International Proceedings of Economics Development and Research
IPEDR vol.86 (2016) © (2016) IACSIT Press, Singapore

Forecasting Foreign Exchange Rate during Crisis - A Neural Network Approach

K. V. Bhanu Murthy [+]

Department of Commerce, Delhi School of Economics, University of Delhi, India

Abstract. This paper attempts to use an artificial neural network for exchange rate forecasting. With the liberalisation of the exchange rate regime in India, there was an interest in forecasting exchange rates. In recent years, there is renewed interest exchange rate on account of the added volatility due to the Global Financial Crisis. Thus, this paper examines foreign exchange rates in India during the period of crisis and does within sample and out of sample forecasting.

This paper analyses the daily USD/INR rates with the help of a neural networks and presents their usefulness even in the times of extreme volatility like the current recessional period. It predicts the one-step-ahead value of the USD/INR exchange rate using a Feed Forward Back Propagation neural network with gradient descent approach using Levenberg-Marquardt Algorithm. It measures the performance using three evaluation criteria, i.e. MSE, MAE and DA. MSE and MAE are both small. But directional accuracy is only 51.67%. This is rather large in the case of out of sample forecasting. The results show that neural networks are a useful technique of forecasting exchanges rate in a period of crisis. The findings in the study have implications for both policy makers and investor's in the foreign exchange market.

Keywords: Neural Networks, Forecasting, Foreign Exchange Rate in India

Introduction

Forecasting foreign exchange has been an interesting and intriguing subject. This paper analyses the USD/INR rates with the help of a neural network and presents their usefulness even in the times of extreme volatility like the current recessionary period. Previous researches have shown that neural networks are a better forecasting tool than linear models. Time series data is fed into the feed forward back propagation neural network to capture the underlying "rules" of the network and thus create a model which can predict future exchange rate given the present rate. This paper aims to verify the effectiveness of neural networks in such times of crisis. This assumes significance because a neural network is as good as the data fed into it. The errors of prediction are measured and analysed. After the presentation of the results, a summary and its implications conclude the paper.

1. Policy Background

India moved from pegged exchange rate to a floating exchange rate in 1993. Along this came the problems of complexity. While fixed exchange rate is seen to have the advantage of a nominal anchor for "importing" credibility, providing transparency, reducing unpredictable volatility and transactions costs, floating exchange rate has the benefits of monetary independence, insulation from real shocks and a less disruptive adjustment mechanism in the face of nominal rigidities. Monetary policy in a floating exchange rate interacts and controls the real activity and inflation. The monetary authority has an obligation to keep inflation at bay and accelerate real economic activity. In order to achieve this, the monetary authority must understand the nuances of the exchange rate movement and having a mechanism to predict it helps. With all these benefits, predicting the future exchange rate seems very attractive proposition.

[+] Corresponding author. Tel.: +00918911601867.
E-mail address: bhanumurthykv@yahoo.com

2. Past Studies

In 1983, Meese and Rogoff analysed many time series and structural models of exchange rate prediction and came to the conclusion that these models are no better than a random walk method. This strengthened the random walk hypothesis and discouraged many researchers in the area of time series modelling. In 1990, Klein, Mizarch and Murphy demonstrated that foreign exchange rates do follow key fundamentals. But with a shorter horizon of days, forecasting using fundamentals become unrealistic because of unavailability of inflation, trade balance etc data on daily basis. Thus for short term forecasting, random walk method is used.

In 1970, the Box-Jenkins ARIMA forecasting method was introduced (Pankratz, A., 1983). It is based on a linear relationship equation between dependent and time lagged independent variables. Economic time series data contain high noise, volatility and complex market environment. This pushed the researchers towards the next level i.e. neural networks as they are capable of fitting non-linear relations on a data series and perform the task of classifying, recognition and prediction. Wei et al (2004) in review work state that "Research efforts on ANNs for forecasting exchange rates are considerable. In this paper, we attempt to provide a survey of research in this area. Several design factors significantly impact the accuracy of neural network forecasts. These factors include the selection of input variables, preparing data, and network architecture. There is no consensus about the factors. In different cases, various decisions have their own effectiveness".

Lean Yu et al (2007).The book discusses the most important advances in foreign-exchange-rate forecasting and then systematically develops a number of new, innovative, and creatively crafted neural network models that reduce the volatility and speculative risk in the forecasting of foreign exchange rates. Pacelli, V. et al, (2011) states that the variable of output of the ANN designed is the daily exchange rate Euro/Dollar and the frequency of data collection of variables of input and the output is daily. By the analysis of the data it is possible to conclude that the ANN model developed can largely predict the trend to three days of exchange rate Euro/USD. Chakradhara, P., & Narasimhan, V. (2007), Narendra, J. (2005) and Panda, C., & Narasimhan, V. (2003) are three studies that have used neural networks for estimating exchange rate in India. While there is lot of literature on forecasting foreign exchange rate these are some of the papers that throw light on ANN as a prediction tool for forecasting foreign exchange.

3. Methodology

Artificial neural networks are made up of interconnecting artificial neurons (programming constructs that mimic the properties of biological neurons). In a neural network model simple nodes, which can be called variously "neurons", "neurodes", "Processing Elements" (PE) or "units", are connected together to form a network of nodes — hence the term "neural network". While a neural network does not have to be adaptive *per se*, its practical use comes with algorithms designed to alter the strength (weights) of the connections in the network to produce a desired signal flow.

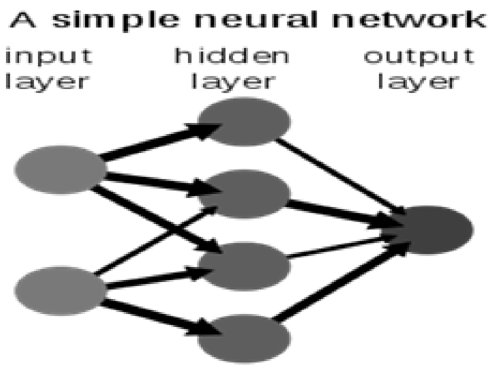

Figure 1: A Simple Neural Network

Neural networks are data-driven self adaptive methods in that there are few a priori assumptions about the model form for a problem under study. These unique features make them valuable for solving many practical forecasting problems. Any time series forecasting model assumes that there is an underlying process from which data are generated and the future value of a time series is solely determined by the past and current observations. Neural networks are able to capture the underlying pattern or autocorrelation structure within a time series even when the underlying law governing the system is unknown or too complex to describe.

3.1 Feed Forward Back Propagation Neural Network

A Feed Forward neural network[1] has a layered structure. It contains an input layer with a number of neurons, a hidden layer which processes the inputs and an output layer for a simple 2 layer network (including the input layer). But a simple feed forward network suffers from the handicap of how to adjust biases and weights. This was overcome by the introduction of back-propagation rule. In Back Propagation, the errors for the units of the hidden layer are determined by back-propagating the errors of the units of the output layer. If properly trained back-propagation networks tend to give reasonable answers when presented with new out of sample inputs. A new input leads to an output similar to the correct output for input vectors used in training that are similar to the new input being presented. This generalization property makes it possible to train a network on a representative set of input/ target pairs and get good results without training the network on all possible input/output pairs. A typical Feed Forward Back Propagation network is shown below.

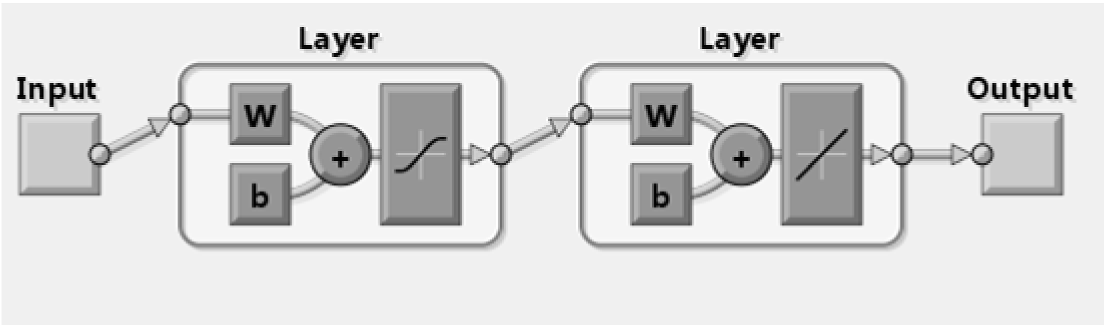

Figure 2: Feed Forward Back Propagation Network

This is the neural network used here. The model used here follows the equation:

$$Y\ (t+1) = f\{Y\ (t),\ Y\ (t-1),\ Y\ (t-2),\ ...,\ Y\ (t-a)\} + e$$

Where Y (t+1) is tomorrows exchange rate. Y (t), ...Y (t-a) are the time lagged spot exchange rate for past “a” days in sequence and “e” is the error

Suppose we want to train a neural network with “a” time lagged observations in the training set, we need a network of “an” input nodes, one output node and N-a training patterns. The initial training pattern will be Y (1), Y (2),...Y (a) and the output will be Y (a+1). This is repeated with patterns like Y (2), Y (3), ...Y (a+1) with output as Y (a+2) and so on. In this way one step ahead foreign exchange rate can be forecasted. The performance measure used is the mean square value.

4. Data

The data taken here is the daily spot rates of USD/INR (US Dollar / Indian National Rupee). The time period of data taken is 1st November 2006 to 8th December 2009. This period is chosen particularly because of the fact that it witnessed some very volatile times i.e. a period of slump from November 2006 to January 2008, then a growth period till late 2008 and then the volatile rise and falls. This effect is due to the period of recession since July 2007 which somewhat abated in 2009. Given below is the graph of volatility of FE data.

[1] Martin T. Hagan, et al (1996). Neural network design, Orlando De Jesus, Consultant, Texas, 2nd Ed.

As evident from the graph, the period of recession introduced extreme volatility in the forex rates. The data in question is taken from the free data section of www.kshitij.com, a foreign exchange trading and analysis organisation based in Mumbai. The rates are minimum daily spot rates (bid). Usually when a neural network is used, the data fed into it is normalised. But keeping in view some previous studies [Shanker, et al, 1996] which state that the normalisation process does not have any impact on predication as compared to the raw data fed into the network, we user aw data into the network.

Figure 3: Forex Spot Rates

5. Design

The aim of this paper is to study the efficiency of neural networks in turbulent times. This is done by creating experimental conditions under which we study the network. Without the hidden nodes, a neural network would simply be the linear statistical model. Thus the hidden nodes capture non linear time series patterns and detect complex relationships in the data. But the numbers of hidden nodes do matter. If we use too few hidden nodes, we risk the networks ability to capture the non linear relationships and thus would not be able to capture the patterns as it is nearly a linear model. On the other hand, with too many nodes, there is the problem of over fitting thus leading to poor forecasting ability. For a linear time series, is has been established that usually autoregressive terms of order 1 or 2 are sufficient for linear time series. But for non linear time series there is no such order. Thus we enquire with autoregressive terms of order 5. In our experiment, we use a total of 10 input sets with a total of 800 observations. They are as follows:

Set	Output	Inputs
1	Y(t+1)	Y(t)
2	Y(t+1)	Y(t), Y(t-1)
3	Y(t+1)	Y(t), Y(t-1), Y(t-2)
4	Y(t+1)	Y(t), Y(t-1), Y(t-2), Y(t-3)
5	Y(t+1)	Y(t), Y(t-1), Y(t-2), Y(t-3), Y(t-4)
6	Y(t+1)	Y(t), Y(t-1), Y(t-2), Y(t-3), Y(t-4),Y(t-5)
7	Y(t+1)	Y(t), Y(t-1), Y(t-2), Y(t-3), Y(t-4),Y(t-5), Y(t-6)
8	Y(t+1)	Y(t), Y(t-1), Y(t-2), Y(t-3), Y(t-4),Y(t-5), Y(t-6),Y(t-7)
9	Y(t+1)	Y(t), Y(t-1), Y(t-2), Y(t-3), Y(t-4),Y(t-5), Y(t-6),Y(t-7), Y(t-8)
10	Y(t+1)	Y(t), Y(t-1), Y(t-2), Y(t-3), Y(t-4),Y(t-5), Y(t-6),Y(t-7), Y(t-8),Y(t-9)

Figure 4: Autoregressive Scheme

As can be seen, there are ‘n’ time lagged inputs for each one step ahead output. For each set, neural networks with 5, 10, 15, 20, 25, 30 hidden layers are used. Thus a total of 60 feed forward back propagation neural networks are analysed.

To feed the data into the network, the data is divided into 70:15:15 ratio for Training, Validation, and out of sample testing. This is done by allocating the training procedure the first 70% of data, then the next 15% goes through the validation of the network just created and the last 15% for testing the performance of the network. All the three sets are divided in blocks and are sequential.

The network used in the study is a Feed Forward Back Propagation Network. It uses a tan sigmoid function in hidden layer and a pure linear function in output layer. This network can simulate any function with a finite number of discontinuities given a sufficient number of neurons. It is as shown in the diagram.

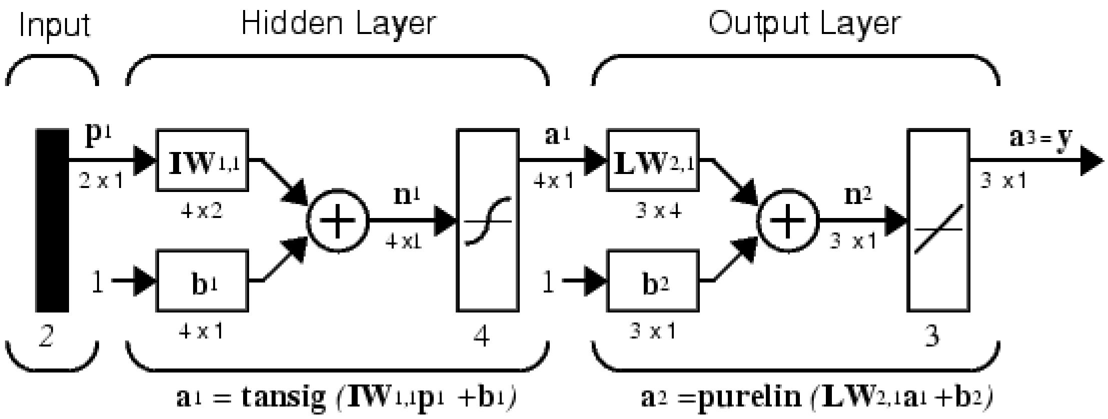

Figure 5: Three Layers of a Neural Network

The tan sigmoid function is given by:

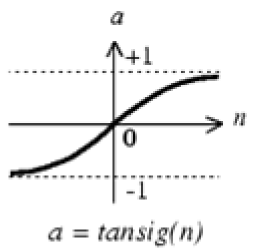

Figure 6: Tan Sigmoid Function (a = tansig (n) = 2/ (1+exp (-2*n))-1)

And the Linear function is given by: a = n

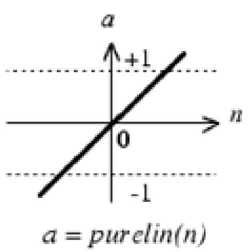

Figure 7: Linear Function

"For training purposes, Levenberg-Marquardt algorithm is used. The primary application of the Levenberg–Marquardt algorithm is in the least squares curve fitting problem: given a set of m empirical datum pairs of independent and dependent variables, (x_i, y_i), optimize the parameters $\boldsymbol{\beta}$ of the model curve $f(x,\boldsymbol{\beta})$ so that the sum of the squares of the deviations is minimal. Back propagation is used to calculate derivatives of performance *p* with respect to the weight and bias variables X. Each variable is adjusted according to gradient descent with momentum",

dX = mc*dXprev + lr*(1-mc)*dperf/dX

where dXprev is the previous change to the weight or bias.

The training parameters are as follows:

1. Number of Epochs: 1000
2. µ: 0.001
3. Minimum Gradient: 1 e-10
4. Goal: 0 error

All these initial conditions and activation functions etc are the standards for neural networks approximating a function. When any of the conditions (1, 3, 4) is reached, the network stops training and a stable point is reached. But sometimes this can cause the problem of over fitting. To overcome it, the network stops to train when a maximum of 6 validation errors occur in a row. This overcomes the over fitting problem of the neural network as validation stops the network before it starts to over fit i.e. continue training even when the performance criteria stops decreasing. Here the focus is on one step ahead forecast i.e. tomorrows exchange rate is forecasted using past "n" days data. This helps in minimising errors as one bad forecast does not affect the future predictions. For e.g. if Y (t+1) is a bad forecast, the Y (t+2) is not affected as it uses a dataset Y (t), ... Y (t-n) which is given and not forecasted. Thus for every forecast, there is a given data set.

6.0 Forecast Procedure

As stated earlier, we use up to 10 time lagged inputs for 5, 10, 15, 20, 25, and 30 number of hidden nodes. The software used here is MATLAB form Mathworks Inc. (Version R2009b). We then train the networks to get the MSE values for each set of testing, validation and training. After the networks are trained, we simulate to get output and calculate the Mean Absolute Error. The findings are presented in the following manner in Table 1. For each input node, the corresponding number of hidden nodes is shown and then their MSE values for each of training, validation and testing and then the values of MAE are presented.

The Validation MSE is the in-sample prediction error. This value is used to prevent the network from over fitting as well as authenticating how well the data had been fitted to the network. Since we have used the ratio 7:1.5:1.5, the first 560 data elements are used for training, next 120 are used for validation and last 120 are used for out of sample testing.

The testing set is used for out-of –sample testing and the MSE (testing) shows how well the network forecasts. The lesser the MSE, the better the forecasting capabilities of the model. Also, Directional Accuracy is calculated for the chosen network.

7. Performance Measure

7.1. MSE

To measure the performance of the network, we use the RMSE i.e. Root Mean Square Error. It is given by:

MSE = $(\grave{Y} - Y)^2 / T$

It compares the target output with the predicted values. The lower the value, the better the prediction is

7.2. MAE

The mean absolute error is a quantity used to measure how close forecasts or predictions are to the eventual outcomes. The mean absolute error (MAE) is given by

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|f_i - y_i| = \frac{1}{n}\sum_{i=1}^{n}|e_i| \,.$$

The mean absolute error is an average of the absolute errors $e_i = f_i - y_i$, where f_i is the prediction and y_i the true value.

7.3. DA

$$\mathrm{DA} = \frac{1}{N}\sum_{i=1}^{N} a_i \quad \text{where } a_i = \begin{cases} 1 & \text{if}(y_{i+1} - y_i)(\hat{y}_{i+1} - y_i) > 0 \\ 0 & \text{otherwise} \end{cases}.$$

The Directional Accuracy is important in terms of the fact that it tells about the movement of exchange rate rather than its exact magnitude.

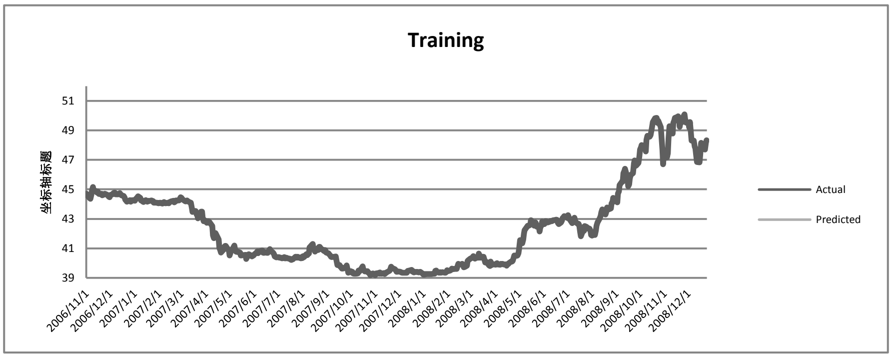

Figure 8: Training

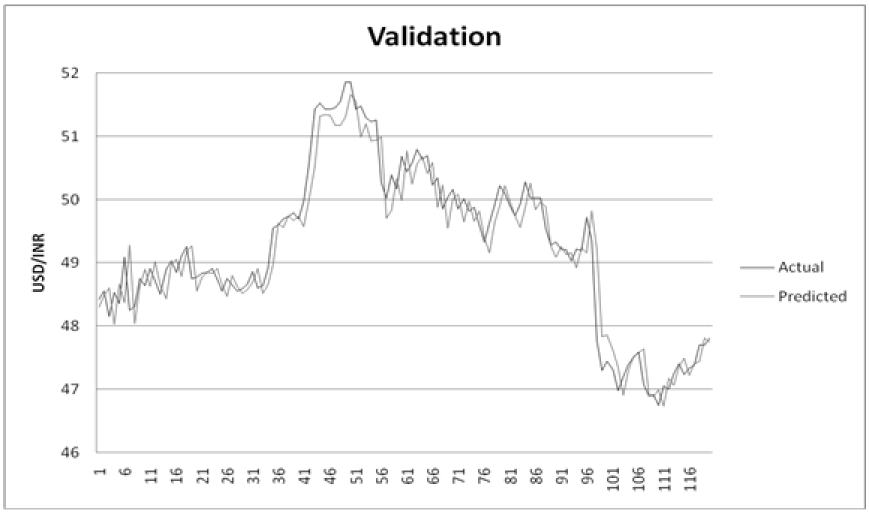

Figure 9: Validation

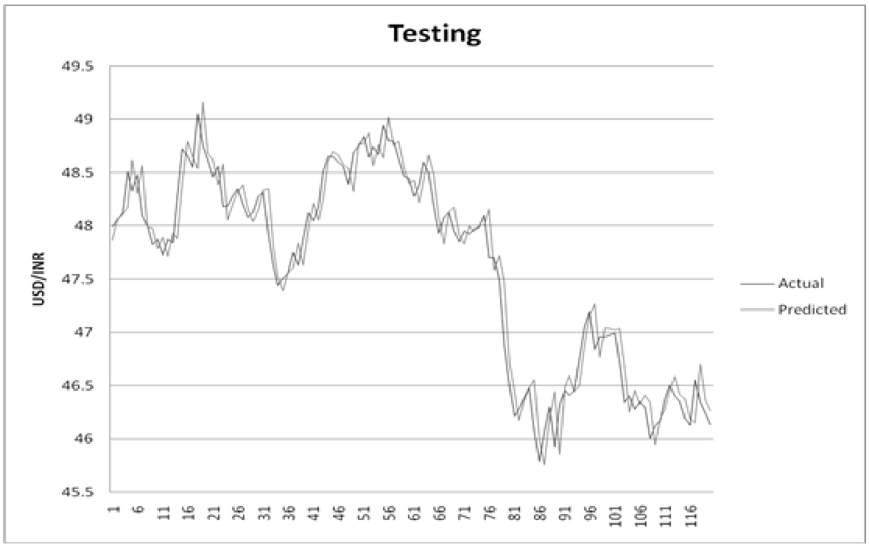

Figure 10: Testing

Table 1: Forecasting & Predictive Errors in stages of prediction.

No. Of Inputs	No. Of Hidden Nodes	Mean Square Error			Mean Absolute Error
		Training	Validation	Testing	
1	**5**	0.039846	0.144116	0.059715	0.162016
	10	0.056514	2.899485	0.057476	0.272052
	15	0.045189	0.3743	0.075889	0.20679
	20	0.055544	40.50274	0.115642	0.603764
	25	0.037712	15.48672	0.072604	0.395418
	30	0.033928	2.606558	0.076219	0.231283
2	**5**	0.03869	0.160091	0.04982	0.163224
	10	0.037515	1.006962	0.063143	0.20695
	15	0.04276	0.525755	0.087396	0.201328
	20	0.081118	0.383109	0.09343	0.265377
	25	0.03272	0.439819	0.065926	0.173315
	30	0.041904	0.421338	0.124542	0.207872
3	**5**	0.037463	0.159512	0.05852	0.163329
	10	0.037888	0.145588	0.056193	0.16294
	15	0.032014	0.215225	0.072275	0.168868
	20	0.066985	1.45679	0.09487	0.27531
	25	0.041236	0.326178	0.084961	0.198986
	30	1.463297	2.387329	2.502357	1.253272
4	**5**	**0.038099**	**0.120066**	**0.046964**	**0.155545**
	10	0.062116	0.168402	0.064563	0.190151
	15	0.062962	0.367272	0.20082	0.263351
	20	0.07103	0.261646	0.125164	0.261605
	25	0.031232	0.64187	0.081991	0.19667
	30	0.037233	0.187149	0.062676	0.177251
5	**5**	0.038791	0.129761	0.043883	0.157819
	10	0.054944	0.146742	0.061537	0.205685
	15	0.036703	0.123068	0.056756	0.157559
	20	0.032855	0.183281	0.081301	0.168061
	25	0.034983	0.133666	0.082011	0.169266
	30	0.162794	0.750025	0.233157	0.399519
6	**5**	0.140335	0.122378	0.051781	0.278535
	10	0.067153	0.184044	0.089492	0.236271
	15	0.034625	0.210294	0.0987	0.181614
	20	0.026154	0.347675	0.175511	0.194075
	25	0.039928	0.486818	0.141723	0.235344
	30	0.029862	0.216285	0.226069	0.197012
7	**5**	0.033466	0.1637	0.063696	0.165616
	10	0.041107	0.228863	0.103421	0.201341
	15	0.032692	0.158099	0.08687	0.168116

	20	0.035215	0.151256	0.092907	0.173368
	25	0.03557	0.332496	0.325062	0.232879
	30	0.256567	0.290375	0.244096	0.422504
8	5	0.032381	0.162581	0.094407	0.17001
	10	0.031957	0.243671	0.110017	0.183513
	15	0.064549	2.512016	0.443569	0.339436
	20	0.031603	0.181293	0.08626	0.173673
	25	0.324049	56.89263	0.148212	0.826163
	30	0.049895	0.30012	0.284829	0.246205
9	5	0.034227	0.174363	0.056372	0.167312
	10	0.031341	0.248513	0.103919	0.181633
	15	0.03203	0.176278	0.091721	0.172258
	20	0.307123	0.384829	0.256829	0.47431
	25	0.090682	1.070216	0.188688	0.342761
	30	0.02172	2.069917	3.012458	0.422607
10	5	0.030473	0.171818	0.083214	0.16666
	10	0.040242	0.166031	0.060342	0.177582
	15	0.028994	0.145636	0.071349	0.159122
	20	0.029627	0.150953	0.158602	0.176026
	25	0.026898	0.231498	0.103967	0.175999
	30	0.085945	0.143189	0.089037	0.255408

8. Results

Presented below are the values of MSE and MAE for the corresponding networks. It is of note that with any number of inputs, the validation error is least in case of 5 hidden nodes in layer 1 with the exception of cases when the number of inputs is 5, 7 and 10. Also, as the number of hidden nodes increase, the MAE values increase i.e. the MAE values are minimum at number of nodes equalling 5.

The optimal network structure according to the experiment turns out to be 4-5-1 i.e. 4 input nodes, 5 hidden nodes and one output node as the MSE values are minimum for this architecture at 0.012 . The accuracy of prediction of the ANN model in terms of MSE, MAE and DA are given below in the table. Presented below are also some graphical representations of the 4-5-1 structure's predictive performance as compared to actual values, in three phases.

Table 2: Prediction Error of a Neural Network (4-5-1)

S. No.	Measure	Neural Network
1.	MSE	0.05037
2.	MAE	0.155458
3.	DA	51.75%
4.	CORRELATION	0.998142

An analysis of complete data set reveals that the MSE values for both neural network is 0.05037. The neural network has a MAE of 0.155458l. A point to be noted here is the DA i.e. directional accuracy values. For neural network it is 51.75%. This measure of DA is of particular interest to players in the financial markets who are more interested in gaining insights to the directional change of tomorrow's exchange rate rather than its absolute value. Out of Sample forecast shows that the neural network's MSE 0.0469, MAE 0.1708 and the Pearson correlation coefficient between actual and predicted is 0.973 for the neural network. But it lags in terms of directional accuracy which is 51.67% for the neural network.

Table 3: Out of Sample Forecast - Prediction Error

S. No.	Neural Network (4-5-1)
MSE	**0.046964**
MAE	**0.170895**
DIRECTIONAL ACCURACY	**51.67 %**
CORRELATION	**0.973395**

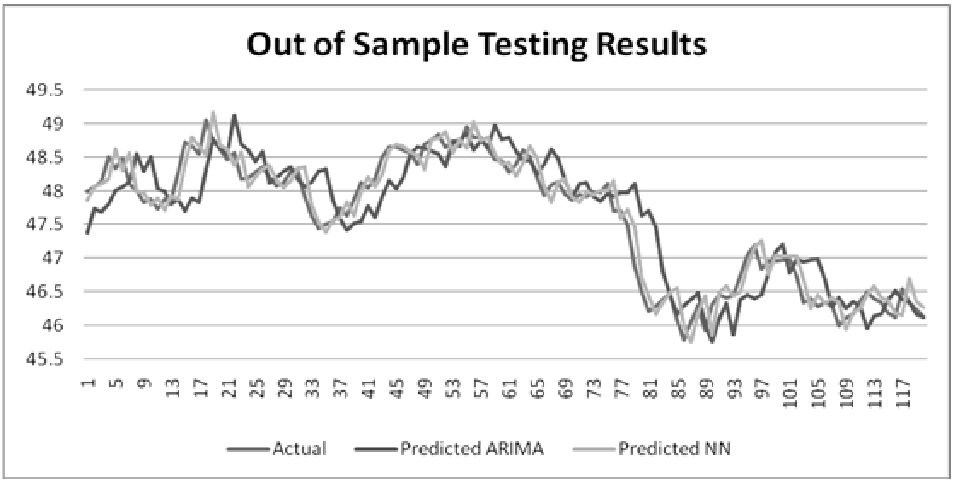

Fig. 10: Out of Sample Forecasting

9. Summary and Implications

This study gives an insight into the utility of artificial neural networks as a tool of forecasting in periods of recession. The three phases in neural networks, i.e. training, validation and testing are different in character, which is a limitation while using neural networks. During the earlier two periods the sign and weight of the parameters differ. And, yet the Neural Network is being trained to use this pattern to predict the out of sample forecast of foreign exchange rate.

This paper studies the forecasting of foreign exchange rate using neural networks and finds out predictive accuracy of forecasting foreign exchange rates in India with the help of MSE, MAE, DA and Correlation Coefficient. This finding also indicates that the exchange market participants' expectations are better modelled by neural network as compared to linear techniques. MSE and MAE are both small. But directional accuracy is only 51.67%. This is rather large in the case of out of sample forecasting.

The varying patterns in forex rates that are due to different phases precipitated by the Global Financial Crisis do not deter the neural networks from making accurate predictions of forex rates, whether 'within sample' or 'out of sample'. Thus, Neural Networks are an appropriate and fairly accurate method for forecasting foreign exchange rate during crisis.

10.References

[1] P. Chakradhara, & V. Narasimhan, Forecasting exchange rate better with artificial neural network, Journal of Policy Modelling, 2007, 29: 227–236.

[2] M. Klien & B. Mizarch, &Murphy, R.G. Managing the Dollar: has the plaza agreement mattered? *Journal of Money Credit and Banking*, 1991, **23:** 742-751

[3] Lean Yu et al *Foreign-Exchange-Rate Forecasting with Artificial Neural Network,* Springer Publishing Company, 2007, ISBN:0387717196 9780387717197.

[4] T. Martin Hagan et al. *Neural Network Design*, Orlando De Jesus, Consultant, Texas, 1996, 2nd Ed.

[5] R. A. Meese, & K. Rogoff,. Empirical exchange rate models of the seventies: Do they fit out of sample? *Journal of International Economics*, 1983, **14**: 3–24.

[6] J. Narendra, Exchange rate regime and capital flows: the Indian Experience, *Chief economist workshop*, April 4-6, 2005, Bank of England.

[7] V. Pacelli, Bevilacqua and M. Azzollini, "An Artificial Neural Network Model to Forecast Exchange Rates," Journal of Intelligent Learning Systems and Applications, 2011, 3(2): 57-69. doi: 10.4236/jilsa.2011.32008.

[8] C. Panda, & Narasimhan, V. Forecasting daily foreign exchange rate in India with artificial neural network. *Singapore Economic Review*, 2003, **48**(2): 181–199.

[9] A. Pankratz, *Forecasting with Univariate Box-Jenkins Models: Concepts and Cases*, Wiley, New York, 1983.

[10] Shanker, Murali, Y. Michael Hu and M.S. Hung. Effect of Data Standardization on Neural Network Training, *Omega*, 1996, **24**(4), 385-397.

[11] Shanker, Murali, Y. Michael Hu and M.S. Hung. Effect of Data Standardization on Neural Network Training, *Omega*, 1996, **24**(4), 385-397.