

Modeling the Financial Market with Multiple Prices

Zhao Danting, Tan Tingfeng, Jin Zhongwei, Xuan Huichao, Li Xian and Wang Qing-Guo

Department of Electrical and Computer Engineering, National University of Singapore, Singapore, 117576.

sendy87716@163.com; a0123784@nus.edu.sg; jinzhongwei@nus.edu.sg; xuanhuichao@nus.edu.sg;
lixian@nus.edu.sg; elewqg@nus.edu.sg

ABSTRACT

An effective financial market trading decision is usually dependent on superior forecasting. Forex market as the largest financial market is chosen in this study. The main objective of this paper is to explore the forecasting performance of the proposed multiple-price model which integrates close, low and high price information, by using Artificial Neural Network (ANN). The architecture of the network and the related algorithms are described. The effects due to different choices of preprocessing methods, combinations of input variables and different time intervals of forecasting are examined. By using the best multiple-price model, trading strategies with high and low prices are developed as well. The results have shown that in terms of both absolute values and trends of the prices, forecasting accuracy has improved compared with single price model. This is especially so for low and high prices whose directional accuracies are much higher. The trading performance is also proven to have much better total return than buy & hold strategy, and trading with high price has the best overall performance considering both return and risk.

Keywords: *Foreign Exchange Rate, Artificial Neural Network, Trading, Technical Indicator, Fundamental Indicator*

1 Introduction

Forecasting future value of financial market time series has been extensively explored and developed. Foreign Exchange market as the largest and most liquid financial market [1] is studied in this paper. Studies making use of Hurst Exponent [2], a measure of the bias in fractional Brownian motion, have indicated that most of the financial markets are not random walk and are not highly efficient, thus forecasting of financial indices is possible. Technical and fundamental approaches are two of the most popular disciplines in financial market forecasting. As a result, in this paper, a combination of fundamental and technical factors will be taking into consideration in financial market forecasting.

Artificial Neural Network (ANN) known as universal function approximators can map any nonlinear function [3]. It is a non-parametric model which is data driven. So, no parametric modeling assumptions need to be made and the entire data set can be utilized. Furthermore, neural networks are less sensitive to error term assumptions and they can tolerate noise, chaotic components, and heavy tails which are common characteristics in financial market, better than most other methods [4].

With regards to existing studies on forex market forecasting using ANN, only the closing currency exchange rate is used. Apart from the closing price, the high and low prices can play important roles in

making investment decisions. More importantly, High, Low and Close prices should all have effects on one and another. Forecasting accuracy might improve in the multiple-price model. Therefore, the major objective of this paper is to explore the accuracy and trading performance of financial forecasting using models with integration of multiple prices and other inputs. Also, different choices of preprocessing methods on the raw data, combinations of input variables with technical and fundamental indicators, as well as different time intervals of forecasting may affect forecasting performance of the multiple-price model and will be investigated in this paper.

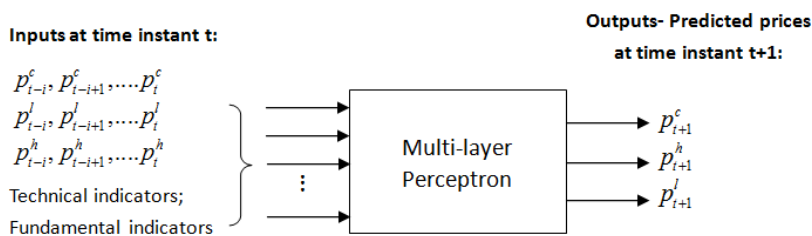
2 Artificial Neural Network

The Back propagation (BP) multilayer feed forward neural network is the most widely used type of neural networks in financial time series forecasting [5]. It has been shown that if sufficient number of hidden neurons is used, standard Back propagation Neural network using an arbitrary transfer function can approximate almost any measurable function in a very precise manner [6].

2.1 The Proposed Multiple-price Model (MPM) Structure

The standard ANN structure for financial time series forecasting only has one output which represents the price when the market closes, and it is called Single-price Model (SPM) in this paper. On the other hand, in the proposed Multiple-price Model (MPM), there will be three outputs which are for close, low and high prices at time instant $t+1$ respectively. Figure 1 demonstrates the structure of the MPM.

Fig. 1. Multiple-price model structure



2.2 Multilayer Perceptron Architecture

The graphical representation of an MLP with one hidden layer is shown in Figure 2. There are two types of links in the network, Synaptic link and Activation link [7]. Synaptic links connect the neurons from the

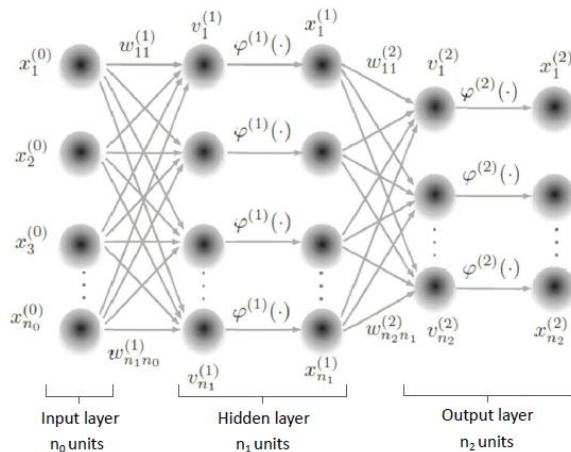


Fig. 2. Three layer MLP architecture

previous layer to the neurons in the next layer, associated with the weight $w_{ji}^{(l)}$ - weight from unit i in layer $(l-1)$ to unit j in layer (l) .

Input signal from unit j in layer (l) to neurons in layer $(l+1)$ is denoted as $x_j^{(l)}$, and it can also be viewed as the output signal of unit j in layer (l) . The bias terms at layer (l) which are always equal to one are noted as $x_0^{(l)}$. The input signals passing through the synaptic links will then be summed together with their associated weights at the junction. The activation link connects the summing junction and the respective output neuron. Therefore, the outputs from the hidden layer and output neurons are:

$$x_j^{(1)} = \varphi^{(1)}\left(\sum_{i=0}^{n_0} w_{ji}^{(1)} x_i^{(0)}\right), \text{ where } j = 1, 2, \dots, n_1 \tag{1}$$

$$x_j^{(2)} = \varphi^{(2)}\left(\sum_{i=0}^{n_1} w_{ji}^{(2)} x_i^{(1)}\right), \text{ where } j = 1 \text{ for Single-price Model,} \tag{2}$$

and } j = 1, 2, 3 \text{ for Multiple-price Model.}

2.3 Data Division and Normalization

In Neural network applications, it is a common practice to split all the available sample data into training, validation and testing sets. The training set is used to adjust the weights of neural network model during the training process. The validation set is used in the training process to monitor the performance of the network after each update of the weights and stops the training when the performance gets worse, which is a way to avoid the over-fitting problem. The testing set is used to evaluate the generalization ability of the trained network and it is evaluated via the performance measures. According to the work of Yao and Tan [8], the training, validation and testing data sets shall be set 70%, 20% and 10% of the collected sample data respectively. In this project, this data division method will be adopted.

Data normalization is one of the most important and basic data preprocessing steps when use ANN. Different inputs could have different ranges. The transfer function in the neurons might become saturated if it is presented with a very large input. To avoid this situation, the input and target data has to be normalized. The normalization process will map all the input and target data into $[-1 \ 1]$ using the maximum and minimum of the particular data set. In other words, each of the data set will be normalized independently of the other sets.

2.4 Training

The ANN is trained with a training set of the form as follows:

$$S = \{(X_1, Y_1), (X_2, Y_2), (X_3, Y_3), \dots, (X_N, Y_N)\}, \tag{3}$$

where } X_k = [x_1, x_2, \dots, x_m]^T, Y_k = [p] \text{ or } [p^c, p^h, p^l]

X_k is the input data vector, which consists of m elements. Y_k is the desired output vector when X_k is applied as the input, and it has one element for Single-price Model and three elements for Multiple-price Model. N is the total number of data pairs in the training set. During the training, each data pair is presented to the network in turn by setting $X^{(0)}(k) = X_k$, where $X^{(0)}(k) = [x_1^{(0)}(k), \dots, x_m^{(0)}(k)]^T$ is the input to the MLP network. So the number of input layer neuron n_0 equals to m . The energy function for network at iteration k is defined as:

$$E = \sum_{k=1}^{k=N} [E(k)] = \sum_{k=1}^{k=N} \left[\frac{1}{2} \sum_{j=1}^{n_L} (y_j - x_j^{(L)}(k))^2 \right], \tag{4}$$

where $L = 2$ in the case of 3 layer MLP

The aim of the training process is to minimize the energy function E at the output layer by adjusting the weights systematically. Here we will adopt the batch training mode, which will only update the weights after all the N data pairs have been presented to the network, and N iterations is called 1 epoch. The energy function depends on only the weights $w_{ji}^{(1)}$ and $w_{ji}^{(2)}$. The standard Back propagation algorithm by Rumelhart et al. uses the steepest gradient descent technique to minimize the energy function E at each epoch. The weight adjust rule is defined as:

$$\Delta w_{ji}^l = -\eta \left(\frac{\partial E}{\partial w_{ji}^l} \right) \quad (5)$$

where $\eta \in (0,1]$ is the learning rate. The learning rate is critical, because it determines how fast the function converges. The training process repeats, epoch by epoch. The training process normally will stop if one of the following criteria is met: 1) the maximum number of training epoch has been reached; 2) the error E has dropped to some minimum value; or 3) the performance for the validation set gets worse for a certain number of consecutive updates.

2.5 Performance Measure

With this network, predictions are made with the input data in the test set. The network's performance is measured by Root Mean Square Error (RMSE) and Directional Accuracy (DA) methods. Both RMSE and DA will be used to measure the prediction performance for Close, Low and High prices separately. Taking Close price as an example (Low and High prices' RMSE and DA are calculated similarly), the formulas of RMSE and DA for Close price are as shown in equation (6) and (7). In the equations, N is the number of iterations, p_{t+1}^c is

the actual price at day $t+1$. \hat{p}_{t+1}^c is the predicted price at day $t+1$, i.e. the output from the network.

$$RMSE^c = \sqrt{\frac{1}{N} \sum_{i=1}^N (p_{t+1}^c - \hat{p}_{t+1}^c)^2} \quad (6)$$

$$DA^c = \frac{1}{N} \sum_{i=1}^N \alpha_i, \text{ where } \alpha_i = \begin{cases} 1 & \text{if } (p_{t+1}^c - p_t^c) \times (\hat{p}_{t+1}^c - p_t^c) > 0 \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

RMSE is calculated based on the difference of the actual price and the predicted price. It shall be used only when the aim of the network performance is to minimize the squared error. In the financial context, it is essential to predict the trend of the currency rate too. Hence in this study, both the RMSE and directional accuracy will be used to evaluate the network performance. Directional Accuracy is the percentage of correct predictions in terms of direction.

3 Simulation Study with Multiple-price Model

The fundamental objective of the simulation is to check the capability of the Multiple-price model compared to the Single-price model. On top of this fundamental objective, the model will be varied from different aspects, namely number of hidden layer neurons, currency pairs, preprocessing methods, time interval of forecasting and most importantly different input variable combinations. By comparing their performances, the best model will finally be chosen.

3.1 Overview of Simulation Study

Bearing in mind the above objectives, all simulations will be carried out for both Single-price and Multiple-price models, and the number of hidden layer neurons n_1 is varied from 4 to 20 with step of 4. The reason for simulating each model with different numbers of hidden layer neuron is to avoid the bias due to the fact that ANN with a certain number of hidden layer neuron normally performs differently when the input changes.

The number of input neurons n_0 , depends on different combination of inputs to be applied to the MLP. In each different model, the basic input variables are the multiple time-delayed close, low and high prices. In particular, they include 5 time-delayed close price data at day t $\{p_t^c, p_{t-1}^c, p_{t-2}^c, p_{t-3}^c, p_{t-4}^c\}$, and similar format for low and high price inputs. These input data will be used as part of the input variables to predict close, low and high prices at $t+1$. We will investigate the usefulness of the multiple-price model compared to the conventional single-price model where the number of output neurons n_2 is set as 1 for the single price model, and 3 for the multiple-price model.

Other than the variations of SPM versus MPM and number of hidden layer neurons, in each of the simulations, the model will be modified in certain aspects.

First, different currency pairs will be simulated. This is to select the most predictable currency pair to be studied in this project. Three currency pairs are studied in this first simulation, USD/SGD, AUD/USD and GBP/USD. In the first simulation, the most basic forecasting model with purely time-delayed inputs using first differencing and normalization preprocessing methods will be adopted.

Second, the simulation model will vary in terms of the preprocessing method after choosing the best currency pair. Thus, simulation model with first differencing ($p_i - p_{i-1}$), natural Log transformation ($\log(p_i)$), or ratio transformation (p_i / p_{i-1}), or natural Log on ratio transformation ($\log(p_i / p_{i-1})$), or price-1 ($p_i - 1$) transformation or no transformation will be carried out. Next, the simulation model is varied in terms of the input combinations. Instead of the purely time-delayed inputs, different combinations of technical indicator and fundamental indicator variables will be fed into the network. Lastly, due to the fact that the daily data might contain relatively high level of noise compared to the weekly data, the time interval of forecasting will be varied from daily to weekly in the simulation to investigate network performance.

3.2 Simulation Results and Discussions

The simulation results for different simulation tests are presented in Table 1 to 5.

Table 1. GBP/USD Averaged Results with Different Currency Pairs for Daily Time-delayed Inputs with First Differencing Preprocessing

Currency Pair	Close Low High			Average	Close Low High			Average
	Testing RMSE				Testing DA			
<i>Multiple-Price Model</i>								
USD/SGD	0.0125	0.0119	0.0273	0.0172	0.5158	0.6188	0.6194	0.5847
AUD/USD	0.0083	0.0061	0.0051	0.0065	0.5018	0.6314	0.6590	0.5974
GBP/USD	0.0100	0.0072	0.0065	0.0079	0.5223	0.6819	0.6802	0.6281
<i>Single-Price Model</i>								
USD/SGD	0.0053	0.0046	0.0366	0.0155	0.5109	0.5316	0.5297	0.5241
AUD/USD	0.0082	0.0079	0.0064	0.0075	0.5271	0.4952	0.5591	0.5271
GBP/USD	0.0101	0.0096	0.0088	0.0095	0.5214	0.5545	0.5410	0.5390

Table 2. GBP/USD Averaged Results for Daily Time-delayed Inputs with Different Preprocessing Methods

Preprocessing Method	Testing RMSE				Testing DA			
	Close	Low	High	Average	Close	Low	High	Average
<i>Multiple-Price Model</i>								
First Differencing	0.0100	0.0072	0.0065	0.0079	0.5223	0.6819	0.6802	0.6281
Natural Log	0.0101	0.0071	0.0063	0.0078	0.5199	0.6850	0.6928	0.6325
Ratio	0.0101	0.0073	0.0065	0.0080	0.5151	0.6735	0.6615	0.6167
Logged Ratio	0.0102	0.0074	0.0066	0.0080	0.5253	0.6632	0.6729	0.6205
Price-1	0.0101	0.0071	0.0064	0.0079	0.5271	0.6837	0.6928	0.6345
No Transformation	0.0100	0.0071	0.0063	0.0078	0.5199	0.6813	0.6873	0.6295
<i>Single-Price Model</i>								
First Differencing	0.0101	0.0096	0.0088	0.0095	0.5214	0.5545	0.5410	0.5390
Natural Log	0.0102	0.0098	0.0088	0.0096	0.5227	0.540	0.5541	0.5407
Ratio	0.0101	0.0097	0.0088	0.0095	0.5263	0.5311	0.5450	0.5341
Logged Ratio	0.0101	0.0097	0.0089	0.0095	0.4592	0.4121	0.4060	0.4258
Price-1	0.0102	0.0088	0.0098	0.0096	0.5396	0.5456	0.5402	0.5418
No Transformation	0.0102	0.0098	0.0088	0.0096	0.5354	0.5396	0.5474	0.5408

Table 3. GBP/USD Averaged Results for Different Technical Indicators and Daily Time-delayed Input with Natural Log Preprocessing

Technical Indicators	Testing RMSE				Testing DA			
	Close	Low	High	Average	Close	Low	High	Average
<i>Multiple-Price Model</i>								
None*	0.0101	0.0071	0.0063	0.0078	0.5199	0.6850	0.6928	0.6325
Moving Averages	0.0101	0.0071	0.0064	0.0079	0.5187	0.6711	0.6934	0.6277
Natural Log MAs	0.0102	0.0071	0.0064	0.0079	0.5246	0.6869	0.7018	0.6378
MA+MACD+RSI+SO	0.0102	0.0072	0.0066	0.0080	0.5185	0.6916	0.6880	0.6327
<i>Single-Price Model</i>								
None	0.0102	0.0098	0.0088	0.0096	0.5227	0.5400	0.5541	0.5407
Moving Averages	0.0102	0.0098	0.0089	0.0096	0.5166	0.5359	0.5565	0.5363
Natural Log MAs	0.0102	0.0097	0.0089	0.0096	0.5239	0.5305	0.5504	0.5349
MA+MACD+RSI+SO	0.0104	0.0087	0.0079	0.0090	0.5118	0.6194	0.6199	0.5837

Notes: * "None" represents the network model with purely time-delayed inputs

Table 4. GBP/USD Averaged Results for Different Fundamental Indicators and Daily Time-delayed Input with Natural Log Preprocessing

Fundamental Indicators	Testing RMSE				Testing DA			
	Close	Low	High	Average	Close	Low	High	Average
<i>Multiple-Price Model</i>								
None*	0.0101	0.0071	0.0063	0.0078	0.5199	0.6850	0.6928	0.6325
Stock Exchange Index	0.0100	0.0070	0.0063	0.0078	0.5211	0.6873	0.7012	0.6365
Gold Price	0.0105	0.0093	0.0071	0.0090	0.5163	0.6723	0.6934	0.6273
Interest Rate	0.0101	0.0070	0.0064	0.0078	0.5211	0.6958	0.6904	0.6358
All Three Indicators**	0.0107	0.0093	0.0079	0.0093	0.5217	0.6862	0.6970	0.6349
All without Gold Price	0.0102	0.0072	0.0065	0.0080	0.5257	0.6873	0.6987	0.6372
<i>Single-Price Model</i>								
None	0.0102	0.0098	0.0088	0.0096	0.5227	0.5400	0.5541	0.5407
Stock Exchange Index	0.0102	0.0097	0.0089	0.0096	0.5281	0.5486	0.5547	0.5438
Gold Price	0.0123	0.0131	0.0108	0.0120	0.5366	0.5390	0.5492	0.5416
Interest Rate	0.0104	0.0098	0.0091	0.0098	0.5390	0.5517	0.5511	0.5472
All Three Indicators	0.0125	0.0139	0.0135	0.0133	0.5299	0.5619	0.5680	0.5533

Notes: * "None" represents the network model with purely time-delayed inputs

** "All Three Indicators" represents the network model with stock exchange index, gold price, interest rate and time-delayed inputs.

Table 5. GBP/USD Averaged Results for Daily Time-delayed Input with Natural Log Preprocessing Using Different Time Interval of Forecasting

Time Interval	Testing RMSE			Average	Testing DA			Average
	Close	Low	High		Close	Low	High	
<i>Multiple-Price Model</i>								
None*	0.0101	0.0071	0.0063	0.0078	0.5199	0.6850	0.6928	0.6325
Weekly Forecasting**	0.0139	0.0115	0.0088	0.0114	0.6243	0.7091	0.6667	0.6667
Daily with Same Amount	0.0093	0.0052	0.0063	0.0070	0.5394	0.7333	0.6611	0.6446
<i>Single-Price Model</i>								
None	0.0102	0.0098	0.0088	0.0096	0.5227	0.5400	0.5541	0.5407
Weekly Forecasting	0.0222	0.0206	0.0181	0.0203	0.6500	0.6038	0.6188	0.6242
Daily with Same Amount	0.0097	0.0077	0.0069	0.0081	0.5243	0.5375	0.5625	0.5414

Notes: * "None" represents the network model originally carried out with data from Jan 1998 to Nov 2010 with daily forecasting.

** "Weekly Forecasting" represents the network model with data from Jan 1998 to Nov 2010 with weekly forecasting.

*** "Daily with Same Amount of Data" represents the network model with data from Apr 2008 to Nov 2010 with daily forecasting, which has same number of data sets as model **.

Referring to Table 1, three currency pairs' performances are examined. The USD/SGD has the worst RMSE and DA performances in both MPM and SPM. One possible reason is that SGD is not an actively traded currency in the forex market, the higher uncertainties make it more unpredictable, hence the simulation results are not as good as AUD/USD and GBP/USD, which are two of the most actively traded currency pairs. Between AUD/USD and GBP/USD, the later has higher low and high price directional accuracy. Since trend forecasting is crucial in forex trading and to better assess the predictive ability of the neural network model, GBP/USD is chosen for the rest of the simulations. Next, the effects of the MPM and SPM are investigated. By looking into each of the simulation test results presented in the above tables, we find that the forecasting performance for close price is similar between MPM and SPM in terms of RMSE and DA. However, forecasting performance is largely improved by using MPM. By considering the relatively better performance models in the above results, the average RMSE values for low and high prices of MPM are around 0.0072 and 0.0064, which perform 25.8% and 27.3% better than the relevant average RMSE values of SPM, which are around 0.0097 and 0.0088. The average DA values for low and high prices of MPM are above 0.68, which are 30.8% higher than SPM with values around 0.52. Therefore, it is proven that the proposed Multiple-price model performs better than Single-price model in terms of both the absolute value (RMSE) and direction (DA) prediction.

By looking at the different preprocessing methods, the RMSE values of all five simulations are very similar. Thus we can infer that the choice of preprocessing method has minimal effect on the prediction capability in terms of the absolute value. However, preprocessing method has significant effect on the prediction capability in terms of the trend, which is reflected by DA, the average DA of Close, Low and High prices in the MPM using Price-1 transformation is 0.6345, which is the highest among all the preprocessing methods. Moreover, Price-1, Natural Log and No Transformation methods have better performance than the other three methods. Our motivation to use the three methods, First Differencing, Ratio and Logged Ratio preprocessing is to highlight the relationship between two consecutive days' prices, by either feeding the price difference or ratio into the network instead of using price directly. This might, however, provide less price information than using no transformation or natural Log transformation. Furthermore, 5 time-delayed transformed price data are used as the NN input for all different models. Thus, in the price-1, natural Log or no transformation models, the relationship between consecutive days' prices can be easily learned by the network using these time-delayed data. Therefore, the price-1, natural Log and no transformation models which provide information for both single day price and consecutive days' price

relationship, are expected to have better performance than the other three preprocessing transformation techniques.

The effects of using technical indicators, which include moving averages (MA10, MA20, MA30, MA60, MA120), moving average convergence divergence (MACD), relative strength index with period of 14 (RSI_{14}) and stochastic oscillator with period 14 (SO_{14}), are examined as well. By looking into the MPM results section in Table 3, we can find out the effect on MPM prediction performance after adding technical indicator inputs compared to the purely time-delayed model. In terms of RMSE, the performances are very similar for all four input combinations. Thus, different type of input variable combinations has little effect on the absolute value prediction accuracy. In terms of DA, the model with natural Log moving averages input has average DA value of 0.6378 which is 0.84% higher than the purely time-delayed inputs model with average DA value, 0.6325. The model with all technical indicators also performs slightly better than the purely time-delayed input model. Thus we can conclude that after adding the technical indicator moving averages, the network's direction prediction performance improves.

Moreover, the effects on MPM prediction performance after adding fundamental indicator inputs compared to the purely time-delayed model are examined. In terms of RMSE, the performances are very similar for input combinations without Gold price. Thus, adding fundamental indicator – Stock Exchange Index and Interest Rate, has little effect on the absolute value prediction accuracy. However, the model with gold price as well as all three indicators inputs performs exceptionally worse, which indicates gold price is not a good input for GBP/USD exchange rate forecasting. In terms of DA, the model with stock exchange rate has average DA value of 0.6365 which is 0.63% higher than the purely time-delayed inputs model with average DA value, 0.6325. Without using gold price, the models with fundamental indicators perform slightly better than the purely time-delayed input model. Thus, we can conclude that after adding fundamental indicators such as stock exchange index and interest rate, the network's direction prediction performance improves.

Finally, by testing over different time intervals and time lengths, it is found that the weekly forecasting model performs 62.9% and 150.6% worse than the daily model with same amount data which has RMSE value of 0.0070 and 0.0081 for MPM and SPM correspondingly. Therefore, the weekly forecasting model performs much worse than the daily forecasting model in terms of absolute value, i.e. RMSE. However, the weekly forecasting model improves the network performance on close price prediction's directional accuracy significantly, which might be because there is less noise in the weekly data than the daily data. When comparing daily model with different total time length, for both RMSE and DA, the performance of the model with data over shorter time length is improved, which indicates the dated data may contain less information of the current market exchange rate compared to more recent data.

3.3 Study of Multiple-price Model

In order to investigate further in the high directional accuracy in low and high prices prediction in the MPM, this simulation is carried out for a multiple-price model without close price time series. It means the historical low and high prices information are fed into the network, and the output will be the predicted high and low prices in the next time period. The simulation result is shown in Table 6.

Table 6. GBP/USD High and Low Prices' Averaged Results of Multiple Price Model with and without Close Price Time Series Information

Multiple Price Model	Low High		Low High		
	Testing RMSE	Average	Testing DA	Average	
With Three Prices' Information	0.0071	0.0063	0.0067	0.685	0.6928 0.6889
With Only Low and High Prices' Information	0.0090	0.0082	0.0086	0.5446	0.5687 0.5566

After excluding close price time series in the multiple-price model, the prediction results of the low and high prices in terms of both RMSE and DA values are much worse than the multiple-price model with all three prices. Moreover, without close price time series, the result of the multiple-price model becomes similar with the single-price model. Therefore, it implies the important role of close price time series information played in the forecasting of low and high prices, which means that the high and low prices are highly related with the close price information.

4 Testing for Model Performance

Stemming from the fact that the multiple-price model shows a high degree of directional accuracy in the prediction of the movement of the high and low GBP/USD exchange rates on daily or weekly basis, a trading strategy utilizing the high and low prices is desired. Moreover, after examining the different variations on the neural network performance in the previous sections, the multiple-price model with 5 natural logged daily time-delayed prices and natural logged moving averages inputs with 20 hidden layer neurons has relatively higher training directional accuracy than other models. Therefore, this model with the highest training performance is fed with the testing data and whose output is used for the trading.

In this simulation, a normal ask/bid spread of 3 pips will be used. It means whenever there is a “buy and sell” or “sell and buy” transaction carried out, the return will be deducted by the transaction cost 3 pips, which is 0.0003.

On day t , BUY signal is triggered if (day $t+1$)'s predicted high price is greater than today's close price for more than 60 pips. The choice of 60 pips is due to the absolute value accuracy of the multiple-price model is around 0.0060 which is 60 pips, therefore, a signal larger than 60 pips might give us more meaningful BUY signal.

On day $t+1$, SELL signal is triggered if (day $t+1$)'s actual high price is higher than (day $t+1$)'s predicted high price. Otherwise, a STOP signal will be triggered if (day $t+1$)'s actual low price is more than 50 pips lower than the lowest of 3 previous days' low prices, and then the position will be closed out by selling all GBP at the STOP price. Mathematically, STOP price is calculated as:

$$\text{StopPrice} = \min(p_t^l, p_{t-1}^l, p_{t-2}^l) - 50 \text{ pips} \quad (8)$$

The stop signal check rule is to ensure no big loss is incurred when the market price falls dramatically within a day, which may often be caused by big disasters or big events such as financial crisis. Whenever a SELL decision is triggered, the return will also be deducted by the bid/ask spread or transaction cost which is 3 pips. When both the SELL and STOP signals are not triggered on day $t+1$, the position will be hold and carried over to the next day.

The decisions will be continuously carried out on every day throughout the whole testing period, which consists of 331 days in this study's simulation. Similar trading decisions are carried out for trading strategy

using low price. Their trading results together with using the normal buy & hold trading strategy is shown in Table 7.

Table 7 Trading Performance of Three Trading Strategy Measured by Trading Performance Indicators

Trading Performance Indicators	Trading with High Price	Trading with Low Price	Buy & Hold Strategy
Total trades	173	141	1
% Positive trades	71.10%	74.47%	-
Best trade*	0.0345	0.0641	0.0292
Worst trade*	(0.0830)	(0.1143)	(0.0316)
Max drawdown*	(0.2037)	(0.2793)	(0.2093)
Mean return per trade*	0.00120	0.00097	(0.00005)
Total Return	0.2062	0.1371	(0.0162)
Annualized Return	0.2297	0.1522	(0.0177)
Standard Deviation	0.0187	0.0300	0.0101
Annualized Standard Deviation	0.0203	0.0325	0.0109
Sharpe ratio	11.23	4.63	(1.77)

Notes: * For Buy & Hold strategy, as only one trade is carried out (i.e. buy on the first day and sell on the last day), the indicators in the table are calculated for profit and loss of the balance on each day throughout the trading period.

Buy & Hold strategy has negative total return, which is obviously the worst strategy in terms of trading return. Trading with high price has slightly lower positive trade percentage, however; it has higher mean return as well as total return than trading with low price. More specifically, its annualized return is 50.9% higher than trading with low price strategy. Therefore, in terms of trading total return, trading with high price has the best performance.

Risk is another important factor for trading decisions. When we look at the range of return, which is simply best trade minus worst trade. Trading with low price has the highest range of return, while buy & hold strategy has the smallest. Moreover, as the standard deviation of returns is commonly used as a measure of investment risk, by comparing three strategies' standard deviation, we can find that trading with low price has the highest risk, which buy & hold has the lowest, which matches with the observations using return range.

Sharpe ratio is a return-risk measure, which is intuitive for a risk-adjusted performance measure. More precisely, it measures the reward, in terms of mean excess return, per unit of risk. For positive sharp ratio, the larger the value, the smaller the risk associated, which indicates, the trading strategy with high price has better risk-adjusted performance. For negative sharp ratio, increasing risk results in numerically larger value; however, one limitation about Sharpe ratio is that it is difficult to compare between positive and negative value. Here, after we examine returns, risk as well as risk-adjusted performance, it's reasonable to conclude that the trading strategy with high price has the best performance.

5 Conclusion

The major objective of the study is to examine whether the proposed multiple-price model has better prediction capability than the traditional single price model. And all simulations are carried out for both models to examine the effects. The simulation results have clearly shown that the multiple-price model has better performance than the single-price model in terms of both absolute error and directional accuracy based on the two performance measure indicators, RMSE and DA. Different variations on the proposed Multiple-price Model have been investigated. By utilizing the high forecasting performance in

low and high prices, trading strategies have also been carried out, and are proven to have better returns than buy & hold strategy.

ACKNOWLEDGMENT

The author thanks Professor Wang Qing-Guo from the Electrical and Computer Engineering department of National University of Singapore for his support and valuable advice. Without him, this project would not have been possible. The author also thanks Qin Qin from the Electrical and Computer Engineering department of National University of Singapore for his contributions to this project.

REFERENCES

- [1]. J.Yao, C.L. Tan, A case study on using neural networks to perform technical forecasting of forex, *Neurocomputing* 34 (2000) 79-98
- [2]. E.E. Peters, *Chaos and Order in the Capital markets: A New View of Cycles, Prices, and Market Volatility*, Wiley, New York, 1991.
- [3]. R.R. Trippi and E. Turban, eds. *Neural Networks in Finance and Investing: Using Artificial Intelligence to Improve Real- World Performance*, (Probus, Chicago, 1993).
- [4]. H. White, Learning in neural networks: A statistical perspective, *Neural Computat.* 4 (1989) 42.5-464.
- [5]. E.E. Peters, *Chaos and Order in the Capital markets: A New View of Cycles, Prices, and Market Volatility*, Wiley, New York, 1991.
- [6]. R. Hecht-Nielsen, *Neurocomputing*, (Addison Wesley, Menlo Park, CA, 1989).
- [7]. M. Adya, F. Collopy, How elective are neural networks at forecasting and prediction? A review and evaluation, *J. Forecasting*, 17 (1998) 481-495.
- [8]. J. T. Yao and C. L. Tan, A case study on using neural networks to perform technical forecasting of forex, *Neurocomputing* 34 (2000) 79–98.