Stock Market Prediction Using Artificial Neural Networks

Bing Yang¹, a, Hao Jiankun¹, b and Zhang Sichang¹, c

¹ School of Mathematics and Statistics, Shandong University at Weihai, Weihai 264209, China
a bingyang@sdu.edu.cn, b haojiankun1209@163.com, c zhangsch125@163.com

Keywords: Learning Algorithm, Gradient Search Technique, Neural Networks

Abstract. In this study we apply back propagation Neural Network models to predict the daily Shanghai Stock Exchange Composite Index. The learning algorithm and gradient search technique are constructed in the models. We evaluate the prediction models and conclude that the Shanghai Stock Exchange Composite Index is predictable in the short term. Empirical study shows that the Neural Network models is successfully applied to predict the daily highest, lowest, and closing value of the Shanghai Stock Exchange Composite Index, but it can not predict the return rate of the Shanghai Stock Exchange Composite Index in short terms.

Introduction

Within the last two decades, there have been many attempts to predict stock prices and price fluctuations with neural networks. [1-5, 7-9]. Financial market is a highly complicate nonlinear system. It is not only variation that has its own regulation, but also influenced by many other factors such as politics, economic situation and investors’ psychology. Neural network enjoys the virtue of self-organization and adaptability and can learn economical knowledge from historical datasets, so the neural network is suitable implement to predict stock prices and price fluctuations.

According to the efficient market hypothesis, in the financial market, stock prices rapidly adjust to new information once it becomes public, making the prediction of stock market’s movements impossible [2]. This judgment is correct for traditional studies using multiple linear regression analysis. However, stock prices will become predictable if the dynamic and non-linear relationships in stock markets are shown. This is the motivation for applying neural networks to financial time series analysis.

Until now, application of neural networks to predict the movements of financial markets has attracted many researchers' attention. White [13] proposed neural network modeling and learning techniques to search for and decode nonlinear regularities of asset price movements and predicted IBM daily stock prices, Chenoweth [4] studied the trading system based on the future values of daily S&P 500 index. Zhang [15] carried out various analyses in four kinds Stock including Lujiazui, Fuhua Industrial, Changchun Hualian, Shanghai Petrochemical using BP network with special input data. Wang [12] introduced the sliding window technique and RBF network into non-linear time series, the result was good. Some of these researches merely used the past values of the stock index as the input of neural networks so as to obtain forecast, while others made use of additional financial factors as inputs.

In this study we apply back propagation Neural Network models to predict the daily Shanghai Stock Exchange Composite Index. The learning algorithm and gradient search technique are constructed in the models. We evaluate the prediction models and conclude that the Shanghai Stock Exchange Composite Index is predictable in the short term. Empirical study shows that the Neural Network models is successfully applied to predict the daily highest, lowest, and closing value of the Shanghai Stock Exchange Composite Index, but it can not predict the return rate of the Shanghai Stock Exchange Composite Index in short terms.

This paper is organized as follows. In the next section, we introduce the model of neural networks. Section 3 gives the data description and analysis. In section 4, we evaluate the prediction models. Section 5 summarizes the conclusions.
Models of Neural Networks

In this section we describe general features of applications of the Neural Networks to data analysis. The Neural Network is not a standard tool for statistical analysis, which can be regard as an algorithm for Artificial Intelligence and Machine Learning.

The Neural Network has different styles as connection styles are various. Our study chooses Back Propagation Neural Network which uses error back propagation as weight training, to train and test the sample data.

The BP Neural Network is a kind of one-way transmission of multilayer feed forward neural network, with one or more layers of hidden nodes, besides input and output nodes in its structure. There is no connection between nodes on the same level. We can treat it as a highly nonlinear mapping from input to output. Our model uses learning algorithm, gradient search techniques in the learning process and the error back propagation to modify weights, to achieve the minimum of the output error. The following diagram shows a usual BP neural network model with one hidden layer.

Fig. 1. Structure of BP Neural Networks

Where $O_1$ is output unit, $V_i$ is the hidden unit, $\xi_k$ is input unit, $w_{jk}$ is connection weight from the input unit $k$ to hidden unit $j$, $W_j$ is connection weight from the hidden unit $j$ to output unit $i$, $\omega = (W, w)$ are all the connections weight. $\mu$ means different input mode, and $p$ is the number input mode, where $\mu = 1, 2, \cdots, P$, $g_1$, $g_2$ correspond to the appropriate activation functions of hidden and output layer.

For a given input pattern $\mu$, the input to hidden unit $j$ is $h_j^\mu = \sum_{k=1}^{K} w_{jk} \xi_k^\mu$, and the output is

$$V_j^\mu = g_1(h_j^\mu) = g_1\left(\sum_{k=1}^{K} w_{jk} \xi_k^\mu\right).$$

The input to output unit $i$ is

$$H_i^\mu = \sum_{j=1}^{J} W_j V_j^\mu = \sum_{j=1}^{J} W_j g_1\left(\sum_{k=1}^{K} w_{jk} \xi_k^\mu\right),$$

and the final output is

$$O_i^\mu = g_2(H_i^\mu) = g_2\left[\sum_{j=1}^{J} W_j g_1\left(\sum_{k=1}^{K} w_{jk} \xi_k^\mu\right)\right].$$

Corresponding to any input mode $\mu$ and output unit $i$, the instantaneous error function is defined as

$$E_i^\mu(\omega) = \frac{1}{2} (\xi_i^\mu - O_i^\mu)^2 = \frac{1}{2} \{\xi_i^\mu - g_2\left[\sum_{j=1}^{J} W_j g_1\left(\sum_{k=1}^{K} w_{jk} \xi_k^\mu\right)\right]\}^2,$$

and total error function for the output unit $i$ is defined as

$$E(\omega) = \sum_{\mu=1}^{P} E_i^\mu(\omega) = \frac{1}{2} \sum_{\mu=1}^{P} (\xi_i^\mu - O_i^\mu)^2.$$
in whole or in part. The gradient descent algorithm can be carried out on the right to learn. For the connection weight matrix $W$ between hidden units and output units and the input mode $\mu$, we can use the rules of the gradient descent to obtain

$$\Delta W = -\eta \frac{\partial E^\mu}{\partial W} = \eta \delta^\mu V^\mu,$$

where $\eta > 0$ is the appropriate selection of learning step, and $\delta^\mu = g'(h^\mu)(\xi^\mu - O^\mu)$.

Similarly, for the connection weight matrix between hidden units and input unit $w$, we can use the chain rule to obtain

$$\Delta w_{jk} = -\eta \frac{\partial E^\mu}{\partial w_{jk}} = -\eta \frac{\partial E^\mu}{\partial V^\mu} \frac{\partial V^\mu}{\partial w_{jk}} = \eta \delta^\mu \xi^\mu,$$

where $\delta^\mu = g'(h^\mu) \sum_i W_{ij} \delta_i^\mu$.

Activation functions $g_1$, $g_2$ shall be single-valued function, to make neurons reversible. Commonly used functions are the Sigmoid function and the log-Sigmoid function, and the mathematical expressions are

- Sigmoid function: $f(x) = \frac{2}{1 + e^{-2x}} - 1 = \frac{1 - e^{-2x}}{1 + e^{-2x}}$,
- Log-Sigmoid function: $f(x) = \frac{1}{1 + e^{-x}}$.

**Data Description and Analysis**

Number section and subsection headings consecutively in Arabic numbers and type them in bold. Avoid using too many capital letters. If any further subdivision of a subsection is needed the titles should be 10 point and flushed left.

The data used in this study are obtained from the web site of Shanghai Stock Exchange. Our data covers the horizon from March 17, 2010 to April 28, 2010, including the daily prices of The Shanghai Stock Exchange Composite Index.

The main purpose is to predict the daily highest, lowest, closing, and opening prices of individual stock and the daily return. In order to make the information more comprehensive and more predictable, we choose following indicators as the input variables to neural network: for The Shanghai Stock Exchange Composite Index, the input include the daily opening, highest, lowest, and closing prices, trading volume, difference between the highest and lowest prices, and the daily return; and chose the input include the highest, lowest, opening and closing prices, trading volume, difference between the highest/lowest prices and return for one stock.

The excess return is calculated as: $\sqrt{(\pi/2)} |R_t - \mu_t|$, where $p_t$ is the closing price of the $t$ day, and $R_t$ is daily return of the Shanghai Stock Exchange Composite index.

The data should be normalized due to the different dimensions. The normalization method for each variable is to divide its maximum value. Thus, we obtain the normalized data within the limit of $[-1, 1]$.

Names of each variable after normalization at time $t$ are as following:

For the Shanghai Stock Exchange Composite index: highest price $H_t$, lowest price $L_t$, opening price $O_t$, closing price $C_t$, trading volume $V_t$, difference between the highest and lowest prices $H_t - L_t$, and return $R_t$.

Assuming

$$x_t = (O_t, H_t, L_t, C_t, V_t, H_t - L_t, R_t)^T$$
The input matrix of neural network is \( X_t \), and the outputs are as follows:
\[
Y_1 = (H_{t-n+2}, H_{t-n+3}, \ldots, H_{t+1}), \quad Y_2 = (L_{t-n+2}, L_{t-n+3}, \ldots, L_{t+1}),
\]
\[
Y_3 = (C_{t-n+2}, C_{t-n+3}, \ldots, C_{t+1}), \quad Y_4 = (R_{t-n+2}, R_{t-n+3}, \ldots, R_{t+1}).
\]

**Neural Network Predicting**

**Network Settings.** The principles of choose the number of hidden nodes are:

1. \( \log_2 (\text{The dimension of input neurons}) + 1; \)
2. Generally, select \( \sqrt{(n_0 + n_i)} + n \), where \( n_0 \) is the number of input neurons, \( n_i \) is the number of output neurons, and \( n \) is an integer from 1 to 10;
3. The number of nodes in hidden layer = (input nodes + output nodes) / 2.

The network has 75 nodes in hidden layer, one node of output units, and then the number of hidden nodes is 38. We take \( 10^{-3} \) as the training error, and 1000 as the training steps, choose the log Sigmoid function as hidden layer activation function, and the linear function as output function.

**Network Training.** Take \( n = 30 \), and train the network respectively. We get the error curves:

While training step is almost 200 steps, errors of \( H \) and \( C \) have reached \( 10^{-3} \) respectively, and when training step is 1000 steps, error of \( L \) has reached \( 10^{-3} \). That is a good training effect. The relative error curves of these four indicators are shown as follows:

![Fig. 2. Error curves of Highest price](image1.png)

![Fig. 3. Error curves of Lowest price](image2.png)

![Fig. 4. Error curves of Closing price](image3.png)

![Fig. 5. Relative error of highest price](image4.png)

![Fig. 6. Relative error of lowest price](image5.png)

![Fig. 7. Relative error of Closing price](image6.png)

![Fig. 8. Relative error of Return](image7.png)
Access specific data on the training of network test, from Figure 5 to Figure 8 we can know that, the relative errors of $H_t$, $L_t$ and $C_t$ are very small, indicating the accuracy of the network after training is better due to testing, and could be used for prediction. As for $R_t$, the relative error is large, indicating that this network after training can not be used to predict.

As Figure 9, Figure 10, and Figure 11 show, the network after training can fit the original data, and results of forecast are good. Thus, we conclude that the trained neural network is very suitable for predicting the highest price $H_t$, lowest price $L_t$, opening price $O_t$ and closing price $C_t$ of the Shanghai Stock Exchange Composite Index; but the result for the return $R_t$ is bad.

Conclusions
From the present analysis, we can conclude that:

(1) The Neural Network model is an appropriate tool to predict the Shanghai Stock Exchange Composite Index, which means that the Chinese Securities Market is not an efficient market.

(2) Based on the history data, the Neural Network model is successfully applied to predict the daily highest/lowest price and Closing price of the Shanghai Stock Exchange Composite Index in short time, but it is ineffective for predicting the return rate of the Shanghai Stock Exchange Composite Index.

(3) The Neural Network model may help in the further research on derivative products pricing, portfolio management and financial risk management.

References


