STOCK PRICE PREDICTION IN KLSE USING NEURAL NETWORKS

Mohammad Faidzul Nasrudin,
Khairuddin Omar, Masri Ayob & Miswan Surip
Faculty of Technology and Information Sciences
Universiti Kebangsaan Malaysia
43600 Bangi, Selangor Darul-Ehsan
contact email: ko@ftsm.ukm.my

ABSTRACT

The use of neural networks in financial market prediction presents a major challenge to the design of effective neural network predictors. This paper presents a study to evaluate capabilities of five prediction approaches that use backpropagation neural networks model in predicting stock prices of Kuala Lumpur Stock Exchange (KLSE). The approaches considered are the Standard, Volatility Data, Technical and Fundamental Data, Data Aggregation, and the Sector-Counter Approach. We found that the network that considers Sector-Counter indices is the best performing network in the prediction. The complexity of the financial market data probably explains why some of the approaches cannot provide any significant improvement in the prediction.

Keywords
Neural Network, Stock Market, Prediction, Time Series, KLSE.

ABSTRAK

Penggunaan Rangkaian Neural dalam meramal pasaran saham telah menghasilkan cabaran kepada rekabentuk peramal rangkaian neural yang berkesan. Kertas ini memaparkan satu kajian penilaian keupayaan kepada lima pendekatan peramal menggunakan model rangkaian neural Perambat-balik dalam meramal harga saham di Bursa Saham Kuala Lumpur (BSKL). Pendekatan-pendekatan ini ialah Biasa, Data Volatiliti, Data Teknikal dan Fundamental, Agregasi Data, dan Sektor-kaunter. Kami mendapati bahawa rangkaian dari pendekatan Sektor-Kaunter merupakan kaedah yang terbaik berbanding kaedah-kaedah yang lain. Kekompleksian data sektor pasaran saham kewangan merupakan faktor yang menyebabkan sebahagian daripada kaedah tersebut gagal dalam ramalannya.

Katakunci
Rangkaian Neural, Pasaran Saham, Ramalan, Siri Masa, BSKL.

INTRODUCTION

The application of Neural Networks to finance sector has experienced continuous growth in research and publication since the pioneering research by White (1988). Different financial applications have been explored by researchers to proof the success of Artificial Neural Network (ANN) in financial domain. Some of them are stock market by Op't Landt (1997), gold market by McCann & Kalman (1994), future market by Trippi & DeSieno (1992), currency exchange by Refenes *et al.* (1993), corporate bond rating by Moody & Utans (1995), and bankruptcy by Odom & Sharda (1990).

The rapid progress of the study in the last few years has generated a lot of interest in the short-term stock market prediction by using neural networks. The recent study in this area have been done by Yao *et al.* (1998). They conducted a case study on the forecasting of the Kuala Lumpur Composite Index (KLCI) using backpropagation neural networks. They found that some of the results were not so compromising due to the high volatility of the KLSE market and neural networks' capabilities maybe can be improved by using a mixture of technical and fundamental factors. They concluded that useful prediction still can be made without the use of extensive market data or knowledge.

The research above is an example of evidance that can be considered a violation of two major market trading theories which are the *Random Walk Hyphotesis* and the *Efficient Market Hyphotesis* (Hellström & Hellström, 1998). Both of the theories agree that the prices on the stock market wander in random and unpredictable manner. Another research that against these theories would be Md. Nasir & Mohamed (1993). They found the presence of *day-of-the-week effects or weekend effect* on KLSE market. They concluded that Monday and Tuesday average returns are negative while Friday returns is the highest. Other studies such as Nassir (1983), Laurance (1986), Barnes (1986), and Neoh (1986) had reported the presence of weak state of the hyphotesis in KLSE. These findings show that stock prices in KLSE can be predicted using historical data.

In this paper, we are interested in studying five prediction approaches that use backpropagation neural networks model in predicting stock prices in KLSE. Each approach has different input data but same output data which is tomorrow returns. Basically, the input data is taken from delayed time-series data from actual data of the KLSE, major technical indicators and economic fundamental indicators. We trained the neural networks to approximate the relationships between the input dan output data. Then, we use the same trained neural network to predict the tomorrow price of the particular stock.

After performing all the approaches, we found that some of the issues in Yao *et al.* (1998) and Md. Nasir & Mohamed (1993) really need to be considered when building a predicting model. The network that considers sector indices is the best performing network in the prediction.

HISTORICAL DATA CHARACTERISTICS

The initial data as provided by Netlink Technology, Sibu, Sarawak consists of 22 sets. 21 directories represents sectors data. Another directory represents KLSE indices data. In those sectors directories stored all counters historical daily data of that particular sector. All counter data sets cover stock market prices from 28 May 1992 until 9 June 2000. Each data set has

- P_O, price of open,
- P_H, price of high,
- P_L, price of low,
- P_C, price of close,
- V, volume.

We divide the data into two parts which are training and testing. For training, we use up to 5 years pre-processed historical data from 1 October 1993 until 30 September 1998, which is 5/6 of total data and consists of 1232 patterns. For testing, we run the remaining 1/6 of the data, which is 1 year data from 1 October 1998 to 30 September 1999.

NEURAL NETWORKS PREDICTION APPROACHES

Many different approaches can be used for stock price prediction. In this paper, we have tested five of them. For every stock we have tested, all results of the following approaches have been observed:

- Standard Approach. Stock price prediction using commonly used technical indicators.
- Prediction using standard approach indicators and stock price volatility indicator. This will be called the Volatility Data Approach.
- Stock price prediction using Sector-counter Approach. This approach is based on the idea that the individual stock is influenced by the index of its

sector.

- Technical and Fundamental Data Approach. Stock price prediction using indicators of Standard approach and some macroeconomic indicators.
- Stock price prediction that deals with the weekend effect. This approach is called Data Aggregation Approach.

Each approach offers a different set of inputs to the Feedfoward Neural Network (FFN). These inputs are values from combination of technical data, fundamental data, and derived data of a company. These data are situated in one-day time-window. In the training stage, a target output will be provided and network weight will be suited based on the different between the computed and the expected output. Then, using the back-propagation algorithm input values are processed through the network to generate a particular predicted output.

Standard Approach

The standard approach is using commonly used technical indicator derived from the daily price of certain stock. The indicators are returns (R), and moving average (MA). The role of indicator MA is to compare the relationship between a moving average of the stock's price with the stock's price itself. According to Achelis (2000), a buy signal is generated when the stock's price rises above its moving average and a sell signal is generated when the stock's price falls below its moving average.

In this approach the input are P_t, MA_5, and MA_{10}. The output is P_{t+1}. Here P_t is the close price of t*th* period, MA_i is the moving average after i*th* period, and P_{t+1} is the next day close price. The indicators are defined as follows:

$$MA = \sum_{1}^{n} \frac{closing\ price}{n}, \qquad (3.1)$$

where
n = the number of time periods in the moving average.

Volatility Data Approach

The volatility data approach uses technical indicators, like in the standard approach, and yesterday volatility indicator as the inputs to the FFN. So, the inputs are P_t, MA_5, MA_{10}, and VI_{t-1}. The output is P_{t+1}. Here VI_t is the volatility indicator of t*th* period . The volatility indicator, *VI,* is calculated based on the four fiqures for daily prices, which are open, high, low, and close. The indicator will be positive for prices that rise and negative for falls. According to Azoff (1994), volatility indicator is a useful signal of the psychological state of investors at certain time. It is a straightforward monitor of the extent of the market volatility, defined as follows:

$$VI = \frac{(P_H - P_L)(P_C - P_O)}{|\,average\ of\ numerator\,|} \tag{3.2}$$

where
P_H = highest traded price during the day
P_L = lowest traded price during the day
P_C = close price
P_O = open price

Sector-Counter Approach

The approach is based on the idea that the individual stock is influenced by the index of its sector. The FFN is trained to react with increasing or decreasing of the sector index of the stock which being trained. The rise and fall of the counter will be predicted, mainly based on the sector index. Therefore, the input of the FNN are P_t, MA_5, MA_{10}, VI_{t-1} and P_s. The output is P_{t+1}. P_s is the closing price of sector index in which the counter belongs to. In the case of TNB counter, its sector index is the Trade and Services index.

Technical and Fundamental Data Approach

This approach incorporates technical and fundamental data into the FFN. The similar approach had been used by Refenes *et al.* (1994), and Ormoneit & Neuneier (1996) in Paris Stock Exchange and Deutscher Aktienindex (DAX) respectively. They found that combination of technical and fundamental data in their neural networks improves their stock price prediction.

The input data in the sector-counter approach will be used again here. Then, we have included two general economy indicators, which are real interest rate and inflation rate, as the fundamental data. Both are annual indicators in percentage. Hence, the input of the FFN are P_t, MA_5, MA_{10}, VI_{t-1}, P_s, real interest, and inflation. Like other approach, the output is also P_{t+1}.

Data Aggregation Approach

The purpose of this approach is to deal with weekend-effect in stock market. In the data aggregation approach we restrict the model to train using one-week data, Monday to the next, to predict the next week. In this approach, the network will learn the patterns that associated with each day of the week. The input of the FNN are P_t, MA_5, MA_{10}, VI_{t-1} and P_s. The output is P_{t+1}. For training and prediction, we have selected randomly 10 networks as Table 1.

Table 1: Training and Prediction Period for Aggregation Data Approach.

No	Training Start	Training End	Prediction Start	Prediction End
1	11/10/93	15/10/93	18/10/93	22/10/93
2	6/12/93	10/12/93	13/12/93	17/12/93
3	4/4/94	8/4/94	11/4/94	15/4/94
4	11/7/94	15/7/94	18/7/94	22/7/94
5	7/11/94	11/11/94	14/11/94	18/11/94
6	13/2/95	17/2/95	20/2/95	24/2/95
7	15/1/96	19/1/96	22/1/96	26/1/96
8	2/6/97	6/6/97	9/6/97	13/6/97
9	11/8/97	15/8/97	18/8/97	22/8/97
10	17/8/98	21/8/98	24/8/98	28/8/98

NEURAL NETWORKS INPUT NORMALIZATION

The input data in stock market require normalizing to standardising the possible numerical range that the input vector elements can take. We normalize the inputs, x, using the following function taken from Callan (1999):

$$x' = \frac{x_{\max} - x_{\min}}{r_{\max} - r_{\min}} \times x + \left(x_{\max} - \frac{x_{\max} - x_{\min}}{-r_{\min}} \times r_{\max} \right), \qquad (4.1)$$

where x_{max} is the highest value in the inputs, x_{min} is the lowest value in the inputs, r_{max} is the maximum range, and r_{min} is the minimum range.

NEURAL NETWORKS SETUP

We develop a neural network program as a dynamic link library (DLL) using Visual Basic. Using the DLL we create feed forward, layered and fully connected networks. The training algorithm is the standard backpropagation with momentum term. The value of momentum and learning rate are 0.9, and 0.3 respectively. The activation function is the symmetrical sigmoid, $S(x)$ with a [-1,1] range that is defined as follows:

$$S(x) = \frac{2}{1+e^{-x}} - 1 = \frac{1-e^{-x}}{1+e^{-x}}. \tag{5.1}$$

The cost function is the absolute function which is defined by

$$E = |T - O| \qquad \frac{\partial E}{\partial O} = -sgn(T - O). \tag{5.2}$$

All simulations run on Intel Pentium III – 433 personal computer. Convergence is reached, depending on the network architechture and amount of training data, in 5,000 to 10,000 iterations, requiring accordingly half to one day of CPU time.

NEURAL NETWORKS PERFORMANCE MATRICS

To measure the convergence and generalization performance of the network we use *Root Mean Square* (RMS) Error and *Normalized Mean Squared Error* (NMSE) which are calculated from the common *Mean Squared Error* (MSE). MSE, RMS, and NMSE are defined as follows:

$$MSE = \frac{1}{N}\sum_{\mu=1}^{N}(O^{\mu} - T^{\mu})^2, \tag{6.1}$$

$$RMS = \sqrt{MSE}, \tag{6.2}$$

$$NMSE = \frac{MSE}{\frac{1}{N}\sum_{\mu=1}^{N}(O^{\mu} - P^{\mu})^2}, \tag{6.3}$$

where O is the produced output and T is the expected output, P is mean of T, and N is the total number of patterns.

Other evaluation measure include the calculation of the correctness of Prediction of Change in Direction, or POCID for short.

$$POCID = 100\frac{\sum_{\mu=1}^{N} D_{\mu}}{N}, \qquad (6.4)$$

where

$$D_{\mu} = \begin{cases} 1 & if (T_t - T_{t-1}) \cdot (O_t - O_{t-1}) < 0 \\ 0 & otherwise \end{cases}$$

NEURAL NETWORKS PREDICTION RESULTS

Standard Approach

The result that we obtained after training the network on the standard approach can be seen in the Table 2. The first column indicates the network number. The second column shows amount of samples in each training set. There are 7 training sets that start from 1 month until 5 years. The third column represents the length of training process in term of epochs and time. The next columns give the mean squared error (MSE), root mean squared error (RMS), normalized mean squared error (NMSE), and percentage of correct predicted direction (POCID) for each network.

The values obtained by training the network, which has 3-2-1 architecture, with the standard approach show that network 7 produces the lowest value in almost all error measures. The best MSE, RMS, NMSE, and POCID values of the approach are 0.069546779, 0.263717233, 0.02403779, and 70.73% consecutively. The actual stock prices and predicted stock prices that attained from the training can be seen in Figure 1.

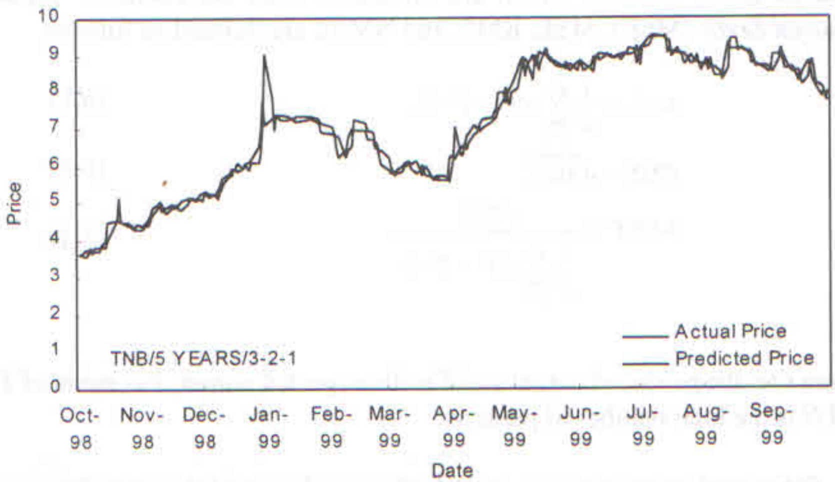

Figure 1: The actual and the predicted stock prices of the standard approach

The result also shows that the average MSE, RMS, and NMSE decrease when the amount of samples in each training set increase. This indicates that the prediction improves when more price patterns learned by the network. Although the best NMSE value of this approach, which is 0.02403779, is quite low, the POCID still performs poor. Hopefully the best network using volatility data approach, discussed on next section, improves the prediction.

Volatility Data Approach

The result that we obtained after training the network on the volatility data approach can be seen in the Table 3. The format of the result table is similar as the standard approach's. But here, the quantity of the input is 4. Therefore, the chosen architecture of the network is 4-2-1.

The values obtained by training the network with the volatility approach show that network 7 produces the lowest value of MSE, RMS, and NMSE. The best MSE, RMS, and NMSE values of this approach are 0.043644270, 0.208912111, and 0.01508498 consecutively. It also produces the highest POCID value, which is 77.64%. The actual stock prices and predicted stock prices that attained from the training can be seen in Figure 2.

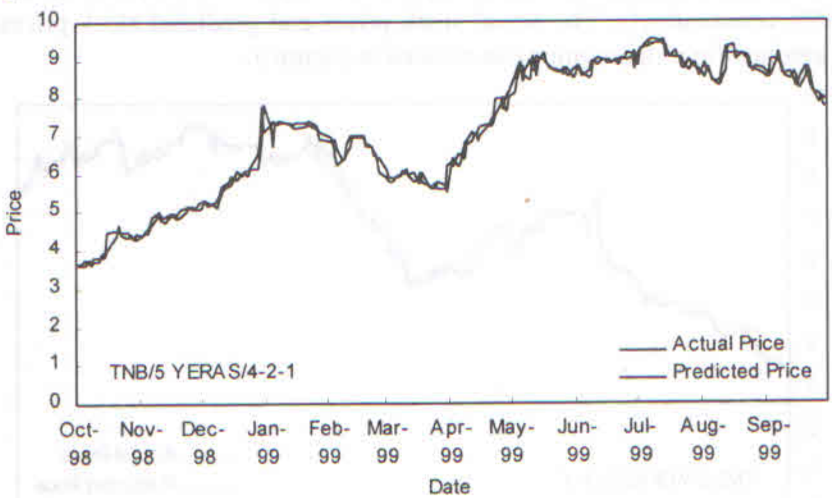

Figure 2 : The actual and predicted stock prices of the volatility data approach.

The sector-counter data approach produces slightly better results in all error measures. The best MSE, RMS, and NMSE values of this approach are lower 0.73%, 0.37%, and 0.49% consecutively than the previous

The volatility data approach produces greatly better results. The best MSE, RMS, and NMSE of this approach are lower 37.24%, 20.78%, and 37.24% consecutively than the standard approach. The best POCID error measure increases 6.91% from 70.73 to 77.64%. These results show that yesterday volatility data certainly influence the movement of the next day prices and improve the prediction in term of all error measures. In next section, we will discuss sector-counter approach that might produce better prediction results.

Sector-Counter Approach

The result that we obtained after training the network on the sector-counter approach can be seen in the Table 4. The format of the result table is similar as the standard approach's. Here, the amount of the input is 5. Therefore, the selected architecture of the network turns to 5-2-1.

The values obtained by training the network with this approach show that network 7 produces the lowest value of MSE, RMS, and NMSE. It also produces the highest POCID value. The best MSE, RMS, NMSE and POCID values of the approach are 0.043324166, 0.208144580, 0.01497434, and 78.05% consecutively. The actual stock prices and predicted stock prices that attained from the training can be seen in Figure 3.

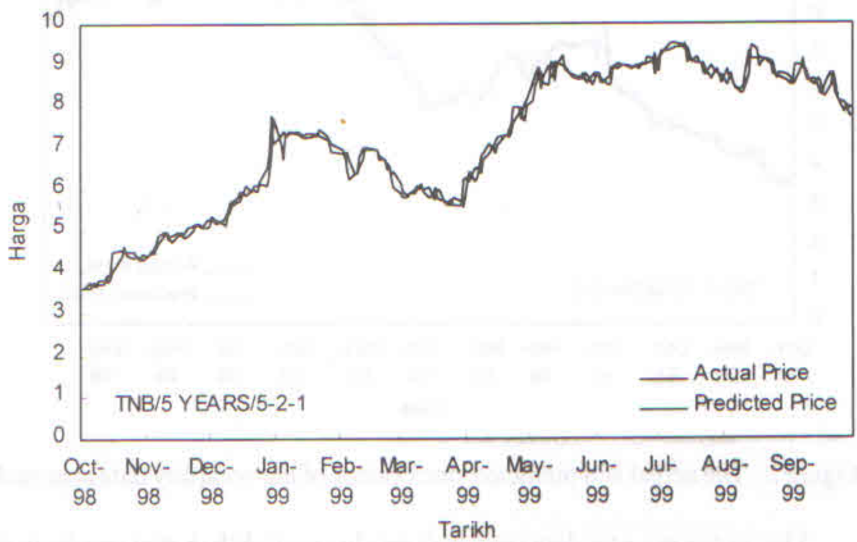

Figure 3 : The actual and predicted stock prices of the sector-counter approach.

approach's. The best POCID error measure however increases 0.41% from 77.64% to 78.05%. These results indicate that sector index of a stock certainly influence its price movements. Altogether this approach somehow improves the prices prediction.

Technical and Fundamental Data Approach

The result that we obtained after training the network on the technical & fundamental data approach can be seen in the Table 5. The format of the result table is similar as the standard approach's. But here, the sum of the input is 7. Therefore, the architectures of the network become 7-3-1.

The values obtained by training the network with the volatility approach show that network 4 produces the lowest value of MSE, RMS, and NMSE. The best MSE, RMS, and NMSE values of this approach are 0.052199525, 0.228472154, and 0.01804198 consecutively. The highest POCID value is 69.92%. The actual stock prices and predicted stock prices that attained from the training can be seen in Figure 4.

The technical and fundamental data approach does not improve results that we have obtained from sector-counter approach. All error measures perform poor. Besides that, the overall results of this approach are weeker than standard, and volatility data approach. The best MSE, RMS, and NMSE values of this approach are higher 20.49%, 9.77%, and 20.49% consecutively than the sector-counter approach. The best POCID error measure decreases 8.13% from 78.05% to 69.92%. These results show that by adding some fundamental data into the input make the price prediction results worsen.

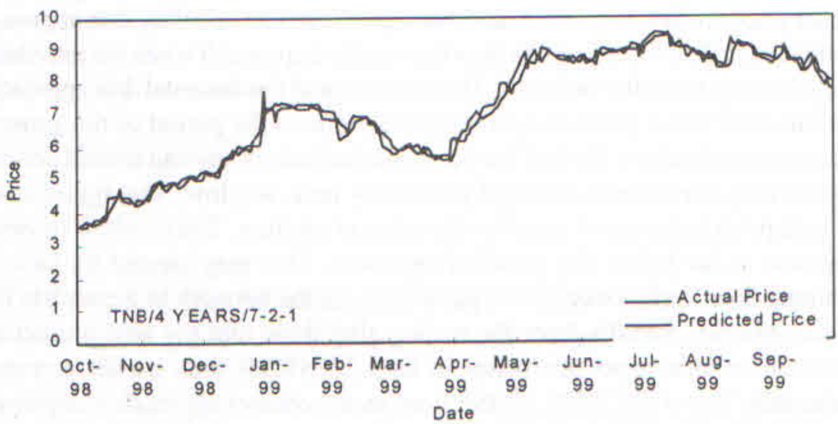

Figure 4 : The actual and predicted stock prices of the technical and fundamental data approach.

Data Aggregation Approach

The result that we obtained after training the network on the sector-counter approach can be seen in the Table 6. The first column indicates the network number. These networks are randomly selected, see Table 1. The second column shows the length of training process in term of epochs and time. The next columns give the mean squared error (MSE), root mean squared error (RMS), normalized mean squared error (NMSE), and percentage of correct predicted direction (POCID) for each network.

The lowest average values of MSE, RMS, and NMSE obtained by testing the 6-2-1 network with this approach are 0.239406741, 0.455385683, and 8.302880038 consecutively. While the best average POCID is 62.00%.

The data aggregation approach do not improve results that we have obtained from sector-counter approach and even worse than all approaches. The best average MSE, RMS, and NMSE values of this approach are higher 244.24%, 72.68%, and 34440.95% consecutively than the best MSE, RMS, and NMSE values in the standard approach. The best POCID error measure decreases 10.73% from 70.73% to 62.00%. The NMSE value is so high that makes the prediction prices totally unusable. These results show that by using the aggregating data approach to add weekend effect pattern into the network can't improve the price prediction.

CONCLUSION

The results from the testing have the following conclusions: Based on Table 7, the presented results show that the sector-counter approach produces better price prediction results than other approachs. The volatility data approach produces slightly better results than the standard approach when we introduce the yesterday volatility indicator. The technical and fundamental data approach, on the other hand, performs poor probably due to the period of the general economic indicators. Both of the economic indicators have an annual period, which the patterns are too lengthy for a day time-window. The aggregation data approach also can't improve the price prediction. The result of its error measure is far below the standard approach. This may caused by lack of training data. Only a week price patterns make the network to generalize the prices blindly. Results from the testing also show that the best prediction result from this paper are better in term of NMSE than earlier research especially Yao *et al.* (1998). NMSE from sector-counter approach is improved by 0.01423502 which is 44.1%. This evidence suggests that the sector index of a stock certainly influence its price movements.

REFERENCES

Achelis, Steven. 2000. *Technical Analysis From A to Z: Covers Every Trading Tool...From the Absolute Breadth Index to the Zig Zag*. 2ed. McGraw-Hill Professional Book Group.

Azoff, E. M. 1994. *Neural Network Time Series Forecasting of Financial Markets*. Chichester: John Wiley & Sons.

Barnes, P. 1986. Thin trading and Stock Market Efficiency: The Case of the Kuala Lumpur Stock Exchange. *Journal of Business and Accounting*. pp. 609-617.

Callan, R. 1999. The Essence of Neural Networks, Hertfordshire: Prentice Hall Europe.

Hellström, T. & Hellström, K. 1998. Predicting the Stock Market. *Technical Report Series IMa-TOM-1997-09*: pp. 6.

Laurance, M. M. 1986. Weak Form Efficiency in the Kuala Lumpur and Singapore Stock Exchange. *Journal of Economic*. **10**: 431-435.

McCann P. J.; & Kalman B. L. 1994. Parallel Training of Simple Recurrent Neural Networks, *Proceedings of the IEEE World Congress on Computational Intelligence(WCCI '94), Orlando, FL, June 1994*, **I**: pp. 167-170.

Md. Nasir, A.; & Mohamed, S. 1993. Stock Returns and the Weekend Effect: The Malaysian Experience, in Ibrahim, Ismail and Yong Othman (ed.), *Understanding the Behavioral Patterns of Stock Prices*, Kuala Lumpur: Leeds Publications.

Moody, J.; & J. Utans 1995. *Architecture Selection Strategies for Neural Networks: Application to Corporate Bond Rating Prediction*, in A.-P. Refenes (ed.), Neural Networks in the Capital Market, New York: John Wiley.

Nassir Lanjong. 1983. Efficiency of the Malaysian Stock Market and Risk-Return Relationship. Tesis Dr. Fal.Katholieke Universitet, Leuven, Belgium.

Neoh, S. K. 1986. An Examination of the Efficiency of the Malaysian Stock Market. Tesis Dr. Fal. Edinburgh.

Odom, M.D.; & Sharda, R. 1990. A Neural Network Model for Bankruptcy Prediction. *Proceedings of the IEEE International Joint Conference on Neural Networks (San Diego 1990)*, **II**, pp. 163-168. New York: IEEE.

Op't Landt, F. W. 1997. Stock Price Prediction Using Neural Networks, Master Thesis, Leiden University.

Ormoneit, D.; & Neuneier, R. 1996. Experiments in Predicting the German Stock Index DAX with Density Estimating Neural Networks. *Proceedings of the IEEE/IAFE 1996 Conference on Computational Intelligence for Financial Engineering (CIFEr)*, pp. 66 –71.

Refenes A. N.; Azema-Barac M.; Chen L.; & Karoussos, S. A. 1993. Currency Exchange Rate Prediction and Neural Network Design Strategies, *Neural Computing and Applications*, 1(**1**): pp. 46-58

Refenes, A. N.; Zapranis, A. D.; & Bentz, Y. 1994. Modelling Stock Returns With Neural Network. *Proceedings of 1995 Conference on Computational Intelligence for Financial Engineering (CIFEr)*. pp. 39-50.

Trippi, R.; & DeSieno, D. 1992. Trading Equity Indeks Futures with a Neural Network. *Journal of Portfolio Management,* (Fall 1992), pp. 27-33.

White, H. 1988. Economic Prediction Using Neural Networks: The Case of IBM Daily Stock Returns. *Proceedings of the IEEE International Conference on Neural Networks. (July 1988),* **II**, pp. 451-458.

Yao J.; Tan, C. L.; & Poh H. L. 1998. Neural Networks For Technical Analysis: A Study on KLCI. *International Journal of Theoritical and Applied Finance*, 2 (**2**): pp. 221 – 241.

Table 2: Standard Approach Training Results.
(Set: Length of Training Set, MSE: Mean Squared Error, RMS: Root Mean Squared error, NMSE: Normalized Mean Squared Error; POCID: Prediction of Change in Direction error)

No	Set	Training		Testing			
		Epoch	Time (s)	MSE	RMS	NMSE	POCID (%)
1	1 month	5000	25	0.477722666	0.691174845	0.16511761	57.72
2	6 months	2314	56	0.077739583	0.278818190	0.02686951	55.28
3	1 year	1210	50	0.082843201	0.287824949	0.02863350	60.57
4	2 years	618	50	0.058489120	0.241845239	0.02021588	72.36
5	3 years	408	49	0.059666103	0.244266458	0.02062269	67.48
6	4 years	358	57	0.061558304	0.248109461	0.02127670	71.54
7	5 years	260	53	**0.069546779**	**0.263717233**	**0.02403779**	**70.73**

Table 3: Volatility Data Approach Training Results.
(Set: Length of Training Set, MSE: Mean Squared Error, RMS: Root Mean Squared error, NMSE: Normalized Mean Squared Error; POCID: Prediction of Change in Direction error)

No	Set	Training		Testing			
		Epoch	Time (s)	MSE	RMS	NMSE	POCID (%)
1	1 month	5000	49	0.550754651	0.742128460	0.19036001	50.41
2	6 months	2150	72	0.120518837	0.347158230	0.04165551	55.28
3	1 year	1206	135	0.069707094	0.264021011	0.02409320	69.51
4	2 years	649	72	0.049099664	0.221584440	0.01697056	74.39
5	3 years	472	63	0.043948153	0.209638147	0.01519001	75.61
6	4 years	347	64	0.044781455	0.211616293	0.01547803	75.20
7	5 years	293	69	**0.043644270**	**0.208912111**	**0.01508498**	**77.64**

Table 3: Volatility Data Approach Training Results.
(Set: Length of Training Set, MSE: Mean Squared Error, RMS: Root Mean Squared error, NMSE: Normalized Mean Squared Error; POCID: Prediction of Change in Direction error)

No	Set	Training		Testing			
		Epoch	Time (s)	MSE	RMS	NMSE	POCID (%)
1	1 month	5000	60	0.489462351	0.699615860	0.16917525	51.22
2	6 months	2164	59	0.156684056	0.395833368	0.05415547	54.47
3	1 year	1228	62	0.076650856	0.276858910	0.02649321	62.60
4	2 years	645	123	0.051035372	0.225910097	0.01763960	69.92
5	3 years	446	115	0.045594321	0.213528268	0.01575899	73.98
6	4 years	346	83	0.045671501	0.213708917	0.01578566	74.80
7	5 years	280	78	**0.043324166**	**0.208144580**	**0.01497434**	**78.05**

Table 5: Technical and Fundamental Data Approach Training Results. (Set: Length of Training Set, MSE: Mean Squared Error, RMS: Root Mean Squared error, NMSE: Normalized Mean Squared Error; POCID: Prediction of Change in Direction error)

No	Set	Training		Testing			
		Epoch	Time (s)	MSE	RMS	NMSE	POCID (%)
1	1 year	1248	122	0.581833409	0.762780053	0.20110191	55.69
2	2 years	635	110	0.564066404	0.751043543	0.19496102	54.88
3	3 years	430	108	0.089047018	0.298407469	0.03077775	60.98
4	4 years	341	114	**0.052199525**	**0.228472154**	**0.01804198**	**69.92**
5	5 years	260	111	0.082901692	0.287926539	0.02865372	61.79

Table 6: Aggregation Data Approach Training Results. (Σ: Average, MSE: Mean Squared Error, RMS: Root Mean Squared error, NMSE: Normalized Mean Squared Error, POCID: Prediction of Change in Direction error)

No	Training		Testing			
	Epoch	Time (s)	MSE	RMS	NMSE	POCID (%)
1	5000	36	0.492907656	0.702073825	10.2689095	60.00
2	5000	26	0.125865003	0.354774580	3.34747348	80.00
3	5000	23	0.406791938	0.637802429	9.59414948	80.00
4	5000	29	0.140514720	0.374852931	10.3319647	80.00
5	5000	33	0.061237175	0.247461461	5.88818986	60.00
6	5000	31	0.134835935	0.367200130	4.71454319	60.00
7	5000	30	0.072545358	0.269342455	4.12189534	60.00
8	5000	30	0.490534221	0.700381482	7.96321787	40.00
9	5000	23	0.395238430	0.628679910	9.10687626	60.00
10	5000	22	0.073596976	0.271287626	17.6915807	40.00
Σ			0.239406741	0.455385683	8.302880038	62.00

Table 7: Technical and Fundamental Data Approach Training Results. (Set: Length of Training Set, MSE: Mean Squared Error, RMS: Root Mean Squared error, NMSE: Normalized Mean Squared Error; POCID: Prediction of Change in Direction error)

Approach	MSE	RMS	NMSE	POCID (%)
Standard	0.069546779	0.263717233	0.02403779	70.73
Volatility Data	0.043644270	0.208912111	0.01508498	77.64
Sector-Counter	**0.043324166**	**0.208144580**	**0.01497434**	**78.05**
Technical and Fundamental Data	0.052199525	0.228472154	0.01804198	69.92
Aggregation Data	0.125865003	0.354774580	3.34747348	80.00