Forecasting of Indian Rupee (INR) / US Dollar (USD) 
Currency Exchange Rate Using Artificial Neural Network

Yusuf Perwej 
M. Tech., MCA 
Computer Science & Information System 
Jazan University, Jazan 
Kingdom of Saudi Arabia (KSA) 

Asif Perwej 
P. hD, MBA 
Department of Management 
Singhania University 
Rajasthan, India 

yusufperwej@gmail.com 
asifperwej@gmail.com

Abstract

A large part of the workforce, and growing every day, is originally from India. As one of the largest populations in the world, they have a lot to offer in terms of employment. The sheer number of IT workers makes them a formidable traveling force as well, easily picking up employment in English speaking countries. However, since the beginning of the economic crises, many Indians have opted to stay or return home, and this has had a solid effect on the Indian Rupee (INR) as compared to the US Dollar (USD). In the real world global economy, accuracy in forecasting the foreign exchange rate or at least predicting the trend correctly is of crucial importance for any future investment. The use of computational intelligence based techniques for forecasting has been proved extremely successful in recent times. In this paper, Artificial Neural Network has successfully been used for exchange rate forecasting. This paper examines the effects of the number of input and hidden nodes and the size of the training sample on the in-sample and out-of-sample performance. The Indian Rupee (INR) / US Dollar (USD) is used for detailed examinations. The number of input nodes has a greater impact on performance than the number of hidden nodes, while a larger number of observations do reduce forecast errors.

Keywords: Artificial Neural Networks (ANN), Data, Prediction, Forecasting, Foreign Exchange Rate, Autoregressive Integrated Moving Average (ARIMA).

1. Introduction

Many Indians travel abroad for work. The IT workforce from Indian is easily employed in countries like United States, United Kingdom, Singapore and Canada. While the global financial/economic crisis that started in Sep 2008 has stemmed part of this workforce flow, the situation is now picking up globally and Indians abroad are taking a breather, knowing that their jobs are secure and they can still remit money back home.

The Indian Rupee (INR) is pegged partly to the US Dollar (USD), one of the key employers of Indian workforce. Currently, 1 US Dollar (USD) is trading at 45.6 Indian Rupees. There has been so wild fluctuations since early 2009, with the exchange rate peaking at 1 USD = 52.1 INR in March 2009, to a low of 1 USD = 46.9 INR in May 2009. This is a difference of 5.2INR/US Dollar and those who remitted money from United States back to India in March 2009 must be laughing all the way to the bank.

There are a few currency theories that state that the increasingly powerful Rupee will depreciate against the dollar. Most turn to the economy as a reason for this, citing the industrialized world's shrinking need for a global workforce as the economy contracts. There are a few examples of such coming from some corporate announcements, indicating that they do not plan to outsource any more than they have to. There are even more who believe that the emerging strength of the Indonesian Rupiah,
the Rupee, and the Chinese Renminbi are all likely to continue to strengthen against the dollar simply because of the continuing strength of the emerging markets when compared to the wilting American economy. The foreign exchange market has experienced unprecedented growth over the last few decades. The exchange rates play an important role in controlling dynamics of the exchange market. As a result the appropriate prediction of exchange rate is a crucial factor for the success of many businesses and fund managers. Although the market is well known for its unpredictability and volatility, there exist a number of groups (like Banks, Agency and other) for predicting exchange rates using numerous techniques. The many type of theoretical models including both econometric and time series approaches have been widely use to model and forecast exchange rates such as autoregressive conditional heteroskedasticity (ARCH), general autoregressive conditional heteroskedasticity (GARCH), chaotic dynamics, and self-exciting threshold autoregressive models applied to financial forecasting. While these models may be good for a particular situation they perform poorly for other applications.

The Artificial Neural Networks have received increasing attention as decision-making tools. The Artificial Neural Networks, the well-known function approximates in prediction and system modeling, has recently shown its great applicability in time series analysis and forecasting. Artificial Neural Networks assists multivariate analysis. Multivariate models can rely on grater information, where not only the lagged time series being forecast, but also other indicators (such as technical, fundamental, inter-marker etc. for financial market), are combined to act as predictors. In addition, Artificial Neural Networks is more effective in describing the dynamics of non-stationary time series due to its unique non-parametric, non-assumable, noise-tolerant and adaptive properties. Artificial Neural Networks are universal function approximates that can map any nonlinear function without a priori assumptions about the data [1]. Artificial Neural Networks as models for forecasting exchange rates have been investigated in a number of studies. Weigend et al. [2] have found that neural networks are better than random walk models in predicting the Deutsche mark/US dollar (DEM/USD) exchange rate. In comparison with the traditional forecasting methods such as Box -Jenkins ARIMA models or regression models, there are many more modeling factors to be considered in neural networks. Zhang et al. [3] the neural network modeling issues for forecasting. Kaasta and Boyd [4] propose an eight step method in designing a neural network model for forecasting financial time series. The purpose of this research is to provide an in-depth study of the effects of several important factors on the performance of neural networks in exchange rate forecasting. Specifically we will examine two neural net- work factors. The number of input nodes and the number of hidden nodes on the forecasting performance of exchange rate between Indian Rupee (INR) / US Dollar (USD).

2. Artificial Neural network

Artificial Neural network are simplified models of the biological neuron system, is a massively parallel distributed processing system made up of highly interconnected neural computing elements that have the ability to learn and thereby acquire knowledge and make it available for use. Artificial Neural network learn by example. They can therefore be trained with known examples of a problem to acquire knowledge about it. Once appropriately trained, the network can be put to effective use in solving unknown or untrained instances of the problem.

2. 1 Neurons

A neuron is a processing unit that takes a number of inputs and gives a distinct output. The figure 1 below depicts a single neuron with R inputs p1,p2, …, pR, each input is weighted with a value w11, w12 , ..., w1R and the output of the neuron a equals to f (w11 p1+ w12 p2 + ... + w1R pR).

![Figure - 1 A simple neuron with R inputs](image)

Each neuron apart from the number of its inputs is characterized by the function f known [12] as transfer function. The most commonly used transfer functions are: the hard limit, the pure linear, the sigmoid and the tansigmoid function. The preference on these functions derives from their characteristics. Hard limit maps any value that belongs to (-∞,+∞) into two distinct values [0,1], thus it is preferred for networks that perform classification tasks (multiplayer perceptions MLP ). Sigmoid and tansigmoid, known as squashing functions, map any value from (-∞,+∞ ) to the intervals [0,1] and [-1,1] respectively. Lastly pure linear is used due to its ability to return any real value and is mostly used at the neurons that are related with the output of the network.
Hard limit $f(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases}$

Pure linear $f(x) = x$

Sigmoid $f(x) = \frac{1}{1+e^{-x}}$

Tansigmoid $f(x) = \frac{2}{1+e^{-2x}}$

$f(x) \in [0,1]$ $f(x) \in (-\infty, +\infty)$ $f(x) \in [0,1]$ $f(x) \in [-1,1]$

Table - 1 the most commonly used Transfer functions

2. 2 Layers

Artificial Neural network is defined as a data processing system consisting of a large number of simple highly interconnected processing elements (artificial neurons) is an architecture inspired by the structure of the cerebral cortex of the brain below figure 2. Each network has got exactly one input layer, zero or more hidden layers and one output layer. All of them apart from the input layer consist of neurons. The number of inputs to the Artificial Neural Networks equals to the dimension of our input samples, while the number of the outputs we want from the Artificial Neural Networks defines the number of neurons in the output layer. In our case the output layer will have exactly one neuron since the only output we want from the network is the prediction of tomorrow’s excess return.

![Neural Network Diagram](image)

Figure - 2 Neural Network Diagram

The mass of hidden layers as well as the mass of neurons in each hidden layer is proportional to the ability of the network to [11] approximate more complicated functions. Of course this does not imply by any means that networks with complicated structures will always perform better. The reason for this is that the more complicated a network is the more sensitive it becomes to noise or else, it is easier to learn apart from the underlying function the noise that exists in the input data. Therefore it is clear that there is a trade off between the representational power of a network and the noise it will incorporate.

2. 3 Weights

The weights used on the connections between different layers have much significance in the working of the neural network and the characterization of a network.

The procedure of adjusting the weights of a Artificial Neural Networks based on a specific dataset is referred as the training of the network on that set (training set). The basic idea behind training is that the network will be adjusted in a way that will be able to learn the patterns that lie in the training set. Using the adjusted network in future situations (unseen data) it will be able based on the patterns that learnt to generalize giving us the ability to make inferences. In our case we will train Artificial Neural Networks models on a part of our time series (training set) and we will measure their ability to generalize on the remaining part (test set). The size of the test set is usually selected to be 10% of the available samples [5]. Each sample consists of two parts the input and the target part is called supervised learning. Initially the weights of the network are assigned random values (usually within [-1 1]). Then the input part of the first sample is presented to the network. The network computes an output based on: the values of its weights, the number of its layers and the type and mass of neurons per layer.

3. Artificial Neural network For Time Series Forecasting

Time series forecasting is highly utilized in predicting economic and business trends. Many forecasting methods have been developed in the last few decades. Recently, artificial neural networks that serve, as a powerful computational framework,
Artificial Neural Networks have gained much popularity in business applications. Artificial Neural Networks have been successfully applied to loan evaluation, signature recognition, time series forecasting, classification analysis and many other difficult pattern recognition problems [6]. However, the use of neural networks is rapidly increasing, and in recent years they have been successfully used in prediction in economic business and hydrology. Artificial Neural Networks models have been proposed and used for the forecasting purpose. The most popular and successful one is the feed forward multilayer network or the multi layer perceptron (MLP). An MLP is typically composed of several layers of nodes. The first or the lowest layer is an input layer where external information is received. The last or the highest layer [8] is an output layer where the problem solution is obtained. The input layer and output layer are separated by one or more intermediate layers called the hidden layers. The nodes in adjacent layers are usually fully connected by acyclic arcs from a lower layer to a higher layer.

Artificial Neural Networks models have been proposed and used for the forecasting purpose. The most popular and successful one is the feed forward multilayer network or the multi layer perceptron (MLP). An MLP is typically composed of several layers of nodes. The first or the lowest layer is an input layer where external information is received. The last or the highest layer [8] is an output layer where the problem solution is obtained. The input layer and output layer are separated by one or more intermediate layers called the hidden layers. The nodes in adjacent layers are usually fully connected by acyclic arcs from a lower layer to a higher layer.

![Typical Fully Connected Feed Forward Neural Network for Time Series Forecasting](image)

The knowledge learned by a network is stored in the arcs and the nodes in the form of arc weights and node biases which will be estimated in the neural network training process. Figure 3 is an example of a fully connected MLP with one hidden layer. For a univariate time series forecasting problem, the inputs of the network are the past, lagged observations [9] of the data series and the outputs are the future values. Each input pattern is composed of a moving window of fixed length along the series. The network represented is a mapping function of the form

\[ y_{t+1} = f(y_t, y_{t-1}, \ldots, y_{t-p}) \]

Where \( y_t \) is the observation at time \( t \) and \( p \) is the dimension of the input vector or the number of past observations related to the future value. In this sense, the feed forward network used for time series forecasting is a general autoregressive model. Suppose we have \( N \) time-lagged observations \( y_1, y_2, \ldots, y_N \) in the training set and we need the one-step-ahead forecasts, then using a network with \( p \) input nodes and one output node, we have \( N-p \) training patterns. The first training pattern is composed of \( y_1, y_2, \ldots, y_p \) as the inputs and \( y_{p+1} \) as the target output. The second training pattern contains \( y_1, y_2, \ldots, y_{p+1} \) for the inputs and \( y_{p+2} \) for the desired output. Finally, the last training pattern is \( y_{N-p}, y_{N-p+1}, \ldots, y_{N-1} \) for the inputs and \( y_N \) for the target. The neural network training objective is to find the weights in order that some overall predictive error measure such as the sum of the squared errors (SSE) is minimized [11]. For this network structure, SSE can be written as

\[ SSE = \sum_{j=p+1}^{N} (y_j - a_j)^2 \]

Where \( a_j \) is the output from the network.

4. Proposed Design

The exchange rates between Indian Rupee (INR) / US Dollar (USD) are obtained from Datastream International. Datastream provides daily pricing and volume information on over 37,000 equities from 57 countries, 12,000 market stock [10] and bond indices, 87,000 macroeconomic series from the International Financial Statistics of the World Bank, OECD and other central statistical offices, over 2,000 daily foreign exchange rates, 1,000 daily and weekly interest rate series, 68,000 fixed income instruments from 23 countries and more than 260 financial and commodity futures and options contracts. Data is composed
of daily rates from the beginning of 1989 through the end of 2009. These papers examine the effects of several factors on the in sample fit and out of sample forecasting capabilities of neural networks. The neural network factors investigated are the number of input and hidden nodes which are two critical parameters in the design of a neural network. The number of input nodes is perhaps the most important factor in neural network analysis of a time series since it corresponds to the number of past lagged observations related to future values. It also plays a role in determining the autocorrelation structure of a time series. The number of hidden nodes allows neural networks to capture nonlinear patterns and detect complex relationships in the data.

We experiment with a relatively large number of input nodes. There is no upper limit on the possible number of hidden nodes in theory. However, it is rarely seen in the literature that the number of hidden nodes is more than double the number of input nodes. In addition, previous research [7] indicates that the forecasting performance of neural networks is not as sensitive to the number of hidden nodes as it is to the number of input nodes. Thus, five levels of hidden nodes, 6, 12, 18, 24 and 30 will be experimented. The combination of ten input nodes and five hidden nodes yields a total of 50 neural network architectures being considered for each in-sample training data set. A comprehensive study of neural network time series forecasting, finds that neural network models do not necessarily require large data set to perform well. To test if there is a significant difference between large and small training samples in modeling and forecasting exchange rates, we use two training sample sizes in our study. The large sample is consisted of 1043 observations from 1989 to 2009 and the small one includes 365 data points from 2003 to 2009. The test sample for both cases is the 2010 data which has 52 observations. The random walk model will be used as a benchmark for comparison. The random walk is a one-step-ahead forecasting model since it uses the current observation to predict the next one.

Three-layer feed forward neural networks are used to forecast the Indian Rupee (INR) / US Dollar (USD) exchange rate. Logistic activation functions are employed in the hidden layer and the linear activation function is utilized in the output layer. We are interested in one-step-ahead forecasts, one output node is deployed in the output layer. The use of direct optimization procedure in neural network training. To be more certain of getting the true global optima, a common practice is to solve the problem using a number of randomly generated initial solutions. We train each network 50 times by using 50 sets of different initial arc weights. The best solution among 50 runs is used as the optimal neural network training solution. The forecasting performance of the model is evaluated against a number of widely used statistical metric, we use three popular metrics RMSE, MAE, and MAPE, to evaluate the predictive performance of neural networks. These forecasting accuracy measures are listed as follows.

\[
RMSE = \sqrt{\frac{\sum(y_t - \hat{y}_t)^2}{T}}
\]

\[
MAE = \frac{\sum|y_t - \hat{y}_t|}{T}
\]

\[
MAPE = \frac{1}{T} \sum \left| \frac{y_t - \hat{y}_t}{y_t} \right| \times 100
\]

Where \( y_t \) the actual observation is \( \hat{y}_t \) is the predicted value, and \( T \) is the number of predictions. These criteria are mean based and are frequently used performance measures in this paper.

5. Experimental Results of Forecasting the Indian Rupee (INR) / US Dollar (USD)

The goal of this paper has been to study is to investigate the effects of neural network factors on the modeling and forecasting performance of neural networks, both in sample (training set) and out-of-sample (test set) results. The focus will be on the out-of-sample analysis because it is the forecasting capability that researchers and practitioners are most interested in. For neural network method, due to the potential problem of over fitting, we need to study the conditions under which over fitting may occur. An over fitted model gives good in-sample fit to the training data, yet poor predictive out-of-sample performance. Therefore, the examination of both in- sample and out-of-sample behaviors will provide us information on when and how over fitting occurs.

| Input | Hidden | RMSE    | MAE    | MAPE     |
|-------|--------|---------|--------|----------|
| 1     | 6      | 0.08493210 | 0.08452030 | 2.85874590 |
| 1     | 12     | 0.08493210 | 0.08451020 | 2.85871860 |
| 1     | 18     | 0.08493210 | 0.08451020 | 2.85872360 |
| 1     | 24     | 0.08493200 | 0.08451020 | 2.85873890 |
| 1     | 30     | 0.08493200 | 0.08451020 | 2.85872910 |
The table 2 shows the in-sample results for the large training sample of 1043 observations. It is quite evident that as the number of hidden nodes increases, RMSE decreases. This pattern is observed consistently in each level of the input node. The more hidden nodes are used the neural network becomes more powerful in modeling the data. We observe a different pattern for the effects of the input node and the hidden node with MAE and MAPE. MAE and MAPE do not decrease in general as the number of hidden nodes increases within each level of input node. The number of input nodes increases from 1 to 5, the mean MAE steadily decreases from 0.08451020 to 0.08449540 . When the number of input nodes is in the range of 6 to 10, overall MAE increases first and then decreases. MAPE does not show a clear input node effect. This is reasonable since our purpose of neural network training is to minimize the RMSE and not MAE and MAPE. It is important to note that there is less variation among different hidden node levels within each input node level than among different input node levels, suggesting that the number of input nodes has greater impact on the model fitting process of Artificial Neural Networks. The out-of-sample analysis to examine the predictive capabilities of Artificial Neural Networks as the Artificial Neural Networks structure changes Artificial Neural Networks performance will also be compared with that of the forecasting models selected by SCA which happen to be the random walk model in our paper.

Table-2 Artificial Neural Networks factors on training performance (training period 1989 - 2009) of effects

| Avgr | 0.08493210 | 0.08451020 | 2.85873120 |
|------|-------------|-------------|-------------|
| 2    | 0.08493190 | 0.08449910 | 2.85660740 |
| 2    | 0.08493120 | 0.08449500 | 2.85951250 |
| 2    | 0.08493110 | 0.08449940 | 2.85913650 |
| 2    | 0.08493080 | 0.08449970 | 2.85926800 |
| 2    | 0.08493080 | 0.08449950 | 2.85912570 |
| Avgr | 0.08493120 | 0.08449940 | 2.85913000 |
| 3    | 0.08483930 | 0.08449790 | 2.85828000 |
| 3    | 0.08483910 | 0.08449840 | 2.85966620 |
| 3    | 0.08483820 | 0.08449890 | 2.85969500 |
| 3    | 0.08483810 | 0.08449730 | 2.85810600 |
| 3    | 0.08483800 | 0.08449830 | 2.85912780 |
| Avgr | 0.08483850 | 0.08449820 | 2.85916850 |
| 4    | 0.08490000 | 0.08449680 | 2.85934930 |
| 4    | 0.08483780 | 0.08449550 | 2.85750980 |
| 4    | 0.08483650 | 0.08449650 | 2.85932080 |
| 4    | 0.08483570 | 0.08449550 | 2.85755380 |
| 4    | 0.08483440 | 0.08449710 | 2.85933500 |
| Avgr | 0.08483690 | 0.08449630 | 2.85861370 |
| 5    | 0.08483720 | 0.08449780 | 2.85864780 |
| 5    | 0.08483500 | 0.08449550 | 2.85682010 |
| 5    | 0.08483360 | 0.08449500 | 2.85720080 |
| 5    | 0.08483030 | 0.08449640 | 2.85868170 |
| 5    | 0.08473780 | 0.08449240 | 2.85635120 |
| Avgr | 0.08482390 | 0.08449540 | 2.85754030 |
| 6    | 0.08483510 | 0.08449920 | 2.85790900 |
| 6    | 0.08483210 | 0.08449930 | 2.85762540 |
| 6    | 0.08483900 | 0.08449230 | 2.85473670 |
| 6    | 0.08473730 | 0.08449750 | 2.85814660 |
| 6    | 0.08473370 | 0.08449520 | 2.85655290 |
| Avgr | 0.08473950 | 0.08449670 | 2.85699940 |
| 7    | 0.08483600 | 0.08449990 | 2.86019840 |
| 7    | 0.08483190 | 0.08450040 | 2.86007590 |
| 7    | 0.08483120 | 0.08449590 | 2.85815280 |
| 7    | 0.08473700 | 0.08449770 | 2.85879190 |
| 7    | 0.08473570 | 0.08449650 | 2.85722920 |
| Avgr | 0.08480300 | 0.08449810 | 2.85894830 |
| 8    | 0.08483480 | 0.08450020 | 2.85772100 |
| 8    | 0.08483250 | 0.08449870 | 2.85956610 |
| 8    | 0.08473980 | 0.08449910 | 2.85873100 |
| 8    | 0.08473730 | 0.08449600 | 2.85672850 |
| 8    | 0.08473480 | 0.08449650 | 2.85537220 |
| Avgr | 0.08473990 | 0.08449810 | 2.85762380 |
| 9    | 0.08483430 | 0.08449990 | 2.85891170 |
| 9    | 0.08483350 | 0.08449930 | 2.85875390 |
| 9    | 0.08473640 | 0.08449830 | 2.85776730 |
| 9    | 0.08473520 | 0.08449590 | 2.85696390 |
| 9    | 0.08463750 | 0.08449000 | 2.85134280 |
| Avgr | 0.08473720 | 0.08449670 | 2.85647970 |
| 10   | 0.08473700 | 0.08449230 | 2.85370770 |
| 10   | 0.08473730 | 0.08449400 | 2.85579050 |
| 10   | 0.08463990 | 0.08448660 | 2.85245050 |
| 10   | 0.08473150 | 0.08449480 | 2.85637290 |
| 10   | 0.08463900 | 0.08449450 | 2.85531560 |
| Avgr | 0.08473290 | 0.08449280 | 2.85472740 |
The random walk model in three time horizons. That is one network models the random walk model in terms of all three measures across the three time horizons not only for the best neural network architecture. The observed pattern in RMSE and MAE is quite consistent with three input nodes, indicating that the specification of the number of input nodes may be sensitive to the performance measure and the forecast horizon. The random walk model in three performance measures across the three time horizons are reported at the bottom of table 3. It is clear that neural networks predict much better than the random walk model in terms of all three measures across the three time horizons not only for the best neural network models but also for most other network architectures.

### Table - 3 Artificial Neural Networks effects of input nodes (training period 1989 - 2009) Out-of-sample analysis

The prediction of out-of-sample results from using the large training sample contains a table 3. The using 1, 2 and 3 for each performance measure of the three time horizons. That is RMSE1, MAE1 and MAPE1 are used for the one-month time horizon and RMSE2, MAE2 and MAPE2 are for the six-month time horizon and RMSE3, MAE3, and MAPE3 are for the 12-month horizon. The one-month horizon, all three measures of performance indicate 6 input nodes produce the best predictions. Average RMSE, MAE and MAPE take on values of 0.08449670, 0.08382900 and 2.80883310 respectively. As the length of time horizon increases, the effects of input nodes on MAE and MAPE are quite consistent. The network with six input nodes is still the overall best architecture. The observed pattern in RMSE for 6- and 12-month horizons is not the same as in the case of one-month horizon. For 6- and 12-month time horizons, the lowest average RMSE is 0.08477910 and 0.08467020 which occur with three input nodes, indicating that the specification of the number of input nodes may be sensitive to the performance measure and the forecast horizon.
The effects of hidden nodes on the out-of-sample performance for both the large and small samples contain table 4. Because of the similarity in the effects with different levels of input nodes, only the results with five input nodes for the large sample and one input node for the small sample are reported. Results show no clear effects of hidden nodes across different time horizons along different performance measures. The differences in performance measures across the levels of hidden nodes are very small, indicating the number of hidden nodes does not occupy an important role in the out-of-sample performance of neural networks. We observe that within each input node level, the results are not very sensitive to the change in the number of hidden nodes. Hence, correctly identifying the number of input nodes is more important than identifying the number of hidden nodes. Our results are in line with the findings reported by Lachtermacher and Fuller [7] who apply neural networks to predict one-step-ahead river flow. The performance measures for the neural network model in the 1989 2009 series (the large training sample) take on larger values. This suggests that the Artificial Neural Networks is a better choice for long term forecasting.

6. Conclusion

Finally, the forecasting of Exchange rates using Artificial Neural Networks models. In this paper we investigate the effects of three important factors on Artificial Neural Networks modeling and forecasting performance for Indian Rupee (INR) / US Dollar (USD) exchange rate. The number of input nodes and the number of hidden nodes are the experimental Artificial Neural Networks factors. Both in-sample fitting ability and out-of-sample predictive performance with three forecast horizons are evaluated along three criteria, RMSE, MAE, and MAPE. The effects of two training sample sizes are examined with the identical forecast horizons. The purpose of this paper is to examine the effects of several important neural network factors on model fitting and forecasting the behaviors. The forecasting purpose it is not appropriate to evaluate the Artificial Neural Networks capability with the training sample alone. There are no broadly accepted methods for construction of the best predictive model using strictly in-sample training data. The selection of the optimal network architecture should be based on test sample results. We have also found that the number of input nodes plays an important role in neural network time series forecasting. The Scope of this study in the Indian Rupee (INR) / US Dollar (USD) exchange rate is selected for detailed examination. The many other studies in exchange rate forecasting show that there is little difference in in-sample fitting and out-of-sample predictive results from one exchange rate to another. We investigate only one-step-ahead forecasting strategy. Robustness of neural networks to the changing structures, it can easily handle the inaccuracy and any degree of nonlinearity in the data.

References

[1] L. Cao and F. Tay, “Financial Forecasting Using Support Vector Machines,” Neural Comput & Applic, vol. 10, pp.184-192, 2001.

[2] Weigend, A. S., Huberman, B. A. and Rumelhart, D.E. (1992) Predicting sunspots and exchange rates with connectionist networks. In Nonlinear modeling and forecasting, eds. M. Casdagli and S. Eubank. Addison-Welsey, Redwood City, 395-432.

[3] Zhang, G., Patuwo, B. E. and Hu, M. Y. (1997)Forecasting with artificial neural networks: The state of the art. International Journal of Forecasting, forthcoming.

[4] Kastra, I. and Boyd, M., Designing a neural network for forecasting financial and economic time series. Neurocomputing, 1996, 10, 215-236.

| Sample Size | Input Levels | Hidden Levels | RMSE3 | MAE3 | MAPE3 |
|-------------|--------------|---------------|-------|------|-------|
| Large       | 5            | 6             | 0.08475240 | 0.08576500 | 3.83429130 |
|             | 5            | 12            | 0.08496670 | 0.08754000 | 4.84778130 |
|             | 1            | 18            | 0.08496510 | 0.08752600 | 4.84686520 |
|             | 1            | 24            | 0.08496270 | 0.08750500 | 4.84588060 |
|             | 1            | 30            | 0.08496590 | 0.08753300 | 4.84737830 |

Table - 4 Artificial Neural Networks Effects of hidden nodes
[5] Mitchell M.T. (1997). Machine learning. New York: The McGraw-Hill Companies.

[6] R. Sharda and R. Patil, Neural Network as Forecasting Experts: An Empirical Test. International Joint Conference on Neural Networks, Vol. 1, 491-494, 1990.

[7] Lachtermacher, G. an Fuller, J. D., Backpropagation in time-series forecasting. Journal of Forecasting, 1995, 14, 381-393.

[8] J. Connors, D. Martin and L. Atlas, “Recurrent Neural network and robust time series prediction”, IEEE Trans. Neural Network, Vol. 5, PP. 240 – 254

[9] J. Yao, Y. Li, C. I Tan, Option Price Forecasting Using Neural Networks. The International Journal of Management Science, Vol. 28, 455-466, 2000.

[10] http://www.lib.uwo.ca/business/data.html