International Journal of Innovative Research and Advanced Studies (IJIRAS)
Volume 3 Issue 8, July 2016

ISSN: 2394-4404

Hybrid Prediction Models For Stock Market

B. Dhamayanthi
MCA., M.Phil.,
Department of Computer Applications,
Madurai Kamaraj University, Madurai, Tamil Nadu

Dr. M. Ramaswami
M.Sc., M.C.A., M.Phil., Ph.D.,
Associate Professor, Department of Computer Applications,
Madurai Kamaraj University, Madurai, Tamil Nadu

Abstract: In this proposed work, three types of times series prediction models are discussing in share trading environment. The forecasting techniques includes Box-Jenkins method (ARIMA), as a classical statistical method and Artificial Neural Networks and hybrid methodology. Autoregressive integrated moving average (ARIMA) is one of the popular linear model in time series forecasting. In forecasting with artificial neural network (ANNs) suggests that ANNs can be promising alternative to the traditional linear methods like ARIMA models. ANN of Back Propagation neural network algorithm used to predict the stock price by establishing a three-tier structure of neural the neural network, namely input layer, hidden layer and output layer. The efficacy of these models are compared with several measures commonly used in forecasting statistical evaluation, for fitness and prediction phases. ARIMA and ANNs are often compared with mixed conclusions in terms of the superiority in forecasting performance accuracy.

Keywords: ARIMA, artificial neural network (ANN), Back Propagation, performance accuracy

I. INTRODUCTION

The stock market is one of the most recognizable and talked about areas of financial trading. Each day billions of dollars are exchanged by traders, buying and selling shares of individual companies on the stock market. Share is nothing but the Ownership of the company divided into small parts and each part is called as Share or Stock. Share is also called by different names like equity, financial security and so on. A person carrying a share of a company holds that part of ownership in that company. A person holding maximum shares carry maximum ownership and designated like director, chairman etc. A Share market is the place where buying and selling of shares takes place.

Forecasting is an approach to determine what the future holds. The need for forecasting significantly increases in the period of time due to the rapid changes in technology, government involvement in the econ, social and political changes, and globalization. It is essential to obtain an estimate of the changes as accurately as possible for companies to survive, to strive for operational excellence and to have a competitive advantage. Forecasting involves the generation of a number, set of numbers, or scenario that responds to a future occurrence. Based on the definition, forecasting is mainly based on past data.

There are three types of times series prediction models are discussing for prediction of share prices in share trading. The forecasting techniques which includes Box-Jenkins method (ARIMA) a classical statistical method and two soft computing models using Artificial Neural Networks and a hybrid method. The Hybrid methodology uses the combination of rough set and Artificial Neural Network model. To compare efficacy of these three models obtained performance fitness various computational measures were used.

DATA PREPARATION

Main source of data for entire study, we use mid-cap database. The data have been collected from nifty National Stock Exchange (NSE-50) auto index of TVS MOTORS, covering the period from January 2011 to December 2015. The data set covering trading details of daily Symbol, Series, Date, previous close, opening price, high price, last price, low price, closing price and total number of share traded (no.). A sample of 1200 records have taken for our study. After collecting data, the data are preprocessed and incorrect data are cleaned. The data set was divided into two groups for training and testing of proposed models.

II. ARIMA MODEL

In statistics and econometrics, and in particular in time series analysis, an Autoregressive integrated moving average (ARIMA) model is used which is a generalization of an autoregressive moving average (ARMA) model. These models are fitted to time series data for two specific reasons, one is to better understand the behavior or trend of data and to predict future points in the series (forecasting). In some cases when there is any evidence of non-stationarity, where an initial differencing step (corresponding to the "integrated" part of the model) can be applied to reduce the non-stationarity. Non-Seasonal ARIMA models are generally denoted ARIMA(p,d,q) where parameters p, d, and q are non-negative integers, p is the order of the Autoregressive model, d is the degree of differencing, and q is the order of the moving-average model.

A. MODEL IDENTIFICATION

The first step is to determine whether the time series data is stationary or non-stationary. A stationary time series is one whose properties do not depend on the time at which the series is observed. So time series with trends, or with seasonality, are not stationary - the trend and seasonality will affect the value of the time series at different times.

On the other hand, a white noise series is stationary - it does not matter when we observe it, it should look much the same at any period of time. Some cases can be confusing - a time series with cyclic behaviour (but not trend or seasonality) is stationary. That is because the cycles are not of fixed length, so before we observe the series we cannot be sure where the peaks and troughs of the cycles will be. If the original series is non-stationary, the series is converted into stationary by differencing the series. The order of differencing is zero for a stationary series and is greater than zero for the non-stationary series.

a. DIFFERENCING

Transformation the series to a new time series where the values are the differences between consecutive values. This procedure may be applied consecutively more than once, giving rise to the “first differences”, ”second differences”, etc.

✓ *First-order differencing*

The differenced series is the *change* between consecutive observations in the original series, and can be written as

$$y'_t = y_t - y_{t-1}.$$

The differenced series will have only $t-1$ values since it is not possible to calculate a difference y'_1 for the first observation.

✓ *Second-order differencing*

Occasionally the differenced data will not appear stationary and it may be necessary to difference the data a second time to obtain a stationary series:

$$y''_t = y'_t - y'_{t-1} = (y_t - y_{t-1}) - (y_{t-1} - y_{t-2}) = y_t - 2y_{t-1} + y_{t-2}.$$

In this case, y''_t will have $t-2$ values. Then we would model the “change in the changes” of the original data. In practice, it is almost never necessary to go beyond second-order differences.

b. TESTING FOR STATIONARITY

One way to determine more objectively if differencing is required is to use a *unit root test*. These are statistical hypothesis tests of stationarity that are designed for determining whether differencing is required or not.

A number of unit root tests are available, and they are based on different assumptions and may lead to conflicting answers. One of the most popular tests is the *Augmented Dickey-Fuller (ADF) test*. For this test, the following regression model to be estimated:

$$y't = \phi y_{t-1} + \beta_1 y'_{t-1} + \beta_2 y'_{t-2} + \cdots + \beta_k y'_{t-k},$$

where y'_t denotes the first-differenced series, $y'_t = y_t - y_{t-1}$ and k is the number of lags to include in the regression. If the original series, y_t, needs differencing, then the coefficient ϕ should be approximately zero. If y_t is already stationary, then $\phi < 0$.

The null-hypothesis for an ADF test is that the data are non-stationary. So large p-values are indicative of non-stationarity, and small p-values indicate stationarity. Using the usual 5% threshold, differencing is required if the p-value is greater than 0.05. The autocorrelation function (ACF) and partial autocorrelation (PACF) plots of the differenced series, can tentatively identify the numbers of AR and/or MA terms.

c. AUTO CORRELATION FUNCTION (ACF)

The time plot of the data, the ACF plot is also useful for identifying non-stationary time series. For a stationary time series, the ACF will drop to zero relatively quickly, while the ACF of non-stationary data decreases slowly. Also, for non-stationary data. After a time series has been stationarized by differencing, the next step in fitting an ARIMA model is to determine whether AR or MA terms are needed to correct any autocorrelation that remains in the differenced series.

Autocorrelation refers to the way the observations in a time series are related to each other and is measured by the simple correlation between current observation (Y_t) and observation from p periods before the current one(Y_{t-p}). That is for a given series Y_t, autocorrelation at lag p=correlation(Y_t,Y_{t-p}) and is given by

$$r_p = \frac{\sum_{t=1}^{n-p} (Y_t - \overline{Y})(Y_{t-p} - \overline{Y})}{\sum_{t=1}^{n} (Y_t - \overline{Y})^2}$$

The lag at which the ACF cuts off is the indicated number of MA terms.

d. PARTIAL AUTO CORRELATION FUNCTION (PACF)

The PACF plot is the partial correlation coefficients between the series and lags of itself. A partial autocorrelation is the amount of correlation between a variable and a lag of itself that is not explained by correlations at all lower-order-lags. The autocorrelation of a time series Y at lag 1 is the coefficient of correlation between Y(t) and Y(t-1), which is presumably also the correlation between Y(t-1) and Y(t-2). But if Y(t) is correlated with Y(t-1), and Y(t-1) is equally correlated with Y(t-2), then we should also expect to find correlation between Y(t) and Y(t-2). (In fact, the amount of correlation we should expect at lag 2 is precisely the square of the lag-1 correlation.) Thus, the correlation at lag 1 "propagates" to lag 2 and presumably to higher-order lags. The partial autocorrelation at lag 2 is therefore the difference between the actual correlation at lag 2 and the expected correlation due to the propagation of correlation at lag 1. The lag at which the PACF cuts off is the indicated number of AR terms.

B. MODEL PARAMETER ESTIMATION

The estimation of parameters is very important in the model building. Therefore, the parameters obtained are estimated statistically by least square method.

ARIMA (Auto Regressive Integrated Moving Average) model [1] is a generalization of an autoregressive moving average (ARMA) model. An ARMA model expresses the conditional Y_t mean of as a function of both past observations $Y_{t-1}\ldots Y_{t-p}$ and past innovations $\in_{t-1}\ldots \in_{t-q}$. The number of past observations that Y_t depends on, p, is the AR degree. The number of past innovations that $\in_t$ depends on, is the MA degree.

In general, these models are denoted by ARMA (p, q). The form of the ARMA (p, q) model is

$$y_t = C + \emptyset_1 y_{t-1} + \emptyset_2 y_{t-2} + \ldots + \emptyset_p y_{t-p} + \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \ldots + \theta_q \epsilon_{t-q}$$

Where ∈ is an uncorrelated innovation process with mean zero

C – Constant term

AR – Nonseasonal AR coefficients ($\emptyset_1,\ldots.\emptyset_p$)

MA - Nonseasonal MA coefficients ($\theta_1,\ldots.\theta_q$)

ARLags – Lags corresponding to nonzero, nonseasonal AR coefficients

MALags – Lags corresponding to nonzero, nonseasonal MA coefficients

D - Degree of nonseasonal differencing, D [if D has value 0 meaning no nonseasonal integration]

C. MODEL DIAGNOSTICS

Before forecasting the series, it is necessary to check the adequacy of the tentatively identified model. The model is declared adequate if the residuals cannot improve forecast anymore.

a. AKAIKE INFORMATION CRITERION (AIC)

The Akaike Information Criterion (AIC) is a way of selecting a model from a set of models. It is defined as:

AIC = -2 (log (L)) + 2 (p+q+k+1)

Where L is likelihood of the data, k=1 if (constant term) c ≠ 0,k=0 if c=0. The smaller AIC value is the better model. To check the residuals of estimated ARIMA model(s), if they are white noise.

Maximum Likelihood estimation:

This technique used to finds the value of the parameters which maximize the probability of the data that what we have observed. For ARIMA models, Maximum Likelihood Estimation (MLE) is very similar to the least squares estimates that would be obtained by minimizing the squared error.

b. LOG LIKELIHOOD

It is used to finds logarithm of the probability of the observed data from the estimated model. The values of p, d, q to maximize the log likelihood when finding the parameter estimation.

c. GOODNESS OF FIT

For choosing the best model for forecasting, used the measures for, Root Mean Square Error (RMSE) and Mean absolute error (MAE).

D. FORECASTING

Once the model adequacy is established, the series is forecasted for a specified period of time. It is always advisable to keep track of the forecast errors. Thus, depending on the magnitude of the errors, the Model shall be re-evaluated.

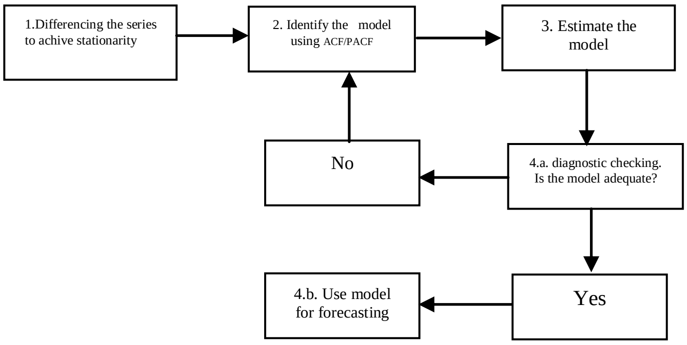
1.Differencing the series to achive stationarity
2. Identify the model using ACF/PACF
3. Estimate the model
No
4.a. diagnostic checking. Is the model adequate?
4.b. Use model for forecasting
Yes

Figure 1: Overall process of arima model

In order to construct the best ARIMA model for exchange rate time series, the autoregressive (*p*) and moving average (*q*) parameters are need to be identified for an effective model. After a time series has been stationarised by lag differencing, the next step in fitting an ARIMA model is to determine whether AR or MA terms are needed to correct any autocorrelation that remains in the differenced series. Autocorrelation (ACF) and partial-autocorrelation (PACF) correlograms were used to identify autoregressive term and moving average term. Since the ACF and PACF coefficients are not significant, we decided to determine the best model based on Akike Information Criterion (AIC) for various orders of autoregressive (p) and moving average (q) terms keeping integrated term (d) as order 1.

In this proposed work, the values of six price values viz of Previous Close, Open Price, High Price, Low Price, Last Price, Close price values are forecasted by ARIMA Model.

III. ARTIFICIAL NEURAL NETWORK

An artificial neural network is an information-processing paradigm that is inspired by the way of biological nervous system, such as the brain, process information. In this information processing system, the elements called as neurons, process the information. The signals are transmitted by means of connection links. The link possess an associated weight, which is multiplied along with the incoming signal (net input) for any typical neural net. The output signal is obtained by applying activations to the net input. The arrangement of neurons into layers and the pattern of connection within and in-between layer are generally called as the architecture of the net. There are various type of network architectures: Feed forward, feed back, fully interconnected net, competitive net, etc.

A. ARCHITECTURE

A multi layer feed forward back propagation network with one layer of Z-hidden units and Y output units.

The Y output unit has W_{ok} bias and Z hidden unit has V_{oj} bias.

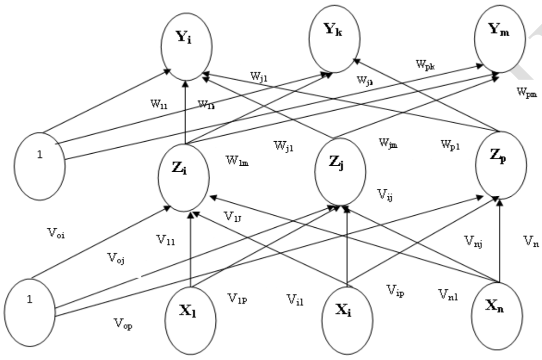

Figure 2: Back propagation Neural Network architecture

B. TRAINING ALGORITHM

The training algorithm of back propagation involves four stages:

✓ Initialization of weights
✓ Feed forward
✓ Back Propagation of errors
✓ Updating of the weights and biases

✓ *Parameters*

x: Input training vector
$x=(x_1,\ldots x_i,\ldots,x_n)$
t:Output target vector
$t=(t_1,\ldots,t_k,\ldots,t_m)$
$\boldsymbol{\delta}_k$ =error at output unit y_k
$\boldsymbol{\delta}_j$ =error at hidden unit z_j
$\alpha_{=}$learning rate
$v_{oj=}$bias on hidden unit j
$z_{j=}$hidden unit j
$w_{ok=}$bias on output unit k
$y_{k=}$output unit k.

The training algorithm used in the back propagation network is as follows. The algorithm is given with various phases:

✓ *Initialization of weights*

STEP 1: Initialize weights to small random values
STEP 2: while stopping condition is false, do steps 3-10
STEP 3: For each training pair do steps 4-9

✓ *Feed Forward*

STEP 4: Each input unit receives the input signal x_i and transmits this signals to all units in the layer above i.e. hidden units.

STEP 5: Each hidden unit (z_j ,j=1…p)sums its weighted input signals

$$Z_{\text{-}inj}=v_{oj+}\sum x_i v_{ij}$$

Applying activation function

$Z_j=f(z_{inj})$

And sends this signal to all units in the layer above i.e. output units

STEP 6: Each output unit (y_k,k=1…m) sum its weighted input signals

$$y_{\text{-}ink}=w_{ok+}\sum z_j w_{jk}$$

and applies its activation function to calculate the output signals

$$y_k=f(y_{\text{-}ink})$$

✓ *BACK PROPAGATION OF ERRORS*

STEP 7: Each output unit (y_k, k=1…m) receives a target pattern corresponding to an input pattern, error information term is calculated

$$\boldsymbol{\delta}_{k=}\ (t_k-y_k)f(y_{\text{-}ink})$$

STEP 8: Each hidden unit (z_j, j=1,..,n) sum its delta inputs from units in the layer above

$$\boldsymbol{\delta}_{in\text{-}j}=\ \sum \boldsymbol{\delta}_{i}w_{jk}$$

The error information term is calculated as

$$\boldsymbol{\delta}_{j=\text{-}}\boldsymbol{\delta}_{in\text{-}j}\ f(z_{\text{-}inj})$$

✓ *UPDATING OF WEIGHT AND BIASES*

STEP 9: Each output unit ($y_{k,}$ k=1,…,m) updates its bias and weights (j=0,..,p)

The weight correction term is given by,

$$\Delta w_{jk}=\alpha\boldsymbol{\delta}_k z_j$$

Antthe bias correction term is given by,

$$\Delta w_{ok=}\alpha\boldsymbol{\delta}_k$$

Therefore, w_{jk}(new)= w_{jk}(old)+ Δw_{jk}, w_{ok}(new)= w_{ok}(old)+ Δw_{ok}

Each hidden unit (z_j, j=1,..,p) updates its bias and weights (i=0,..,n)

The weight correction term

$$\Delta V_{ij}= \alpha\boldsymbol{\delta}_k z_j$$

And the bias correction term

$$\Delta V_{oj}= \alpha\boldsymbol{\delta}_j$$

Therefore, V_{ij}(new)= V_{ij}(old)+ $\Delta V_{ij,}$ ΔV_{oj}(new)= V_{oj}(old)+ ΔV_{oj}

STEP 10: Test the stopping condition.

The stopping condition may be the minimization of the errors, number of epochs etc.

C. NETWORK TRAINING

After the neural network model is constructed, training of the neural network is the next essential step of the forecasting model. Training of neural network is an iterative process of non linear optimization of the parameters like weights and bias of the network. The result of the training process of the network depends on the algorithm used for the purpose. A back propagation network uses a supervised learning method for training.

In one complete cycle of the training process, a set of input data {6 prices of Previous Close, Open Price, High Price, Low Price, Last Price, Close price} is presented to the input node. The corresponding target output, is presented to the output node in order to show the network what type of behavior is expected. The output signal is compared with the desired response or target output and consequently an error signal is produced. In each step of iterative process, the error signal activates a control mechanism which applies a sequence of corrective adjustments of the weights and biases of the neuron. The corrective adjustments continue until the training data attains the desired mapping to obtain the target output as closely as possible. After a number of iterations the neural network is trained and the weights are saved. The test set of data is presented to the trained neural network to test the performance of the neural network. The result is recorded to see how well the net is able to predict the output using the adjusted weights of the network. Two training algorithms, gradient descent adaptive back-propagation and gradient descent with momentum and adaptive learning back-propagation are considered in our model

In artificial neural network model at the hidden and output layer log sigmoid transfer function has been used and its range lies in [0, 1]. So before providing the data to the network, the data is to be normalized. Once the input normalized data is processed through the artificial neural network and the output obtained, it is denormalized. Methods for normalizing and de normalizing data are as described as follows,

D. NORMALIZED VALUES

A=(CURRENT VALUE - MINIMUM VALUE)/(MAXIMUM VALUE - MINIMUM VALUE)

where A is the normalized value which to be converted in the original numeric value,

Val=(A*(MAXIMUM VALUE - MINIMUM VALUE)+ MINIMUM VALUE)

Performance Measurement:

The performance measure of the neural network model relies on the prediction accuracy of the model. In the proposed work we used RMSE and MAE measures for comparing the forecasting accuracy.

E. ROUGH SET THEORY

Rough set theory can be regarded as a new mathematical tool for imperfect data analysis. It can be used for feature selection, feature extraction, data reduction, decision rule generation, and pattern extraction (templates, association rules) etc. The theory has found applications in many domains, such as decision support, engineering, environment, banking, medicine and others.

✓ Information/Decision Systems (Tables)
✓ Indiscernibility
✓ Set Approximation
✓ Reducts and Core

a. INFORMATION SYSTEM/ DECISION SYSTEMS (TABLES)

Rough set philosophy is founded on the assumption that with every object of the universe of discourse some information (data, knowledge) is associated**.** Rough set based data analysis starts from a data table called a decision table, columns of which are labeled by attributes, rows – by objects of interest and entries of the table are attribute values. Attributes of the decision table are divided into two disjoint groups called condition and decision attributes, respectively.

b. INDISCERNIBILITY RELATION

Objects characterized by the same information are indiscernible (similar) in view of the available information about them. The in-discernibility relation generated in this way is the mathematical basis of rough set theory. Any set of all indiscernible (similar) objects is called an elementary set, and forms a basic granule (atom)of knowledge about the universe. Any union of some elementary sets is referred to as a crisp (precise)set – otherwise the set is rough (imprecise, vague).

c. SET APPROXIMATION

The rough sets approach to data analysis hinges on two basic concepts, namely the lower and the upper approximations. Suppose we are given an information system IS=(U,A), X⊆U and B⊆A. Two operations assigning to every X⊆U two sets $\overline{B}$ (X) and $\underline{B}$ (X), called the B- upper and the B- lower approximation of X, respectively, and defined as follows

$$\underline{B}X = \{x \mid [x]_B \subseteq X\},$$

$$\overline{B}X = \{x \mid [x]_B \cap X \neq \phi\}.$$

Accuracy of Approximation

$$\alpha_B(X) = \frac{|\underline{B}(X)|}{|\overline{B}(X)|}$$

where |X| denotes the cardinality of $X \neq \phi$. Obviously

$$0 \leq \alpha_B \leq 1.$$

If $\alpha_B(X)=1$, *X* is *crisp* with respect to *B*.

If $\alpha_B(X)<1$, *X* is *rough* with respect to *B*

d. REDUCT & CORE

Keep only those attributes that preserve the indiscernibility relation and, consequently, set approximation. There are usually several such subsets of attributes and those which are minimal are called *reducts*. The set of all the condition attributes indispensable in *T* is denoted by *CORE(C).*

$$CORE(C) = \cap RED(C)$$

where *RED(C)* is the set of all *reducts* of *C.*

e. RULES GENERATION

Each row of a decision table induces a decision rule, which specifies decision (action, results, outcome, etc.) if some conditions are satisfied. If a decision rule uniquely determines decision in terms of conditions – the decision rule is certain. Otherwise the decision rule is uncertain. Decision rules are closely connected with approximations. Roughly speaking, certain decision rules describe lower approximation of decisions in terms of conditions, whereas uncertain decision rules refer to the boundary region of decisions. With every decision rule two conditional probabilities, called the certainty and the coverage coefficient, are associated. The certainty coefficient expresses the conditional probability that an object belongs to the decision class specified by the decision rule, given it satisfies conditions of the rule. The coverage coefficient gives the conditional probability of reasons for a given decision.

The proposed work, we have used a share trading's & database consists of 5 conditional attributes and one decision attribute. Rough set method have been used and find the lower and upper approximations, and subsequently rules, reduct values obtained.

IV. ROUGH-NEURAL METHOD

Rough neural networks used in this proposed work consist of one input layer, one output layer and one hidden layer of conventional neurons. The input layer neurons accept input from the rough set reduct attributes. The outputs from input layer neurons are fed to the hidden layer neurons. The hidden layer neurons feed their output to the output layer neurons which send their output to the external environment. The output of a rough neuron is a set of stock prices, while the output of a conventional neuron is a single value.

STEP 1:

In rough set theory For each decision attribute, to find the reduct value and produce the input of neural network.

STEP 2:

To train the neural network for using back propagation algorithm.

STEP 3:

To train the network for 1 to 8 hidden layers.

STEP 4:

To find the better accuracy result based on the measures of RMSE &MAE.

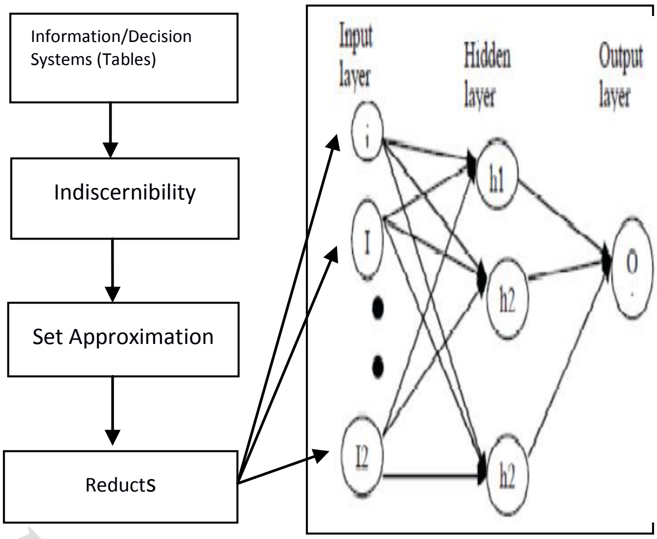

Figure 3: Overall process of rough-neural

V. RESULT AND ANALYSIS

A. COMPUTATIONAL MEASURES

The performance of all three models is analyzed against the share trading database. The performance results were given individually and finally comparative were made and suggests best model for market price prediction.

RMSE:

The Root Mean Square Error (RMSE) (also called the root mean square deviation, RMSD) is a frequently used measure of the difference between values predicted by a model and the values actually observed from the environment that is being modeled. These individual differences are also called residuals, and the RMSE serves to aggregate them into a single measure of predictive power. The RMSE of a model prediction with respect to the estimated variable X_{model} is defined as the square root of the mean squared error:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(X_{obs} - X_{forecast})^2}{n}}$$

where X_{obs} is observed values and $X_{forecast}$ is predicted values at time/place *i*.

MAE:

The MAE measures the average magnitude of the errors in a set of forecasts, without considering their direction. It measures accuracy for continuous variables. The MAE is the average over the verification sample of the absolute values of the differences between forecast and the corresponding observation. The mean absolute error (MAE) is a quantity used to measure how close forecasts or predictions are to the eventual outcomes. The mean absolute error 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.

B. COMPARITIVE STUDY OF ARIMA, ANN AND ROUGH-NEURAL MODELS

In this work, we compare the stock price prediction efficacy of both conventional and soft computing techniques of forecasting models. All three models viz; General Neural Network, Rough Neural Network and Conventional method of ARIMA models are compared against their predictive accuracy measure.

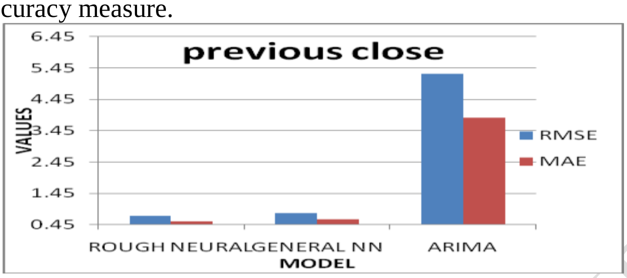

Figure 1

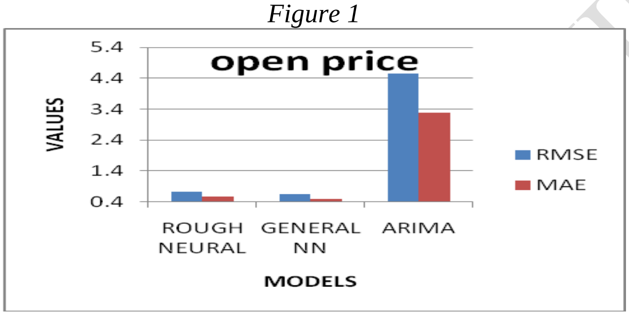

Figure 2

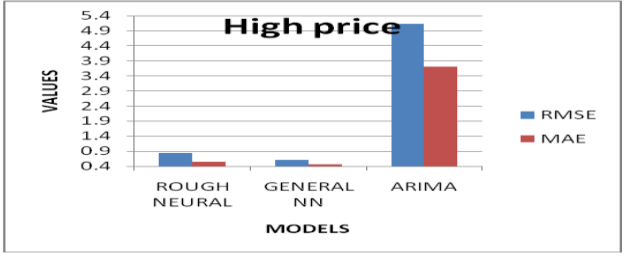

Figure 3

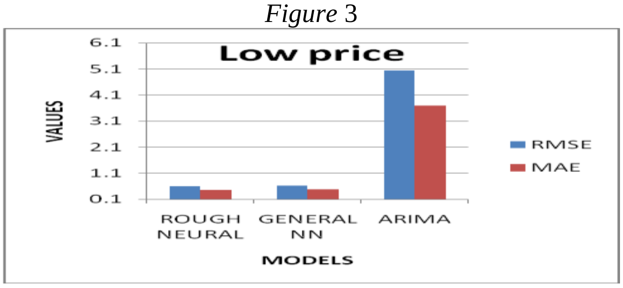

Figure 4

Figure 4 & 5: Comparison of ARIMA, ANN and ROUGH-NEURAL Models

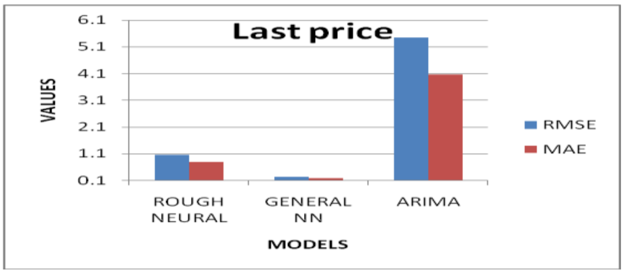

Figure 5

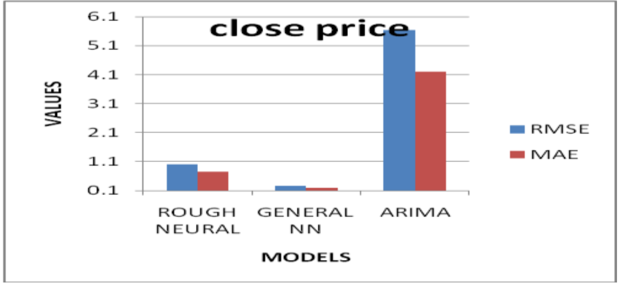

Figure 6

Figure 3,4,5,6: Comparison of ARIMA, ANN and ROUGH-NEURAL Models

VI. CONCLUSION

Thus in this work, three types of times series prediction model are analyzed which includes Box-Jenkins method (ARIMA), as a classical statistical method and Artificial Neural Network and Rough Neural Network –a hybrid methodology. For single price prediction models - ARIMA, ANN and Rough neural, general neural network model outperform other two models for prediction of 4 prices(Open price, High price, Last Price, Close price) and roughneural model take over other two models ARIMA and ANN for predicting prices of previous close price , and low price.

REFERENCES

[1] B. Uma Devi, D. Sundar, and P. Alli. "An Effective Time Series Analysis for Stock Trend Prediction Using ARIMA Model for Nifty Midcap-50.", International Journal of Data Mining & Knowledge Management Process (IJDKP) Vol.3, No.1, January 2013 (page no:65-78).

[2] Prapanna Mondal, Labani Shit and Saptarsi Goswami- “STUDY OF EFFECTIVENESS OF TIME SERIES MODELING (ARIMA) IN FORECASTING STOCK PRICES”, International Journal of Computer Science, Engineering and Applications m(IJCSEA) Vol.4, No.2, April 2014, (page no:13-29)

[3] Ayodele A. Adebiyi., Aderemi O. Adewumi, Charles K. Ayo-“Stock Price Prediction Using the ARIMA Model”, 2014 UKSim-AMSS 16th International Conference on Computer Modeling and Simulation(page no:105-111).

[4] Radga Alwadi- “FORECASTING SHORTTERM FINANCIAL DATA” European Scientific Journal,

edition vol.11 No.25 ISSN: 1857 – 7881, September 2015,(page no:251-255).

[5] Renhao Jin*, Sha Wang, Fang Yan, Jie Zhu-"The Application of ARIMA Model in 2014 Shanghai Composite Stock Price Index, Science Journal of Applied Mathematics and Statistics. Vol. 3, No. 4, August 3, 2015, ISSN: 2376-9491 (Print); ISSN: 2376-9513 (Online), (page no:199-203).

[6] Banhi Guha and Gautam Bandyopadhyay," Gold Price Forecasting Using ARIMA Model", Journal of Advanced Management Science Vol. 4, No. 2, March 2016, doi: 10.12720/joams.4.2.117-121.

[7] Mahdi Pakdaman Naeini, Hamidreza Taremian, Homa Baradaran Hashemi-" Stock Market Value Prediction Using Neural Networks" ,2010 International Conference on Computer Information Systems and Industrial Management Applications (CISIM).

[8] Bhagwant Chauhan, Umesh Bidave, Ajit Gangathade, Sachin Kale-"Stock Market Prediction Using Artificial Neural Networks", Bhagwant Chauhan et al, (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (1) , 2014, 904-907.

[9] Zabir Haider Khan, Tasnim Sharmin Alin, Md. Akter Hussain,"Price Prediction of Share Market using Artificial Neural Network (ANN)", International Journal of Computer Applications (0975 – 8887) Volume 22– No.2, May 2011.

[10] Manna Majumder,MD Anwar Hussian," FORECASTING OF INDIAN STOCK MARKET INDEX USING ARTIFICIAL NEURAL NETWORK" research by National Stock Exchange of India Limited,(page no:1-21).

[11] Sead Hesam Bostani, Sead Ehsan Sherafat, Sead Jamal Bostani, Ali Ranjbar," Stock Price Forecasting Using Artificial Neural Networks in Shiraz city",Journal of Applied Environmental and Biological Sciences, ISSN: 2090-4274, J. Appl. Environ. Biol. Sci., 5(9S)353-358, 2015.

[12] Mahezabin Shaikh, Mrs. G. J. Chhajed," Financial Forecasting using Neural Network", international Journal of Advanced Research in Computer Science and Software Engineering, Volume 5, Issue 1, January 2015,(page.no :66-70).

[13] Sunday Olusanya Olatunji,Mohammad Saad AI-Ahmadi,Moustafa Elshafee, Yaser Ahmed Fallatah, "Saudi Arabia Stock Prices Forecasting Using Artificial Neural Networks",ResearchGate, CONFERENCE PAPER, AUGUST, 2011, DOI:10.1109/ICADIWT.2011.6041425, (page.no :81-86).

[14] ay Desai, Arti Trivedi, Nisarg A Joshi-" Forecasting of Stock Market Indices Using Artificial Neural Network", Shri Chimanbhai Patel Institutes, Ahmedabad, Working Paper No. CPI/MBA/2013/0003,(page.no:1-18).

[15] G. Peter Zhang, "Time series forecasting using a hybrid ARIMA and neural network model", Neurocomputing 50 (2003), (page.no:159–175).

[16] Joanna collantes-duarte, Francklin Rivas-Echeverria" Time Series Forecasting using ARIMA, Neural Networks and Neo Fuzzy Neurons", (page.no :159–175).

[17] Ayodele Ariyo Adebiyi,Aderemi Oluyinka Adewumi,and Charles Korede Ayo-"Comparison of ARIMA and Artificial Neural Networks Models for Stock Price Prediction" Hindawi Publishing Corporation Journal of Applied Mathematics Volume 2014, (page.no :).

[18] Mustain Billah, Sajjad Waheed, Abu Hanifa -" Predicting Closing Stock Price using Artificial Neural Network and Adaptive Neuro Fuzzy Inference System (ANFIS): The Case of the Dhaka Stock Exchange", International Journal of Computer Applications (0975 - 8887), Volume 129 - No.11, November 2015, (page.no :1–5).

[19] Babu AS, Reddy SK , "Exchange Rate Forecasting using ARIMA, Neural Network and Fuzzy Neuron", J Stock Forex Trad 4: 155. doi:10.4172/2168-9458.1000155, 2015, (page.no:1–5).

[20] S. Cankurt and A. Subasi," Developing tourism demand forecasting models using machine learning techniques with trend, seasonal, and cyclic components", BALKAN JOURNAL OF ELECTRICAL & COMPUTER ENGINEERING, 2015, Vol.3, No.1, ISSN: 2147-284X, (page.no 42-49).

[21] IDRIS EL-FEGHI, ZAKARIA ZUBIA,A. ABOZGAYA," Air Temperature Forecasting using Radial Basis Functional Artificial Neural Networks", publications on ResearchGate, DATASET.DECEMBER 2013.

[22] Nitin Merh, Vinod P. Saxena and Kamal Raj Pardasani-" A COMPARISON BETWEEN HYBRID APPROACHES OF ANN AND ARIMA FOR INDIAN STOCK TREND FORECASTING", Business Intelligence Journal, 2010, (page.no 23-43).

[23] Adebiyi Ayodele A., Ayo Charles K., Adebiyi Marion O., and Otokiti Sunday O.,-" Stock Price Prediction using Neural Network with Hybridized Market Indicators", Journal of Emerging Trends in Computing and Information Sciences, VOL. 3, NO. 1, January 2012, (page.no 1-9).

[24] Kraimon Maneesilp, Chutima Prasartkaew-"Price Pattern Detection using Finite State Machines with Fuzzy Transitions", publications on ResearchGate CONFERENCE PAPER NOVEMBER 2014,DOI: 10.1109/ICEBE.2014.31.

[25] Yusuf Perwej, Asif Perwej," Prediction of the Bombay Stock Exchange (BSE) Market Returns Using Artificial Neural Network and Genetic Algorithm", Journal of Intelligent Learning Systems and Applications, 2012, 4, 108-119.

[26] Heino Bohn Nielsen-" Univariate Time Series Analysis ARIMA Models", Econometrics 2 — Fall 2005, 1of41.

[27] L.M. Bhar and V.K. Sharma,"TIME SERIES ANALYSIS",Indian Agricultural statistics Research institute, Library Avenue, page no:1-15

[28] Miroslav Kľúčik, Jana Juriová, Marian Kľúčik," Time Series Modeling with Genetic Programming Relative to ARIMA Models", page no:1-10.

[29] Robert Nau,, Fuqua School of Business, Duke University," The mathematical structure of ARIMA models" - 2014,page no: 1-8.

[30] Robert Nau,Fuqua School of Business, Duke University," Notes on nonseasonal ARIMA models", 2014,page no:1-21

[31] Panchal Amitkumar Mansukhbhai, Dr. Jayeshkumar Madhubhai Patel,” Role of soft computing techniques in predicting stock market direction”, December 2012, Vol. 1, No.2,page no:198-202.

[32] Selvan Simon and Arun Raoot “Accuracy driven artificial neural networks in stock market prediction”, International Journal on Soft Computing (IJSC) Vol.3, No.2, May 2012,page no:35-44.

[33] Brent Doeksen, Ajith Abraham, Johnson Thomas and Marcin Paprzycki,” Real Stock Trading Using Soft Computing Models”

[34] J. Mary Dallfin Bruxella, S. Arun Joe Babulo,” CNX PSU Bank Index Prediction Using Soft Computing - A Study”, International Journal of Computer Science and Mobile Computing Vol.2 Issue. 6, June- 2013, pg. 82-86.

[35] Zbigniew Suraj,” An Introduction to Rough Set Theory and Its Applications A tutorial”, Retrieved on: 29 Ju2015, researchgate, page no:1-39.