

STOCK PRICE FORECASTING BASED ON BACK PROPAGATION NEURAL NETWORK AND MARKOV CHAIN

Azme Bin Khamis & Li Chee Guan

Department of Mathematics and Statistics
Faculty of Applied Science and Technology
University Tun Hussein Onn Malaysia

DOI: 10.31364/SCIRJ/v8.i7.2020.P0720787
<http://dx.doi.org/10.31364/SCIRJ/v8.i7.2020.P0720787>

Abstract: This study focused on the stock price prediction of an oil and gas exploration company in Malaysia. An artificial neural network that trained by back propagation algorithm or back propagation neural network (BPNN) as well as a hybrid model that combining back propagation neural network and Markov chain are used in this study. The main objective of this study is to compare the forecasting performance of the hybrid model and the BPNN model. The sample data is taken from 1st October 2018 to 30th September 2019. The main interest in this study is daily closing price, and the attributes are opening price, highest price, lowest price, trading volume, Brent crude oil price and OPEC oil price. The hybrid model is built by initialized the forecasting using back propagation neural network, followed by Markov state division, transition probability matrix formulation and prediction computation. Accordingly, both predictive models are compared by Mean Absolute Percentage Error (MAPE) and coefficient of determination R². As a result, back propagation neural network is better than hybrid model because it showed a higher level of forecast accuracy with 0.5057% of MAPE and 95.94% of R².

Keywords: Artificial Neural Network, Markov chain, Back Propagation Algorithm, Forecast Accuracy

1.0 Introduction

Stock is an asset which represents the ownership of a company. It is the main source for companies to raise fund from public for the business expansion in which by selling company's shares through centralized exchange (Esbond & Fumilayo, 2015). On the other hands, investor can be a part of owner of the particular company in order to make profit without taking high risk to enter their own businesses with high overheads and start-up costs. Basically, investors favour to purchase shares of prospective companies in order to get return on their investment. Nevertheless, investor will lose money in trading environment if they hold stocks from company that losing value and sell them when the value is low.

The problem of investors is the ability to predict with some level of precision regarding the direction of price movement and the most likely price value of the stock itself. By having a good prediction skill, it can help to generate significance return from stock trading. However, single method would be limited to achieve an ideal precision level due to the complicated influencing factors in stock market. Some new hybrid approaches have showed their potential superiorities in stock market forecasting especially the approach based on adaptive modelling and conditional probability transfer (Dai, et. al, 2014).

In addition, artificial neural network can effectively cover the non-linear relationship that exists in financial market and it is superior in predicting stock market return if compared to regression or the passive buy-and-hold strategy (Qui, et. al. 2016). As the stochastic process followed by stock price, Markov chain is widely used in stock market prediction either for a group of stock as well as for a single stock (Bhusal, 2017). Therefore, a hybrid models that combining artificial neural network with back propagation algorithm or back propagation neural network (BPNN) and Markov chain is suggested. This study compared the forecast accuracy between back propagation neural network and the hybrid model in order to select the best model to perform stock price forecasting.

Stock price of an oil and gas exploration company in Malaysia, from 1st October 2018 to 30th September 2019 are taken from Investing.com. Closing price is the main interest in this study while opening price, lowest price, highest price, and trading volume are independent variables. Brent crude oil price and OPEC oil price which act as the benchmark of global crude oil price are taken as independent variables as well. All the data is recorded in day. And a total of 240 data are collected. The data is assigned into training set and test set by a ratio of 9 to 1.

2. Methodology

2.1 Multi-Layer Perceptron (MLP) Feed Forward Network

MLP constitutes of three layers: input layer, hidden layer and output layer. The signal is transmitted in one direction, from the input layer to the hidden layer and eventually to the output layer. The output layer generalizes the overall responses of the network to the activation pattern supplied by the source nodes. In a neuron model, there are three basic elements: connecting link, adder and activation function. Connecting link is regarded as weight. Specifically, signal x_i at the input layer connected to neuron j at the hidden layer is multiplied by the weight w_{ij} . Meanwhile, adder total up all the input signals that weighted by the respective connecting links. Activation function limits the amplitude of the output of a neuron in the interval of $[0, 1]$ or alternatively $[-1, 1]$.

2.1.1 Back Propagation Learning

Back propagation neural network is illustrated in Figure 2.1. The weight of each neuron is adjusted during the back propagation learning process. After the feed forward propagation, output will be obtained. The difference between the output and the desired value yield error. Chain rule is used to compute the derivatives of error with respect to the weight of the output layer. These derivatives are known as gradients. Subsequently, these gradients are used to calculate the gradients of the hidden layer. This process is repeated until gradients for each neuron in the network are obtained and the overall network error $e_k(n)$ is fall below the predefined value. In short, error back propagation is used to modify the weights in order to minimize the output error.

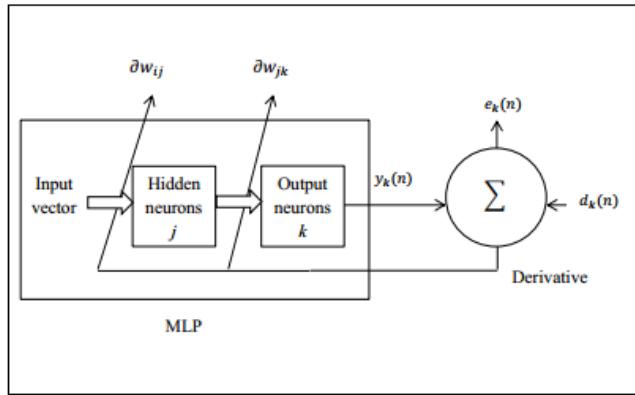


Figure 2.1: Back propagation neural network

2.1.2 Parameter Determination

Initially, data is normalized into the range $[0, 1]$ using min-max normalization. Some parameters need to be identified before executing the training algorithm such as number of hidden layer neurons (n_j), momentum constant (mc), learning rate (lr) and number of iterations for learning or epoch (ep). These parameter values are set by trial and error. Meanwhile, the optimum number used in the study (Inthachot, et. al. 2016) is used as a guideline in determining the parameter values. The researcher getting an ideal level of accuracy with the parameters value as $n_j = 100$, $mc = 0.1$, $lr = 0.1$, and $ep = 8,000$. Apart from that, the network is trained continually until the relative change in training error is less than 0.0001.

2.2 Markov Chain

Discrete-time Markov chain is a collection of random variable which represents by X_t where $t = 0, 1, 2, \dots, n$. It is a stochastic process that possesses Markov property in which the future is conditionally independent of the past given the present (Li & Song, 2015). It can be written as:

$$P(X_t = j | X_0 = i_0, X_1 = i_1, \dots, X_{t-1} = i_{t-1}) = P(X_t = j | X_{t-1} i_{t-1}) \quad (1)$$

p_{ij} is the transition probability moving from state i to state j . The states of Markov chain are introduced in rows and column in a matrix, therefore the transition probability matrix is a square matrix and the sum of row is equal to one (Adesokan, et. al., 2017). The transition probability matrix can be represented by $P = (p_{ij})$ and the n step transition probability can be described as below:

$$P(n) = P(0) \times P^n = P(0) \times \begin{bmatrix} p_{11} & \cdots & p_{1n} \\ \vdots & \ddots & \vdots \\ p_{n1} & \cdots & p_{nn} \end{bmatrix}^n \quad (2)$$

From the transition probability matrix, p_{ij} can be computed by

$$p_{ij} = \frac{n_{ij}}{\sum_i n_{ij}} \quad (3)$$

where n_{ij} is the number of transition from state i to state j .

2.3 Forecast Based on Back Propagation Neural Network and Markov Chain

Back propagation neural network is used to initialize the forecasting. The actual stock's closing prices, x_a and the predicted stock's closing prices, x_p are used to compute the relative error of prediction in day ε_i by using the Equation (4).

$$\varepsilon_i = \left| \frac{x_a - x_p}{x_a} \right| \quad (4)$$

In order to generate forecast using BPNN-Markov chain model, the relative error of testing sample is normalized. Based on the normalized values, sample average-mean square deviation method is used to classify into a five state Markov chain which represented by $(-\infty, \bar{x} - \beta_1 s)$, $(\bar{x} - \beta_1 s, \bar{x} - \beta_2 s)$, $(\bar{x} - \beta_2 s, \bar{x} - \beta_3 s)$, $(\bar{x} - \beta_3 s, \bar{x} - \beta_4 s)$ and $(\bar{x} - \beta_4 s, +\infty)$ where \bar{x} is refers to the mean and s refers to the sample standard deviation. According to [2] β_1 and β_4 are usually in the range from 1.0 to 1.5 while β_2 and β_3 are usually in the range from 0.3 to 0.6. Subsequently, the transition probability matrix is constructed using the number of Markov state transition by refer to the Equation (3). After that, the n step state vector is built using the Equation (2). Yet, the n step state vector yields probability of the movement of stock price from one state to another state for a certain days, the predicted stock price need to be computed by taking the average of the interval from Markov prediction interval that having the maximum probability in n step state vector.

2.4 Forecast Accuracy Measure

Forecasting accuracy is evaluated by mean absolute percentage error (MAPE) which computed by using Equation (5). Smaller MAPE indicates a better precision level of forecast accuracy thus infers a more appropriate model in stock price forecasting.

$$MAPE = \frac{1}{N} \sum_{i=1}^N \left| \frac{Y_i - \hat{Y}_i}{Y_i} \right| \times 100 \quad (5)$$

Another tool is coefficient of determination R^2 which computed by using Equation (6). It illustrates the percentage of variability of dependent variable that can be explained by the independent variable. Higher value of R^2 infers a better forecasting performance since more uncertainty is explained by the mode [8].

$$R^2 = 1 - \frac{\sum_{i=1}^N (Y_i - \hat{Y}_i)^2}{\sum_{i=1}^N (Y_i - \bar{Y}_i)^2} \quad (6)$$

where Y_i represents actual value; \hat{Y}_i represents predicted value and \bar{Y}_i represents the mean of the actual values.

3. Results and Discussions

3.1 Modeling of Back Propagation Neural Network

One hidden layer is used to avoid unnecessary computation complexity. Hence, three-layer neural network with input layer, hidden layer and output layer is built.

The activation functions used in both hidden and output layers are sigmoid function. The stopping rules used to terminate the training process is when the relative change in training error falls under 0.0001. The maximum number of circulation or epoch is set at 10,000 times. Initial learning rate and initial momentum factor are defined at the optimal solution by depending on the experience and repeated training. Initial learning rate and initial momentum factor are substituted by the values from 0.1 to 1.0. An optimal solution is found when initial learning rate is 0.7 and initial momentum factor is 0.4 by comparing the respective sum of square error (SSE) and mean absolute percentage error (MAPE) of the models in both training and testing set.

The number of nodes in the input layer is six which represented by opening price, highest price, lowest price, trading volume, Brent crude oil price and OPEC price. Data on 1st October 2018 is the first input data in the input layer. The number of nodes at output layer is one which represented by the closing prices. The first output data is the closing prices on 1st October 2018. The number of nodes in the hidden layer is four which is determined automatically by the estimation algorithm.

Before the data is applied into the neural network, the data is normalized to restrict its interval from 0 to 1. After that, the network is trained using training set until the relative change falls under 0.0001. The network is trained by updating the synaptic weights after every single training data record. It continuously recorded and updated the weights until the stopping rule is met. Furthermore, gradient descent is selected to use as an optimization algorithm to estimate the synaptic weight of neurons. At the end of training, an ideal model is obtained.

3.2 Forecasting using Back Propagation Neural Network

Figure 3.1 showed the trend of the predicted price and the actual price. The disparity between two lines is very small where the lines are seen to be overlapped. The trend of predicted price is almost similar to the actual price. A decreasing trend is showed before December 2018 and after that, it bounced and fluctuated around RM1.



Figure 3.1: Stock closing price forecasting using BPNN

3.3 Hybridization of BPNN with Markov chain

Relative error on respective predicted days is computed and normalized to develop Markov state. Since the data is limited, Markov state is divided into four states which are state 1: (0 - 0.1749), state 2: (0.1749 – 0.3819), state 3: (0.1819 – 0.5683) and state 4: (0.5683 - 1). As the Markov state transition is built as shown in Table 3.1, a transition probability matrix is formed by divided the number in cell to the sum of the row.

Subsequently, the step state vector of prediction is calculated using equation (2) where $P(0) = [1,0,0,0]$. $P(0)$ is the transition from 216th day to 217th day since 217th day is the first day of prediction. Step referred as the day ahead. For example, step 1 means one day ahead or tomorrow. Thus, the step state vector of 24 trading days is described as shown in Table 3.2. According to the table, started from step 6, the matrix has developed into a steady state where the probability is remained constant and maintains forever. The probability is [0.384615 0.395604 0.181319 0.038462]. This means that in the far future, the probability would not be changing much from one transition to the next. Accordingly, the predicted price is computed from the average prediction interval of the day that having the maximum probability in step state vector.

Table 3.1: Markov state transition

State	1	2	3	4	Total
1	4	3	2	1	10
2	3	5	1	0	9
3	1	1	1	0	3
4	1	0	0	0	1
Total	9	9	4	1	23

Table 3.2: Probability of step state vector

State	Step 1	Step 2	Step 3	Step 4	Step 5	Step 6
1	1	0.4	0.4266	0.3876	0.3846	0.3846
2	0	0.3	0.3533	0.3915	0.3955	0.3956
3	0	0.2	0.1800	0.1819	0.1813	0.1813
4	0	0.1	0.0400	0.0388	0.0384	0.0384
State	Step 7	Step 8	Step 9	...	Step 23	Step 24
1	0.3846	0.3846	0.3846		0.3846	0.3846
2	0.3956	0.3956	0.3956	...	0.3956	0.3956
3	0.1813	0.1813	0.1813		0.1813	0.1813
4	0.0384	0.0384	0.0384		0.0384	0.0384

3.4 Comparison between BPNN and BPNN-Markov Chain Model

The result from BPNN is better than hybrid model according to the MAPE and coefficient of determination R^2 which is shown in Table 3.2. BPNN showed 0.5057% of MAPE which is almost 0.07% lower than BPNN-Markov chain and 95.94% of R^2 which is almost 2% higher than BPNN-Markov chain.

Table 3.2: Comparison of model based on MAPE and R^2

Model	MAPE	R^2
BPNN	0.5057%	95.94%
BPNN-MC	0.5734%	94.22%

2.0 Conclusions

The importance in this study is the high level of forecast accuracy in order to decrease the risk of deficit and increase the chance of profit for every transaction to be issued on the floor of the stock market. BPNN is outperformed. It is an ideal model to use to perform stock price forecasting.

3.0 References

- Adesokan, I., Philip. N. and Abdulhakeem, K. (2017). Analyzing Expected Returns of a Stock Using The Markov Chain Model and the Capital Asset Pricing Model.
- Bhusal, M. K. (2017). Application of Markov Chain Model in the Stock Market Trend Analysis of Nepal. *International Journal of Scientific & Engineering Research* 8, pp. 1733-1745.
- Dai, Y., Dongmei, H and Weihui. D. (2014). Modeling and computing of stock index forecasting based on neural network and Markov chain. *The Scientific World Journal*.
- Esbond, G. I and Fumilayo W. O. S. (2015). Introducing the Use of Markov Chain Model for Informed Decision Making for Investors in a Stock Market Environment. *International Journal of Science and Research*, 6(10), pp. 1013-1019.
- Inthachot, M., Veera, B. and Sarun, I. (2016). Artificial neural network and genetic algorithm hybrid intelligence for predicting thai stock price index trend. *Computational intelligence and neuroscience*.
- Li, X and Jingdong, S. (2015). A new ANN-markov chain methodology for water quality prediction." In *International Joint Conference on Neural Networks (IJCNN)*, pp. 1-6. IEEE.
- Qiu, M., Yu S. and Fumio, A. (2016). Application of artificial neural network for the prediction of stock market returns: The case of the Japanese stock market. *Chaos, Solitons & Fractals* 85, pp. 1-7.