A Wavelet Inspired Neuro-Fuzzy Approach To Forecast Financial Data

Hemlata[1] Anil Kumar[2]
[1,2] Deptt. Of Computer Science and Engineering
[1, 2] P M College Of Engineering, Sonepat

Abstract—*The prediction algorithm always has their advantage in each field. While working with financial areas the importance of such approach significantly improve. An investor or analyst always wants some factor on the basis of which can take his financial decision accurately. The prediction algorithm basically reduces the risk factor. But main concern here is the accuracy level of the prediction, as the wrong prediction can be disaster for the investor. The proposed work is about to predict or forecast the financial outcome based on available statistics. Here we are using a neuro-fuzzy approach to predict the change in financial terms. Neural networks have gained use in economics and finance more recently. The networks have been used in issues like economic prediction, stock picking, and exchange rate prediction. This research will examine and analyse the use of neural networks as a forecasting tool. Specifically a neural network's ability to predict future values. While only briefly discussing neural network theory, this research will determine the feasibility and practicality of using neural networks as a forecasting tool for the individual investor. The general methodology required to build, train, and test a neural network using commercially available software. In this research, network models will be validated using recent data and provided a benchmark for further improvement. Here we are implementing the fuzzy logic as the final decision making. The work is about to optimize the prediction results.*

Keywords— Neural Network, Fuzzy Logic, Prediction, Financial, Forecast, back propagation.

I. INTRODUCTION

Stock market prediction has been an area of strong interest due to the possible of obtaining a very high return on the devoted money in a very short time. However, according to the efficient market theory [1], all such attempts at forecast are useless as all the information that could affect the conduct of stock price or the market index must have been already incorporated into the current market quotation. There have been several studies, for example, in [2], which question the efficient market hypothesis by showing that it is, in fact, possible to predict, with some degree of accuracy, the future behavior of the stock markets. Practical analysis has been used since a very long time but has had incomplete success [3]. Recently, soft computing techniques are being increasingly employed. Artificial neural networks (ANNs) have been widely used forecast of financial time series. Evaluation of the efficiency of time delay, recurrent and probabilistic neural networks for prediction of stock trends based on past data of the daily closing price is done in [4].Mixtures of technical indicators and ANNs have been used in [5],[6],[7] and [8] for forecasting of stock exchanges, exchange traded funds trading, determining buy/sell points for stocks and currency exchange rates, respectively. In [9], a fuzzy rule based system using technical directories as inputs, for stock price forecast, is presented. Technical pointers, which are widely used as input features to the prediction system, are of numerous types [6]. In fact, there are hundreds of different technical directories which are being employed today by technical analysts all over the world. Selection of the appropriate feature set from such a large set of features is a stimulating task. Dimensionality reduction of the selected feature set is of critical importance. It ensures that the information content in the dataset is preserved while optimally reducing the size of the dataset .Once the dimensionality reduction is accomplished; the next step is to input the reduced dataset to the prediction system. The accurateness and robustness of the forecaster is essential.

I. RELATED WORK

Chao Sun[10] performed a work. The article examines whether the stock market is predictable, and provides evidence that several basic financial and economic factors have predictive power for the market excess return.

Joyce Berg[11] performed a work," Accuracy and Forecast Standard Error of Prediction Markets". Author provide the first systematic evidence on the long-run predictive power of these markets by studying ex post accuracy and means of measuring ex ante forecast standard errors. Author use efficient markets theory and some special properties of the markets to develop forecast standard errors. Anthony J.T. Lee [12] performed a work," AN EFFECTIVE CLUSTERING APPROACH TO STOCK MARKET PREDICTION". In this paper, Author propose an effective clustering method, HRK (Hierarchical agglomerative and Recursive K-means clustering), to predict the short-term stock price movements after the release of financial reports. The proposed method consists of three phases.

Luna C. Tjung [13] performed a work," Forecasting Financial Stocks using Data Mining". The purpose of Presented paper is to compare the performance of ordinary least squares model and neural network model to see which model does a better job to predict the changes in the stock prices and identify critical predictors to forecast stock prices to increase forecasting accuracy.

NontokozoMpofu [14] performed a work," Forecasting Stock Prices using a Weightless Neural Network". In this research work, Author proposes forecasting stock prices in the stock market industry in Zimbabwe using a Weightless Neural Network (WNN). Author design and implement a neural network application that is used to demonstrate the application of the WNN in the forecasting of stock prices in the market.

Gabriel Pui Cheong Fung [15] performed a work," The Predicting Power of Textual Information on Financial Markets". In this paper, Author proposes a systematic framework for predicting the tertiary movements of stock prices by analyzing the impacts of the news stories on the stocks.

B. Wuthrich [16] performed a work," Daily Stock Market Forecast from Textual Web Data". In this paper, Author describes such an application driven data mining system. Presented aim is to predict stock markets using information contained in articles published on the Web. Mostly textual articles appearing in the leading and influential financial newspapers are taken as input.

Christian L. Dunis [17] performed a work," Neural Network Regression and Alternative Forecasting Techniques for Predicting Financial Variables". In this paper, Author examines the use of Neural Network Regression (NNR) and alternative forecasting techniques in financial forecasting models and financial trading models.

Mark T. Leung [18] performed a work," Forecasting stock indices: a comparison of classification and level estimation models". Despite abundant research which focuses on estimating the level of return on stock market index, there is a lack of studies examining the predictability of the direction / sign of stock index movement.

Jean-Philippe Peters [19] performed a work," Estimating and forecasting volatility of stock indices using asymmetric GARCH models and (Skewed) Student-t densities". This paper examines the forecasting performance of four GARCH(1,1) models (GARCH, EGARCH, GJR and APARCH) used with three distributions (Normal, Student-t and Skewed Student-t). Author explores and compares different possible sources of forecasts improvements: asymmetry in the conditional variance, fat-tailed distributions and skewed distributions.

Tarika Singh [20] performed a work," Volatilty of Stock Returns: A Case Study of Selected Asian Indices". This paper aims to contribute to the literature by investigating the relationship between trading volume and stock return volatility in selected Asian stock Indices by utilizing a relatively more recent database and extensive dataset.

Marc-André Mittermayer [21] performed a work," TEXT MINING SYSTEMS FOR PREDICTING MARKET RESPONSE TO NEWS". This paper describes the main systems developed to forecast price trends and presents a framework for comparing the approaches.

Georg Schneider [22] performed a work," Event-based Prediction of the Foreign Exchange Market Using News Text Classification". In this thesis, a nouvelle approach is taken to incorporate such information into prediction models.

Lahane Ashish Gajanan [23] performed a work," FINANCIAL FORECASTING Comparison Of ARIMA, FFNN and SVR Models". This study compares statistical models such as ARIMA, and AI models such as Feed Forward Neural Networks (FFNN) and Support Vector Regression (SVR), for long term one-day-ahead forecast of financial indices.

II. Present Work

A. Problem Definition

Ability to forecast market variables is critical to analysts, economists and investors. Among other uses, neural networks are gaining in popularity in forecasting market variables. They are used in various disciplines and issues to map complex relationships. This task gets complex as world financial markets get increasingly interconnected and interdependent. This complexity has created opportunities for neural networks which have the ability to explore interrelationships among a large number of market variables. Hence they are gaining popularity. The proposed work is about to present an efficient and more accurate approach to perform the prediction for the financial data such as gold rate or dollar rates. The approach presented here is wavelet inspired neuro fuzzy approach. Where wavelet provides the data cleaning initially and furthermore neural will perform the prediction task with boundation of logic. The work will give the reliable solution to the defined problem.

B. Objectives

The proposed research work will uses the following research objectives as the milestone:

- Study of existing prediction models on financial data.
- Collect the financial data set from secondary resources
- Implement the wavelet to remove impurities
- Implement neural to predict the variation
- Implement the fuzzy to set rule on prediction
- Analysis of result

C. Proposed Work

There are seven steps to develop financial forecasting model building system and normally involved in the manual approach. Each step deals with an important issue. They are data preprocessing, input and output selection, sensitive analysis, data organization, model construction, post analysis and model recommendation.

Step 1: Variable selection

Too many variables can unnecessarily overload the system. If we omit important variables, then its effect on the performance of the neural network can be significant. Knowing important variable are important for forecasting the critical value. The frequency of the data depends on the objective of the researchers. A typical off-floor trader in stock or commodity future market would likely used daily data if designing of neural network as a component of an overall trading system.

An investor with a long term horizon may use weekly or monthly data as input to the NN to formulate the best asset mix rather than using a passive buy and sold strategy.

Sensitivity Analysis is used to find out which indicator is more sensitive to the outputs. In other words, after a sensitivity analysis, we can easily eliminate the less sensitive variables from the input set. Usually, sensitivity analysis is used to reduce the number of fundamental factors. NN or some other forecasting models are used in forecasting as the forecast target is believed to have relationship with many other series.

Step 2: Data collection

The researcher must consider cost and availability when collecting data for the variables chosen in the previous step. Technical data is readily available from many vendors at a reasonable cost whereas fundamental information is more difficult to obtain. Time spent collecting data cannot be used for preprocessing training and evaluating network performance. Missing observation which often exists can be handled in a number of ways. All missing observation can be dropped or a second option is to assume that the missing observation is remaining the same by interpolating and averaging from nearby values.

Step 3: Data Preprocessing

A general format of data is prepared. Depending on the requirement, longer term data, e.g. weekly, monthly data may also be calculated from more frequently sampled time series. We may think that it makes sense to use as frequent data sampling as possible for experiments. However, researchers have found that increasing observation frequency does not always help to improve the accuracy of forecasting. Inspection of data to find outliers is also important as outliers make it difficult for NNs and other forecasting models to model the true underlying functional. Although NNs have been shown to be universal approximates, it had been found that NNs had difficulty modeling seasonal patterns in time series. When a time series contains significant seasonality, the data need to be depersonalized. Before the data is analyzed, basic preprocessing of data is needed.

Step 4: Data Organization

The next step is Data Organization. In data preprocessing step, we have chosen the prediction goal and the inputs that should be used. The historical data may not necessarily contribute equally to the model building. We know that for certain periods the market is more volatile than others, while some periods are more stable than others. We can emphasize a certain period of data by feeding more times to the network or eliminate some data pattern from unimportant time periods. With the assumption that volatile periods contribute more, we will sample more on volatile periods or vise versa. We can only conclude this from the experiments of particular data set. The basic assumption for time series forecasting is that the pattern found from historical data will hold in the future. Traditional regression forecasting model building uses all the data available. However, the model obtained may not be suitable for the future. When training NNs, we can hold out a set of data, out of- sample set apart from training. After the network is confirmed, we use the out-of-sample data to test its performance. There are tradeoffs for testing and training. One should not say it is the best model unless he has tested it, but once one has tested it one has not trained enough. In order to train NNs better, all the data available should be used. The problem is that we have no data to test the ``best" model. In order to test the model, we partition the data into three parts. The first two parts are used to train (and validate) the NN while the third part of data is used to test the model. But the networks have not been trained enough as the third part is not used in training. The general partition rule for training, validation and testing set is 70%, 20% and 10% respectively according to the authors' experience.

Step 5: Neural Network design

Model Construction step deals with NN architecture, hidden layers and activation function. A back propagation NN is decided by many factors, number of layers, number of nodes in each layer, weights between nodes and the activation function. In a sigmoid function is used as the activation function for a back propagation network. Similar to the situation of conventional forecasting models, it is not necessarily true that a complex NN, in terms of more nodes and more hidden layers, gives a better prediction.

– Selecting the Number of Hidden Layers

For nearly all problems, one hidden layer is sufficient. Two hidden layers are required for modeling data with discontinuities such as a saw tooth wave pattern. Using two hidden layers rarely improves the model, and it may introduce a greater risk of converging to a local minima. There is no theoretical reason for using more than two hidden layers. Three layer models with one hidden layer are recommended.

– Deciding how many neurons to use in the hidden layers

One of the most important characteristics of a perceptron network is the number of neurons in the hidden layer(s). If an inadequate number of neurons are used, the network will be unable to model complex data, and the resulting fit will be poor.

If too many neurons are used, the training time may become excessively long, and, worse, the network may over fit the data. When over fitting occurs, the network will begin to model random noise in the data. The result is that the model fits the training data extremely well, but it generalizes poorly to new, unseen data. Validation must be used to test for this. You specify the minimum and maximum number of neurons you want it to test, and it will build models using varying numbers of neurons and measure the quality using either cross validation or hold-out data not used for training. This is a highly effective method for finding the optimal number of neurons, but it is computationally expensive, because many models must be built, and each model has to be validated. It is important not to have too many nodes in the hidden layer because this may allow the NN to learn by example only and not to generalize.

When building a suitable NN for the financial application we have to balance between convergence and generalization. We use a one hidden layer network for our experiment. We adopt a simple procedure of deciding the number of hidden nodes which is also determined by the number of nodes in the input or preceding layer. For a single hidden layer NN, the number of nodes in the hidden layer being experimented are in the order of n2, n2 ±1, n2 ± 2 ,…, where n2 stands for half of the input number. The minimum number is 1 and the maximum number is the number of inputs, n, plus 1. In the case where a single hidden layer is not satisfactory, an additional hidden layer is added. Then another round of similar experiments for each of the single layer networks are conducted and now the new n2 stands for half of the number of nodes in the preceding layer. Besides the architecture itself the weight change is also quite important. The learning rate and momentum rate can lead to different models.

The crucial point is the choice of the sigmoid activation function of the processing neuron. There are several variations from the standard back propagation algorithm which aim at speeding up its relatively slow convergence, avoiding local minima or improving its generalization ability. e.g. the use of different activation functions other than the usual sigmoid function, the addition of a small positive offset to the derivative of the sigmoid function to avoid saturation at the extremes, the use of a momentum term in the equation for the weight change.

Although the back propagation algorithm does not guarantee optimal solution but observed that solutions obtained from the algorithm come close to the optimal ones in their experiments. The accuracy of approximation for NNs depends on the selection of proper architecture and weights;

however, back propagation is only a local search algorithm and thus tends to become trapped in local optima. Random selection of initial weights is a common approach. If these initial weights are located on local grades, the algorithm will likely become trapped at a local optimum. Some researchers have tried to solve this problem by imposing constraints on the search space or by restructuring the architecture of the NNs. For example, parameters of the algorithm can be adjusted to affect the momentum of the search so that the search will break out of local optima and move toward the global solution. Another common method for finding the best (perhaps global) solution using back propagation is to restart the training at many random points. Another issue with the back propagation network is the choice of the number of hidden nodes in the network. While trial-and-error is a common method to determine the number of hidden nodes in a network, genetic algorithms are also often used to find the optimum number In fact, in recent years, there has been increasing use of genetic algorithms in conjunction with NNs. The application of genetic algorithms to NNs has followed two separate but related paths. First, genetic algorithms have been used to find the optimal network architectures for specific tasks. The second direction involves optimization of the NN using genetic algorithms for search. No matter how sophisticated the NN technology, the design of a neural trading system remains an art. This art, especially in terms of training and configuring NNs for trading, and be simplified through the use of genetic algorithms. Traditional back propagation NNs training criterion is based on goodness-of-fit which is also the most popular criterion for forecasting.

Over fitting is another major concern in the design of a NN. When there is no enough data available to train the NNs and the structure of NNs is too complex, the NN tends to memorize the data rather than to generalize from it. Keeping the NN small is one way to avoid over fitting.

Step 6: Training the ANN

The training of a neural network is the process by which the neural network is presented with actual data from the process and uses this data (either off-line or continuously on-line) to find the most appropriate set of weights for each connection.

The goal of the training process is to find the set of weight values that will cause the output from the neural network to match the actual target values as closely as possible.

Training a NN to learn patterns in the data involves iteratively presenting it with examples of the correct known answer. The objective of training is to find the set of weights between the neurons that determine the global minimum of the error function. Unless the model is over fitted, this set of weights should provide good generalization. The back propagation network uses a gradient descent training algorithm which adjusts the weights to move down the steepest slope of the error surface. Finding the global minimum is not guaranteed since error surface can include many local minima in which the algorithm can become "stuck". A momentum term and the five to ten random sets of starting weights can improves the chances of reaching a global minimum.

III. Simulation And Results

A. *Simulation Tool*

1. MATLAB Editor is used for writing the code to implement our algorithm.
2. The result will be shown in the command window of MATLAB.

B. *Results*

In this present work we have implemented the forecasting process on Gold rates. The results obtained from the presented work are given as under

1) *Gold Price Curve*

As we can see in figure 5.1, it is showing the variation in gold price over the period. Here x axis represents the period or the time instances and the y axis represents the variation. As we can see as the time instances increases the variation is also increased.

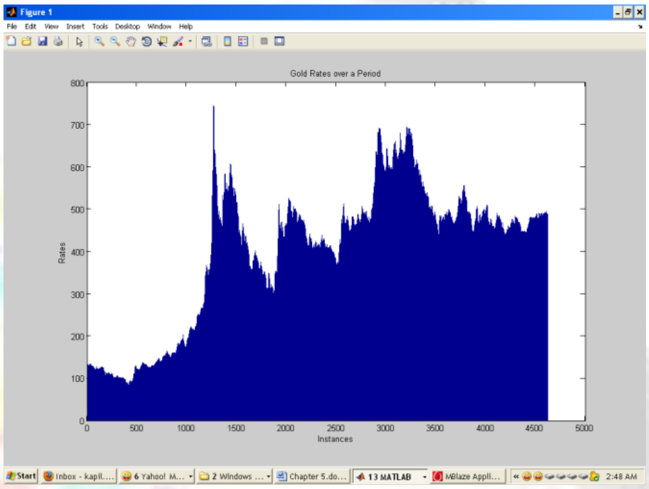

Fig. 1: Gold Price Curve

As we can see in figure 2, it is showing the variation in gold price over the period. Here only the curve is shown to shown the variation in better way. Effectively the results of this figure are same as the previous. Here x axis represents the period or the time instances and the y axis represents the variation. As we can see as the time instances increases the variation is also increased.

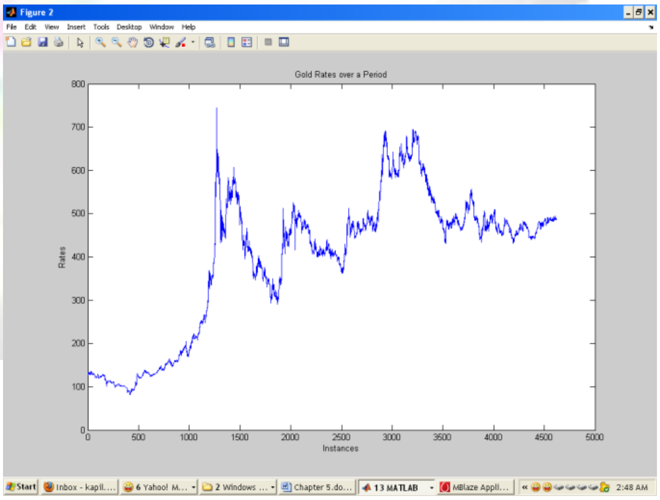

Fig.2 : Efficient Gold Price Curve

2) *Error curve*

As we can see in figure 3, it is showing the Error curve. As we can see in this figure as the training process is performed over the time, the error rate is reduced.

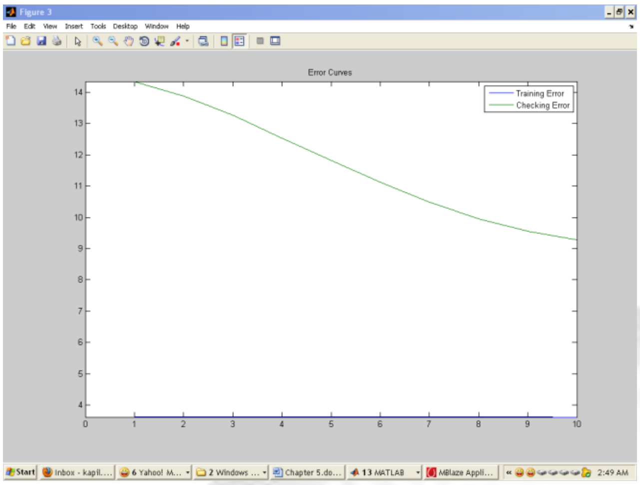

Fig. 3: Error Curve

3) Step curve

As we can see in figure 4, it is showing the step change in gold price over the period. As we can see there are three major changes in gold rates, It showing the major variation in gold rates over the period.

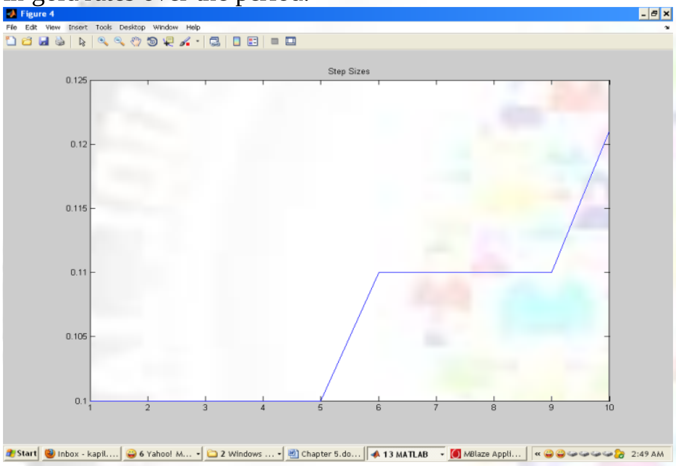

Fig. 4: Step Curve

C. Expected Fuzzy Output

As we can see in figure 5, it is showing the expected variation in the gold rates. As we can see, most of the time the cold rates are stable. The difference between the light and dark green lines showing the fluctuation in rates of gold

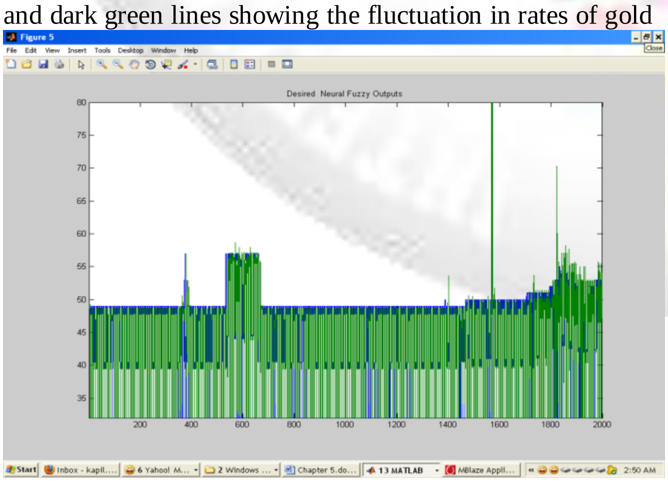

Fig. 5: Expected Fuzzy Output

As we can see in figure 6, here the fuzzy rule is shown that are implemented here to predict the gold rate, prediction error and the step size in the gold rate stability.

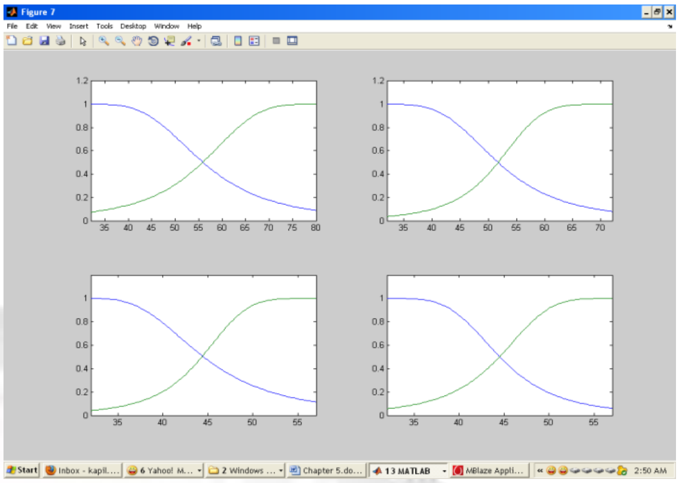

Fig. 6: Expected Fuzzy Output (error curve)

1) Fuzzy error curve

As we can see in figure 7(a) it is showing the Error curve. As we can see in this figure as the training process is performed over the time, the error rate is reduced. The result here obtains after the implementation of neuro fuzzy on current dataset.

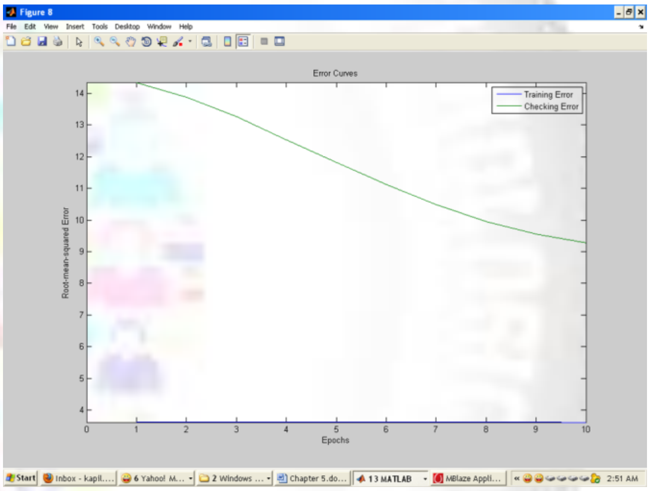

Fig. 7(a): Error Curve 2

As we can see in figure 7(b), it is showing the expected variation in the gold rates. As we can see, most of the time the cold rates are stable. The difference between the light and dark green lines showing the fluctuation in rates of gold. The result obtained here is after implementation of Neuro Fuzzy.

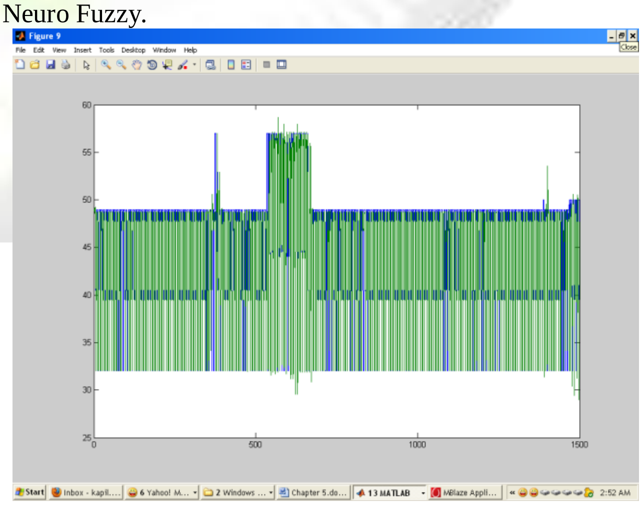

Fig. 7(b): Expected Fuzzy Output

2) Neuro–Fuzzy curve

As we can see in figure 8, it is showing the Analysis performed on gold rate prediction. As we can see most of times the result is stable. The stability here represents the true prediction of rates. Higher and lower lines showing the variation in prediction.

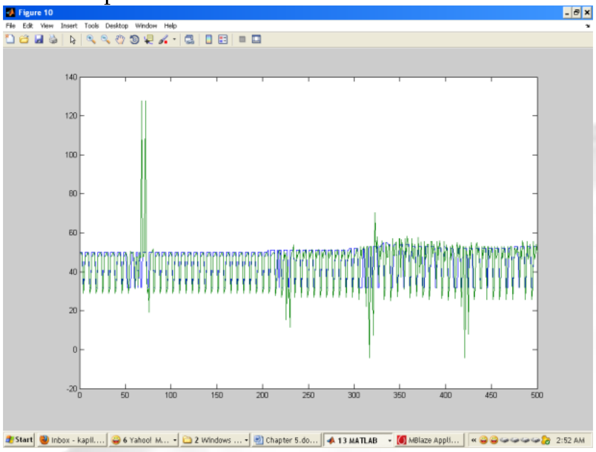

Fig. 8: Neuro Fuzzy Output

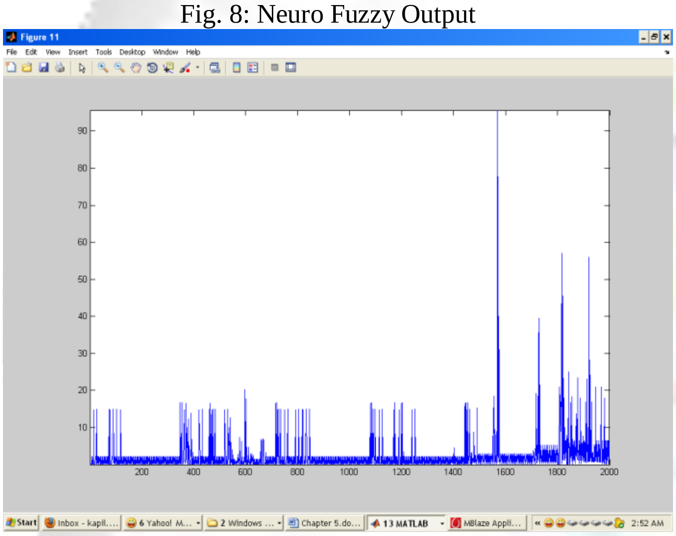

Fig. 9: Actual Neuro-Fuzzy Output

As we can see in figure 9, It is showing the prediction error while analyzing the gold rates. The flat surface is the indication of true estimation. Higher the line, more error in the prediction process. The work is here on the basis of Neuro Fuzzy rule Implementation.

IV. Conclusion And Future Work

Proposed System is suitable for financial forecasting and marketing analysis. They can be used for financial time series, such as foreign exchange rates forecasting. Proposed System is successful applied to the problem of forecasting the foreign currency exchange. When applying proposed model in a real application, attention should be taken in every single step. The architecture selection is a result of a long and time-consuming process of trial-and-error. This process is more an art than a science, more practice than theory.

References

[1] F. Fama , “Efficient capital markets: a review of theory and empirical work”, Journal of Finance, vol. 25, pp.383–417, 1970.

[2] G. S.Atsalakis, and K. P.Valavanis, “Surveying stock market forecasting techniques – part II: soft computing methods”, Expert Systems with Applications, vol.36, pp.5932–5941, 2009.

[3] D.R. Jobman, ed.,Technical Analysis for Futures Traders. New Delhi, India, Vision Books, 1988.

[4] E. W. Saad, D.V. Prokhorov, and, D.C. Wunsch, “Comparative study of stock trend prediction using time delay, recurrent and probabilistic neural networks”, IEEE Transactions on Neural Networks, Vol. 9, No. 6, pp. 1456-1470, 1998.

[5] C-T.Lee and Y-P.Chen, “The efficacy of neural networks and simple technical indicators in predicting stock markets”, International Conference on Convergence Information Technology, pp.2292-2297, 2007.

[6] E.L. de Faria, M.P. Albuquerque, J.L.Gonzalez, J.T.P. Cavalcante, and M. P. Albuquerque, “Predicting the Brazilian stock market through neural networks and adaptive, exponential smoothing methods”, Expert Systems with Applications, Vol. 36, pp.12506–12509, 2009.

[7] P-C. Chang , C-H. Liu, J-L. Lin, C-Y Fan, and Celeste S.P. Ng. , “A neural network with a case based dynamic window for stock trading prediction”, Expert Systems with Applications, vol. 36, pp.6889–6898, 2009.

[8] Y.Chen, S. Mabu, K. Shimada, and K. Hirasawa, “A genetic network programming with learning approach for enhanced stock trading model ”, Expert Systems with Applications, Vol. 36, pp.12537-12546, 2009.

[9] Pei-Chann Chang and Chen-Hao Liu, “A TSK type fuzzy rule based system for stock price prediction”, Expert Systems with Applications, vol. 34, pp.135–144, 2008.

[10] Chao Sun, “Stock Market Returns Predictability:Does Volatility Matter?”, 2008.

[11] Joyce Berg, “Accuracy and Forecast Standard Error of Prediction Markets”.

[12] Anthony J.T. Lee, “An Effective Clustering Approach To Stock Market Prediction”.

[13] Luna C. Tjung, “Forecasting Financial Stocks using Data Mining”.

[14] NontokozoMpofu, “Forecasting Stock Prices using a Weightless Neural Network”.

[15] Gabriel Pui Cheong Fung, “The Predicting Power of Textual Information on Financial Markets”, The Predicting Power of Textual Information on Financial Markets

[16] B. Wuthrich, “Daily Stock Market Forecast from Textual Web Data”.

[17] Christian L. Dunis, “Neural Network Regression and Alternative Forecasting Techniques for Predicting Financial Variables”.

[18] Mark T. Leung, “Forecasting stock indices: a comparison of classification and level estimation models”.

[19] Jean-Philippe Peters, “Estimating and forecasting volatility of stock indices using asymmetric GARCH models and (Skewed) Student-t densities”.

[20] Tarika Singh, “Volatilty of Stock Returns: A Case Study of Selected Asian Indices”, International Research Journal of Finance and Economics
[21] Marc-André Mittermayer, “Text Mining Systems For Predicting Market Response To News”, ISBN: 978-972-8924-40-9 © 2007 IADIS
[22] Georg Schneider, “Event-based Prediction of the Foreign Exchange Market Using News Text Classification”.
[23] Lahane Ashish Gajanan, “Financial Forecasting Comparison Of ARIMA, FFNN And SVR Models”.