Forecasting GDP using ARIMA and Artificial Neural Networks Models under Indian Environment

Sunitha.G#1, Sampath Kumar.K#2, JyothiRani.S. A#3, Haragopal.V.V#4

[1]*Assistant Professor, Dept of Statistics, Vivekananda Govt. Degree College, Vidyanagar, Hyderabad – 500 044*
[2]*Assistant Professor, Dept of Statistics, Telangana University Dichpally, Nizamabad – 503 322*
[3]*Assistant Professor, Dept of Statistics, Osmania University, Hyderabad – 500 007*
[4]*Professor, Mathematics Department, BITS Pilani, Hyderabad Campus, Jawaharnagar, Shamirpet Mandal Telangana,*

Abstract

Gross Domestic Product (GDP) is one of the most important economic factors world over. India's growth majorly depends on the market and economy as a whole. In this paper an attempt is made to forecast the GDP growth. From the past experience it is evident that the variation in the GDP economy was cyclical. To see this behaviour we evaluate the analytics by considering the data drawn from Reserve Bank of India (RBI) for the period 1951 to 2016. Out of a variety of forecasting models, Autoregressive Integrated Moving Average (ARIMA) and Artificial Neural Network (ANN) – Multilayer Perception Model are evaluated to forecast the GDP. In this study Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE) are calculated for ARIMA model and ANN model. Using specifically RMSE and MAPE values, both the models are compared and it is observed from the analytics, that ANN is performing better than the traditional statistical models viz., ARIMA.

Keywords: *GDP, ARIMA, ANN, RMSE, MAPE, Forecasting.*

I. INTRODUCTION

GDP refers to the production of all goods and their services of a country or nation within a period of time and is one of the crucial factors of the economy which is to be measured annually. It is the aggregate statistic of all economic activities and captures the broadest coverage of the economy than other macro economic variables. It is the market value of all final goods and services produced within the borders of a nation in a year. It is often considered the best measure to see how the economy is performing. GDP can be measured in three ways. First, the Expenditure approach, it consists of household, business and government purchases of goods and services and net exports. Second the Production approach, it is equal to the sum of the value added at every stage of production (the intermediate stages) by all industries within the country, plus taxes and fewer subsidies on products in the period. Third is Income approach, it is equal to the sum of all factor income generated by production in the country (the sum of remuneration of employees, capital income, and gross operating surplus of enterprises i.e. profit, taxes on production and imports less subsidies) in a period (Yang, 2009- 2010).

GDP can be calculated by using following formula:

GDP = Consumption + Investment + Government Expenditure + Balance of trade. GDP prediction is a crucial job in the economy and business analysis. It is provides the way to set up future business plan and take timely decision about the financial market and economy.

The issues of GDP has become the most concerned amongst macro economy variables and data on GDP is regarded as the important index for assessing the national economic development and for judging the operating status of macro economy as a whole (Ning et al. 2010) . GDP is the aggregate statistic of all economic activity and captures the broadest coverage of the economy than other macro-economic variables. it is equal to the sum of all factor income generated by production in the country (the sum of remuneration of employees, capital income, and gross operating surplus of enterprises i.e. profit, taxes on production and imports less subsidies) in a period (Yang 2009; Ard 2010) Besides these, it is also the vital basis for government to set up economic developmental strategies and policies. Therefore, an accurate prediction of GDP is necessary to get an insightful idea of future health of an economy since the data on GDP is actually represented the past activities in summary form, which is not very helpful to frame suitable economic development strategies, economic policies and allocation of funds on different priorities for government as well as individual firms in a particular industry.

It needs a reliable estimate of GDP in some period ahead, which is only possible by forecasting GDP as accurately as possible by suitable sophisticated time series modeling, since it is not easy to identify the variables those effects on GDP precisely.

The researcher was motivated to undertake this study dealing with the GDP issues in India as not many studies have been done attempting to forecast the GDP as well as prediction of growth rates in various forms in India. However an attempt to forecast this macro variable only as point estimates has been of very little help for the managers and policy makers since variability is the key in decision making when certain level of risk is involved.

The objective of this paper is to forecast the GDP using Artificial Neural Network models, and to compare the forecasting performance of such non-linear models with traditional linear specifications like ARIMA models.

II. LITERATURE REVIEW

The literature on GDP process and its forecasting is organized into three sections. Section 2.1 is on theoretical literature review, section 2.2 is on ARIMA models while section 2.3 is Artificial Neural Networks Models:

2.1) GDP

Economic growth of a country is measured in terms of an increase in the size of a nation's economy. A broad measure of an economy's size is its output. The most widely-used measure of economic output is the GDP. The three basic ways to determine a nation's GDP are; the Expenditure approach, the Production approach and the Income approach.

The Expenditure Approach of determining GDP adds up the market value of all domestic expenditures made on final goods and services in a single year, including consumption expenditures, investment expenditures, government expenditures, and net exports. Add all of the expenditures together and you determine GDP.

The Production approach, also called the Net Product or Value added method requires three stages of analysis. First gross value of output from all sectors is estimated. Then, intermediate consumption such as cost of materials, supplies and services used in production final output is derived. Then gross output is reduced by intermediate consumption to develop net production.

The Income Approach of determining GDP is to add up all the income earned by households and firms in the year. The total expenditures on all of the final goods and services are also income received as wages, profits, rents, and interest income. GDP is determined by adding together all of the wages, profits, rents, and interest income.

The three methods of measuring GDP should result in the same number, with some possible difference caused by statistical and rounding differences. The credibility of data is always a significant concern in any form of research. An advantage of using the Expenditure Method is data integrity. The source data for expenditure components is considered to be more reliable than for either income or production components.

GDP as examined using the Expenditure Approach is reported as the sum of four components. The formula for determining GDP is:

$$C + I + G + (X - M) = GDP \qquad (1)$$

Where:

C = Personal Consumption Expenditures

I = Gross Private Fixed Investment

G = Government Expenditures and Investment

X = Net Exports

M = Net Imports

The GDP forecasting involves the application of both statistical and mathematical models to predict future developments in the economy. It allows economists to review past economic trends and forecast how recent economic changes will alter the patterns of past trends.

In forecasting macroeconomic time series variables like GDP, one has many possible types of models to choose from: vector error correction models, autoregressive conditional heteroskedasticity (ARCH)-based models, or various possible combinations. However, ARIMA models have proven themselves to be relatively robust especially when generating short-run GDP forecasts and have frequently outperformed more sophisticated structural models in terms of short-run forecasting ability. In a study, (Tsay and Tiao 1984, 1985) ARIMA model was used, which is in fact fitted on non-seasonal data by identifying autoregressive and moving average terms with the help of partial autocorrelation and autocorrelation functions (Box and Jenkins 1970:1976,Pankratz 1991).However, in the case of seasonal data, a number of studies used filtering approach, which in fact very helpful in case of weekly, monthly, quarterly and semiannual data to estimate a model to forecast any macro variable (Liu 1989; Liu and Hudak 1992; Liu 1999). In another research (Reynolds et al. 1995) automatic methods were developed to identify as well as estimate the parameters of ARIMA model by utilizing time-series data for a single variable. Another study (Reilly 1980) used similar methodology to model macroeconomic variable like GDP. The same study was also conducted (Bipasha Maity et al. 2012) for a period till 2020 using ARIMA Model. Automatic methods were developed to identify as well as estimate the parameters of ARIMA model by utilizing time-series data for a single variable (Reynolds et al. 1995). However, both the studies confined themselves only on non-seasonal time series data and restrained to predict the variable in future. However, the above mentioned methods need a long time-series data on the macroeconomic variable in question. To estimate the model for prediction of a macro variable, a number of studies imply analytical neural network techniques, which is very effective in the case of seasonal data (Chiu et al. 1995; Cook and Chiu 1997; Geo et al. 1997; Saad et al. 1998). These types of models have got pace since the seminal paper of Granger and Joyeux (1980) and Hosking (1981). However, this neural networking approach is very difficult to applying in real life situation by the policy makers /managers due to difficult network design, training and testing are required to build the model as well as to estimate the parameters.

In Indian context, not much effort has been seen in using non-linear models for forecasting macroeconomic variables. Genetically optimized neural network was used for forecasting daily exchange rate (Nag and Mitra 2000). Probit model and Artificial Neural Network were used to compare forecasting performance of turning points of business cycle with lead indicators (Bardoloi, presented in TIES 2007 conference). This paper evaluates the forecast accuracy of linear, non-linear time series models along with forecast combination (of linear and non-linear) for forecasting the GDP in respect of India.

2.2) Auto-regressive Integrated Moving Average (ARIMA) Models:

Autoregressive Integrated Moving Average models (ARIMA models) were popularized by George Box and Gwilym Jenkins in the early 1970s. It's an iterative process that involves four stages; identification, estimation, diagnostic checking and forecasting of time series.

ARIMA models are a class of linear models that is capable of representing stationary as well as non-stationary. They do not involve independent variables in their construction, but rather make use of the information in the series itself to generate forecasts. ARIMA models therefore, rely heavily on autocorrelation patterns in the data.

ARIMA methodology of forecasting is different from most methods because it does not assume any particular pattern in the historical data of the series to be forecast. It uses an interactive approach of identifying a possible model from a general class of models. The chosen model is then checked against the historical data to see if it accurately describes the series. Most of the traditional forecasting models therefore, provide a limited number of models relative to the complex behaviour of many time series with little guidelines and statistical tests for verifying the validity of the selected model.

2.2.1) Auto-Regressive (AR)

An autoregressive model of order *p*, an AR (*p*) can be expressed as;

$$y_t = c + \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \cdots + \alpha_p y_{t-p} \quad (3)$$

Where, $u_t \sim wn(0, \sigma^2)$.

The model is expressed in terms of past values and therefore, we wish to estimate the coefficients $\alpha_j, j = 1, \dots, p,$ and use the model for forecasting. In this case, all previous values will have cumulative effects on the current level y_t and thus, it is a long-run memory model. The ACF(s) therefore does not die out easily since it takes a longer time to have ACF close to zero.

Partial Autocorrelation Functions (PACF) measures the correlation between an observation k periods ago and the current observation, after controlling for observations at intermediate lags (i.e. all lags <*k*).

PACF (k) = ACF (k) after controlling the effects of $(y_{t-1}, \dots, y_{t-kt1})$. Thus PACF (k) can be found as the coefficient of y_{t-k} in the regression

$$Y_t = \alpha_0 + \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \cdots + \alpha_{k-1} y_{t-k+1} + \alpha_k y_{t-k} + u_t \quad (4)$$

$$\Rightarrow \alpha_k = \text{PACF (K)}$$

Hence the PACF is useful for telling the maximum order of an AR process.

2.2.2) Moving Average (MA) Process

This is a time series model which uses past errors as explanatory variable. Let u_t(t=1,2,3,...) be a white noise process, a sequence of independently and identically distributed (iid) random variables with E(u_t)=0 and Var(u_t) = σ^2. Then the *q*th order MA model is given as:

$$y_t = \mu + u_t + \theta_1 u_{t-1} + \theta_2 u_{t-2} + \cdots + \theta_q u_{t-q} \quad (2)$$

This model is expressed in terms of past errors and thus we estimate the coefficients $\theta_j, j = 1, \ldots, q,$ and use the model for forecasting. Therefore only q errors will affect the current level y_t but higher order errors do not affect y_t. This implies that it is a short memory model.

Auto-regressive (AR) models can be coupled with moving average (MA) models to form a general and useful class of time series models called Autoregressive Moving Average (ARMA) models. These can be used when the data are stationary.

2.2.3) Autoregressive Moving Average Model (ARMA)

ARMA (p, q) model is as follows:

$$y_t = c + \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \cdots + \alpha_p y_{t-p} + u_t + \theta_1 u_{t-1} + \theta_2 u_{t-2} + \cdots + \theta_q u_{t-q} \quad (5)$$

This is a combination of both AR and MA models. In this case therefore, neither ACF nor PACF can solely provide the information on the maximum orders of p or q.

This class of models can further be extended to non-stationary series by allowing the differencing of the data series resulting to Autoregressive Integrated Moving Average (ARIMA) models.

2.2.4) Autoregressive Integrated Moving Average (ARIMA) Process

There are a large variety of ARIMA models. The general non-seasonal model is known as ARIMA (p, d, q): where p is the number of autoregressive terms, d is the number of differences and q is the number of moving average terms. A white noise model is classified as ARIMA (0, 0, 0) since there exists no AR part because y_t does not depend on y_{t-1}, there is no differencing involved and also there's no MA part since y_t does not depend on e_{t-1}. For instance, if y_t is non-stationary, we take a first-difference of y_t so that Δy_t becomes stationary.

$\Delta y_t = y_t - y_{t-1}$ (d = 1 implies one time differencing)

$$\Delta y_t = c + \alpha_1 \Delta y_{t-1} + \alpha_2 \Delta y_{t-2} + \cdots + \alpha_p \Delta y_{t-p} + \theta_1 u_{t-1} + \theta_2 u_{t-2} + \cdots + \theta_q u_{t-q} + u_t \quad (6)$$

is an ARIMA (p, 1, q) model.

A random walk model is classified as ARIMA (0, 1, 0) because there is no AR and MA part involved and only one difference exists.

2.2.5) Box Jenkins Methodology

The Box Jenkins Methodology uses four iterative stages of Modeling that involves; identification, estimation, diagnostic checking and forecasting (See figure 1 below).

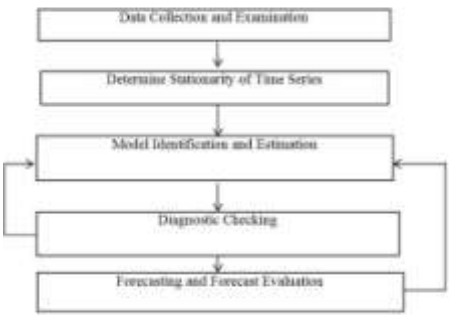

Figure 1. ARIMA forecasting procedure.

2.2.6) Model Identification

A preliminary Box-Jenkins analysis with a plot of the initial data should be run as the starting point in determining an appropriate model. The input data must be adjusted to form a stationary series and identify seasonality in the dependent series (seasonally differencing it if necessary), and using plots of the autocorrelation and partial autocorrelation functions of the dependent time series to decide which (if any) autoregressive (AR) or moving average (MA) component should be used in the model.

2.2.7) Model Estimation

The parameters of the selected ARIMA (p, d, q) model can be estimated consistently by least-squares or by maximum likelihood. Both estimation procedures are based on the computation of the innovations ε_t from the values of the stationary variable. The least-squares methods minimize the sum of squares;

$$min \sum_t \varepsilon^2 \quad (7)$$

The log-likelihood can be derived from the joint probability density function of the innovations $\varepsilon_1, \dots \varepsilon_T$, that takes the following form under the normality assumption, $\varepsilon_t \sim N.I.D(0, \sigma^2)$:

$$f(\varepsilon_1, \dots, \varepsilon_T) \ltimes \sigma^{-T} exp\left\{-\sum_{t=1}^{T} \varepsilon^2 / 2\sigma^2\right\} \quad (8)$$

In order to solve the estimation problem, equations 6 and 7 should be written in terms of the observed data and the set of parameters (Θ, ϕ, σ^2). An ARMA (p, q) process for the stationary transformation Z_t can be expressed as:

$$\varepsilon_t = Z_t - \sigma - \sum_{i=1}^{p} \phi_i Z_{t-1} - \sum_{i=1}^{q} \theta_i \varepsilon_{t-1} \quad (9)$$

Then, to compute the innovations corresponding to a given set of observations $(Z_1, \dots, Z_T)$ and parameters, it is necessary to count with the starting values $Z_0, \dots, Z_{p-1}, \varepsilon_0, \dots, \varepsilon_{q-1}$. More realistically, the innovations should be approximated by setting appropriate conditions about the initial values, giving to conditional least squares or conditional maximum likelihood estimators.

2.2.8) Diagnostic Checking

Before using the model for forecasting, it must be checked for adequacy (diagnostic checking). The model is considered adequate if the residuals left over after fitting the model is simply white noise and also the pattern of ACF and PACF of the residuals may suggest how the model can be improved.

Akaike's Information Criterion (AIC) is one of the most robust methods used in estimating parameters of an identified model.

$$AIC = -2 \log L + 2m \quad (10)$$

Where; L denotes the likelihood and m is the number of parameters estimated in the model such that;

$$m = p + q + P + Q \quad (11)$$

However, not all computer programs produce the AIC or the likelihood L, thus it is not always possible to find the AIC for a given model. A useful approximation to the AIC is therefore denoted as;

$$AIC = n(1 + \log(2\pi) + n \log \sigma^2 + 2m) \quad (12)$$

As an alternative to AIC, the Bayesian Information Criteria (BIC) and the Schwarz- Bayesian Information Criteria (SBC) are also used as model diagnostics. The SBC is given by;

$$SBC = \log \sigma + (m \log n)/n \quad (13)$$

2.2.9) Model Forecasting

Model forecasting states the difference between in-sample forecasting and out-of sample forecasting. In-sample forecasting for instance, explains how the chosen model fits the data in a given sample while Out-of-sample forecasting on the other hand, is concerned with determining how a fitted model forecasts future values of the regressand, given the values of the regressors.

To build a reliable model, the following factors are highly considered in forecasting;

a) The level of accuracy required – forecasts should be prepared as accurately as possible to facilitate the decision making process especially made on the basis of the GDP forecasts.

b) Availability of data and information – a wealth of reliable and up-to-date GDP data results to a reliable model.

c) The time horizon that the GDP forecast is intended to cover. This study for instance, covered a short run period.

2.3) Artificial Neural Networks

In recent years neural computing has emerged as a practical technology, with successful applications in many fields as diverse as finance, medicine, engineering, geology, physics and biology. The excitement stems from the fact that these networks are attempts to model the capabilities of the human brain. From a statistical perspective neural networks are interesting because of their potential use in prediction and classification problems. Artificial neural networks (ANNs) are non-linear data driven self-adaptive approach as opposed to the traditional model based methods. They are powerful tools for modeling, especially when the underlying data relationship is unknown. ANNs can identify and learn correlated patterns between input data sets and corresponding target values. After training, ANNs can be used to predict the outcome of new independent input data. ANNs imitate the learning process of the human brain and can process problems involving non-linear and complex data even if the data are imprecise and noisy. Thus they are ideally suited for the modeling of agricultural data which are known to be complex and often non-linear. A very important feature of these networks is their adaptive nature, where "learning by example" replaces "programming" in solving problems. This feature makes such computational models very appealing in application domains where one has little or incomplete understanding of the problem to be solved but where training data is readily available. These networks are "neural" in the sense that they may have been inspired by neuroscience but not necessarily because they are faithful models of biological neural or cognitive phenomena. In fact majority of the network are more closely related to traditional mathematical and/or statistical models such as non-parametric pattern classifiers, clustering algorithms, nonlinear filters, and statistical regression models than they are to neurobiology models. Neural networks (NNs) have been used for a wide variety of applications where statistical methods are traditionally employed. They have been used in classification problems, such as identifying underwater sonar currents, recognizing speech, and predicting the secondary structure of globular proteins. In time-series applications, NNs have been used in predicting stock market performance. As statisticians or users of statistics, these problems are normally solved through classical statistical methods, such as discriminant analysis, logistic regression, Bayes analysis, multiple regression, and ARIMA time-series models. It is, therefore, time to recognize neural networks as a powerful tool for data analysis.

An artificial neural network is a set of simple computational units that are highly interconnected. The units are also called nodes and loosely represent the biological neuron. A graphical presentation of neuron is given in Figure 1. A neuron is an information processing unit that is fundamental to the operation of a neural network. The connections between nodes are unidirectional and are represented by arrows in the figure. These connections model the synaptic connections in the brain. Each connection has a weight called the synaptic weight associated with it. The synaptic weight is interpreted as the strength of the connection from the j^{th} unit to the k^{th} unit. Unlike a synapse in the brain, the synaptic weight of an artificial neuron may lie in a range that includes negative as well as positive values. If a weight is negative, it is termed inhibitory because it decreases the net input. If the weight is positive, the contribution is excitatory because it increases the net input.

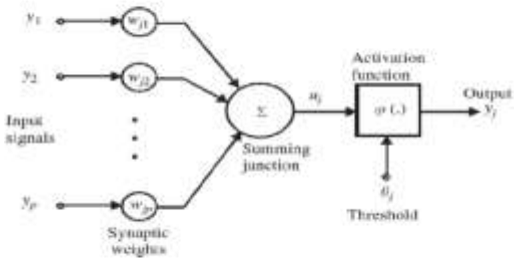

Figure 2: Nonlinear model of a neuron

The input into a node is a weighted sum of the outputs from nodes connected to it. Each unit takes its net input and applies an activation function to it. An activation function which is also known as squashing function, squashes or limits the amplitude range of the output of a neuron. The neuronal model of Figure 1 also includes an externally applied bias. The bias has the effect of increasing or lowering the net input of the activation function depending on whether it is positive or negative respectively. In mathematical terms, we may describe a neuron k by the following equations

$$Y_k = \phi(v_k) = \phi(\sum_{j=1}^{n} w_{kj} x_j + b_k)$$

Where $x_1, x_2, \ldots, x_n$ are the input patterns, $w_{k1}, w_{k2}, \ldots, w_{kn}$ are the synaptic weights of neuron k, b_k is the bias, Ø(.) is the activation function and Y_k is the output of the neuron. The sigmoid function, whose graph is s-shaped, is by far the most common form of activation function used in the construction of artificial neural networks. The neural networks are built from layers of neurons connected so that one layer receives input from the preceding layer of neurons and passes the output on to the subsequent layer.

2.3.1) Characteristics of Neural Networks

The NNs exhibit mapping capabilities, that is, they can map input patterns and also learn by examples. The NNs possess the capability to generalize. Thus, they can predict new outcomes from past trends. The NNs are robust systems and are fault tolerant. They can, therefore, recall full patterns from incomplete, partial or noisy patterns. The NNs can process information in parallel, at high speed, and in a distributed manner.

2.3.2) Neural networks architectures

An artificial neural network is defined as a data processing system consisting of a large number of simple highly inter connected processing elements (artificial neurons) in an architecture inspired by the structure of the cerebral cortex of the brain. There are several types of architecture of neural networks. However, the two most widely used NNs are discussed below:

Feed forward networks

In a feed forward network, information flows in one direction along connecting pathways, from the input layer via the hidden layers to the final output layer. There is no feedback (loops) i.e., the output of any layer does not affect that same or preceding layer.

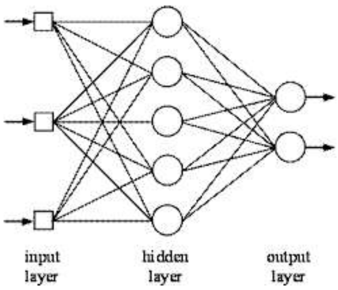

Figure 3: A multi-layer feed forward neural network

Recurrent networks

These networks differ from feed forward network architectures in the sense that there is at least one feedback loop. Thus, in these networks, for example, there could exist one layer with feedback connections as shown in figure below. There could also be neurons with self-feedback links, i.e. the output of a neuron is fed back into itself as input.

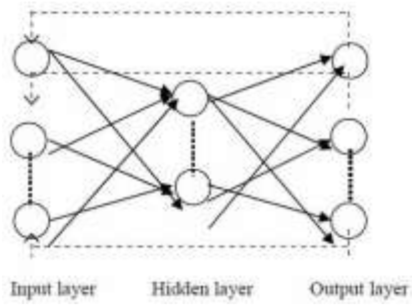

Figure 4: A Recurrent Neural Network

2.3.3) Learning/Training methods in Neural Networks :

Learning methods in neural networks can be broadly classified into three basic types: supervised, unsupervised and reinforced.

Supervised learning

In this, every input pattern that is used to train the network is associated with an output pattern, which is the target or the desired pattern. A teacher is assumed to be present during the learning process, when a comparison is made between the network's computed output and the correct expected output, to determine the error. The error can then be used to change network parameters, which result in an improvement in performance.

Unsupervised learning

In this learning method, the target output is not presented to the network. It is as if there is no teacher to present the desired patterns and hence, the system learns of its own by discovering and adapting to structural features in the input patterns.

2.3.4) Types of neural networks

The most important class of neural networks for real world problems solving includes Multilayer Perceptrons, Radial Basis Function Networks, Kohonen Self Organizing Feature Maps. In this study Multilayer Perceptrons, feed forward networks and Radial Basis Function Networks are used.

Multilayer Perceptron

The most popular form of neural network architecture is the multilayer perceptron (MLP) which is a generalization of the single-layer perceptron. Typically, the MLP network consists of a set of source nodes that constitute the input layer, one or more hidden layers of computation nodes and an output layer of computation nodes. The input signal propagates through the network in a forward direction on a layer by layer basis. MLP have been applied successfully to solve some difficult and diverse problems by training them in a supervised manner with a highly popular algorithm known as the error back-propagation algorithm. A multilayer perceptron has three distinctive characteristics:

The model of each neuron in the network includes a nonlinear activation function which should also be a differentiable everywhere. A commonly used form of nonlinearity that satisfies this requirement is a sigmoidal nonlinearity. The presence of nonlinearities is important because otherwise the input-output relation of the network could be reduced to that of a single layer perceptron.

The network contains one or more layers of hidden neurons that are not part of the input or output of the network. These hidden neurons enable the network to learn complex tasks by extracting progressively more meaningful features from the input pattens.

The network exhibits a high degree of connectivity determined by the synapses of the network. A change in the connectivity of the network requires a change in the population of synaptic connections or their weights. Given enough data, enough hidden units, and enough training time, an MLP with just one hidden layer can learn to approximate virtually any function to any degree of accuracy. (A statistical analogy is approximating a function with nth order polynomials.) For this reason MLPs are known as universal approximators and can be used when we have little prior knowledge of the relationship between inputs and targets. Although one hidden layer is always sufficient provided we have enough data, there are situations where a network with two or more hidden layers may require fewer hidden units and weights than a network with one hidden layer, so using extra hidden layers sometimes can improve generalization.

Radial Basis Function Networks

Radial basis function (RBF) networks have a very strong mathematical foundation rooted in regularization theory for solving ill-conditioned problems. RBF networks, almost invariably, consists of three layers: a transparent input layer, a hidden layer with sufficiently large number of nodes and an output layer. As its name implies, radially symmetric basis function is used as activation function of hidden nodes. The transformation from the input nodes to the hidden nodes is non-linear one and training of this portion of the network is generally accomplished by an unsupervised fashion. The training of the network parameters between the hidden and output layers occurs in a supervised fashion based on target outputs. MLPs are said to be distributed-processing networks because the effect of a hidden unit can be distributed over the entire input space. On the other hand, Gaussian RBF networks are said to be local-processing networks because the effect of a hidden unit is usually concentrated in a local area centered at the weight vector.

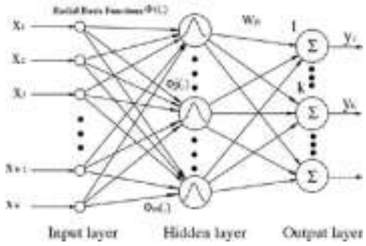

Figure 5: Radial Basis Function Architecture

III. Methodology

In this study, the GDP data analysed is collected from the official website (http://www.rbi.org.in) of Reserve Bank of India (RBI). The yearly aggregate GDP data (Market Price) at current prices is taken for the study from 1951-52 to 2015-16. For forecasting aggregate GDP, Box – Jenkins and Neural Network methods are used. In this section, the results of forecasting using these two methods are presented. The reported results are then analysed and compared. These two methods are compared by Root Mean Squared Error (RMSE) and Mean Absolute Percentage Error (MAPE) which are given by :

$$\text{MAPE} = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{Y_t - F_t}{Y_t}\right| \times 100$$

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n}\left(Y_t - F_t\right)^2}$$

Where Yt is the actual value and Ft is the forecasted value and n is the number of years used as forecasting period.

3.1) Box Jenkins Methodology:

In this methodology ARIMA(p,d,q) model is used which is a non seasonal ARIMA model where

p is the number of autoregressive terms, **d** is the number of nonseasonal differences needed for stationarity, and **q** is the number of lagged forecast errors in the prediction equation.

The ARIMA(p,d,q) model is defined as :-

$$\phi(B)\ \nabla^{d}\ Zt = \theta(B)\ a_t$$

Where,

$$\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \text{-----} - \phi_p B^p$$

is a polynomial in B of order ‘p’ and is known as AR operator

$$\theta(B) = 1 - \theta_1 B - \theta_2 B^2 - - - - - \theta_q B^q$$

is a polynomial in B of order ‘q’ and is known as MA operator

The difference operator is taken as $\nabla = 1 - B$, B is the Backward shift operator $B_k Z_t = Z_{t-k}$ and d is the number of differences required to achieve stationarity.

The forecasting of aggregate GDP(at Market Price) using Box Jenkins method for identifying the ARIMA model is done with SPSS software and the results are as follows:

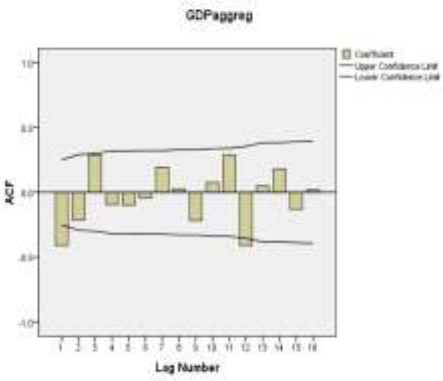

Figure 6 : ACF Plot

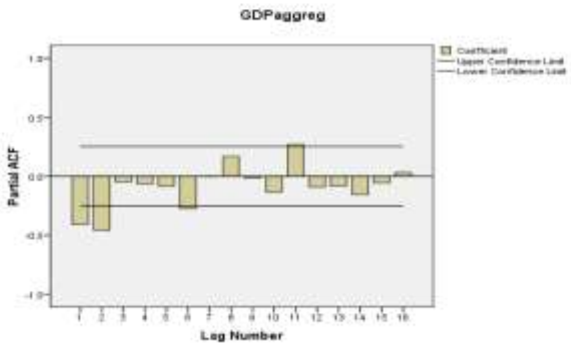

Figure 7 : PACF Plot

In model identification, ACF (Autocorrelation Function) and PACF (Partial Autocorrelation Function) are used. Based on this the tentative models will be assumed. Minimum BIC (Bayesian Information Criterion) value will decide which model will be the best among the tentatively assumed models as shown in the following table:

Box-Jenkins Model	BIC Value
ARIMA(1,2,1)	23.893
ARIMA(2,2,1)	24.013
ARIMA(1,2,2)	**23.140**
ARIMA(2,2,2)	23.967

Table 1: BIC value for tentative ARIMA models

From the above table, it is clear that ARIMA(1,2,2) would be the best model because it has minimum BIC value. Therefore, ARIMA (1,2,2) is the model used for forecasting future values.

$$Y_t = -0.762\ Y_{t-1} + (1 - 0.136\ B + 0.863\ B^2) a_t + 0.001$$

is the ARIMA(1,2,2) model.

3.2) Neural Network Method:

In this methodology, Multilayer Perceptron - Feed Forward Neural Network and Recurrent Neural Network architectures are used and also Radial Basis Function Network architecture is applied to analyse the data. These neural network models are trained with data partition as Training data set – 70%, Testing data set – 20% and Validation set – 10% using Neurosolutions as follows:

Multilayer Perceptron – Feed Forward Neural Network, Network Architecture: Feed Forward Neural Network, Training Algorithm: Backpropagation, Learning rate: 0.6, Momentum rate: 0.4, No. of observations: 65, Activation function: Sigmoid.

1) Multilayer Perceptron – Recurrent Neural Network, Network Architecture: Feed Back Neural Network, Training Algorithm: Backpropagation, Learning rate: 0.5, Momentum rate: 0.35, No. of observations: 65, Activation function: Sigmoid.

2) Radial Basis Function (RBF) Networks, Network Architecture: RBF Network(Feed forward), Training Algorithm: RBF learning algorithm(supervised), Learning rate: 0.5, Momentum rate: 0.3, No. of observations: 65, Activation function: Gaussian.

IV. Conclusion

In this study, two forecasting methods are presented :one is based on statistical models and the other is using ANN. The first method employed Box-Jenkins model which is usually used to forecast time series. In the second method, artificial neural network models (FFNN, RNN and RBF Network) are used to forecast future values. The two methods are applied to forecast the yearly Aggregate GDP (Market prices) of India (at current prices) from 1951-52 to 2015-16. For each method, the experimental results are given and analyzed based on statistical standards such as Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE).

Model	No. of Observations	RMSE	MAPE
ARIMA(1,2,2) Model	65	133.628	6.382
MLP-FFNN (with one hidden layer)	65	96.433	5.293
MLP – RNN (With one hidden layer)	65	73.726	4.811
RBF Network	**65**	**50.023**	**2.035**

Table 2: Comparative performance of Box-Jenkins and Neural Network methods

From the above Table , which gives the comparison between the statistical model and the ANN models which shows that the RBF Network model gives lower errors and higher accuracy for forecasting of yearly aggregate GDP (Market Prices). Therefore, the prediction done by ANN (RBF Network) will be more consistent and gives good forecasted observations.

Forecasted values for the next ten years (2016-17 to 2025-26) using RBF Network are as follows:

Year	**Forecasted values using RBF Network**
2016-17	14226436
2017-18	15382463
2018-19	17668292
2019-20	19036425
2020-21	22143162
2021-22	24496214
2022-23	27624192
2023-24	32142642
2024-25	38746328
2025-26	42692413

References

1. Cheng, B. and Titterington, D. M. (1994). Neural networks: A review from a statistical perspective. Statistical Science, 9, 2-54.
2. Dewolf, E.D., and Francl, L.J., (1997). Neural networks that distinguish in period of wheat tan spot in an outdoor environment. Phytopathalogy, 87, 83-87.
3. Dewolf, E.D. and Francl, L.J. (2000) Neural network classification of tan spot and stagonespore blotch infection period in wheat field environment. Phytopathalogy, 20, 108-113 .
4. Gaudart, J. Giusiano, B. and Huiart, L. (2004). Comparison of the performance of multi-layer perceptron and linear regression for epidemiological data. Comput. Statist. & Data Anal., 44, 547-70.
5. Girish K. Jha, Parimala Thulasiraman and Ruppa K. Thulasiram (2009). PSO based neural network for time series forecasting. In proceeding of the IEEE International Joint Conference on Neural Networks, USA, pp 1422-1427.
6. Hervai, S., Osborn, D. R., and Birchenhall. C. R. (2004). Linear versus neural network forecast M IV: 3: Artificial Neural Networks M IV-20 for European industrial production series. International Journal of Forecasting, 20, 435-446.
7. Kaastra, I. and Boyd, M.(1996). Designing a neural network for forecasting financial and economic time series. Neurocomputing, 10, 215-236.
8. Kohzadi, N., Boyd, S.M., Kermanshahi, B. and Kaastra, I. (1996). A comparision of artificial neural network and time series models for forecasting commodity prices. Neurocomputing, 10, 169-181.
9. Girish Kumar Jha Indian Agricultural Research Institute ARTIFICIAL NEURAL NETWORKS http://cabgrid.res.in/cabin/publication.
10. IEE Explore Digital Library.
11. Aminian, F., Suarex, E.D., Aminian, M., and Walz, D.T. 2006. Forecasting Economic data with Neural Networks, Computational Economics, 28, 71-88.
12. Binner, J. M., Bissoondeeal, R. K., Elger, T., Gazely, A.M., and Mullineux, A.W. 2005. A comparison of Linear Forecasting Models and Neural Networks: An Application to Euro Inflation and Euro Divisia, Applied Economics, 37, 665-680.
13. Kuan,, C and White, H. 1994. Artificial neural networks : an econometric perspective, Econometric Review, 13(Nov.).