Prediction Stock Market Exchange Prices for the Reserve Bank of Australia Using Auto-Regressive with eXogenous Input Neural Network Model

ALAA SHETA
Computers and Systems Department
Electronics Research Institute
Giza, EGYPT
asheta66@gmail.com

Abstract: Financial forecasting is one of the challenging field of research due to its wide commercial applications and high risks that could happen to courtiers economies if fail to deal with various changes in the market. Stock Market found to be a dynamic, non-linear and complex process in nature. It is usually affected by many factors such as economic conditions, bank exchange rate, investors' expectations, governmental events, and of course Wars in various areas of the world. The process of prediction/forecasting of money exchange rate help organizations, governments and business market to make decisions; it is essential for determining information about future markets. This paper introduces the basic idea of developing mathematical models for currency exchange rate using Artificia Neural Network (ANN) and Multiple Linear Regression (MLR) models. The data set used in the experiments collected during January 4, 2010 to December 31, 2013. Number of criterion were used to validate the developed model's performance. The NN model show promising results.

Key–Words: Prediction, Stock Market Exchange Prices, Reserve Bank of Australia, Artificia Neural Networks.

1 Introduction

Prediction in time series data are clearly beneficia in many areas like forecasting the weather, prices in the stock market, signal processing, Earthquake prediction, control engineering, computer communications and other areas. In the literature, several models have been proposed to give either short-term or long-term prediction, but what make these models supersede the others is the accuracy of their prediction and the minimum of required input data [1]. Accurate forecasting methods are crucial for portfolio management by commercial and investment banks. Assessing expected returns relative to risk presumes that portfolio strategists understand the distribution of returns. Financial expert can easily model the influenc of tangible assets to the market value, but not intangible asset like know-how and trademark. The financia time series models expressed by financia theories have been the basis for forecasting a series of data in the twentieth century. Studies focusing on forecasting the stock markets have been mostly preoccupied with forecasting volatilities. There have been few studies bringing models from other forecasting areas such as technology forecasting.

Forecasting simply means understanding which variables lead to predict other variables [2]. This means a clear understanding of the timing of lead-lag relations among many variables, understanding the statistical significanc of these lead lag relations and learning which variables are the more important ones to watch as signals for predicting the market moves. Better forecasting is the key element for better finan cial decision making, in the increasing financia market volatility and internationalized capital fl ws.

It was suggested that one of the best ways to predict exchange price is the use ANN which can easily adapt the changes of the market and provide a high performance modeling with sufficien available data [2]. In literature many ANN models are evaluated against statistical models for forecasting the market value. It is observed that in most of the cases ANN models give better result than other methods. Exploring the use of feedforward and various recurrent neural network (RNN) architectures in predicting the foreign exchange rates was presented in [3]. Data of future contracts on foreign exchange rates for British pound (BP), Canadian dollar (CD), and Japanese yen (JY) that are traded at the Chicago Mercantile Exchange (CME) against US dollars were used in the experiments. It was found that recurrent NN model offer the most accurate predictions than the Back Propagation (BP) for JP exchange rates. However, none of the RNN models appear to be statistically superior to the benchmark (i.e., linear model) for predicting CD exchange rates. This mean that the type of NN, the learning algorithms, the proposed model and the available data set can significantl affect the results of the prediction models.

In this paper, we introduce a solution to exchange rate prediction problem using NN. Back propagation (BP) training method is used to train the neural network. BP is one of the most popular techniques to train the Feed forward (FF) neural networks. It is a supervised learning type neural network. It uses the steepest gradient descent method to update the ANN weights such that a minimization criterion is reached. BP uses a differentiable activation function. We show that NN can provide better prediction values to the exchange rate than the traditional MLR model. The results provided, significantl outperform the MLR ones. A detailed description of the proposed NN model structure, learning algorithms and method of validating results are presented.

2 Why Stock Market Prediction?

Strong motivations for demanding the prediction of stock market prices exists [2, 4–6]. The most significan motive is the maximization of the financia profit With the growing investing and trading sizes, people urgently searched for an intelligent tool which helps to increase their gains and minimizing their risks. Many approaches were used to achieve these goals. They include statistics and linear regression techniques [4, 5, 7]. The goal is to be able predict and benefi from the market direction. None of these techniques has proved to be the constantly correct prediction tool as desired.

Predicting stock market based time-series models, using past measurements to provide an estimate of future measurements, have been explored in many articles [6, 8, 9]. It is always required to build a model that has a recurrence relation derived from past measurements. The recurrence relation is then used to provide a near accurate new measurement. These measurements are expected to be good enough compared to the actual measurements. Auto-Regressive (AR) models investigated in the past to solve the modeling problem for stock index predication [10–12]. Recently, much effort was directed towards either improving the prediction accuracy of these models using soft computing techniques or using it directly to handle the forecasting problem [13, 14]. In [15], authors presented a nonlinear auto-regressive (NAR) time-series model for forecasting applications. GAs simultaneously optimize all of the Radial Basis Function (RBF) parameters so that an effective time-series model was designed and used for forecasting. A time series model based fuzzy inference system was explored to select the best time lags of a linear model was introduced in [16]. Similar approaches based hybridization of Autoregressive Integrated Moving Average (ARIMA) and soft computing techniques were also presented in [17].

Many Artificia Intelligent techniques were used to build a model structure for stock market prediction. They include the multilayer feed-forward neural networks, expert systems and Genetic Algorithms (GAs) [9, 18–20]. Prediction of stock price index movement of the daily Istanbul Stock Exchange (ISE) National 100 Index using ANN and support vector machines (SVM) was presented in [21]. Ten technical indicators were selected as inputs of the proposed models. Experimental results show that average performance of ANN model (75.74%) was found significantl better than that of SVM model (71.52%). This shows that ANN has the capabilities to provide better results than other types of modeling approaches. Genetic Algorithms and ANN technologies were integrated for developing a stock market forecasting system. GAs was used as a classifie to control the activation of a feedforward ANN to perform local forecasting activity [22]. Recently, authors in [23] provided a prediction model for the Standards & Poors 500 (S&P500) index based Genetic Programming (GP). The experiments and analysis conducted show some unique advantages of using GP over other soft computing techniques in stock market modeling.

3 Modeling based MLR

Common types of casual models are the regression analysis and the econometrics models. An auto regression analysis is a technique that uses multiple related variables (x) to predict the outcome of a dependent variable (y) [24], which will be discussed further in the coming sections of the research. An essential part in the forecasting process is the collection of data, when collecting historical or recent data it is important to make sure that the data is reliable, valid, and also relevant to get the most accurate estimations. Another important part is the selection of the method, it is important to select the method that is best fi or more effective to the situation, and that could depends on several factors like for example the level of uncertainty and the feasibility of the selected method [25].

In most business and economic applications, we need to solve the problem of findin the relationship which exists between two different random variables X and Y. This type of relationship is known as a *bivariate* relationship. Thus, we need to predict the value of a dependent variable Y based upon the value of one independent variable X. For example, sales manager might want to predict the expected sales price based on the historical sales during the past weeks. A financia specialist may be interested in developing a relationship between the investment during certain time frame and the future returns [26].

In simple regression analysis, we are always interested in findin the pattern of the functional nature of the relationship between numbers of variables. This relation could be a simple one so a simple linear regression model is adequate or it may need a higher complex relation where more advanced relation need to be produced. When adopting regression analysis to solve the forecasting/estimation problem, it is important to determine the appropriate mathematical model that properly explains the relationship between the variables of interest.

The general MLR model can be described as given in Equation 1:

$$y^k = \beta_0 + \sum_{i=1}^{n} \beta_i y^{k-i} \quad (1)$$

Equation 1 is called *regression function* and the model parameters γ_i's ($i = 0, \ldots, n$) are called *regression coefficients*.

4 Least-Squares Estimation

The general least-squares estimation (LSE) problem can be formulated as follows. Assume a linear system has:

- **Input signals:** y^{k-i}, $i = 1, \ldots, n$ and
- **Output signal:** y^k as given in Equation 1.

To show how the parameter estimation process work, we assume we have a system with four input variables $y^{k-1}, y^{k-2}, y^{k-3}, y^{k-4}$ and single output y^k as in the case under study. Thus, the linear mathematical model can be represented as in Equation 2:

$$\begin{aligned} y^k &= \beta_0 + \beta_1 y^{k-1} + \beta_2 y^{k-2} \\ &+ \beta_3 y^{k-3} + \beta_4 y^{k-4} \end{aligned} \quad (2)$$

According to statistical terminology, fittin data using linear model structure is called *linear regression*. To estimate the model parameters we need to collect number of measurements m for the input-output pair which is called training data set.

To fin the values of the model parameters β's we need to build what is called the *regression matrix* Ω. This matrix is developed based on the experiment collected measurements. Thus, Ω can be presented as follows given there is a set of measurements m. Given that y_1^{k-1} is the collected measurements number one at the instance of time $k-1$.

$$\Omega = \begin{pmatrix} 1 & y_1^{k-1} & y_1^{k-2} & \vdots & y_1^{k-n} \\ 1 & y_2^{k-1} & y_2^{k-2} & \vdots & y_2^{k-n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & y_m^{k-1} & y_m^{k-2} & \vdots & y_m^{k-n} \end{pmatrix} \quad (3)$$

The parameter vector λ and the output vector y can be presented as follows:

$$\lambda = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{pmatrix} \mathrm{y} = \begin{pmatrix} y_1^k \\ y_2^k \\ \vdots \\ y_m^k \end{pmatrix} \quad (4)$$

The least squares solution of yields the normal equation:

$$\Omega^T \lambda = \mathrm{y} \quad (5)$$

which has a solution:

$$\lambda = \Omega^{-1} \mathrm{y} \quad (6)$$

But since, the regression matrix Ω is not a symmetric matrix, we have to reformulate the equation such that the solution for the parameter vector λ is as follows:

$$\lambda = (\Omega^T \Omega)^{-1} \Omega^T \mathrm{y} \quad (7)$$

The second order polynomial model is given in Equation 8. This model can provide a better accuracy than the firs order model since it provides more dynamics and non-linearity.

$$y^k = \beta_0 + \sum_{i=1}^{n} \beta_i\, y^{k-i} + \sum_{i=1}^{n} \sum_{j=0}^{n} \beta_{ij}\, y^{k-i} y^{k-j} \quad (8)$$

5 Prediction and ANN

ANN is an information-processing system which is inspired by the biological neural system. ANN is composed of interconnected neurons which send signals to each other over a large number of weighted connections. McCulloch and Pitt in 1943 started by a simple binary neuron model then moved to large number of interconnected neuron. ANNs and prediction of exchange price were heavily investigated by researchers such as in [27] where the prediction of currency exchange rate between US$ to British Pound, Indian Rupees and Japanese Yen were presented. The performance of the proposed models was evaluated through simulation and has been compared with those obtained from standard LMS based forecasting model.

5.1 Neural Network Architecture

ANN neurons are grouped into layers: input layer, hidden layer, and output layer. Neurons can be highly connected or fully connected. Each connection is represented by a weight which is define based on the learning algorithm. Connection weights reflec the relative importance of each input to the neuron. In input layer, each input represents a single attribute, while the fina output of the output layer represents a result or solution. These neurons are grouped together to form a layer. Each neuron has a number of inputs and a single output, in our case study. Each input has an assigned factor or parameter called *the weight*. The way how a neuron is working is as follows: input signal to each neuron is multiplicity by the corresponding weight then the result from the multiplication is summed and passes through a transfer function, most likely to be a sigmoid function. If the result of the summation is over a certain threshold, the neuron output will be activated else the output is not. The most simple unit of any ANN is the perceptron which is shown in Figure 1.

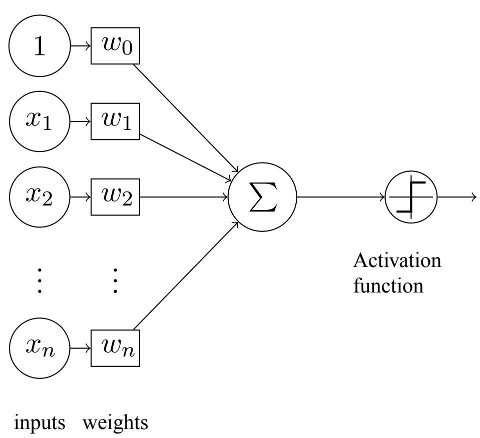

Figure 1: The simple perceptron architecture

5.2 Learning Algorithm

Learning process in ANN is the algorithm used to adjust the weights of the network in order to minimize the difference between the actual and predicted values by the network. Usually, the weights if the network is initialized randomly. There are four basic types of learning rule: Error Correlation Learning (ECL), Boltzmann learning (BL), Hebbian Learning (HL), and Competitive Learning (CL). The detailed descriptions of these learning rules are referred to the work of [28]. Among all the training algorithms, Back-Propagation (BP) which follows ECL rule is the most popular choice. The famous BP is essentially a gradient steepest descent method, searching at error surface. Basically, BP involved two steps in each iteration:

- Forward calculation to produce a solution and based on the error,
- Backward propagation to adjust weights.

However, the standard BP suffers from several weaknesses such as slow convergence, lack of robustness, and inefficien y [29]. To address the slow convergence rate problem of the BP algorithm, researchers like, Leonard and Kramer [30]; Hunt and Sbarbaro [31]; Nahas et al. [32]; Karim and Rivera [33] proposed the use of conjugate gradient method to provide a faster convergence.

Suppose we have ANN with n neurons in the input layer and m neurons in the hidden layers, and one output neuron. The learning process can be divided into number of stages and described as follows:

1. **Process in the Hidden layer**: Given a set of inputs x_i and a set of corresponding weights w_{ij} between the input and hidden neurons, the outputs of all neurons in the hidden layer are calculated by the summation function in Equation 9 where the activation function is shown in Equation 10.

$$\begin{aligned} S_i &= \sum_{i=0}^{n} w_{ij} x_i \\ y_j &= \phi(S_j) \end{aligned} \tag{9}$$

where $i = 1, 2 \ldots, n$ and $j = 1, 2, \ldots, m$. ϕ and y_j are the activation function and output of the j^{th} node in the hidden layer, respectively. ϕ is usually a sigmoid function given Equation 10.

$$\phi(x) = \frac{1}{1 + e^{-x}} \tag{10}$$

2. **Process in the Output layer**: The outputs of all neurons in the output layer are given Equation

11. l is define as the number of neurons in the output layer. For simplicity, l is one.

$$Y = f(\sum_{j=0}^{m} W_{jl} y_j) \tag{11}$$

f is the activation function of the output layer which is usually a line function. Y is the neural network output from the single neuron in the output layer as in our case study.

3. **Stopping process**: The learning process continues till the error minimization criterion (the difference between the actual and the result of the network) is reached.

6 NN AutoRegressive, eXternal (NNARX) Input Model

The ARX model is a simple model structure. This structure can be presented as given in [34–36] and in Equation (12):

$$\begin{aligned} G(q^{-1},\theta) &= q^{-k}\frac{B(q^{-1})}{A(q^{-1})} \\ H(q^{-1},\theta) &= \frac{1}{A(q^{-1})} \end{aligned} \tag{12}$$

The predictor $\hat{y}(t|\theta)$, which is the prediction of the observed response using ARX model can be computed as:

$$\begin{aligned} \hat{y}(t|\theta) &= q^{-k}B(q^{-1})u(t) + [1 - A(q^{-1})]y(t) \\ &= \varphi^T(t)\theta \end{aligned} \tag{13}$$

where:

$$\begin{aligned} \varphi(t) &= [y(t-1), \ldots, y(t-nA), \\ &\quad u(t-k), \ldots, u(t-k-nB)]^T \\ \theta &= [-a_1 \ldots - a_{nA}, b_0 \ldots b_{nB}]^T \end{aligned} \tag{14}$$

General nonlinear input-output ARX models can be described as given in Equation (15):

$$\begin{aligned} y(t) &= \hat{y}(t|\theta) + e(t) \\ &= g[\varphi(t,\theta),\theta] + e(t) \end{aligned} \tag{15}$$

where $\varphi(t,\theta)$ is the regression vector, θ is the neural network adjustable parameters vector known as the weights W and g is a nonlinear function realized by the neural network. The NNARX model structure is depicted in Figure 2.

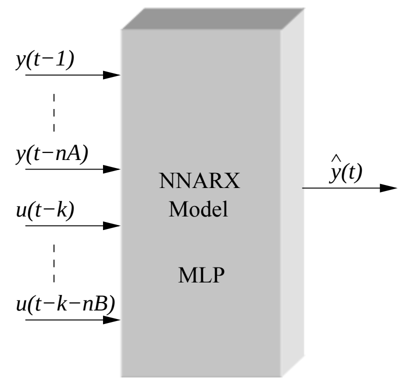

Figure 2: The NNARX Model Structure [34–36]

To develop a prediction model for the Reserve Bank of Australia, we need to use a set of training data Z^N which is used to train the ANN. This process determine the ANN weights matrix W which is specifie by the adjustable parameter θ. The MLP network can be trained to learn the nonlinear relationship from the set of training data Z^N to the set of possible weights $\hat{\theta}$.

$$Z^N \to \hat{\theta}$$

The developed ANN model will predict the values of $\hat{y}(t)$, which should be "close" to the true output $y(t)$. The produced NN should be able to pass the acceptable performance test [37, 38]. The regression vector [34, 37] is define as in Equation (14). Notice that the regression vector is formed with the past values of the input and output of the system.

7 Performance Criterion

We explored various criteria to measure the performance of the developed model. They are given in following equations:

1. Variance-Accounted-For (VAF):

$$VAF = [1 - \frac{var(y-\hat{y})}{var(y)}] \times 100\% \tag{16}$$

2. Mean Square Error (MSE):

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_j - \hat{y}_j)^2 \tag{17}$$

3. Euclidean distance (ED):

$$ED = \sqrt{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2} \qquad (18)$$

4. Manhattan distance (MD):

$$MD = (\sum_{i=1}^{n} |y_i - \hat{y}_i|) \qquad (19)$$

R refers to the Pearson product-moment correlation coefficient often called simply correlation coefficien between the two variables y and $\hat{y}$ is define as:

$$R = \frac{\sum_{i=1}^{n} (y_i - \overline{y})(\hat{y}_i - \overline{\hat{y}})}{\sqrt{\sum_{i=1}^{n} (y_i - \overline{y})^2 \sum_{i=1}^{n} (\hat{y}_i - \overline{\hat{y}})^2}} \qquad (20)$$

where y, $\hat{y}$ and $\overline{y_i}$ are the actual and estimated model

output of the developed model and the mean value of the signal y, respectively.

8 Reserve Bank of Australia

The Australian Government and its agencies require specifi bank services and the Reserve Bank of Australia (RBA) provide such services. The foundation of the bank was in 1912. This central bank takes the duty of managing the Australian gold and foreign exchange reserves. There are also many central banks and some formal organization help in doing so. The central bank of Australia is the RBA, its functions and power is taken from the Reserve Bank Act 1959. The records of the Reserve Banks contain: the central banking, the financia markets and system surveillance, the economy and monetary policy, and the bank designs and issue functions. The bank's mission involves measuring the changing occurs in the currency and its effect on the full employment, and welfare and economic prosperity of the Australian people. The bank policy is to archives all information or attributes which affect the exchange rate value which can help in monitoring and control the economy.

9 Experimental Results

A data set from January 2010 till the end of December 2013 was used in our experiments. The Daily exchange rates of the Australian Dollar against the Euro was used in our experiments. In the following subsection, we shall provide the mathematical relationship which was developed using MLR model and the values of the estimated parameters using LSE. We will also show graphically the relationship in both training and testing cases between the actual exchange price and exchange estimated price. We shall also provide the developed ANN model to be used in predicting the exchange rate.

9.1 Euro MLR Model

The developed Euro model is presented in Equation 21. The study of residuals (i.e. error) is essential to MLR modeling process and deciding the performance of the proposed model. The best model with the highest R^2 (0.97956) in training case is obtained and shown in Figure 3. The measured and estimated exchange rate between the Australian and Euro Rate Using MLR model in both training and testing cases is shown in Figure 4. The computed evaluation criteria is given in Table 1.

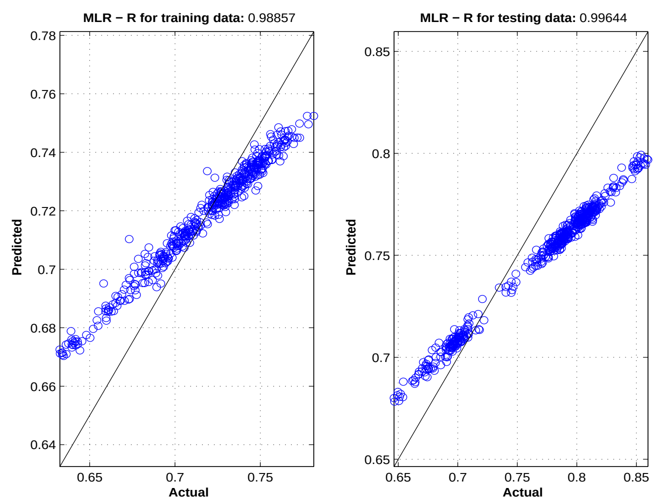

Figure 3: Computed R for the MLR model

$$\begin{aligned} y(k) &= 0.5264\, y(k-1) - 0.0038\, y(k-2) \\ &+ 0.0015\, y(k-3) + 0.0441\, y(k-4) \\ &+ 0.3108 \end{aligned} \qquad (21)$$

Table 1: Evaluation Criteria-Euro Model

Case	VAF	MSE	ED	MD
Training	80.7344%	21×10^{-5}	0.3274	0.0118
Testaing	80.1604%	97×10^{-5}	0.7065	0.0284

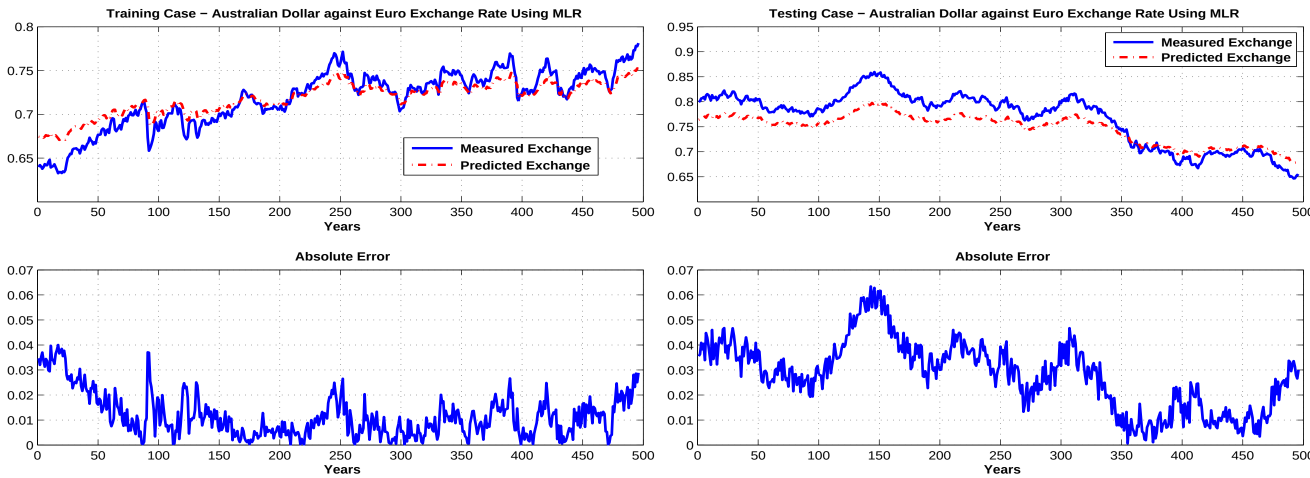

Figure 4: Measured and Estimated MLR Euro Exchange Rate in (a) Training (b) Testing Cases

9.2 Euro ANN Model

In this section, we provide our results based ANN model. The simulated prediction process was performed using ANN Based System Identificatio Toolbox provided by M. Norgaard under Windows system [37]. The ANN model was trained based LM algorithm.

- **ANN Architecture:** The proposed ANN architecture developed for the stock market model is shown in Figure 5 (a). The ANN has four inputs $y(k-1), \ldots, y(k-4)$ with the ANN output is $y(k)$ which is the current estimated value of the exchange rate. We run the back-propagation (BP) learning algorithm to estimate the ANN weights which minimize the error difference between the actual exchange rate and the estimated one. The ANN convergence is shown in Figure 5 (b). After adequate number of epochs the ANN error reached a pre-specifie threshold of minimum error.

- **Simulation:** To show the performance of the ANN, we draw the results of the actual exchange rate and the output of the ANN in both training and testing cases (See Figure 6 (a), (b)). It can be seen that the two curves are very close which indicate an outstanding performance of the developed ANN model. The actual and estimated values are better coinciding as given in the plot. This means the ANN was able to improve the computed $\hat{y}$ than in the case of MLR.

- **Computed R and Auto-Correlation:** According to Equation 20, we also measured the values of R and show its performance in Figure 7 (a). To guarantee a robust performance, we measured the auto-correlation function of the prediction error and presented it in Figure 7 (b).

- **Error Criterion:** We computed the performance of the developed ANN model using the evaluation criterion, VAF, RMSE, ED, and MD. The values are given in Table 2. It can be seen that the performance of the ANN model is better compared to the results developed using the MLR model.

Table 2: Evaluation criteria of the Euro based ANN Model

Case	VAF	MSE	ED	MD
Training	98.08%	2×10^{-5}	0.101	0.003
Testing	96.00%	16×10^{-5}	0.287	0.009

10 Conclusions and Future Work

In this paper, we provide a comparison between ANN and MLR models in predicting the currency exchange rate. Two models for the daily exchange rates of the Australian Dollar against the Euro w ere developed. The data set used in the experiments collected during the period of January 4, 2010 to December 31, 2013. Many evaluation criterion were used to evaluate the developed model. It was found that the ANN is superior in providing better prediction capabilities. We plan to explore other soft computing techniques to solve this problem since these techniques were promising.

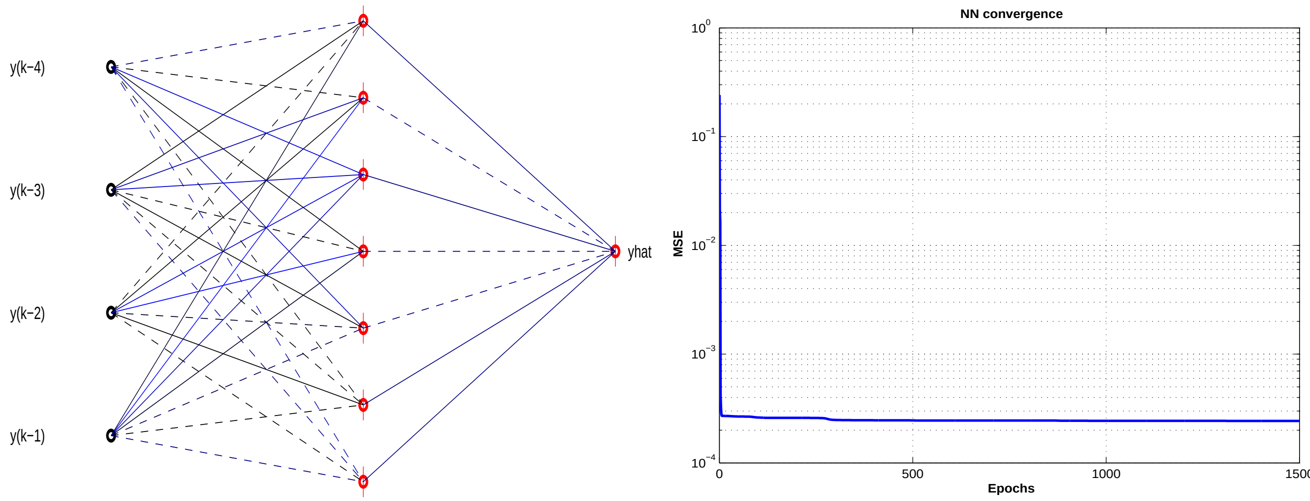

Figure 5: (a) Proposed ANN Architecture (b) Convergence of the proposed ANN

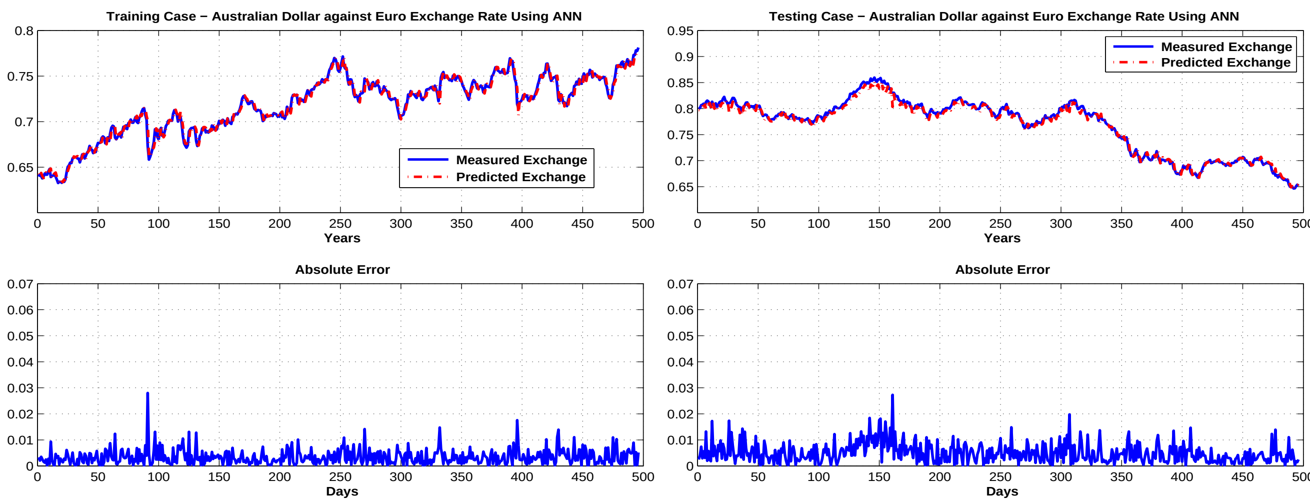

Figure 6: Measured and Estimated ANN Euro Exchange Rate in (a) Training (b) Testing Cases

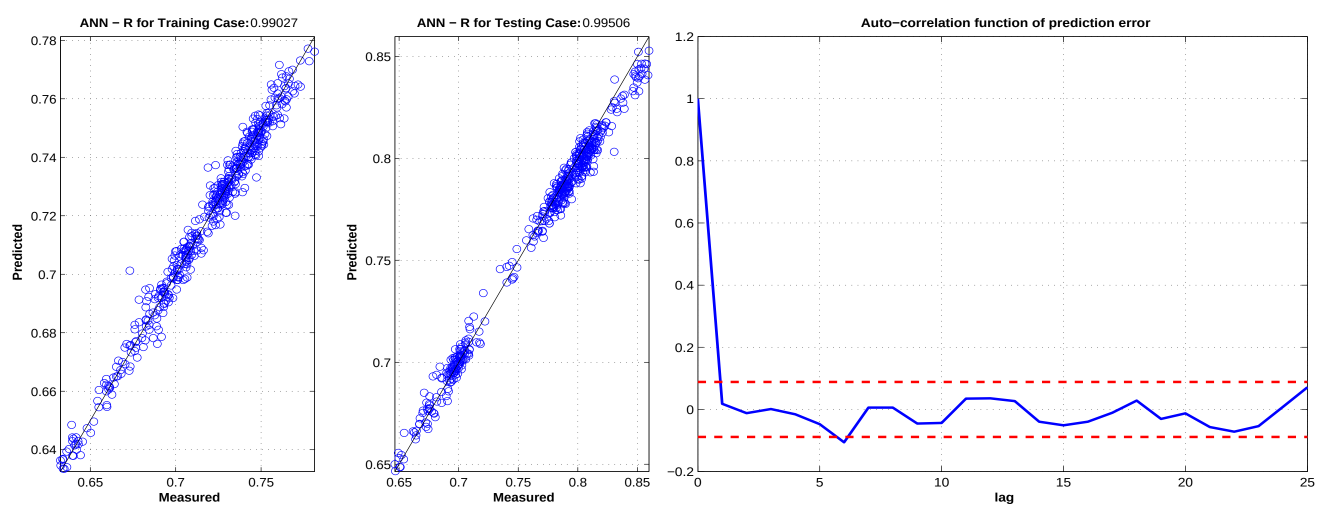

Figure 7: (a) Computed R for the ANN model (b) Auto-Correlation function of prediction error in ANN

References:

[1] B. Premanode and C. Toumazou, "Improving prediction of exchange rates using differential EMD," *Expert Syst. Appl.*, vol. 40, pp. 377–384, Jan. 2013.

[2] E. Guresen, G. Kayakutlu, and T. U. Daim, "Using artificia neural network models in stock market index prediction," *Expert Syst. Appl.*, vol. 38, pp. 10389–10397, Aug. 2011.

[3] K. M. Kiani and T. L. Kastens, "Testing forecast accuracy of foreign exchange rates: Predictions from feed forward and various recurrent neural network architectures," *Comput. Econ.*, vol. 32, pp. 383–406, Nov. 2008.

[4] J. J. Murphy, *Technical Analysis of the future Market*. NYIF, New York, 1986.

[5] A. Colin, "Exchange rate forecasting at citibank london," in *Proceedings of the Neural Networks Computing, London*, 1991.

[6] D. E. Baestanens and W. M. van den Bergh, *Tracking the Amsterdam stock index using neural networks*. In Neural Networks in the Capital Markets, chapter 10. John Wiley and Sons, 1995.

[7] E. Y. Li, "Artificia neural networks and their business applications," *Information and Managements*, vol. 27, pp. 303–313, 1994.

[8] K. Bergerson and D. Wunsch, *A commodity trading model based on a neural network-expert system hybrid*. In Neural Networks in financ and investing, chapter 23. Probus Publishing Company, 1993.

[9] R. J. Van Eyden, *The Application of Neural Networks in the Forecasting of Share Prices*. Finance and Technology Publishing, 1996.

[10] L. Ljung, ed., *System Identification (2Nd Ed.): Theory for the User*. Upper Saddle River, NJ, USA: Prentice Hall PTR, 1999.

[11] M. Salimifard, M. Jafari, and M. Dehghani, "Identificatio of nonlinear mimo block-oriented systems with moving average noises using gradient based and least squares based iterative algorithms," *Neurocomput.*, vol. 94, pp. 22–31, Oct. 2012.

[12] B. Majhi, M. Rout, R. Majhi, G. Panda, and P. J. Fleming, "New robust forecasting models for exchange rates prediction," *Expert Syst. Appl.*, vol. 39, pp. 12658–12670, Nov. 2012.

[13] M. Khashei, M. Bijari, and G. A. Raissi Ardali, "Improvement of auto-regressive integrated moving average models using fuzzy logic and artificia neural networks (anns)," *Neurocomput.*, vol. 72, pp. 956–967, Jan. 2009.

[14] C.-F. Liu, C.-Y. Yeh, and S.-J. Lee, "Application of type-2 neuro-fuzzy modeling in stock price prediction," *Appl. Soft Comput.*, vol. 12, pp. 1348–1358, Apr. 2012.

[15] A. F. Sheta and K. A. DeJong, "Time-series forecasting using GA-tuned radial basis functions.," *Information Science Journal*, vol. 133, no. 3-4, pp. 221–228, 2001.

[16] K. Lukoseviciute and M. Ragulskis, "Evolutionary algorithms for the selection of time lags for time series forecasting by fuzzy inference systems," *Neurocomput.*, vol. 73, pp. 2077–2088, June 2010.

[17] M. Khashei, M. Bijari, and G. A. Raissi Ardali, "Hybridization of autoregressive integrated moving average (arima) with probabilistic neural networks (pnns)," *Comput. Ind. Eng.*, vol. 63, pp. 37–45, Aug. 2012.

[18] F. Zahedi, *Intelligent Systems for Business: Expert Systems with Neural Networks*. Wadsworth Publishing Company, Belmont, USA, 1993.

[19] A. Sheta and K. De Jong, "Time-series forecasting using GA-tuned radial basis functions," in *Information Science Journal*, pp. 221–228, 2001.

[20] C. S. Vui, K. S. Gan, C. K. On, R. Alfred, and P. Anthony, "A review of stock market prediction with artificia neural network (ann).," in *ICCSCE*, pp. 477–482, IEEE, 2013.

[21] Y. Kara, M. Acar Boyacioglu, and O. K. Baykan, "Predicting direction of stock price index movement using artificia neural networks and support vector machines: The sample of the istanbul stock exchange," *Expert Syst. Appl.*, vol. 38, pp. 5311–5319, May 2011.

[22] G. Armano, M. Marchesi, and A. Murru, "A hybrid genetic-neural architecture for stock indexes forecasting," *Inf. Sci.*, vol. 170, pp. 3–33, Feb. 2005.

[23] A. F. Sheta, H. Faris, and M. Alkasassbeh, "A genetic programming model for s&p 500 stock market prediction," *International Journal of Control and Automation*, vol. 6, no. 6, pp. 303–314, 2013.

[24] J. Blocher, V. Mabert, A. Soni, and M. Venkataramanan, *Forecasting: Including an Introduction to Forecasting using the SAP R/3 System*. Indiana University Kelley School of Business, 2004.

[25] J. Armstrong, *Principles of Forecasting: A Handbook for Researchers and Practitioners*. Norwell, MA: Kluwer Academic Publishers, 2001.

[26] A. Hoshmand, *Business Forecasting: A practical approach*. NY: Routledge, 2010.

[27] M. A. Hernandez Medina and C. M. Mendez, "Modelling and prediction of the mxnusd exchange rate using interval singleton type-2 fuzzy logic systems [application notes]," *Comp. Intell. Mag.*, vol. 2, pp. 5–8, Feb. 2007.

[28] A. K. Jain, J. Mao, and K. K. Mohiuddin, "Artificia neural networks: A tutorial," *IEEE Computer Special Issue on Neural Computing*, pp. 31–44, 1996.

[29] D. Rumelhart, G. Hinton, and R. Williams, *Learning Internation Representations by Error Propagation*. In parallel Distributed Processing: Exploration in the Microstructure of Cognition, ed. D.E. Rumelhart and McCleland, vol.1, Chap.8 MIT Press, 1986.

[30] J. Leonard and M. A. Kramer, "Improvement of the backpropagation algorithm for training neural networks," *Computer Chemical Engineering*, vol. 14, pp. 337–343, 1990.

[31] K. J. Hunt and D. Sbarbaro, "Neural networks for nonlinear interna1 model control," *IEEE Proc. Part D 138*, vol. 43, pp. 1–438, 1991.

[32] E. P. Nahas, M. A. Henson, and D. E. Seborg, "Nonlinear interna1 model control strategy for neural network models," *Computer Chemical Engineering*, vol. 16, pp. 1039–1057, 1992.

[33] M. N. Karim and S. L. Rivera, "Artifi cial neural networks in bioprocess state estimation," *Advances in Biochemical Engineering/Biotechnology*, vol. 46, 1992.

[34] M. Norgaard, O. Ravn, Poulsen, and L. K. Hansen, *Neural Networks for Modelling and Control of Dynamic Systems*. Springer, London, 2000.

[35] H. Hiary, "Identificatio and model predictive controller design of nonlinear systems using artificia neural networks," Master's thesis, Balqa Applied University, Salt, Jordan, 2005.

[36] A. F. Sheta, N. Ghatasheh, and H. Faris, "Forecasting global carbon dioxide emission using auto-regressive with eXogenous input and evolutionary product unit neural network models," in *Proceedings of the 2015 6th International Conference on Information and Communication Systems (ICICS)*, pp. 182–187, April 2015.

[37] M. Norgaard, "Neural network based system identificatio toolbox," *Department of Automation. Technical University of Denmark*, 2000.

[38] M. Gibson, E.Ferreira, X.Cheng, T.Knight, D.Greve, and B.Krogh, "System identificatio methods for plasma enhanced chemical vapor deposition," *ECE Department, Carnegie Mellon University*, 1997.