CSCanada
Management Science and Engineering
Vol. 8, No. 1, 2014, pp. 22-34
DOI:10.3968/j.mse.1913035X20140801.4274

ISSN 1913-0341 [Print]
ISSN 1913-035X [Online]
www.cscanada.net
www.cscanada.org

Experimenting With Three Different Input-Output Mapping Structures of ANN Models for Predicting CSI 300 Index

ZHANG Chengzhao[a],*; PAN Heiping[b]

[a]Ph.D. School of Management and Economics, University of Electronic Science and Technology, Chengdu, China.
[b]Chongqing Institute of Finance, Chonqing Technology and Business University, Chongqing, China.
*Corresponding author.

Received 10 December 2013; accepted 9 February 2014

Abstract

Forecasting the stock market price index is a challenging task. Many scholars have tried on many kinds of models to predict the stock index, mainly autoregressive integrated moving average model (ARIMA), artificial neural networks (ANN) with genetic algorithms (GA). This paper documents a set of thorough empirical tests of ANN's with different choices of inputs and different numbers of hidden neurons for forecasting the CSI 300-the benchmark stock index of China. The prediction accuracy is measured in terms of hit rate and mean square error. The trend of the hit rate is observed by adjusting the window length and the number of hidden neurons. The results show that the hit rate is highest when the window length is between 14 days to 20 days.

Key words: ARMIA; ANN; GA; CSI 300; Hit rate; Mean square error

ZHANG Chengzhao, PAN Heiping (2014). Experimenting With Three Different Input-Output Mapping Structures of ANN Models for Predicting CSI 300 Index. *Management Science and Engineering*, *8*(1), 22-34. Available from: http://www.cscanada.net/index.php/mse/article/view/j.mse.1913035X20140801.4274
DOI: http://dx.doi.org/10.3968/j.mse.1913035X20140801.4274

INTRODUCTION

The stock market has become a popular investment channel in recent years owing to the relative low return rates of other investment instrument in a long-term view. Most of the investors, either individual or institutional investors, are interested in the prediction of the stock index. However, making accurate prediction of the stock index is a challenging task owing to inherently noisy and non-stationary nature of the stock index (Yaser & Atiya, 1996, pp.205-213; Zhu, & Ogihara, 2002, pp.49-68). Many macro-economical factors affect the stock index such as political events, general economic conditions, commodity price changes, corporate policies, interest rates and exchange rates, and investors' expectations and mass psychological factors.

Forecasting models are used to forecast the future trends of the stock index based on historical observations - time series of the stock index. There are many approaches to model the financial time series according to a theory or assumptions about the hypothesized relationship or dynamics in the data. Traditional methods are based on linear models such as time series regression, exponential smoothing and autoregressive integrated moving average (ARIMA) (Brooks, 2002, p.289). All these methods assume linear relationships among the past values of the forecast variable and therefore non-linear patterns cannot be captured by these models. Approximation of linear models to complex real-world problems is not always satisfactory. Non-linear models for stock index forecasting are also developed in the literature, mainly including artificial neural networks (ANN), genetic algorithms (GA) and support vector machine (SVM) and so on. A large number of successful applications have shown that ANN can be a very useful tool for financial time-series modeling and forecasting (Bodyanskiy & Popov, 2006, pp.1357-1366; Freitas & Rodrigues, 2006, pp.801-814). ANN can approximate any linear and nonlinear functions because of its own nonlinear and connectionist characteristics. ANN let the data speak for themselves and have the capability to identify the underlying mapping among the data.

1. LITERATURE REVIEW

Interdisciplinary financial scholars have done a lot of research on ANN-based models for forecasting the stock index. Pan, Titakaratne, and Yearwood (2005, pp.43-45) used ANN to predict Australian stock index exploiting dynamical swings and inter-market influences. A basic neural network with limited optimality on these aspects from them achieved correctness in directional prediction of 80%. Pan, Haidar, and Kulkarni (2009, pp.177-191) predicted short-term trends of crude oil prices with neural networks exploiting multimarket dynamics. The best out-of-sample hit rate is produced by using the spot prices and the heating oil prices for input. Several transformations on the original price data were tested, it was found that 3-day moving averaging to the original data as preprocessing leads to much higher hit rate of prediction. Wang, Wang, and Zhang (2012, pp.758-766) predicted a stock index based on a hybrid model combining the exponential smoothing model (ESM), autoregressive integrated moving average model (ARIMA), and the back propagation neural network (BPNN). Their results showed that the proposed hybrid model outperforms all traditional models, including ESM, ARIMA, BPNN, and the random walk model. Bekiros (2010, pp.285-293) introduced a hybrid neuro-fuzzy system for decision-making and trading under uncertainty. The efficiency of a technical trading strategy based on the neuro-fuzzy model is investigated, in order to predict the direction of the market for 10 of the most prominent stock indices of U.S.A, Europe and Southeast Asia. The total profit of the proposed neuro-fuzzy model is consistently superior to a recurrent neural network and a Buy & Hold strategy for all indices including transaction fee, particularly for the highly speculative, emerging Southeast Asian market. Barbulescu, and Bautu (2012, pp.327-335) proposed a novel method for time series forecasting based on a hybrid combination of ARMA and Gene Expression Programming (GEP) induced models. The investigations showed a definite improvement in the accuracy of forecasts of the hybrid method over pure ARMA and GEP used separately. Zhang, and Berardi (2001, pp.652-664) presented a detailed investigation of the effectiveness of neural network ensembles for exchange rate forecasting. Results show that by appropriately combining different neural networks, forecasting accuracy of individual networks can be largely improved. Although their ensemble methods showed considerable advantages over the traditional KTB approach, they did not have significant improvement compared to the widely used random walk model in exchange rate forecasting. Ruxanda (2010, pp.37-54) used multilayer perceptron neural networks to predict the exchange rate time series. Jacquier, Polson and Rossi (2004, pp.185-212) develop a MCMC algorithm to conduct inference in an extended SVOL (stochastic volatility) model, featuring fat-tails and a leverage effect. Methods for computation of Bayes Factors are introduced to assess the weight of the sample evidence. Ding Shifei, Su Chunyang, and Yu Junzhao (2011, pp.153-162) combined the BP neural networks and GA and showed the method that use GA to optimize the connection weights of neural network. The BP algorithm improves the convergence rate of the network and reduces the training failure, and the neural network's generalization ability is better than the algorithms that only use GA. Kwon Yung-Keun, Moon, Byung-Ro (2007, pp.851-864) proposed a GA combined with a recurrent neural network having one hidden layer for the daily stock trading. The proposed method was tested with 36 companies in NYSE and NASDAQ for 13 years from 1992 to 2004 and showed significantly better performance than the "buy-and-hold" strategy. Koulouriotis, Diakoulakis, Emiris, and Zopounidis (2005, pp.157-179) went to the methodological extension and application of dynamic cognitive networks, an emerging technique in the field of cognitive mapping and systems analysis. Pai and Lin (2005, pp.497-505) proposed a hybrid model of ARIMA and SVM which is believed to greatly improve the prediction performance of the single ARIMA model or the single SVM model in forecasting stock prices. Theoretically as well as empirically, hybridizing two dissimilar models reduces forecasting errors (Granger CWJ, 1989, pp.167-173). Melin, Mancilla, and Lopez (2007, pp.1217-1226) proposed modular neural networks for simulation and forecasting time series of consumer goods in the U.S. market. They applied monolithic and modular neural networks with different training algorithms to compare the results and found the Levenberg-Marquardt learning algorithm produced the best result. Chavarnakul and Enke (2008, pp.1014-1017) proposed a generalized regression neural network (GRNN) combined with the VAMA (volume adjusted moving average) and EMV (ease of movement) for stock trading. Result from all trading strategies showed that VAMA and EMV with the neural network can improve the performance of the VAMA and EMV alone by providing earlier trading signals. The results utilized from VAMA and EMV with the neural network outperform other benchmarking tools, including those without neural network assistance, the MA, VAMA used alone, and the buy-and-hold strategy. Refenes, Zapranis and Francis (1994, pp.375-388) found that neural networks can provide a reasonable explanation of their predictive behavior and can model their environment more convincingly than regression models. Chen Wun-Hua, Shi Jen-Ying, and Wu Soushan (2006, pp.49-67) examined the feasibility of applying two AI models, SVM and BP, to financial time-series forecasting for the Asian stock markets. Their experiments demonstrated that both two AI models perform better than the benchmark AR (1) model in the deviation measurement criteria. Kanas (2003, pp.299-315) extends recent research on non-linear present-value by exploring the relative out-of-sample forecast performance of two parametric and two non-parametric non-linear models of stock return. The Markov regime switching model is the most preferable non-linear empirical extension of the present-value model for out-of-sample forecasting of stock returns. Moreno and Olmeda (2007, pp.436-454) have analyzed the daily and weekly fore-castability of stock returns of a large number of markets and several years. They employed different information sets as well as model specifications. Their results suggested that nonlinear models do not provide superior predictions than the linear ones and that emerging and developed stock markets are generally nonpredictable when total transaction cost are considered. Chen, Leung & Daouk (2003, pp.901-923) proposed the probabilistic neural network (PNN) in predicting the direction of index returns. Their results showed that the PNN has a stronger predictive power than both the GMM-Kalman (generalized methods of moments) filter and the random walk forecasting models.

2. MODEL SETTING AND ANALYSIS

In this paper we choose to use multilayer Feed forward Neural Networks (FNN) as our computational model to forecast the stock index. The neural network has three layers - the input layer, the hidden layer, and the output layer. The number of neurons in the hidden layer is predefined to the range from 2 to 10 according to our experience. To be precise, let us start with our notation: Assuming variable $X(t)$ indicates the stock market price (including index) data at time period t. In general, we have:

$$X(t) = (X.O(t), X.H(t), X.L(t), X.C(t), X.V(t)) \quad (1)$$

$X.O(t)$,$X.H(t)$,$X.L(t)$,$X.C(t)$ and $X.V(t)$ are the opening price, the highest price, the lowest price, the closing price and the volume traded during the given time period, respectively. Usually the time period t may take one of 8 levels: 1 minute, 5 minutes, 15 minutes, 30 minutes, 1 hour, 4 hours, 1 day, 1 week and 1 month. The investigation in this paper is limited to daily data only. Here we assume the availability of the sufficient historical data sets $FX(t)$ with the following form:

$$FX(t) = \{X(k) \mid k = t - N + 1, t - N + 2, ..., t - 2, t - 1, t\} \quad (2)$$

N is the length of the historical time series data set. In view of the short-term trend of feed-forward neural network forecasting model, after assuming the historical data set, the main task is to determine the input and output mapping. The input to the model is a feature set extracted from the historical data set $FX(t)$. The input set of the model is denoted as $I.X(t)$ in this paper and the output of the model is denoted as $O.X(t)$. The output of the model is generally one step or multi-step prediction of the future price data. Thus, the simplest input-output mapping of the prediction model based on a multilayer Feedforward Neural Network (FNN) can be expressed as:

$$FX(t) \Rightarrow I.X(t) \Rightarrow FNN \Rightarrow O.X(t) = \{X(t+1), X(t+2), ..., X(t+T)\} \quad (3)$$

The prediction target is the future of the opening price, the highest price, the lowest price and the close price. For simplicity, the prediction target of this paper has following structure:

$$I.X(t) \Rightarrow FNN^T \Rightarrow \{X.O(t+T), X.H(t+T), X.L(t+T), X.C(t+T)\} \quad (4)$$

Particular, when the T = 1, it is one step prediction:

$$I.X(t) \Rightarrow FNN^1 \Rightarrow \{X.O(t+1), X.H(t+1), X.L(t+1), X.C(t+1)\} \quad (5)$$

As the change of the opening price is mostly due to non-predictable information from outside of the market, it is hardly predictable from historical data. Therefore, the opening price is excluded from the target of prediction here; one step prediction model can be reduced into:

$$FX(t) \Rightarrow I.X(t) \Rightarrow FNN^1 \Rightarrow \{X.H(t+1), X.L(t+1), X.C(t+1)\} \quad (6)$$

The input data of neural network is not the price (index), but the logarithmic return of the price (index) in this paper. The formula of the logarithmic return is:

$$LX(t) = \ln \frac{X(t)}{X(t-1)} \approx \frac{X(t) - X(t-1)}{X(t-1)} \quad (7)$$

We use three models to predict the CSI 300 index in this article: 1) single input single output sets model, 2) multiple input single output sets model and 3) multiple input multiple output sets model.

2.1 Model One: Single Input Single Output Sets Model

We respectively use three neural networks to predict the logarithmic returns of the closing price, the highest and the lowest price. Each neural network has three layers: The input layer, the hidden layer and the output layer. The number of neurons in the hidden layer spans from 2 to 10 according to experience. The input data of the three neural networks are the logarithmic return of the closing price, the highest price and the lowest price respectively. The look-back window length is an input parameter of the neural network which is from 1 day to 22 days considering a month has about 22 trading days. Our main goal here is two-fold: i.e. to find the optimal window length and the optimal number of hidden neurons. According to formula (6) the Model One can be expressed in three neural networks for predicting the high, low and closing price respectively each with its own input:

$$\begin{cases} \{FX(t) \Rightarrow I.LX_h(t)\} \Rightarrow FNN_h \Rightarrow LX_h(t+1) \\ \{FX(t) \Rightarrow I.LX_l(t)\} \Rightarrow FNN_l \Rightarrow LX_l(t+1) \\ \{FX(t) \Rightarrow I.LX_c(t)\} \Rightarrow FNN_c \Rightarrow LX_c(t+1) \end{cases} \quad (8)$$

2.2 Model Two: Multiple Input Single Output Sets Model

In model two, we still use three neural networks. Each neural network has the same input containing the logarithmic returns of the highest, the lowest and the closing prices. But three neural networks are different in different outputs. Each neural network has three layers: The input layer, the hidden layer and the output layer. The number of neurons in the hidden layer varies from 2 to 10 according to experience. Unlike model one, the input data of the neural network are the vector which simultaneously contains the logarithmic return of the closing price, the highest price and the lowest price in a sequence. The output data set of the neural network is the prediction of the logarithmic return of the closing price, the highest price and the lowest price respectively. The look-back window length is an input parameter of the neural network which varies from 1 day to 22 days. The goal here is to find the optimal window length and the optimal number of hidden neurons. According to the formula (6), model two is expressed in a single neural network:

$$\left\{ FX(t) \Rightarrow I.LX(t) = \begin{pmatrix} I.LX_h(t) \\ I.LX_l(t) \\ I.LX_c(t) \end{pmatrix} \right\} \Rightarrow FNN \Rightarrow LX_h(t+1) \quad (9)$$

$$\left\{ FX(t) \Rightarrow I.LX(t) = \begin{pmatrix} I.LX_h(t) \\ I.LX_l(t) \\ I.LX_c(t) \end{pmatrix} \right\} \Rightarrow FNN \Rightarrow LX_l(t+1) \quad (10)$$

$$\left\{ FX(t) \Rightarrow I.LX(t) = \begin{pmatrix} I.LX_h(t) \\ I.LX_l(t) \\ I.LX_c(t) \end{pmatrix} \right\} \Rightarrow FNN \Rightarrow LX_c(t+1) \quad (11)$$

2.3 Model Three: Multiple Input Multiple Output Sets Model

In model three, we use only one neural network to predict the logarithmic returns of the closing price, the highest and the lowest price. The neural network has three layers: The input layer, the hidden layer and the output layer. The number of neurons in the hidden layer varies from 2 to 10 according to experience. The input data of the neural network are the vector which simultaneously contains the logarithmic return of the highest, the lowest, and the closing price sequentially. Unlike model one and model two, the output data set of the neural network is the prediction of the logarithmic return of the highest, the lowest, and the closing price simultaneously. The look-back window length is an input parameter of the neural network which varies from 1 day to 22 days. The goal here is to find the optimal window length and the optimal number of hidden neurons. According to the formula (6), model three can be expressed as:

$$\left\{ FX(t) \Rightarrow I.LX(t) = \begin{pmatrix} I.LX_h(t) \\ I.LX_l(t) \\ I.LX_c(t) \end{pmatrix} \right\} \Rightarrow FNN_M \Rightarrow \{LX_h(t+1), LX_l(t+1), LX_c(t+1)\} \quad (12)$$

3. HISTORICAL DATA AND STATISTICAL PROPERTIES OF THE CSI 300 INDEX

We study the daily time series of CSI index from April 8, 2005 to April 3, 2013, with 3 components: The highest, lowest, and closing prices. So the data set includes 1942 daily data points. All the data were obtained from the RESSET financial database (www.resset.cn). The data set was divided into three subsets: 70% of the data for training set, 15% of the data for validation set and 15% of the data for testing set. This 3-set division is a popular approach to avoid overfitting. The training set is used for estimating the weights of the FNN model, the validation set is used for model selection, and the testing set is used for out-of-sample evaluation. It is important to clarify that the performance of the testing set must not influence the choice of the FNN architecture. The logarithmic returns for each component time series is normalized to fit into interval [-1, 1] as preprocessing before subsequent steps in all the experiments.

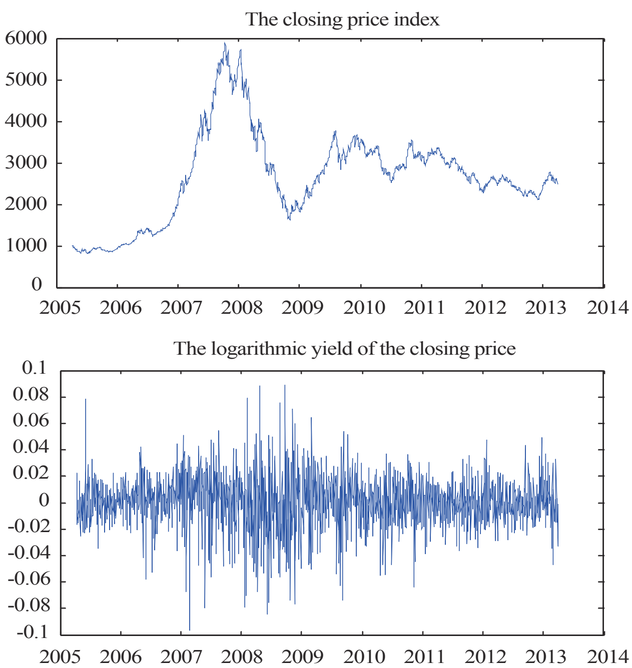

Figure 1
The Closing Price of the Index (Upper) **and Its Logarithmic Return** (Lower)

Figure 1 shows the closing price of CSI 300 index in the upper section and the logarithmic return of the closing price in the lower section. The figure of the closing price of the index shows that the behavior of the closing price time series has started to change from 2007. The level of price has increased significantly from 2006 to 2008.

The index has retraced deeply after 2008. Moreover, the figure of the logarithmic return reveals that the index has high volatility on daily basis. Furthermore, it also shows evidence of volatility clustering large changes are followed by large changes and small changes are more often followed by small changes.

In order to detect whether there may be any stochastic dynamics in the three component time series, we have investigated on the statistical properties of the data using a Ljung-Box Q-test for autocorrelation. Ljung-Box Q-test was done for autocorrelation of the input data and the formula is as follow:

$$Q_{LB} = T(T+2)\sum_{j=1}^{p}\frac{r_j^2}{T-j} \quad (13)$$

Where r_j is autocorrelation coefficient of the residual sequence at lag j, T is the sample size, P is the number of lags tested for autocorrelation. Under this model the null hypothesis is that there is no significant correlation.

The results of the Ljung-Box Q-test for the logarithmic return of the highest, lowest price and closing prices are shown in Table 1 to Table 3 respectively. Clearly, significant correlations are detected for all the lags tested(5, 10, 15 and 20) at 5% significant level. Thus, there is strong evidence (P-value all less than 0.05) that the present and past information could be useful to predict the futures direction.

Table 1
Ljung-Box Q-Test for the Logarithmic Return of the Highest Price

Lag	*P-value*	*LQ*	Critical value
5	0.0075E-10	65.8455	11.0705
10	0.2794E-10	71.0374	18.3070
15	0.0050E-10	91.5734	24.9958
20	0.0574E-10	96.3049	31.4104

Table 2
Ljung-Box Q-Test for the Logarithmic Return of the Lowest Price

Lag	P-value	LQ	Critical value
5	0.0437E-5	37.6836	11.0705
10	0.1252E-5	46.3251	18.3070
15	0.0033E-5	65.1177	24.9958
20	0.0057E-5	73.0598	31.4104

Table 3
Ljung-Box Q-Test for the Logarithmic Return of the Closing Price

Lag	P-value	LQ	Critical value
5	0.0450	11.3410	11.0705
10	0.0180	21.4800	18.3070
15	0.0003	41.1303	24.9958
20	0.0003	48.9219	31.4104

Therefore, we can use the information (feature vector) extracted from the past prices as input to the FNN models. The pairwise correlation coefficients of the 3 component price time series are calculated and shown in tables 4-6. These statistical properties support the usefulness of the three component prices as input to the FNN models. (CCC: Cross Correlation Coefficient)

Table 4
The Correlation Coefficient of the Closing Price and the Highest Price

Lag	CCC	Lag	CCC	Lag	CCC	Lag	CCC
1	0.0043	10	0.0370	19	-0.0469	28	0.0091
2	0.0192	11	0.0621	20	-0.0013	29	-0.0018
3	0.0534	12	0.0386	21	0.0075	30	0.0099
4	0.0523	13	0.0545	22	0.0376	31	0.0014
5	-0.0232	14	0.0265	23	-0.0039	32	-0.0246
6	-0.0162	15	0.0357	24	-0.0143	33	-0.0374
7	0.0176	16	-0.0204	25	0.0143	34	0.0556
8	-0.0149	17	0.0117	26	0.0154	35	0.0166
9	0.0186	18	0.0449	27	0.0311	36	0.0130

Table 5
The Correlation Coefficient of the Closing Price and the Lowest Price

Lag	CCC	Lag	CCC	Lag	CCC	Lag	CCC
1	-0.0411	10	0.0723	19	-0.0206	28	-0.0032
2	0.0181	11	0.0350	20	-0.0013	29	0.0104
3	0.0702	12	0.0424	21	-0.0014	30	-0.0053
4	0.0407	13	0.0311	22	0.0509	31	0.0291
5	-0.0262	14	0.0107	23	-0.0034	32	-0.0699
6	-0.0469	15	0.0412	24	-0.0195	33	0.0038
7	0.0454	16	-0.0178	25	-0.0039	34	0.0313
8	-0.0288	17	-0.0123	26	0.0345	35	-0.0003
9	-0.0005	18	0.0431	27	0.0260	36	0.0196

Table 6
The Correlation Coefficient of the Highest Price and the Lowest Price

Lag	CCC	Lag	CCC	Lag	CCC	Lag	CCC
1	0.2861	10	0.0343	19	0.0351	28	-0.0040
2	-0.0408	11	0.0617	20	0.0001	29	0.0309
3	0.0824	12	0.0453	21	-0.0223	30	-0.0046
4	0.0839	13	0.0292	22	0.0564	31	0.0083
5	0.0060	14	0.0202	23	-0.0030	32	-0.0114
6	-0.0541	15	0.0220	24	-0.0181	33	-0.0278
7	0.0341	16	0.0388	25	0.0170	34	0.0157
8	-0.0064	17	-0.0237	26	-0.0077	35	0.0117
9	-0.0028	18	0.0037	27	0.0403	36	0.0219

4. PERFORMANCE MEASURES TO THE PREDICTION MODELS

Here we use two performance measures for the prediction accuracy of a FNN model. They are the Hit Rate (HR) and Mean Squared Error (MSE). Suppose at time t, the output of the prediction model is $y(t)$, the actual value of market price is $x(t)$, so we define these measures as follows:

(1) HR: Hit Rate measures the right number to hit the target of short-term trend direction over the total number of trials. On the financial market price prediction research, the ultimate goal is to find a robust predictive model and apply it into the actual market investment. If the objective is simply to predict the price value of financial market, the utility to the investment decision-making would be limited. So in order to be more effectively linked with the actual trading, the price trend of the market price is the most concerned to the investors. So HR will be the most important criterion to measure the effectiveness of predictive models. HR is calculated as follows:

$$HR = \frac{1}{n}\sum_{k=1}^{n}\left[x(t_k)y(t_k) > 0\right] \tag{14}$$

Where *[f]* represents the number satisfying the condition *f*.

(2) MSE: Mean Squared Error is used to measure the predictive accuracy of the model output value. MSE is defined as follows:

$$MSE = \frac{1}{n}\sum_{k=1}^{n}\left(x(t_k) - y(t_k)\right)^2 \tag{15}$$

It could be argued that HR of prediction is more relevant than MSE, since the trading decisions are usually made based on the trend direction of the market. For this reason, to evaluate the performance of our predictions in a different way, we compute the proportion of correct forecasted directions. Under the Efficient Market Theory, the null hypothesis is that the market is not predictable, thus HR should be equal to 0.5. A number higher than 0.5 with statistical significance indicates that the corresponding model outperforms the random walk. In addition, Root Mean Square Errors (RMSE) and Mean Absolute Error (MAE) can also be considered.

5. EXPERIMENTING FNN MODELS AND RESULT INTERPRETATION

In order to analyze the model performance in a way closer to the investors preference, we focus on prediction hit rate when analyzing the prediction results of various models.

5.1 Experimenting Model One: Single Input Single Output Sets Model

As defined by equation (8), model one is limited to use the logarithmic returns of the highest, lowest, and closing prices to predict the next day of those logarithmic returns respectively. This model one is used as the benchmark for the other two models. A number of data-based experiments are carried out for virtually exhaustive search for the optimal input-output mapping and the optimal architecture of the FNN. In view of the window length as an input parameter to the FNN, a specific form of model one for one-step prediction can be expressed as:

$$\begin{cases} \{D(FX(t),m) \Rightarrow D(I.LX_h(t),m)\} \Rightarrow FNN_h \Rightarrow LX_h(t+1) \\ \{D(FX(t),m) \Rightarrow D(I.LX_l(t),m)\} \Rightarrow FNN_l \Rightarrow LX_l(t+1) \\ \{D(FX(t),m) \Rightarrow D(I.LX_c(t),m)\} \Rightarrow FNN_c \Rightarrow LX_c(t+1) \end{cases} \tag{16}$$

The m refers to the window length. The performances of model one with different window length and optimal number of hidden neurons are shown in tables 7-9. (ONoHN means Optimal Number of Hidden Neurons)

Table 7
Performance of Model One on Closing Prices With Different Window Length

Window length	ONoHN	Hit Rate			MSE		
		Training	Validation	Testing	Training	Validation	Testing
1	9	53.17%	50.52%	60.82%	3.52129e-4	3.76675e-4	3.89020e-4
2	7	52.1%	53.61%	55.33%	3.63096e-4	3.55476e-4	3.43759e-4
3	10	56.27%	52.92%	60.14%	3.60875e-4	2.96701e-4	3.65037e-4
4	5	56.09%	50.86%	58.08%	3.34453e-4	4.18647e-4	3.65468e-4
5	10	57.46%	52.23%	57.04%	3.04496e-4	3.62815e-4	4.14003e-4
6	5	56.25%	50.52%	57.73%	3.55446e-4	3.37648e-4	3.57551e-4
7	5	53.92%	52.58%	56.36%	3.43072e-4	3.70846e-4	3.77024e-4
8	9	56.40%	54.30%	59.45%	3.32548e-4	3.88198e-4	3.75788e-4
9	8	56.96%	50.86%	56.70%	3.40189e-4	3.77682e-4	3.88720e-4
10	7	56.71%	50.17%	56.70%	3.48337e-4	4.07475e-4	3.29281e-4
11	10	59.27%	49.14%	59.11%	3.18207e-4	3.63658e-4	4.07475e-4

To be continued

Continued

Window length	ONoHN	Hit Rate			MSE		
		Training	Validation	Testing	Training	Validation	Testing
12	8	56.42%	48.80%	57.73%	3.23160e-4	4.02470e-4	3.87910e-4
13	10	59.58%	49.48%	60.82%	2.86703e-4	3.82611e-4	4.11030e-4
14	2	57.47%	48.45%	58.08%	3.65658e-4	3.98572e-4	3.07156e-4
15	5	55.65%	48.45%	60.14%	3.42236e-4	3.45432e-4	4.12928e-4
16	6	55.17%	50.17%	59.45%	3.56824e-4	3.96445e-4	3.66718e-4
17	6	58.27%	51.20%	61.86%	2.99686e-4	4.16327e-4	4.96115e-4
18	9	58.76%	50.86%	59.79%	2.93196e-4	4.03575e-4	4.40188e-4
19	8	59.33%	50.86%	57.73%	3.35050e-4	3.21888e-4	4.22516e-4
20	4	59.60%	50.52%	57.39%	3.44814e-4	3.1261e-4	4.06787e-4
21	6	54.93%	53.95%	57.04%	3.49959e-4	3.74004e-4	3.31936e-4
22	9	61.63%	52.23%	59.45%	3.11954e-4	3.93944e-4	3.25792e-4

The hit rate of the closing price gets the maximum value 61.86% when the window length is 17 days and gets the minimum value 55.33% when the window length is 2 days in model one.

Table 8
Performance of Model One on Highest Prices With Different Window Length

window length	ONoHN	Hit Rate			MSE		
		Training	Validation	Testing	Training	Validation	Testing
1	8	54.57%	54.30%	56.70%	2.54547e-4	2.79401e-4	2.27677e-4
2	6	54.61%	54.64%	56.36%	2.61257e-4	2.50389e-4	2.27186e-4
3	6	54.65%	53.61%	57.04%	2.41685e-4	2.75466e-4	2.69499e-4
4	10	58.75%	55.67%	57.73%	2.34543e-4	2.42987e-4	2.87495e-4
5	7	60.19%	51.89%	57.39%	2.36565e-4	2.52333e-4	2.72230e-4
6	9	59.28%	58.42%	57.73%	2.25132e-4	2.37096e-4	2.68462e-4
7	7	61.09%	53.95%	56.70%	2.27076e-4	2.77825e-4	2.38737e-4
8	8	59.22%	52.58%	58.08%	2.46527e-4	2.74508e-4	2.21595e-4
9	5	57.19%	49.48%	59.11%	2.51579e-4	2.21579e-4	2.64053e-4
10	8	59.15%	51.20%	57.73%	2.27983e-4	2.91501e-4	2.45268e-4
11	6	58.53%	53.26%	57.39%	2.48738e-4	2.30739e-4	2.44858e-4
12	10	59.09%	54.98%	58.08%	2.18141e-4	2.83407e-4	2.66670e-4
13	3	59.36%	58.76%	57.39%	2.32730e-4	2.96756e-4	2.85181e-4
14	5	61.26%	51.55%	56.36%	2.26152e-4	1.89732e-4	3.18264e-4
15	6	52.60%	51.20%	58.08%	2.59712e-4	3.12607e-4	2.72610e-4
16	9	60.16%	55.33%	58.42%	2.22001e-4	2.71332e-4	2.51517e-4
17	9	61.85%	54.64%	57.39%	1.91822e-4	2.44879e-4	3.70028e-4
18	10	55.26%	50.86%	57.39%	3.16232e-4	2.48771e-4	3.05663e-4
19	7	58.73%	51.20%	63.23%	2.25489e-4	3.02258e-4	3.18219e-4
20	7	59.90%	60.14%	58.42%	2.21574e-4	3.04939e-4	2.88186e-4
21	3	60.09%	51.89%	57.04%	2.38534e-4	2.35780e-4	2.43795e-4
22	9	58.41%	50.52%	57.04%	2.18824e-4	2.69984e-4	3.30646e-4

The hit rate of the highest price gets the maximum value 63.23 % when the window length is 19 days and gets the minimum value 56.36% when the window length is 2 days in model one.

Table 9
Performance of Model One on Lowest Prices With Different Window Length

Window length	ONoHN	Hit Rate			MSE		
		Training	Validation	Testing	Training	Validation	Testing
1	6	53.31%	54.30%	57.73%	3.81597e-4	3.33257e-4	2.93626e-4
2	10	53.43%	54.30%	57.04%	3.59042e-4	3.28253e-4	3.86663e-4
3	8	56.42%	49.48%	58.08%	3.46532e-4	3.15385e-4	4.20502e-4
4	10	54.02%	53.26%	57.73%	3.49624e-4	4.34130e-4	2.65633e-4
5	2	54.51%	52.58%	57.04%	3.83739e-4	3.33918e-4	2.49695e-4
6	4	55.65%	48.80%	58.76%	3.3729e-4	3.43114e-4	4.37878e-4
7	3	58.21%	50.52%	58.08%	3.34557e-4	3.30219e-4	4.64936e-4
8	4	56.48%	51.55%	58.42%	3.51288e-4	3.69112e-4	2.99850e-4
9	7	54.96%	49.83%	58.76%	3.96435e-4	3.46107e-4	3.88599e-4
10	2	56.71%	47.77%	60.48%	3.22032e-4	4.64510e-4	3.84609e-4
11	2	57.42%	49.48%	59.79%	3.30891e-4	4.36131e-4	3.56998e-4
12	4	57.68%	47.77%	60.48%	3.47817e-4	3.45892e-4	2.57587e-4
13	10	60.33%	47.08%	59.11%	3.35819e-4	3.15794e-4	3.42412e-4
14	5	58.29%	50.86%	57.73%	3.36197e-4	3.77400e-4	3.48407e-4
15	10	57.51%	51.20%	58.08%	3.04626e-4	4.53350e-4	5.36721e-4
16	7	54.43%	53.26%	59.45%	3.40161e-4	4.00233e-4	4.54210e-4
17	4	58.12%	52.23%	60.48%	3.30095e-4	4.08135e-4	3.53261e-4
18	10	60.33%	53.61%	57.39%	2.81008e-4	3.88021e-4	4.12486e-4
19	9	59.55%	47.77%	62.54%	3.07975e-4	3.42655e-4	5.02655e-4
20	10	61.02%	51.89%	58.08%	2.53019e-4	4.31926e-4	5.89697e-4
21	5	58.30%	49.14%	57.73%	2.95786e-4	3.28023e-4	5.49061e-4
22	7	55.57%	56.36%	59.79%	3.29766e-4	3.79454e-4	4.36860e-4

The hit rate of the lowest price gets the maximum value 62.54% when the window length is 19 days and gets the minimum value 57.04% when the window length is 2 days in model one.

The relationships of the hit rate of the closing price, the highest price, the lowest price and the window length in model one are shown in figure 2 to figure 4 respectively.

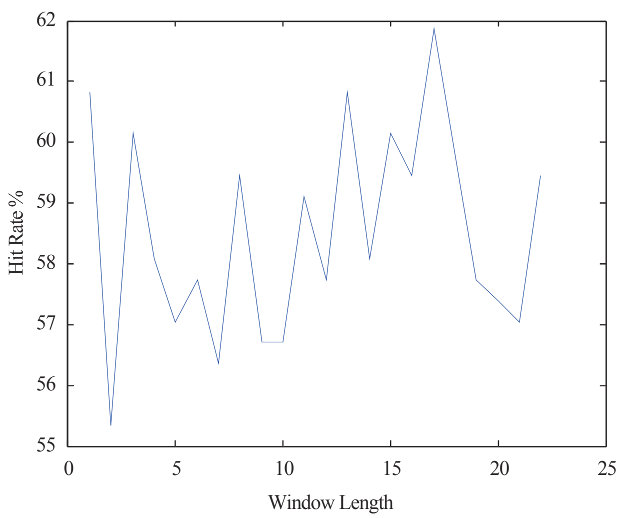

Figure 2
Variation of Hit Rate of the Closing Price With the Window Length of Model One

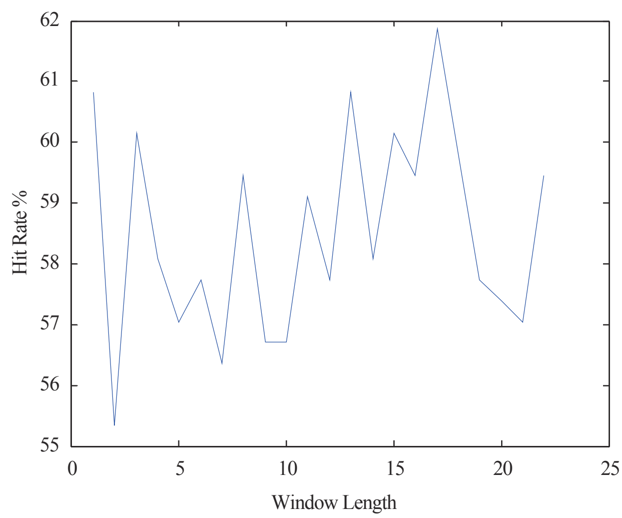

Figure 3
Variation of Hit Rate of the Highest Price With the Window Length of Model One

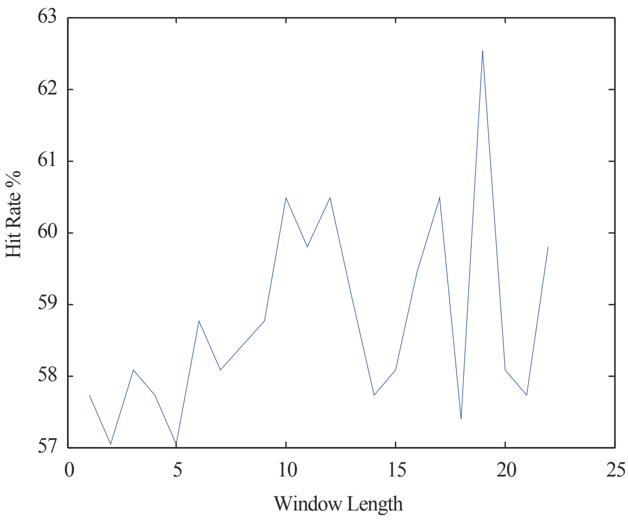

Figure 4
Variation of Hit Rate of the Lowest Price With the Window Length of Model One

5.2 Experimenting Model Two: Multiple Input Single Output Sets Model

As defined by equation (9) in model two, the input data are the logarithmic returns of the highest, lowest, and closing prices in parallel and the output data are the logarithmic return of the highest, lowest, and closing prices respectively. In view of the window length as an input parameter to the FNN, a specific form of model two for one-step prediction can be expressed as

$$\left\{ D(FX(t),m) \Rightarrow D(I.LX(t),m) = \begin{pmatrix} I.LX_h(t) \\ I.LX_l(t) \\ I.LX_c(t) \end{pmatrix} \right\} \Rightarrow FNN \Rightarrow LX_h(t+1) \quad (17)$$

$$\left\{ D(FX(t),m) \Rightarrow D(I.LX(t),m) = \begin{pmatrix} I.LX_h(t) \\ I.LX_l(t) \\ I.LX_c(t) \end{pmatrix} \right\} \Rightarrow FNN \Rightarrow LX_l(t+1) \quad (18)$$

$$\left\{ D(FX(t),m) \Rightarrow D(I.LX(t),m) = \begin{pmatrix} I.LX_h(t) \\ I.LX_l(t) \\ I.LX_c(t) \end{pmatrix} \right\} \Rightarrow FNN \Rightarrow LX_c(t+1) \quad (19)$$

The *m* refers to the window length. The performances of model two with different window length and optimal number of hidden neurons are shown in tables 10-12.

Table 10
Performance of Model Two on Closing Prices With Different Window Length

Window length	ONoHN	Hit Rate			MSE		
		Training	Validation	Testing	Training	Validation	Testing
1	2	51.03%	53.26%	56.36%	3.82168e-4	3.20277e-4	3.31910e-4
2	10	56.60%	51.20%	57.39%	3.34976e-4	3.57384e-4	3.74832e-4
3	2	48.67%	56.01%	57.73%	3.54071e-4	3.66695e-4	4.07640e-4
4	3	56.90%	56.36%	58.08%	3.64709e-4	3.05567e-4	3.23112e-4
5	6	56.35%	50.52%	58.08%	3.32484e-4	4.23687e-4	3.88613e-4
6	7	56.69%	51.20%	57.39%	3.34296e-4	3.29845e-4	4.26930e-4
7	8	58.95%	54.30%	57.04%	3.17648e-4	3.41916e-4	4.68498e-4
8	7	58.92%	52.92%	58.76%	3.10617e-4	3.73129e-4	4.03523e-4
9	2	56.52%	52.58%	59.45%	3.36562e-4	3.72649e-4	3.89011e-4
10	8	60.34%	51.55%	59.11%	2.82452e-4	5.19103e-4	5.40366e-4
11	9	63.06%	51.55%	59.79%	2.69414e-4	4.67574e-4	4.51917e-4
12	10	58.72%	52.92%	58.76%	3.16730e-4	4.52288e-4	3.12760e-4
13	3	56.17%	52.58%	60.14%	3.41694e-4	4.17981e-4	3.87093e-4
14	10	59.55%	52.23%	62.20%	2.79402e-4	5.01174e-4	4.33848e-4
15	5	55.28%	48.80%	60.14%	2.89891e-4	5.16607e-4	5.06561e-4
16	7	61.36%	52.58%	60.82%	3.10661e-4	3.83050e-4	4.69075e-4
17	10	62.52%	53.95%	58.08%	2.87019e-4	5.18931e-4	6.92034e-4
18	5	60.10%	52.92%	58.76%	3.20983e-4	3.56281e-4	4.24469e-4
19	3	52.61%	56.70%	58.42%	3.28369e-4	4.49511e-4	3.91666e-4
20	7	61.54%	52.92%	60.14%	3.26316e-4	4.64240e-4	4.15666e-4
21	8	62.56%	56.01%	60.48%	2.55050e-4	5.52164e-4	4.54652e-4
22	9	63.65%	51.89%	61.17%	2.69748e-4	3.62594e-4	5.03816e-4

The hit rate of the closing price gets the maximum value 62.20% when the window length is 14 days and gets the minimum value 56.36% when the window length is 1 day in model two.

Table 11
Performance of Model Two on Highest Prices With Different Window Length

Window length	ONoHN	Hit Rate			MSE		
		Training	Validation	Testing	Training	Validation	Testing
1	6	75.18%	66.32%	64.60%	1.48051e-4	1.49309e-4	1.73804e-4
2	4	76.42%	71.13%	69.76%	1.31172e-4	1.83784e-4	1.68885e-4
3	6	77.95%	70.79%	68.38%	1.19880e-4	1.69684e-4	1.51642e-4
4	4	77.27%	70.10%	69.76%	1.35530e-4	1.47402e-4	1.09925e-4
5	7	77.10%	70.10%	70.45%	1.07299e-4	1.74104e-4	1.44205e-4
6	2	77.46%	72.51%	70.79%	1.26880e-4	1.56477e-4	1.38564e-4
7	2	77.88%	74.23%	71.13%	1.38328e-4	1.30881e-4	1.29089e-4
8	10	75.94%	69.42%	71.13%	1.42693e-4	1.42222e-4	1.40998e-4
9	6	78.22%	73.20%	70.45%	1.11117e-4	1.54468e-4	1.38601e-4
10	9	78.06%	70.45%	71.13%	1.21120e-4	1.45299e-4	1.42071e-4
11	6	77.23%	69.76%	71.48%	1.30150e-4	1.69462e-4	1.44098e-4
12	5	77.95%	72.16%	71.48%	1.10980e-4	1.55346e-4	1.71566e-4
13	10	78.68%	75.60%	71.82%	9.63796e-5	1.64295e-4	2.18201e-4
14	3	78.29%	72.16%	71.13%	1.16102e-4	1.46280e-4	1.64910e-4
15	6	78.42%	74.23%	70.45%	9.33318e-4	1.49653e-4	1.76748e-4
16	4	76.40%	70.10%	70.10%	1.07660e-4	1.72580e-4	1.93625e-4
17	3	77.20%	71.13%	72.85%	1.28590e-4	1.57347e-4	1.37641e-4
18	6	77.11%	70.79%	71.82%	1.15076e-4	1.74878e-4	1.42785e-4
19	4	77.31%	72.85%	71.48%	1.17116e-4	1.73045e-4	1.73976e-4
20	8	78.27%	69.42%	70.10%	1.03277e-4	2.21400e-4	1.74849e-4
21	2	79.15%	69.42%	71.13%	1.20182e-4	1.30443e-4	1.41492e-4
22	7	77.94%	71.82%	70.45%	1.04999e-4	3.17281e-4	1.88989e-4

The hit rate of the highest price gets the maximum value 72.85% when the window length is 17 days and gets the minimum value 64.60% when the window length is 1 day in model two.

Table 12
Performance of Model Two on Lowest Prices With Different Window Length

Window length	ONoHN	Hit Rate			MSE		
		Training	Validation	Testing	Training	Validation	Testing
1	6	76.14%	72.85%	69.42%	1.92709e-4	2.17211e-4	2.12661e-4
2	5	76.93%	73.54%	71.13%	1.86560e-4	2.02380e-4	1.76039e-4
3	6	76.92%	74.91%	72.51%	1.72566e-4	2.06200e-4	1.71506e-4
4	9	76.61%	75.95%	73.20%	1.62052e-4	1.71259e-4	2.68697e-4
5	6	78.51%	76.29%	74.23%	1.72607e-4	1.50612e-4	2.33008e-4
6	4	77.38%	76.63%	73.20%	1.81263e-4	1.66336e-4	1.80598e-4
7	9	77.81%	77.32%	73.20%	1.49396e-4	2.14816e-4	1.81619e-4
8	5	76.46%	76.63%	73.88%	1.68217e-4	2.10662e-4	1.87520e-4
9	8	77.93%	75.26%	74.57%	1.46251e-4	2.09204e-4	2.54428e-4
10	2	76.72%	76.63%	73.54%	1.66257e-4	2.11446e-4	1.88132e-4
11	8	78.12%	75.95%	74.57%	1.35461e-4	2.17738e-4	2.71278e-4
12	8	76.76%	74.23%	75.60%	1.54367e-4	2.04216e-4	2.03817e-4
13	6	77.79%	74.91%	73.88%	1.53178e-4	1.71964e-4	2.30193e-4
14	4	78.44%	75.95%	73.88%	1.60054e-4	1.65123e-4	2.52842e-4
15	10	78.27%	73.20%	75.95%	1.41446e-4	2.05640e-4	2.02193e-4

To be continued

Continued

Window length	ONoHN	Hit Rate			MSE		
		Training	Validation	Testing	Training	Validation	Testing
16	9	79.52%	75.26%	73.88%	1.07973e-4	2.79525e-4	3.92056e-4
17	3	77.50%	74.57%	74.91%	1.40879e-4	2.54352e-4	2.79682e-4
18	3	77.70%	72.85%	73.54%	1.71704e-4	2.03129e-4	2.21761e-4
19	6	77.09%	73.20%	74.23%	1.62638e-4	2.00595e-4	2.07050e-4
20	8	78.64%	75.26%	74.23%	1.21622e-4	2.79563e-4	2.12462e-4
21	7	77.43%	75.95%	74.23%	1.32440e-4	3.08882e-4	2.63189e-4
22	6	75.99%	74.57%	73.54%	1.44551e-4	2.72193e-4	3.24253e-4

The hit rate of the lowest price gets the maximum value 75.95% when the window length is 15 days and gets the minimum value 69.42% when the window length is 1 day in model two.

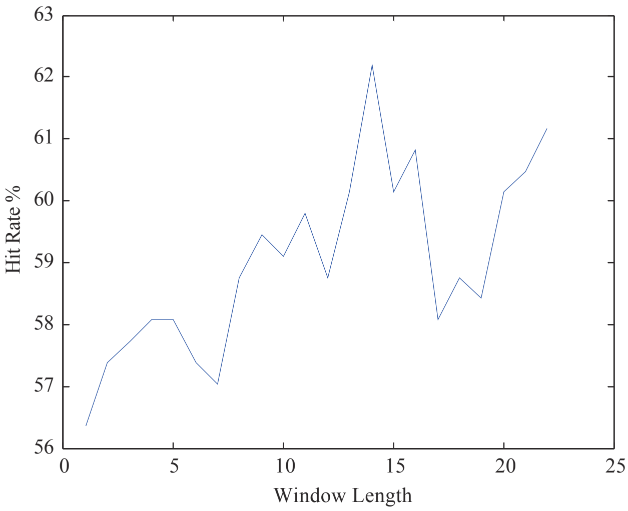

Figure 5
Variation of Hit Rate of the Closing Price With the Window Length of Model Two

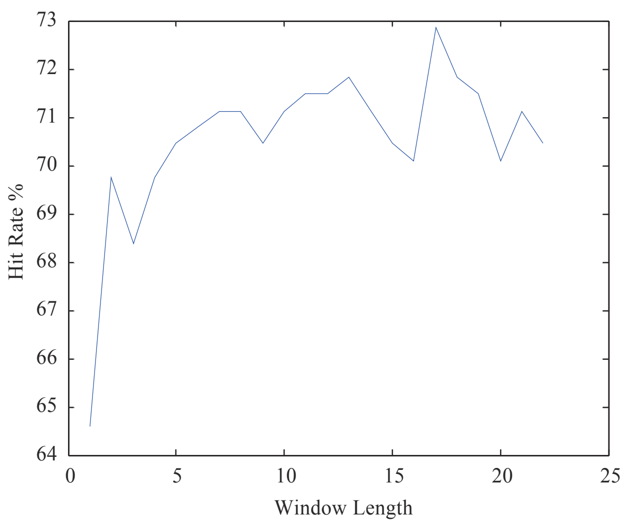

Figure 6
Variation of Hit Rate of The Highest Price With the Window Length of Model Two

The relationships of the hit rate of the closing price, the highest price, the lowest price and the window length in model two are shown in figures 5-7 respectively.

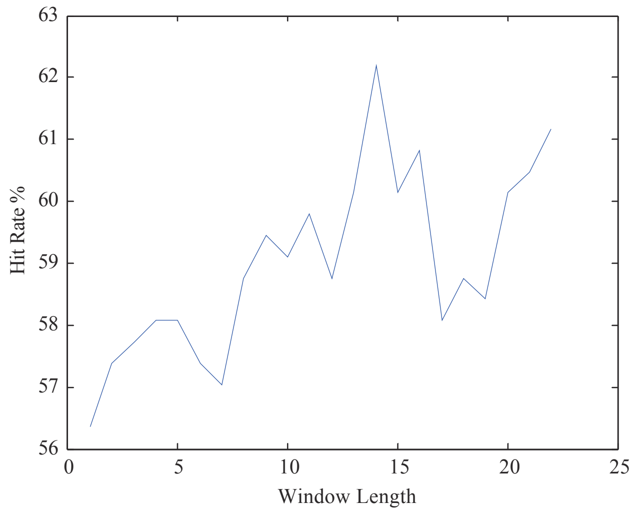

Figure 7
Variation of Hit Rate of the Lowest Price With the Window Length of Model Two

5.3 Experimenting Model Three: Multiple Input Multiple Output Sets Model

As defined by equation (12) in model three, the input data the logarithmic returns of the highest, lowest and closing prices in parallel and the output data are the logarithmic return of the highest, lowest, and closing prices simultaneously. In view of the window length as an input parameter to the FNN, a specific form of model three for one-step prediction can be expressed as

$$\left\{ D(FX(t),m) \Rightarrow D(I.LX(t),m) = \begin{pmatrix} I.LX_h(t) \\ I.LX_l(t) \\ I.LX_c(t) \end{pmatrix} \right\} \Rightarrow FNN \tag{20}$$

$$\Rightarrow \{LX_h(t+1), LX_l(t+1), LX_c(t+1)\}$$

The m refers to the window length. Here we define the variable HR_{com} as the comprehensive hit rate and the formula of it is as follow:

$$HR\_COM = \left(HR_{highest} + HR_{lowest} + 2 \times HR_{close}\right)/4 \quad (21)$$

HR_{com} is regarded as the typical hit rate of a trading day. In view of forecasting the trend of the latest trading days, we focus on the hit rate of the testing set (out of sample test). Table 13 shows the hit rate of the testing set of the closing, highest and lowest prices and the comprehensive hit rate in model three.

Table 13
Hit Rates of Model Three With Different Window Length

Window length	ONoHN	Hit Rate of the testing set			
		Closing price	Highest price	Lowest price	Comprehensive hit rate
1	10	54.30%	65.64%	67.01%	60.31%
2	8	57.39%	69.42%	70.45%	63.66%
3	9	56.01%	67.01%	72.51%	62.89%
4	10	56.01%	67.70%	74.23%	63.49%
5	10	57.39%	68.73%	72.16%	63.92%
6	5	57.04%	67.70%	74.57%	64.09%
7	10	58.76%	67.70%	71.48%	64.18%
8	9	58.08%	70.10%	73.20%	64.86%
9	9	61.86%	69.76%	71.48%	66.24%
10	6	56.70%	68.04%	71.82%	63.32%
11	10	60.82%	66.32%	71.82%	64.95%
12	10	59.45%	67.70%	72.16%	64.69%
13	6	58.76%	70.45%	74.23%	65.55%
14	6	59.79%	69.76%	71.13%	65.12%
15	4	61.86%	66.32%	70.45%	65.12%
16	10	61.86%	65.89%	72.16%	65.44%
17	5	58.42%	68.73%	72.85%	64.61%
18	8	60.82%	69.07%	71.13%	65.46%
19	3	62.89%	68.38%	73.20%	66.84%
20	7	58.76%	67.35%	71.13%	64.00%
21	3	58.42%	64.95%	71.48%	63.32%
22	7	60.82%	67.01%	72.16%	65.20%

The comprehensive hit rate gets the maximum value 66.84% when the window length is 19 days and gets the minimum value 60.31% when the window length is 1 day in model three.

Figure 8 shows these hit rates with different window length. We can see from the figure that the close price is least predictable and the lowest price is the most predictable, with the highest price the next.

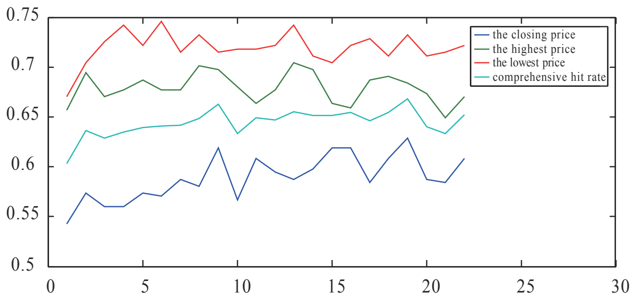

Figure 8
Variation of the Hit Rates With the Window Length of Model Three

CONCLUSION

In this paper we have reported on a systematic study for developing ANN-based models to predict the short-term trends of the CSI 300 index – the Chinese benchmark stock index – using the current information set from the three component prices – the highest, lowest and closing prices – of the stock index. Three models are designed and tested: 1) single input single output sets model, 2) multiple input single output sets model, and 3) multiple input multiple output sets model. From the experiment results (section 6), two important conclusions can be drawn: 1) the stock index CSI 300 is probabilistically predictable, as all the three models produced the hit rate of prediction significantly higher 50%; 2) the second model consisting of three FNN's each with multiple input single output sets produced remarkably high hit rates: 72% on the highest price, 75% on the lowest price, and 62% on the closing price. Obviously, this kind of hit rates is already very useful in terms of economic profitability.

REFERENCES

Barbulescu, A., & Bautu, E. (2012). A ybrid approach for modeling financial time series. *International Arab Journal of Information Technology, 9*(4), 327-335.

Bekiros, S. D. (2010). Fuzzy adaptive decision-making for boundedly rational traders in speculative stock market. *European Journal of Operational Research, 202*(1), 285-293.

Bodyanskiy, Y., & Popov, S. (2006). Neural network approach to forecasting of quasiperiodic financial time series. *European Journal of Operational Research*, *175*(3), 1357-1366.

Brooks, C. (2002). Introductory Econometrics for Finance. Cambridge, UK: Cambridge University Press.

Chavarnakul, T., & Enke, D. (2008). Intelligent technical analysis based equivolume charting for stock trading using neural network. *Expert Systems With Applications, 34*(2), 1004-1017.

Chen, W. H., Shi, J. Y., & Wu, S. S. (2006). Comparison of support-vector machines and back propagation neural networks in forecasting the six major Asian stock markets. *International Electronic Finance, 1*(1), 49-67.

Chen, A. S, Leung, M. T., & Daouk, H. (2003). Application of neural networks to an emerging financial market: Forecasting and trading the Taiwan Stock Index. *Computers & Operations Research, 30*(6), 901-923.

Ding, S. F., Su, C. Y., & Yu, J. Z. (2011). An optimizing BP neural network algorithm based on genetic algorithm. *Artificial Intelligence Review, 36*(2), 153-162.

Freitas, P. S. A., & Rodrigues, A. J. L. (2006). Model combination in neural-based forecasting. *European Journal of Operational Research, 173*(3), 801-814.

Granger CWJ. (1989). Invited review combining forecasts—twenty years later. *Journal of Forecasting 8*(3),167-173.

Jacquier, E., Polson, N., & Rossi, P. (2004). Bayesian analysis of stochastic volatility models with fat-tails and correlated errors. *Journal of Econometrics, 122*(1), 185-212.

Kanas, A. (2003). Non-linear forecasts of stock returns. *Journal of Forecasting, 22*(4), 299-315.

Koulouriotis, D. E., Diakoulakis, I. E., Emiris, D. M., & Zopounidis, C. D. (2005). Development of dynamic cognitive networks as complex systems approximators: Validation in financial time series. *Applied Soft Computing Journal, 5*(2), 157-179.

Kwon, Y. K., & Moon, Byung-Ro. (2007). A hybrid neurogenetic approach for stock forecasting. *IEEE Transactions on Neural Networks, 18*(3), 851-864.

Li, T., Li, Q., Zhu, S., & Ogihara, M. (2002). A survey on wavelet applications in data mining. *SIGKDD Explorations 2002,4*(2), 49-68.

Melin, P., Mancilla, A., Lopez, M. (2007). A hybird modular neural network architecture with fuzzy Sugeno integration for time series forecasting. *Applied Soft Computing, 7*(4), 1217-1226.

Moreno, D. & Olmeda, I. (2007). Is the predictability of emerging and developed stock markets really exploitable?. *European Journal of Operational Research, 182*(1), 436-454.

Pai, P. F., & Lin, C. S. (2005). A hybrid ARIMA and support vector machines model in stock price forecasting. *OMEGA-International Journal of Management Science, 33*(6), 497-505.

Pan, H., Haidar, I., & Kulkarni, S. (2009). Daily prediction of short-term trends of crude oil prices using neural networks exploiting multimarket dynamics. *Frontiers of Computer Science in China, 3*(2), 177-191.

Pan, H., Titakaratne, C., & Yearwood, J. (2005). Predicting Australian stock index using neural networks exploiting dynamical swings and intermarket influences. *Journal of Research and Practice in Information Technology, 37*(1), 43-55.

Refenes, A. N., Zapranis, A., & Francis, G. (1994). Stock performance modeling using neural networks: A comparative study with regression models. *Neural Networks, 7*(2), 375-388.

Ruxanda, G. (2010). Learning perceptron neural network with backpropagation algorithm. *Economic Computation and Economic Cybernetics Studies and Research, 44*(4), 37-54.

Wang, J. J., Wang, J. Z. & Zhang, Z. (2012). Stock index forecasting based on a hybrid model. *Omega-International Journal of Management Science, 40*(6), 758-766.

Yaser S., & Atiya A. (1996). Introduction to financial forecasting. *Applied Intelligence*, *6*(3), 205-213.

Zhang, G., & Berardi, V. (2001). Time series forecasting with neural network ensembles: An application for exchange rate prediction. *Journal of the Operational Research Society, 52*(6), 652-664.