish "Ibex-35" stock index. *Journal of Empirical Finance, 12*(3), 490-509.

Rao, P., & Miller, R. L. (1971). Applied Econometrics. Belmont: Wadsworth Publishing Co. Inc. *Rao Applied Econometrics 1971.*

Schierholt, K., & Dagli, C. H. (1996, March). Stock market prediction using different neural network classification architectures. *Proceedings of the IEEE/IAFE Conference on Computational Intelligence for Financial Engineering* (pp. 72-78). IEEE.

Shahidehpour, M., Yamin, H., & Li, Z. (2002). *Frontmatter and index* (pp. i-xiv). John Wiley & Sons, Inc.

Sharma, A. K., & Sharma, R. K. (2009). Effectiveness of heuristic rules for model selection in connectionist models to predict milk yield in dairy cattle. *TECHNIA–Int. J. Comput. Sci. Commun. Technol, 2*(1).

Sivanandam, M. & Paulraj, M. (2009). *Introduction to artificial neural networks*. Vikas publishing House PVT LTD.

Sutheebanjard, P., & Premchaiswadi, W. (2010, April). Stock exchange of Thailand index prediction using back propagation neural networks. *Second International Conference on Computer and Network Technology (ICCNT)* (pp. 377-380). IEEE.

Tan, T. G., Teo, J., & Chin, K. O. (2013). Single-versus multiobjective optimization for evolution of neural controllers in Ms. Pac-Man. *International Journal of Computer Games Technology, 2013*.

Thomsett, M. C. (1998). *Mastering fundamental analysis*. Dearborn Financial Pub.

Tilakaratne, C. D., Morris, S. A., Mammadov, M. A., & Hurst, C. P. (2007, September). Predicting stock market index trading signals using neural networks. In *Proceedings of the 14th Annual Global Finance Conference (GFC'07)* (pp. 171-179).

Tong, C. K., On, C. K., Teo, J., & Kiring, A. M. J. (2011, September). Evolving neural controllers using GA for Warcraft 3-real time strategy game. Sixth International Conference on *Bio-Inspired Computing: Theories and Applications (BIC-TA)* (pp. 15-20). IEEE.

Vogl, T. P., Mangis, J. K., Rigler, A. K., Zink, W. T., & Alkon, D. L. (1988). Accelerating the convergence of the back-propagation method. *Biological Cybernetics, 59*(4-5), 257-263.