FORECASTING OF INFLATION IN BANGLADESH USING ARIMA AND ANN MODELS

by

Rumana Hossain

A Project submitted in partial fulfillment of the requirements for the degree of Master of Engineering in Advanced Engineering Management

Department of Industrial and Production Engineering

Bangladesh University of Engineering and Technology

JUNE 2013

CERTIFICATE OF APPROVAL

The Project titled **"FORECASTING OF INFLATION IN BANGLADESH USING ARIMA AND ANN MODELS"** submitted by Rumana Hossain, Student No: 0411082106, Session: April, 2011, has been accepted as satisfactory in partial fulfillment of the requirement for the degree of MASTER OF ENGINEERING IN ADVANCED ENGINEERING MANAGEMENT on June 26, 2013.

BOARD OF EXAMINERS

____

Dr. Abdullahil Azeem
Professor — **Chairman**
Department of Industrial and Production Engineering, — (Supervisor)
Bangladesh University of Engineering and Technology (BUET)
Dhaka – 1000, Bangladesh.

____

Dr. M. Ahsan Akhtar Hasin
Professor — **Member**
Department of Industrial and Production Engineering
Bangladesh University of Engineering and Technology (BUET)
Dhaka – 1000, Bangladesh.

____

Dr. A. K. M. Masud
Professor — **Member**
Department of Industrial and Production Engineering
Bangladesh University of Engineering and Technology (BUET)
Dhaka – 1000, Bangladesh.

CANDIDATE'S DECLARATION

It is hereby declared that this thesis or any part of it has not been submitted elsewhere for the award of any degree or diploma.

Signature of the Candidate

____

Rumana Hossain

Dedication

To my parents, husband and son.

ACKNOWLEDGEMENT

First of all, I would like to thank Allah for giving me the ability to complete this thesis work.

I would like to express my sincere gratitude to my supervisor, Dr. Abdullahil Azeem. This project would not have been completed without his support and guidance. I would like to express my great thanks and gratefulness for his instructions, continuous encouragement, valuable discussions, and careful review during the period of this research. His keen sight and a wealth of farsighted advice and supervision have always provided me the precise guiding frameworks of this research. I have learned many valuable lessons and concepts of forecasts from him through my study, which I have utilized to develop my abilities to work innovatively and to boost my knowledge. His constant encouragement gave me the confidence to carry out my work.

I would like to thank all my teachers. They gave the knowledge and directions that have helped me throughout my life. I express my gratitude to my teachers from Bangladesh University of Engineering and Technology. The knowledge I learned from the classes in my M. Engineering level were essential for this project.

Last but not the least, I would like to thank by parents and my family. Their unconditional support made it possible for me to finish this project.

ABSTRACT

This study is to investigate forecasts of Bangladesh's inflation by the linear forecasting method like Autoregressive Integrated Moving Average (ARIMA) and Neural Network (NN) model as Nonlinear Autoregressive network with exogenous inputs (NARX). Inflation forecast is used as guide in the formulation of the monetary policy by the money policy makers worldwide. Monetary policy decisions are based on inflation forecast extracted from the information from different models and other indicators, which influence the macroeconomics conditions of the economy.

ARIMA method is an extrapolation method for forecasting, based on probability theory and statistical analysis with a certainty of distributions assumed in advance and like any other such methods, it requires only the historical time series data for the variable under forecasting. Five different plausible ARIMA estimated models are selected by various diagnostic and selection & evaluation criteria. On the basis of in sample and out of sample forecast and forecast evaluation statistics two candid models among the five models which have sufficient predictive powers and the findings are well compared to the other models are proposed. Artificial neural network (ANN) models are data-driven self- adaptive methods in that there are few a priori assumptions about the models for problems under study. An ANN model is developed to forecast the inflation of Bangladesh as a function of its own previous value. The model selects a feed-forward back-propagation ANN with the input of previous inflation and an exogenous variable of exchange rate, five hidden neurons and one output as the optimum network. The model is tested with actual time series data of inflation in case of Bangladesh and forecast evaluation criteria. The forecast performance of the ANN model is compared with ARIMA based model and observed that RMSE of ANN based forecasts is much less than the RMSE of forecasts based on ARIMA models. So it can be said that forecasting of inflation with ANN offers better performance in comparison with ARIMA methods.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Background

There are many attributes of the system to be forecast that impinge on the choice of method by which to forecast and on the likelihood of obtaining reasonably accurate forecast. The main purpose for constructing a time series model is to forecast. Such forecasts enable the policy makers to judge whether it is necessary to take any measure to influence the relevant economic variables. Broadly speaking there are four approaches to economic forecasting based on the time series data:

1) Single-equation regression model.

2) Simultaneous equation regression model.

3) Autoregressive integrated moving average (ARIMA) models.

4) Vector auto regression (VAR) models.

Scalar version of the time series models usually take the form proposed by Kalman(1960) or Box and Jenkins(1976). There are good reasons why integrated autoregressive moving average models (ARIMAs) might be regarded as the dominant class of the scalar time series models. The Wold decomposition theorem states that any purely indeterministic stationary time series has an infinite moving-average (MA) representation. Moreover any infinite MA can be approximated to any required degree of accuracy by an ARMA model. Typically the order of the AR and MA polynomials (p and q) required to adequately fit the series may be relatively low. Many economic time series may be non-stationary, but provided they can be made stationary by differencing, after that they are amendable to analysis within the Box-Jenkins framework, in which case we obtain ARIMA rather than ARMA models (where the order of the integrated component is d, the minimum number of times the series has to be differenced to be stationary).

A univariate autoregression is a single-equation, single-variable linear model in which the current value of a variable is explained by its own lagged values whereas a VAR (vector auto regression provided by Christopher Sims (1980)) is a n-equation, n variable linear model in which each variable is in turn explained by its own lagged values, plus current and past values of the remaining n-1 variables. VARs hold out the promise of providing a coherent and credible approach to data description, forecasting, structural inference, and policy analysis.

Around the world keeping a strong control over Inflation has turned out to be one of the primary objectives of the regulators as inflation increases uncertainty both in consumer's and producer's mind. As the economic effect of monetary policy have time lag policy makers and financial authorities require frequent updates to the path of inflation. Policy makers can get prior indication about possible future inflation through Inflation forecasting using univariate time series auto regressing integrated moving average (ARIMA) models. The intrinsic nature of a time series is that successive observations are dependent or correlated and therefore, statistical methods that rely on independent assumptions are not applicable. Time series analysis studies the stochastic mechanism of an observed series. The study of time series helps to understand and describe the underlying generating mechanisms of an observed series. This analysis assists in forecasting future values and to estimate the impact of events or policy changes. Results from analysis can give valuable information when formulating future policies.

Traditional models that have been used to inflation forecasting are all based on probability theory and statistical analysis with a certainty of distributions assumed in advance. In most cases these assumptions are unreasonable and nonrealistic. Also the linear structure of these models doesn't guarantee accuracy of prediction. Recent studies have addressed the problem of inflation rate forecasting by using different methods including artificial neural network (ANN) and model based approaches due to the significant properties of handling non-linear data with self learning capabilities.

There has been a great interest in studying the artificial neural network (ANN) forecasting in economics, financial, business and engineering applications including GDP growth, stock returns, currency in circulation, electricity demand, construction demand and exchange rates.

This thesis intends to contribute to a better understanding of the problem related to the inflation prediction. To accomplish this goal we will try to compare linear models (for instance, ARIMA models) with non-linear models (for instance, Artificial Neural Networks), namely, if the forecast and prediction power improves from one technique to other. Therefore, we will compare various ARIMA models and find out the best one. Then I will compare that ARIMA model with ANN model (a more conventional technique based on seasonal and trends models with a non-linear methodology) and check which of the methods is the most effective.

There has been increasing interest in the application of neural networks to the field of finance. Several experiments have been carried out stating the success of neural networks for time series prediction. Most of the existing systems recommend single neural network architecture to be used for a particular time series. Empirical results show that Artificial Neural Networks (ANNs) can be more effectively used to make better forecasts than the traditional methods since inflation of Bangladesh have a complex structure, nonlinear, dynamic and even chaotic. Due to these reasons, ANNs can increase the forecast performance due to a learning process of the underlying relationship between the input and output variables and their ability to discover nonlinear relationships. Despite of all, ANNs also have some limitations, for instance, error functions of ANNs are usually complex, cumulative, and commonly they have many local minima, unlike the traditional methods. So, each time the network run with different weights and biases it arrives at a different solution.

Despite of all, the choice between one method and other is not an easy task. A literature review point out that the increasing complexity does not necessary increase the accuracy. Sometimes, the traditional statistics can be applied and present a higher performance. Gooijer and Hyndman (2006) refers that some authors stress the importance that future research needs to be done in order to define the frontiers were ANNs and the traditional methods can be more effective with a greater accuracy in relation to each other. For some tasks, neural networks will never replace traditional methods; but for a growing list of applications, the neural architecture will provide either an alternative or a complement to these other techniques.

The question is whether this Neural Networks technique is fundamentally different form linear regression analysis. Particular attention is paid in this research to similarities and differences between these two approaches. When different, we must find out which will be the appropriate to employ. Neural Networks and ordinary regression analysis can sometimes be combined to create a powerful forecasting and learning tool.

1.2 Inflation Behavior

Inflation has become a well-entrenched phenomenon in many countries. Somehow it seems that the general price level can only rise implying that there is an inflationary bias in society. Consensus has it that inflation is likely to impose considerable economic costs (Fischer and Modigliani, 1978). Types of costs are, for instance, menu costs, the decrease in real money balances and decreased efficiency of the price system. There is, however, a lack of understanding of the process which systematically generates inflation (Davis, 1991). One of the central objectives of traditional monetary policy is inflation control since the belief is that, among others; low inflation helps to improve resource allocation and fosters rapid and stable economic growth. The view also holds that inflation is primarily a monetary phenomenon so that low inflation is to be achieved mainly through aggregate demand control by pursuing concretionary monetary (and fiscal) policies. It is argued that these policies should also be supported by liberalization, privatization, and other macroeconomic reforms to create a more open and competitive economy driven by the private sector. In short, the argument is that there exists a tradeoff between inflation and growth (alternatively between inflation and unemployment) that makes inflation targeting as the dominant paradigm in monetary policy. Macroeconomists and central bankers pay close attention to inflation behavior, both in their theoretical debates and in empirical studies. The importance of inflation behavior results from the fact that they influence the behavior of economic agents, i.e., in terms of consumption, savings and investment decisions. Moreover, to the extent that they provide an unbiased predictor of future inflation, quantitative measures of expected inflation may constitute an important information variable taken into account in forward-looking considerations and monetary policy decisions (Forsells and Kenny, 2002). Finally, the inflation behavior of different groups of economic agents indicate the degree of confidence enjoyed by the central bank, the credibility of inflation targets, and whether these targets seem to be attainable. Depending on their nature, inflation behavior may play an important role in price formation. By affecting real interest rates, changes in inflation behavior may lead to changes in aggregate demand, which may then influence prices. As regards cost-push effects, an increase in the expected rate of inflation may make employees demand higher wage settlements. Companies, anticipating higher cost to be faced in the future, may see incentives to increase prices and may be more willing to pay higher wages. Even if prices are not adjusted immediately, companies may temporarily put off the sale of their products. All these interactions, combined with each other, may result in an increase in demand and a simultaneous decrease in supply. In this way, a rise in inflation may generate an increase in prices. The growing popularity of the strategy of direct inflation targeting stems from the conviction that central banks ought to influence inflation behavior. Monetary policy transparency and central bank credibility – key elements in direct inflation targeting – allow monetary policy to meet its ultimate objective of price stability, and by increasing the forward-lookingness of inflation behavior may reduce the sacrifice ratio (Gomez, 2002).

1.3 Why inflation forecasting is so important

Inflation forecast is used as guide in the formulation of the monetary policy by the monetary authorities in the world. Monetary policy decisions are based on inflation forecast extracted from the information from different models and other information suggested by relevant economic indicators of the economy. Inflation tends to be a relatively persistent process, which means that current and past values should be helpful in forecasting future inflation. Forecasts of inflation are important because they affect many economic decisions. Investors need good inflation forecasts, since the returns to stocks and bonds depend on what happens to inflation. Businesses need inflation forecasts to price their goods and plan production. Homeowners' decisions about refinancing mortgage loans also depend on what they think will happen to inflation.

A high and sustained economic growth in conjunction with low inflation is the central objective of macroeconomic policy. Low and stable inflation along with sustainable budget deficit, realistic exchange rate, and appropriate real interest rates are among the indicators of a stable macroeconomic environment. Thus, as an indicator of stable macroeconomic environment, the inflation rate assumes critical importance. It is therefore important that inflation rate be kept stable even when it is low. The primary focus of monetary policy, both in Bangladesh and elsewhere, has traditionally been the maintenance of a low and stable rate of aggregate price inflation as defined by commonly accepted measures such as the consumer price index.

1.4 Bangladesh's Inflation: *A Brief History*

The experience of high inflation is not new in Bangladesh. Over the last five years, Inflation increased several times due to the contractionary monetary policies, orthodox exchange rate management, the rise in import bills and internationally price hikes in food. The average inflation in 2001 was 1.90% while it is found 9.07 % in 2007. After the 2007 global financial crisis, Bangladesh Bank decided to ease monetary policy in order to limit the impact of the crisis on the domestic economy. As a result, in 2009 the average inflation declined to 5.42%. But it went up again 10.68% in 2011. Further, to control this hyper inflation Bangladesh Bank took more restrained monetary policies in 2011. In the national budget and monetary policy of FY 2011-12, the rate of inflation was targeted at 7.5 percent whereas; it stood at 10.6 percent (12-month average) and 8.56 percent (point to point inflation). In FY 2012-13, the government has targeted the rate of inflation at 7.2 percent while the prior experience suggests that it might be hard to maintain inflation below 9% in 2013. Therefore, careful revisions are essential to conduct an effective monetary policy which can successfully control any hyper inflation in 2013.

Currently the financial regulatory authorities in Bangladesh are facing the twin challenge of maintaining price stability while accommodating higher growth in the economy. It is often a tough task to achieve a combination of the two goals. Like other developing countries, Bangladesh has three macroeconomic targets: a growth target to support higher employment and poverty reduction; an inflation target to maintain internal economic stability; and a target for stability of the balance of payments. To achieve these three targets, Bangladesh needs some combination of three policy instruments: monetary policy, fiscal policy, and policies for managing the balance of payments. However, with rising inflation Bangladesh is finding it difficult to properly coordinate all three macroeconomic targets in a sustainable manner.

As the primary objective of monetary policy is to lower inflation and maintain the stability of the exchange rate many expert is currently advocating for the use of monetary policy to control inflation in Bangladesh. But with the long time lag between monetary policy announcement and policy action, it is difficult for policymakers to properly coordinate their strategies. Under such situation, forecasting future inflation can assist policymakers in formulating their strategies. Along with the time lag, in reality inflation is often multi causal and prime cause of inflation can vary from year to year. The possible factors behind excessive inflation can include supply side factors including cost push relationship along with exchange rate effects, excessive borrowing by local government and demand pull inflation.

Given the complexity of inflation controlling and time lag of monetary policy affect many monetary economists strongly advocated for inflation targeting to maintain stable aggregate price inflation. In his writing Svensson (1996) argued for inflation forecasting targeting where central bank tries to stabilize only inflation and resource utilization. However, before formulating strategy based on inflation forecast it is necessary to emphasize the structural soundness of inflation forecasting. This paper is one such attempt towards accurate univariate time series inflation forecasting in Bangladesh using monthly time series data from July 2001 to March 2013.

1.5 Overview

Chapter 2 spells out the details literature of the related work of this thesis. Chapter 3 draws a theoretical framework basic of Box-Jenkins methodology of estimating the ARIMA models and discusses various plausible ARIMA models. Chapter 4 spells out the details of Neural Networks modeling and here the Neural Network is applied in forecasting inflation variables. From this empirical analysis, the forecasting power of NN with that of proposed ARIMA (traditional econometric) models is compared for out-of-sample time series in Chapter 5. Chapter 6 concludes the thesis with a perspective on what has been achieved and identifies some prospective topics for further research.

CHAPTER 2

LITERATURE REVIEW

2.1 Financial Time Series Analysis

Financial markets are complex dynamic systems with a high volatility and a great amount of noise. Due to these and other reasons we might say that forecasting financial time series can be a challenging task. In the past decades, strongest assumptions on financial time-series (namely the Random Walk Hypothesis) have been partially discharged.

A time series is a sequence of variables whose values represent equally spaced observations of a phenomenon over time. We can write a time series as

$$\{x_1, x_2, x_3, \dots \dots \dots, x_t\} \text{or } \{x_t\}, t = 1,2,3, \dots \dots, T \quad (2.1)$$

Where, we will treat x_t as a random variable.

The main objective of time series prediction can be stated as Ho et al. (2002) describes: "given a finite sequence $x_1, x_2, x_3, \dots \dots \dots, x_t$ find the continuation of x_{t+1}, x_{t+2}". The ability to predict time or at least the range within a specific confidence interval it is important in many knowledge areas for planning, decision making, etc, and time series analysis in financial area isn't an exception.

Due to the fact that most of the current modeling techniques are based on linear assumptions there are emerging some authors that think that a non linear analysis of financial time series needs to be considered. A technique that is emerging in this field is the use of neural networks, declared to be a universal approximator for nonlinear models.

2.2 Literature Concerning Linear Time Series analysis of Inflation

It was the major contribution of Yule (1927) which launched the notion of stochasticity in time series by postulating that every time series can be regarded as the realization of a stochastic process. Based on this idea, a number of time series methods have been proposed. George E.P. Box and Gwilym M. Jenkins (1970) integrated the existing knowledge on time series with their book “Time Series Analysis: Forecasting and Control”. First of all, they introduced univariate models for time series which simply made systematic use of the information included in the observed values of time series. This offered an easy way to predict the future development of the variable. Moreover, these authors developed a coherent, versatile three-stage iterative cycle for time series identification, estimation, and verification. George E.P. Box and Gwilym M. Jenkins (1970) book had an enormous impact on the theory and practice of modern time series analysis and forecasting. With the advent of the computer, it popularized the use of autoregressive integrated moving average (ARIMA) models and their extensions in many areas of science. Since then, the development of new statistical procedures and larger, more powerful computers as well as the availability of larger data sets has advanced the application of time series methods. After the introduction by Yule (1921), the autoregressive and moving average models have been greatly favored in time series analysis. Simple expectations models or a momentum effect in a random variable can lead to AR models. Similarly, a variable in equilibrium but buffeted by a sequence of unpredictable events with a delayed or discounted effect will give MA mode.

Granger and Newbold (1977) provide a survey of early comparisons of forecasting performance of univariate and multivariate model, and Zarnowitz and Braun (1993) compare forecasts from univariate and VAR models with forecasts constructed by professional forecasters for the U.S. over the 1968-1990 period. Aidan Meyler, Geoff Kenny and Terry Quinn (1998) outlined ARIMA time series models for forecasting Irish inflation. It considered two alternative approaches, which suggests that ARIMA forecast has outperformed. Toshitaka Sekine (2001) estimated an inflation function and forecasts one-year ahead inflation for Japan. He found that markup relationships, excess money and the output gap are particularly relevant long-run determinants for an equilibrium correction model of inflation. Christiano (1989) used past quarterly changes in the short-term nominal T-bill interest rate as an explanatory variable of the U.S. inflation. Francisco Nadal-De Simone (2000) estimated two time-varying parameter models of Chilean inflation Box-Jenkins models outperform the two models for short-term out-of-sample forecasts; their superiority deteriorates in longer forecasts. Jae J. Lee(2011) used Bayesian inference to forecast time series under Box and Jenkins’ARIMA model. Kanchan Datta(2011) claimed that ARIMA (4, 12, 2, 0) model fits the inflation data of Bangladesh satisfactorily.

2.3 Literature Concerning Nonlinear Time Series analysis of Inflation

Nowadays central banks, such as, CZECH National Bank (Marek Hlavacek, Michael Konak and Josef Cada, 2005), Bank of Canada (Greg Tkacz and Sarah Hu, 1999), Bank of Jamaica (Serju,2002), are currently using their forecasting models based on ANN methodology for predicting various macroeconomic indicators.

Adya and Collopy (1998) review 48 applications of neural networks to business forecasting and prediction. The authors attempt to determine whether these studies adequately compare the neural network forecasts with alternative techniques (effectiveness of validation) and whether the neural network technique is effectively implemented (effectiveness of implementation). Gazely and Binner (1998) employ neural networks to compare the capabilities of Divisia and simple sum measures of broad money as indicators of inflation in United Kingdom. They find Divisia monetary measures produce more accurate forecasts of inflation than standard monetary aggregates. Binner et al. (2002) further apply neural networks to model Taiwan's inflation rate resulting in particularly accurate forecasts when Divisia monetary measures are used.

More recently, Binner, Bissoondeeal, Elger, Gazely, and Mullineux (2005) compare a neural network model to ARIMA and VAR models in predicting the inflation rate of the Euro. They find that the VAR model produces superior out-of-sample forecasts compared to the univariate ARIMA model, and that in every case examined, the neural network model produces superior forecasts relative to the VAR model. Thus, they conclude that linear models such as ARIMA and VAR represent a subset of non-linear models such as neural networks.

Tkacz (2001) has also investigated Canadian data using neural networks. Comparing the forecasts of a neural network model to those of a naive random walk model, an exponential smoothing model, an autoregressive model, and a multivariate linear model, Tkacz finds that the neural network produces superior year-to-year forecasts of real GDP growth relative to all other models.

Neural networks have also been frequently applied to the prediction of currency exchange rates. Shazly and Shazly (1999) develop a genetic neural network model to predict the three-month spot rate for the British pound, German mark, Japanese yen, and Swiss franc. Their results show that the neural network forecasts are superior to those obtained from both future and forward market rates. Nag and

Mitra (2002) also use a genetic neural network to predict daily spot exchange rates. Using the German mark/US dollar, Japanese yen/US dollar, and US dollar/British pound exchange rates as data, they find that their neural network model outperforms six different ARCH and GARCH models.

Taken together, this literature indicates that neural network models can significantly outperform linear forecasting models such as ARIMA, VAR, or GARCH in predicting long-term economic activity (e.g. inflation or real GDP) or short-term financial activity (e.g. exchange rates). The literature also indicates that currently there is no standard technique to measure the relative forecasting advantage of a neural network model relative to a linear model. Hence, current studies which compare the forecasting ability of neural networks and linear models use very different methodologies.

CHAPTER 3

FORECASTING INFLATION BY ARIMA MODELS

3.1 Introduction

The classical linear regression model is the conventional starting point for time series and econometric methods. Peter Kennedy, in A Guide to Econometrics (1985), provides a convenient statement of the model in terms of five assumptions:

(1) The dependent variable can be expressed as a linear function of a specific set of independent variables plus a disturbance term (error);

(2) The expected value of the disturbance term is zero;

(3) The disturbances have a constant variance and are uncorrelated;

(4) The observations on the independent variable(s) can be considered fixed in repeated samples; and,

(5) The number of observations exceeds the number of independent variables and there are no exact linear relationships between the independent variables.

While regression can serve as a point of departure for both time series and econometric models, it is incumbent on the analyst to generate the plots and statistics which will give some indication of whether the assumptions are being met in a particular context.

A time series model is a tool used to predict future values of a series by analyzing the relationship between the values observed in the series and the time of their occurrence. Time series models can be developed using a variety of time series statistical techniques. If there has been any trend and/or seasonal variation present in the data in the past then time series models can detect this variation, use this information in order to fit the historical data as closely as possible, and in doing so improve the precision of future forecasts. There are many traditional techniques used in time series analysis. Some of these include Exponential Smoothing, Linear Time Series Regression and Curvefit, Autoregression, ARIMA (Autoregressive Integrated Moving Average), Intervention Analysis, Seasonal Decomposition, etc.

In this thesis we'll focus our analysis on ARIMA models. Until the 19th century, the study of time series was characterized by the idea of a deterministic world. Here, we can find the contribution of Yule (1927) which launched the notion of stochastic process in time series analysis by postulating that every time series can be regarded as a realization of a stochastic process. Box and Jenkins in the 1970's developed a coherent, versatile three-stage iterative cycle for time series identification, estimation and verification. Many of the ideas that have been incorporated into ARIMA models were by these authors (see Box et al, 1994), and for this reason ARIMA modelling is sometimes called Box-Jenkins modelling. ARIMA stands for AutoRegressive Integrated Moving Average, and the assumption of these models is that the variation accounted for in the series variable can be divided into three components:

- Autoregressive (AR)
- Integrated (I) or Difference
- Moving Average (MA)

An ARIMA model can have any component, or combination of components, at both the nonseasonal and seasonal levels. There are many different types of ARIMA models and the general form of an ARIMA model is ARIMA (p,d,q)(P,D,Q), where:

- p refers to the order of the nonseasonal autoregressive process incorporated into the ARIMA model (and P the order of the seasonal autoregressive process)
- d refers to the order of nonseasonal integration or differencing (and D the order of the seasonal integration or differencing)
- q refers to the order of the nonseasonal moving average process incorporated in the model (and Q the order of the seasonal moving average process).

So for example an ARIMA (2; 1; 1) would be a nonseasonal ARIMA model where the order of the autoregressive component is 2, the order of integration or differencing is 1, and the order of the moving average component is also 1. ARIMA models need not have all three components. For example, an ARIMA (1; 0; 0) has an autoregressive component of order 1 but no difference or moving average component. Similarly, an ARIMA (0; 0; 2) has only a moving average component of order 2.

3.2 Background and Methodology

For the current research my first objective was to find out the most accurate out of different ARIMA models in forecasting inflation of Bangladesh.
A modeling and forecasting of various ARIMA time series models based on Bngladesh's monthly
Inflation data would be carried out.

To realize the objectives of the study, the following steps will be taken in this regard:

- Collection of monthly inflation data from the Central Bank of Bangladesh, Bureau of Statistics (BBS).
- Specification and estimation of various possible types of ARIMA models.
- Obtaining of ex-post forecast after empirically estimating the various types of ARIMA models.
- Comparison of forecasting performance of various types of ARIMA models by using certain statistical measures.
- At last there is an attempt to find out the best model for forecasting purpose.

The following questions allow the research to meet the objectives proposed:

- Which ARIMA models are plausible to predict the inflation of Bangladesh and why?
- What are the similarities and the differences between the ARIMA models?

In the first phase, the statistical properties/summary statistics as well as distribution of all time series will be tested by means of coefficient of skewness and kurtosis, normal probability plots etc. to check presence of typical stylized facts.

3.3 Autoregressive Models

In a similar way to regression, ARIMA models use independent variables to predict a dependent variable (the series variable). The name autoregressive implies that the series values from the past are used to predict the current series value. In other words, an autoregressive process is a function of lagged dependent variables and a moving average process a function of lagged error terms. The integrated in ARIMA takes into account that a time series may be non-stationary before an AR and MA process can be combined into one equation and used for forecasting purposes, the dependent variable must be stationary or made stationary. Otherwise, the underlying trend would falsely be attributed to serial correlation. If a series needs to be differenced d times before it is stationary, the series is said to be integrated of degree d. 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, q is the number of moving average terms.

A pth-order AR model is defined as

$$y_t = C + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots \phi_p y_{t-p} + e_t \tag{3.1}$$

C is the constant term; ϕ_j is the jth auto regression parameter; e_t is the error term at time t. The explanatory variables in this equation are time-lagged values of the variable y.

For example, it might be the case that a good predictor of current monthly sales is the sales value from the previous month.

3.4 Moving Average Models

The autoregressive component of an ARIMA model uses lagged values of the series values as predictors. In contrast to this, the moving average component of the model uses lagged values of the model error as predictors. Some analysts interpret moving average components as outside events or shocks to the system. That is, an unpredicted change in the environment occurs, which influences the current value in the series as well as future values. Thus the error component for the current time period relates to the series' values in the future. The order of the moving average component refers to the lag length between the error and the series variable. For example, if the series variable is influenced by the model's error lagged one period, then this is a moving average process of order one and is sometimes called an MA(1) process.

The general MA model of order q can be written as

$$y_t = C + e_t - \theta_1 e_{t-1} - \theta_2 e_{t-2} + \cdots + \theta_q e_{t-q} \tag{3.2}$$

C is the constant term; θ_j is the jth moving average parameter; e_{t-k} is the error term at time t-k. This model is defined as a moving average of the error series.

3.5 Integration

The Integration (or Differencing) component of an ARIMA model provides a mean of accounting for trend within a time series model. Creating a differenced series involves subtracting the values of adjacent series values in order to evaluate the remaining component of the model. The trend removed by differencing is later built back into the forecasts by Integration (reversing the differencing operation). Differencing can be applied at the nonseasonal or seasonal level. If y_t is non-stationary, we take a first-difference of y_t so that Δy_t becomes stationary.

$$\Delta y_t = y_t - y_{t-1} \tag{3.3}$$

Again if Δy_t is non-stationary, we use a second difference (d = 2), implying a first difference of the first differenced series

$$\Delta^2 y_t = \Delta y_t - \Delta y_{t-1} \tag{3.4}$$

3.6 Stationarity

In time series analysis the term stationarity is often used to describe how a particular time series variable changes over time. Stationarity has three components. First, the series has a constant mean, which implies that there is no tendency for the mean of the series to increase or decrease over time. Second, the variance of the series is assumed constant over time. Finally, any autocorrelation pattern is assumed constant throughout the series. For example, if there is an AR (2) pattern in the series; it is assumed to be present throughout the entire series. Any violation of stationarity creates estimation problems for ARIMA models. It is difficult to detect the true variations in the dependent variable if it is non-stationary. Because the mean of the series is changing over time, correlations and relationships between the variables in the ARIMA model will be exaggerated or distorted. Only if the mean of the dependent variable is stationary will true relationships and correlations be identified.

The Integration component of ARIMA is typically associated with removing trend from the series, which would violate the constant mean component of stationarity. It is often the case in time series analysis that the mean of a variable increases or decreases over time. In order to make a series containing trend stationary, we can create a new series that is the difference of the original series. A first order difference creates a value for the new series which is the difference between the series value in the current period minus the series value in the previous period. Often the differenced series will have a stationary mean. If the differenced series does not have a stationary mean then it might be necessary to take first differences of the differenced series. This transformation is known as second order differencing as the original series has now been differenced twice. The number of times a series needs to be differenced is known as the order of integration. Differencing can be performed at the seasonal (current time period value minus the value from one season ago) or non-seasonal (current time period minus the value from the previous time period) component.

In short,

- If a series is stationary then there is no need to difference the series and the order of integration is zero. In all ARIMA models the dependent variable should be left in its original values. ARIMA models will be of the form ARIMA (p; 0; q) where p is the order of the autoregressive process and q is the order of the moving average process in the model.

- If a series is non-stationary then usually first differencing the series will make it stationary. If first differencing makes the series stationary then the order of integration is one. ARIMA models will be in the form of ARIMA (p; 1; q).

- However if it might be necessary to difference a series twice to make it stationary the form become ARIMA (p; 2; q). In practice, you barely would difference a series more than twice. Most common is a d of one or two.

There are several tests to check the stationarity of a time series model namely Dicky Fuller test, ACF and PACF function examination etc.

3.7 ARIMA models

If a process y_t has an ARIMA (*p; d; q*) representation, then it has an ARMA (*p; q*) representation as presented by the equation below:

$$\Delta^d y_t(1-\phi_1 L-\phi_2 L^2-\ldots-\phi_p L^p)=(1-\theta_1 L-\theta_2 L^2-\ldots-\theta_p L^q)u_t \tag{3.5}$$

SARIMA:

The ARIMA models can be extended to handle seasonal components of a data series. The general shorthand notation is ARIMA (p; d; q)(P; D; Q)s where s is the number of periods per season. In the seasonal component, *P* represents the Seasonal Autoregressive (SAR) term, *D* is the number of seasonal difference(s) performed and *Q* denotes the Seasonal Moving Average (SMA) term.

3.8 The Box-Jenkins Methodology of ARIMA Estimation

The Autoregressive Integrated Moving Average (ARIMA) models, or Box-Jenkins methodology, are a class of linear models that is capable of representing stationary as well as nonstationary time series.

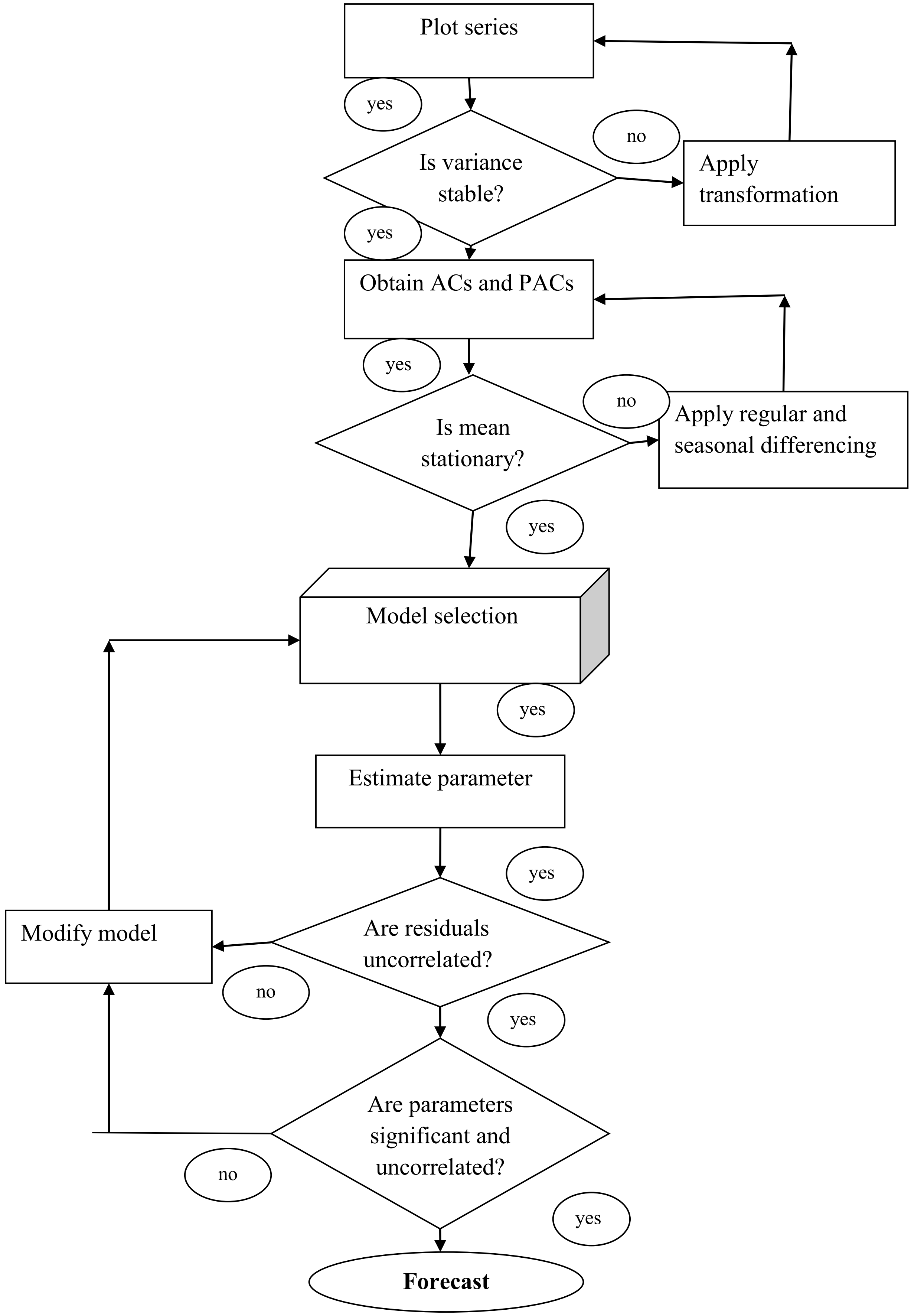


Figure 3.1 Box Jenkins Modeling Approach

3.8.1 The Box-Jenkins methodology uses an iterative approach

- An initial model is selected, from a general class of ARIMA models, based on an examination of the TS and an examination of its autocorrelations for several time lags
- The chosen model is then checked against the historical data to see whether it accurately describes the series: the model fits well if the residuals are generally small, randomly distributed, and contain no useful information.
- If the specified model is not satisfactory, the process is repeated using a new model designed to improve on the original one.
- Once a satisfactory model is found, it can be used for forecasting.
- Autoregressive models are appropriate for stationary time series, and the coefficient Φ0 is related to the constant level of the series. Theoretical behavior of the ACF and PACF for AR(1) and AR(2) models:

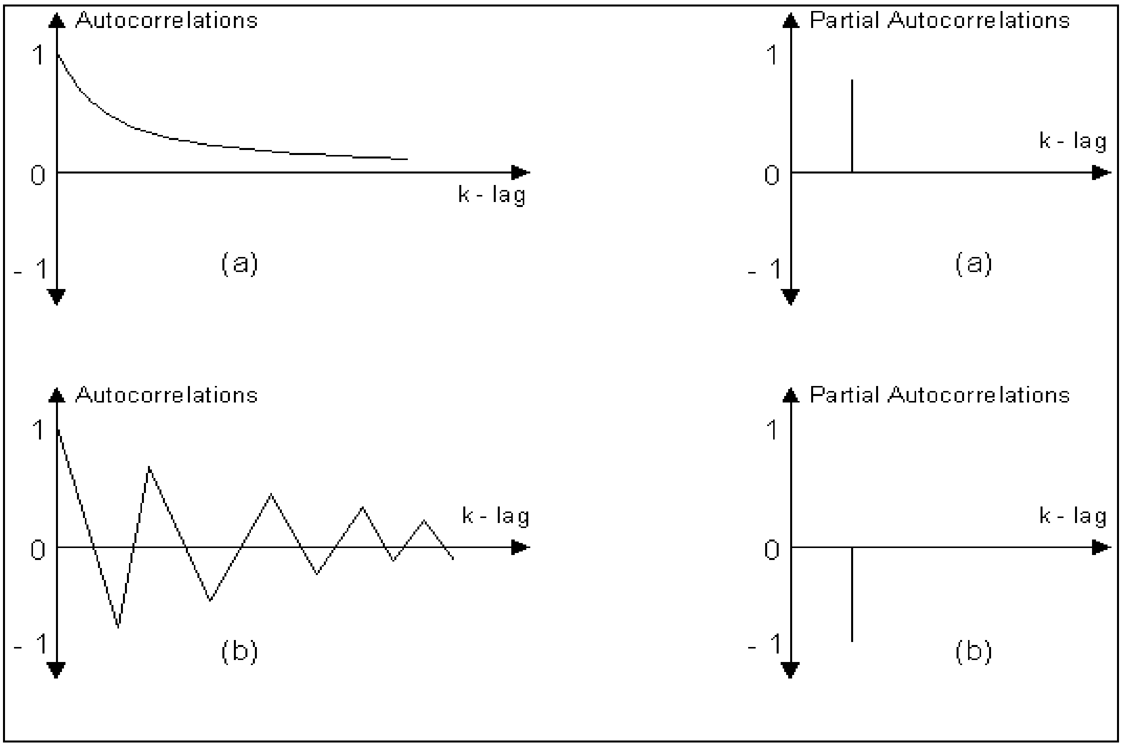


Figure 3.2 behavior of auto correlation and partia autocorrelation of AR(1)

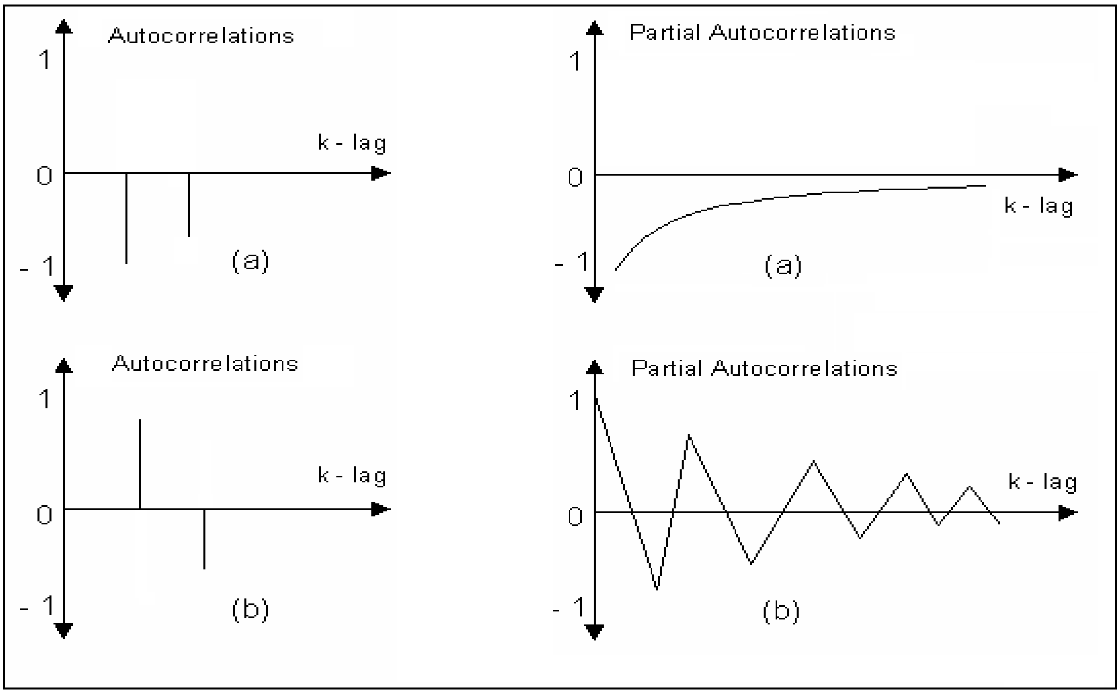


Figure 3.3 behavior of auto correlation and partia autocorrelation of MA(1)

Overall the Box-Jenkins Methodology can be illustrated as the following major steps:

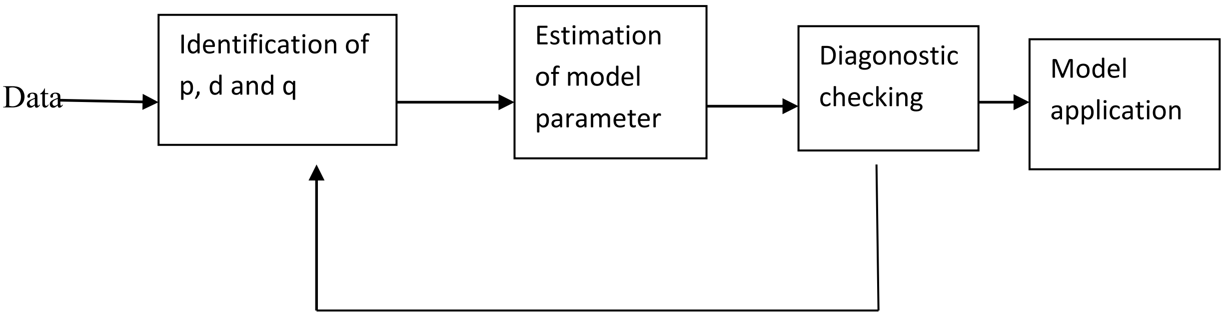


Figure 3.4 Major Steps of Box-Jenkins Methodology

3.8.2 Model identification

3.8.2.1 Identification of Number of Differences

For the identification of the number of differences more commonly used methodology is to see the Autocorrelation Plots (ACF plots) which plots plot correlation coefficients against various time lags. Whenever pattern is of initially "long spikes" that slowly fade out (or tail off), that is an indicator of data is not random. As a rule of thumb, as long as ACF plots reveal a pattern of fading out autocorrelation coefficients, one needs to keep on differencing. In practice, it would be barely differenced a series more than twice. Most common is a d of one or two.

Another method is unit-root test (if d=1).

3.8.2.2 Identification of AR and MA Lag Orders

Identification of degrees of lag orders is often done by "reading" the autocorrelation and partial autocorrelation plot. Both autocorrelation and partial autocorrelation functions are correlations of a series with itself, successively shifted by lags. The partial autocorrelation function (PACF), however, controls for the correlation between the lags and ACF is essentially obtained from bivariate regressions between all variables separated by a certain number of lags. The PACF is essentially obtained from multivariate regressions that also add the previous lagged values. The following decision-making matrix has been developed to assess univariate time series:

Table 3.1 Patterns for identifying ARMA Processes

	Model		
	AR(p)	MA(q)	ARMA(p,q)
ACF	Tails off	Cuts off after q	tails off
PACF	Cuts off after p	Tails off	tails off

3.8.2.3 Seasonal parameters

Whenever you see a cyclical pattern, ARIMA also allows for seasonal parameters of p, d, and q.

The Box-Jenkins method is nothing more but a rough guide on how to identify an ARIMA model, in practice it remains a trial and error process. In this trial and errors process, it is a good strategy to move from the most parsimonious model up to more complex specifications.

3.8.3 Estimation of Model

Once a tentative model has been selected, the parameters for the model must be estimated. The method of least squares can be used for ARIMA model. However, for models with MA components, there is no simple formula that can be used to estimate the parameters. Instead, an iterative method is used. This involves starting with a preliminary estimate, and refining the estimate iteratively until the sum of the squared errors is minimized.

Another method of estimating the parameters is the maximum likelihood procedure. Like least squares methods, these estimates must be found iteratively. Maximum likelihood estimation is usually favored because it has some desirable statistical properties.

After the estimates and their standard errors are determined, t values can be constructed and interpreted in the usual way. Parameters that are judged significantly different from zero are retained in the fitted model; parameters that are not significantly different from zero are dropped from the model.

3.8.4 Diagnostic Check of Residuals

Before using the model for forecasting, it must be checked for adequacy. A model is adequate if the residuals left over after fitting the model are simply white noise. The pattern of ACF and PACF of the residuals may suggest how the model can be improved. A portmanteau test can also be applied to the residuals as an additional test of fit. If the portmanteau test is significant, then the model is inadequate. In this case we need to go back and consider other ARIMA models. So the purpose of the residual test is to check whether the error terms are white noise (uncorrelated). Portmanteau test, also known as Box- Pierce Q statistic defined as:

$$Q = n\sum_{k=1}^{m}\hat{\rho}_k^2 \tag{3.6}$$

k is the k-th order sample autocorrelation of the residuals and T the sample size. Q has a Chi-square distribution with (K-p-q) degrees of freedom. The Q statistic essentially sums up various correlation coefficients. The null hypothesis is that there is no serial correlation in the residuals. The Portmanteau test is mostly meaningful for 12<lags <25.

Any new model will need their parameters estimated and their AIC values computed and compared with other models. Usually, the model with the smallest AIC will have residuals which resemble white noise. Occasionally, it might be necessary to adopt a model with not quite the smallest AIC value, but with better behaved residuals.

3.8.5 Forecast Evaluation Statistics for ARIMA Models

To evaluate forecast accuracy of the models statistics such as mean error (ME), mean absolute error (MAE), root mean squared error (RMSE) and Theil's U are used.

The mean absolute error (MAE)

Measures forecast accuracy by averaging the magnitudes of the forecast errors.

$$\text{MAE} = \frac{1}{n}\sum_{t=1}^{n}\left|y_t - \hat{y}_t\right| \qquad (3.7)$$

The Mean Percentage Error (MPE)

Can be used to determine if a forecasting method is biased (consistently forecasting low or high) Large positive MPE implies that the method consistently under estimates. Large negative MPE implies that the method consistently over estimates. The forecasting method is unbiased if MPE is close to zero.

$$\text{MPE} = \frac{1}{n}\sum_{t=1}^{n}\frac{(y_t - \hat{y}_t)}{y_t} \qquad (3.8)$$

The Mean Absolute Percentage Error (MAPE)

Provides an indication of how large the forecast errors are in comparison to actual values of the series. Especially useful when the y_t values are large can be used to compare the accuracy of the same or different methods on two different time series data.

$$\text{MAPE} = \frac{1}{\text{n}}\sum_{t=1}^{n}\frac{\left|y_t - \hat{y}_t\right|}{y_t} \qquad \text{……..(3.9)}$$

Mean Squared Error

This approach penalizes large forecasting errors.

$$\text{MSE} = \frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2 \qquad \text{………(3.10)}$$

Root Mean Squared Error

The RMSE is easy for most people to interpret because of its similarity to the basic statistical concept of a standard deviation, and it is one of the most commonly used measures of forecast accuracy.

$$\text{RMSE} = \sqrt{\frac{\sum_{t=1}^{n}(y_t - \hat{y}_t)^2}{n}} \qquad \text{……(3.11)}$$

Theil's U-statistic

This statistic allows a relative comparison of formal forecasting methods with naïve approaches and also squares the errors involved so that large errors are given much more weight than smaller errors. The advantage of the Theil statistic is that it is 'unit less' as it compares the RMSE of the chosen model to that of the 'naive' forecast model. Mathematically, Theil's U-statistic is defined as

$$U = \sqrt{\frac{\sum_{t=1}^{n-1}\left(\frac{\hat{y}_{t+1} - y_{t+1}}{y_t}\right)^2}{\sum_{t=1}^{n-1}\left(\frac{y_{t+1} - y_t}{y_t}\right)^2}} \qquad \text{………(3.12)}$$

U = 1, the naive method is as good as the forecasting technique being evaluated.

U < 1, the forecasting technique being used is better than the naïve method.

U > 1, there is no point in using a formal forecasting method since using a naïve method will produce better results.

3.9 Data and Estimation

Monthly Inflation rates of Bangladesh measured by CPI, with the base being 1995-1996 are taken from the Central Bank of Bangladesh, Bureau of Statistics (BBS) cover from July 2001 to April 2013, we use the data from July 2001 to December 2012, and the remaining data is used as out of sample period to check the strength of our prediction.

Time series plot of the inflation rate shows that it had been increasing from the starting point to 2006 than it decreased for a very short period of time and increased abruptly and touched the double digit level during July, August and September of 2008. After that there was a sharp decline throughout the year (up to October 2009). Again it increased sharply and touched the double digit level and reached at the highest point at February 2012, after that it started decreasing.

Summary statistics, using the observations 2001:07 - 2013:01 for the variable 'BD_INFLATION' (139 valid observations) given in the next page.

Table 3.2 Summery statistics of Inflation data

Mean	6.7804
Median	6.9400
Minimum	1.4700
Maximum	10.960
Standard deviation	2.4563
Skewness	-0.38764
Ex. Kurtosis	-0.41060

Time series plot, using the observations 2001:07 - 2013:01 for the variable 'BD_INFLATION' (139 valid observations) given in the next page.

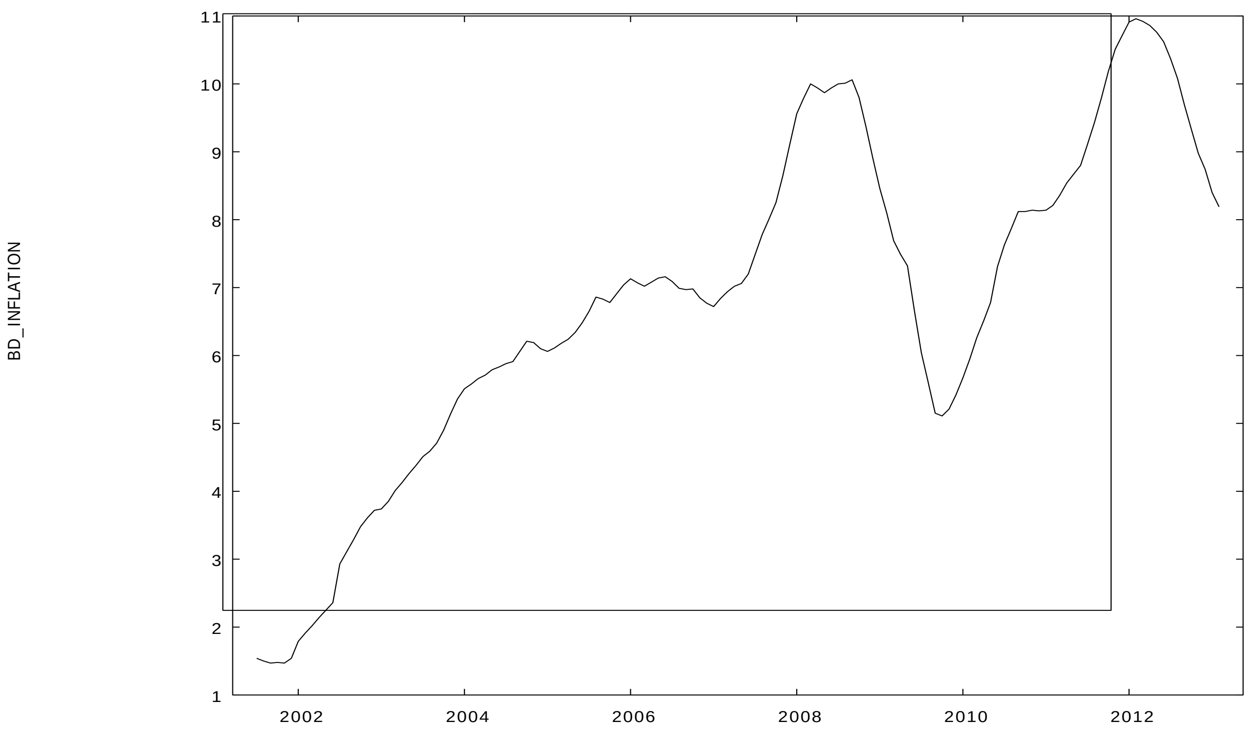


Figure 3.5 Time series plot of inflation

Another way to examine the properties of a time series is to plot its autocorrelogram. The autocorrelogram plots the autocorrelation between differing lag lengths of the time series. Plotting the autocorrelogram is a useful aid for determining the stationarity of a time series, and is also an important input into Box-Jenkins model identification.

3.10 Testing for Stationarity

3.10.1 Stationarity Test for Inflation Data

<u>ACF and PACF Test for Stationarity for Inflation Data</u>

In Figure 3.6 the Correlogram and partial Correlogram are shown where two facts stand out- First, the ACF declines very slowly, almost all lags are individually statistically significantly different from zero. Second, after the first lag, the PACF drops dramatically. And all ACF are statistically insignificant. These phenomena indicate that the data are non stationary. Table 3.3 indicates the numeric values of autocorrelation function.

Table 3.3 Autocorrelation function of inflation of Bangladesh

LAG	ACF	PACF	Q-stat. [p-value]
1	0.9771 ***	0.9771 ***	134.6379 [0.000]
2	0.9464 ***	-0.1837 **	261.8797 [0.000]
3	0.9089 ***	-0.1424 *	380.1027 [0.000]
4	0.8657 ***	-0.1078	488.1582 [0.000]
5	0.8177 ***	-0.0881	585.2880 [0.000]
6	0.7663 ***	-0.0586	671.2381 [0.000]
7	0.7134 ***	-0.0234	746.2954 [0.000]
8	0.6592 ***	-0.0322	810.8750 [0.000]
9	0.6046 ***	-0.0233	865.6135 [0.000]
10	0.5506 ***	-0.0060	911.3792 [0.000]
11	0.4983 ***	0.0024	949.1571 [0.000]
12	0.4488 ***	0.0198	980.0441 [0.000]
13	0.4060 ***	0.1010	1005.5224 [0.000]
14	0.3670 ***	0.0050	1026.5070 [0.000]
15	0.3316 ***	-0.0014	1043.7829 [0.000]
16	0.2999 ***	-0.0004	1058.0295 [0.000]
17	0.2712 ***	-0.0147	1069.7756 [0.000]
18	0.2454 ***	-0.0085	1079.4718 [0.000]
19	0.2220 ***	-0.0099	1087.4719 [0.000]
20	0.2010 **	-0.0048	1094.0867 [0.000]
21	0.1826 **	0.0059	1099.5950 [0.000]
22	0.1665 *	0.0026	1104.2128 [0.000]
23	0.1524 *	0.0042	1108.1168 [0.000]
24	0.1399	0.0003	1111.4358 [0.000]

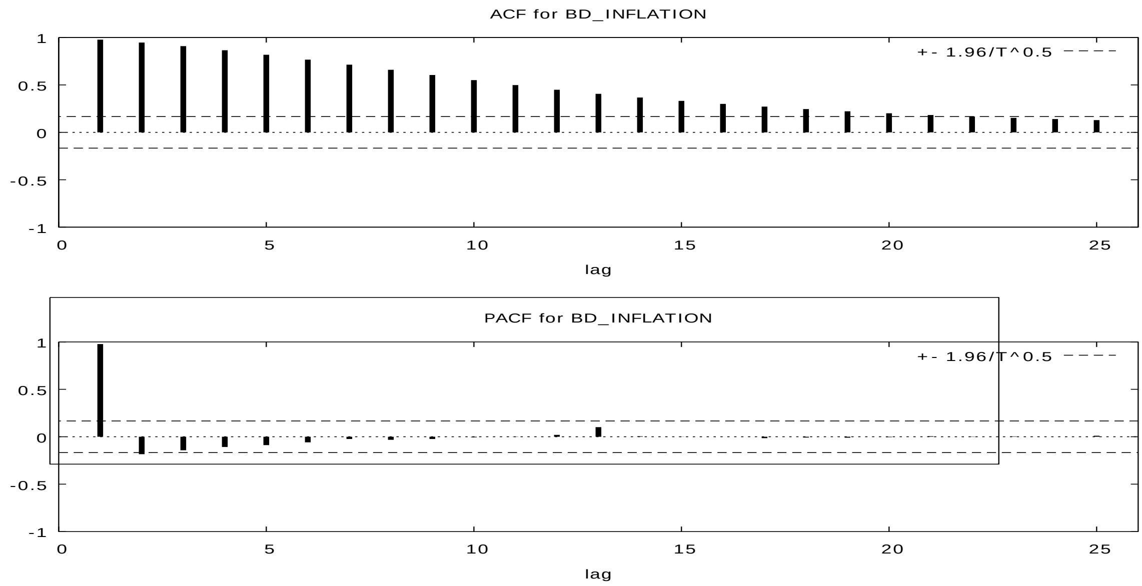


Figure 3.6 ACF and PACF of inflation of Bangladesh

Another way of stationarity test is augmented Dicky-Fular test that is described below.

<u>Unit root test for inflation data:</u>

A Dickey-Fuller test of the variable could not reject the hypothesis that the series has a unit root which is shown below:

Augmented Dickey-Fuller test for BD_INFLATION including 11 lags of (1-L)BD_INFLATION (max was 12)

Sample size= 126, Unit-root null hypothesis: a = 1

Table 3.4 Unit root test for inflation data

Test with constant	Test with constant and trend
Model: (1-L)y = b0 + (a-1)*y(-1) + ... + e 1st-order autocorrelation coefficient for e: 0.015 Lagged differences: F (11, 113) = 33.562 [0.0000] Estimated value of (a - 1): -0.0114448 Test statistic: tau_c(1) = -2.17659 Asymptotic p-value 0.2151	Model: (1-L)y = b0 + b1*t + (a-1)*y(-1) + ... + e 1st-order autocorrelation coefficient for e: 0.022 Lagged differences: F(11, 112) = 33.893 [0.0000] Estimated value of (a - 1): -0.0236624 Test statistic: tau_ct(1) = -2.19773 Asymptotic p-value 0.4903

All the statistical analysis, time series plot, ACF and PACF plot, unit root test indicates that the CPI inflation of Bangladesh is not stationary, it has to be made stationary before applying the Box-Jenkins methodology.

3.10.2 Stationarity Test for First difference of Inflation Data

ACF and PACF Test for Stationarity for the First Difference of Inflation Data

In Table 3.3, the autocorrelation function values for all the lags from 1 to 7 and from 10 to 22 are statistically significant. The partial autocorrelation function of the first, 9^{th} , 11^{th} and 13^{th} are statistically significant. All the test statistics and p value reject the null hypothesis of stationarity.

The ACF plot shows the cyclical pattern in Figure 5.7 and PACF drops dramatically which is an indication of non stationarity.

Table 3.5 Autocorrelation function for d_BD_INFLATION

LAG	ACF	PACF	Q-stat. [p-value]
1	0.8417 ***	0.8417 ***	99.1882 [0.000]
2	0.6873 ***	-0.0723	165.8199 [0.000]
3	0.5635 ***	0.0140	210.9431 [0.000]
4	0.4234 ***	-0.1348	236.6136 [0.000]
5	0.3330 ***	0.0846	252.6125 [0.000]
6	0.2500 ***	-0.0589	261.6969 [0.000]
7	0.1443 *	-0.1191	264.7481 [0.000]
8	0.0408	-0.0994	264.9939 [0.000]
9	-0.0908	-0.1921 **	266.2212 [0.000]
10	-0.2053 **	-0.0617	272.5435 [0.000]
11	-0.3329 ***	-0.2294 ***	289.2931 [0.000]
12	-0.4111 ***	0.0284	315.0374 [0.000]
13	-0.3492 ***	0.3482 ***	333.7632 [0.000]
14	-0.2889 ***	0.0352	346.6854 [0.000]
15	-0.2560 ***	-0.0669	356.9184 [0.000]
16	-0.2257 ***	-0.0771	364.9389 [0.000]
17	-0.2374 ***	-0.0793	373.8797 [0.000]
18	-0.2483 ***	-0.0862	383.7429 [0.000]
19	-0.2343 ***	-0.0763	392.6043 [0.000]
20	-0.2154 **	-0.0735	400.1572 [0.000]
21	-0.1793 **	-0.0512	405.4355 [0.000]
22	-0.1563 *	-0.1009	409.4817 [0.000]
23	-0.1251	-0.0249	412.0964 [0.000]
24	-0.1211	-0.0167	414.5664 [0.000]

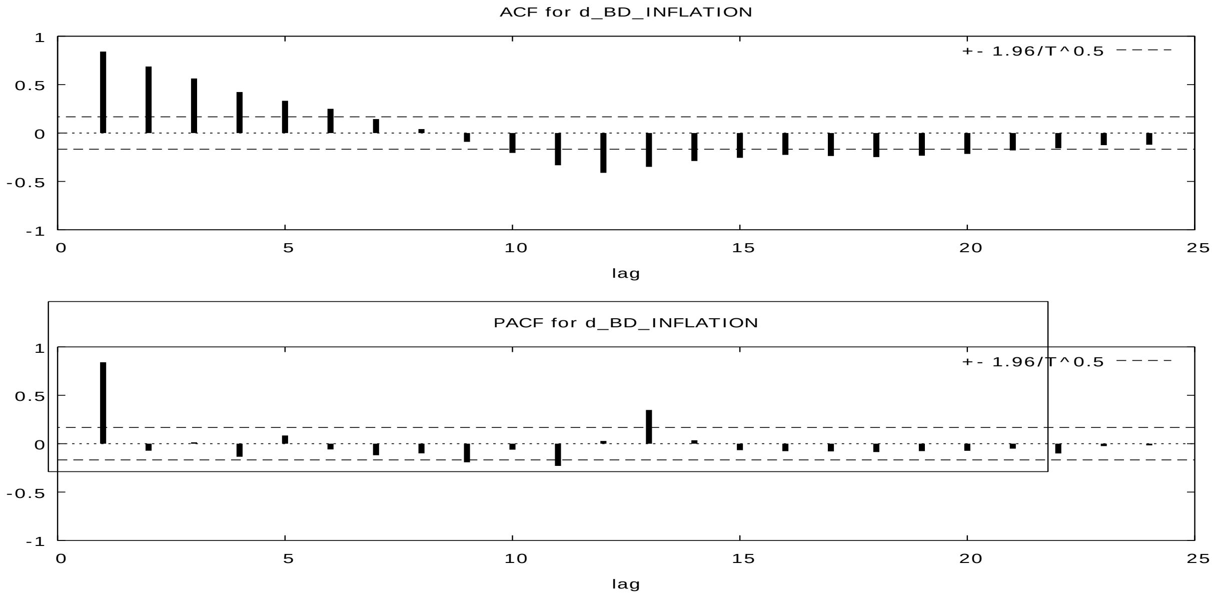


Figure 3.7 ACF and PACF plot for the first difference of inflation

<u>Unit root test for the first difference of inflation:</u>

The unit root test cannot reject the null hypothesis of non stationarity which indicates that the data are still non stationary. Augmented Dickey-Fuller test for first difference of inflation: including 12 lags of (1-L)d_BD_INFLATION (max was 12), Sample size 124,Unit-root null hypothesis: a = 1.

Table 3.6 Unit root test for the first difference of inflation

<u>Test with constant</u>	<u>Test with constant and trend</u>
Model: (1-L)y = b0 + (a-1)*y(-1) + ... + e 1st-order autocorrelation coefficient for e: 0.032 Lagged differences: F(12, 110) = 5.247 [0.0000] Estimated value of (a - 1): -0.182156 Test statistic: tau_c(1) = -2.47022 Asymptotic p-value 0.1229	Model: (1-L)y = b0 + b1*t + (a-1)*y(-1) + ... + e 1st-order autocorrelation coefficient for e: 0.032 Lagged differences: F(12, 109) = 5.236 [0.0000] Estimated value of (a - 1): -0.195085 Test statistic: tau_ct(1) = -2.5856 Asymptotic p-value 0.287

So, to make the data stationary we need to take another difference which is stated below.

3.10..3 Second difference of Inflation Data

ACF and PACF Test for Stationarity for the Second Difference of Inflation Data

Table 3.4 indicates that almost all the ACF and PACF are within the range except lag 4, 11, 12 and 24. So it can be said that the data are stationay now.

Figure 5.7 shows the second difference of inflation of Bangladesh with most significant spike at lag 12. ACF, PACF plot and unit root test indicate that the second \ difference of inflation data is stationary.

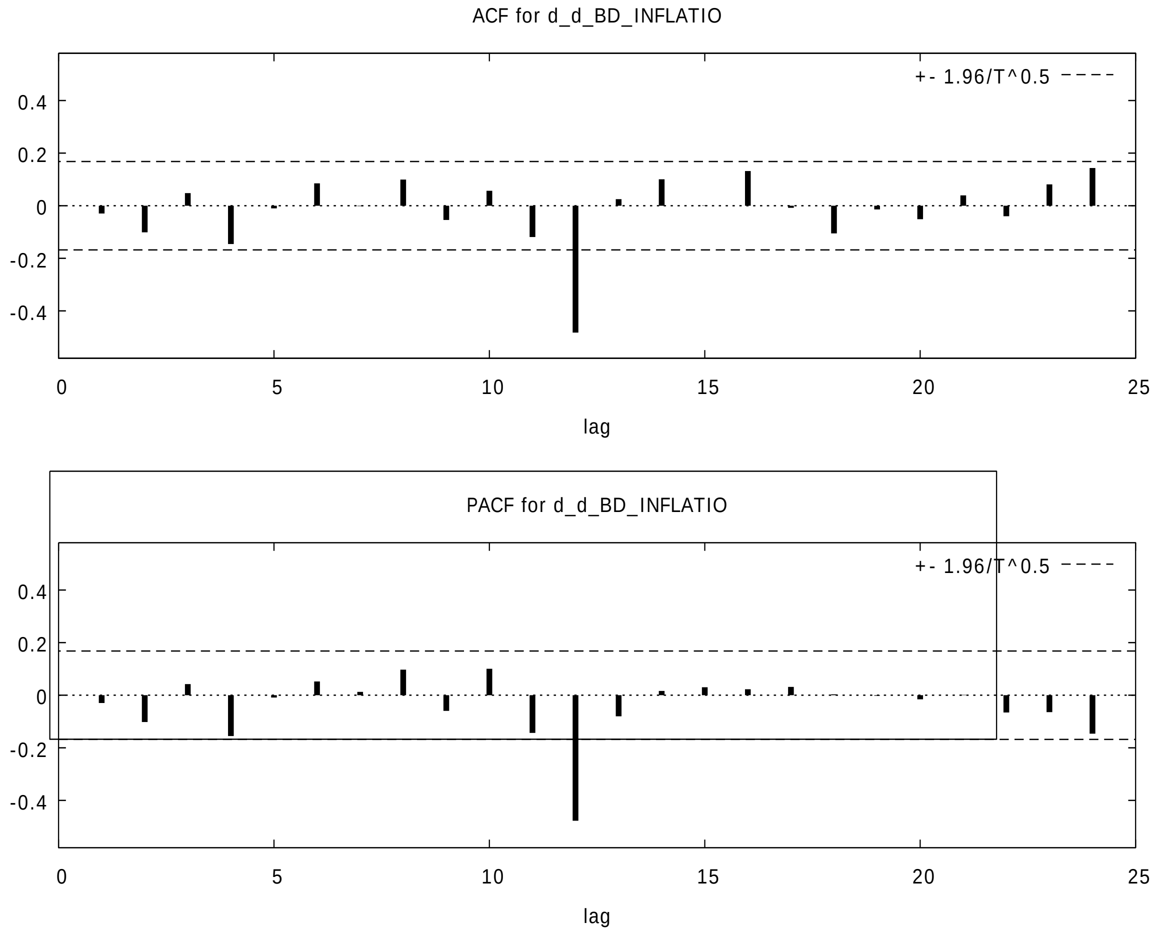


Figure 3.8 Autocorrelation function for the second difference of inflation

Table 3.7 Autocorrelation function for second difference of inflation

LAG	ACF	PACF	Q-stat. [p-value]
1	-0.0296	-0.0296	0.1220 [0.727]
2	-0.1011	-0.1021	1.5531 [0.460]
3	0.0481	0.0422	1.8791 [0.598]
4	-0.1458 *	-0.1554 *	4.9002 [0.298]
5	-0.0102	-0.0088	4.9149 [0.426]
6	0.0852	0.0522	5.9625 [0.427]
7	-0.0017	0.0127	5.9630 [0.544]
8	0.0995	0.0970	7.4151 [0.493]
9	-0.0543	-0.0596	7.8504 [0.549]
10	0.0568	0.1006	8.3307 [0.597]
11	-0.1191	-0.1433 *	10.4621 [0.489]
12	-0.4825 ***	-0.4774 ***	45.7045 [0.000]
13	0.0251	-0.0801	45.8005 [0.000]
14	0.1005	0.0161	47.3550 [0.000]
15	0.0012	0.0304	47.3552 [0.000]
16	0.1319	0.0225	50.0772 [0.000]
17	-0.0080	0.0316	50.0872 [0.000]
18	-0.1052	0.0035	51.8475 [0.000]
19	-0.0141	-0.0022	51.8794 [0.000]
20	-0.0513	-0.0157	52.3049 [0.000]
21	0.0394	0.0014	52.5576 [0.000]
22	-0.0398	-0.0655	52.8189 [0.000]
23	0.0812	-0.0644	53.9150 [0.000]
24	0.1435 *	-0.1462 *	57.3646 [0.000]

Unit root test for the second difference of inflation:

Augmented Dickey-Fuller test for second difference of inflation including 11 lags of (1-L)d_d_BD_INFLATION (max was 12)Sample size 124, Unit-root null hypothesis: a = 1.

Table 3.8 Unit root test for the second difference of inflation

Test with constant	Test with constant and trend
Model: (1-L)y = b0 + (a-1)*y(-1) + ... + e 1st-order autocorrelation coefficient for e: 0.005 Lagged differences: F(11, 111) = 5.760 [0.0000] Estimated value of (a - 1): -1.64844 Test statistic: tau_c(1) = -5.9453 Asymptotic p-value 1.62e-007	Model: (1-L)y = b0 + b1*t + (a-1)*y(-1) + ... + e 1st-order autocorrelation coefficient for e: 0.004 Lagged differences: F(11, 110) = 5.714 [0.0000] Estimated value of (a - 1): -1.654 Test statistic: tau_ct(1) = -5.92798 Asymptotic p-value 1.733e-006

3.11 Identification and Estimation for the Five Models

Having determined the correct order of differencing required to make the series stationary, the next step is to find an appropriate ARMA form to model the stationary series. There are number of alternative identification methods proposed in the literature. Here the the Box-Jenkins procedure is used which follows an iterative process for model identification, model estimation and model valuation. Various ARIMA models are established among which the three suitable models which were satisfying all the properties of residual. Further the parameters were significantly impacting the inflation.

For an ARIMA model if the terms of AR process and MA process are statistically significant then it seems to be plausible.

Table 3.9 Estimation of different ARIMA model

Model		*Coefficient*	*Std. Error*	*z*	*p-value*
Model:1 ARIMA (12,24;2;1,11)	const	-0.000312581	0.00275965	-0.1133	0.90982
	phi_12	-0.719766	0.10906	-6.5997	<0.00001***
	phi_24	-0.280269	0.0998767	-2.8061	0.00501***
	theta_1	-0.220937	0.0972159	-2.2726	0.02305**
	theta_11	-0.209717	0.10416	-2.0134	0.04407**
Model:2 SARIMA(0;2;1,11)(0;0;1)	const	-0.00103306	0.0014559	-0.7096	0.4780
	theta_1	-0.227297	0.0892144	-2.548	0.0108 **
	theta_11	-0.220390	0.0964650	-2.285	0.0223 **
	Theta_1	-0.802930	0.107834	-7.446	9.62e-014 ***
Model:3 SARIMA(24;2;1,11)(1;0;0)	const	-0.000928457	0.00219458	-0.4231	0.6722
	phi_24	-0.360366	0.0995118	-3.621	0.0003 ***
	Phi_1	-0.698170	0.0727884	-9.592	8.66e-022 ***
	theta_1	-0.209299	0.0943697	-2.218	0.0266 **
	theta_11	-0.228959	0.0974791	-2.349	0.0188 **
Model:4 ARIMA (1;2;11,12)	const	-0.00100620	0.00169653	-0.5931	0.5531
	phi_1	-0.195526	0.0890834	-2.195	0.0282 **
	theta_11	-0.154678	0.0738746	-2.094	0.0363 **
	theta_12	-0.845322	0.112436	-7.518	5.55e-014 ***
Model:5 SARIMA (11;2;1)(0;0;1)	const	-0.00104429	0.00163163	-0.6400	0.5222
	phi_11	-0.172749	0.0921209	-1.875	0.0408 **
	theta_1	-0.204409	0.0927563	-2.204	0.0275 **
	Theta_1	-0.833610	0.110316	-7.557	4.14e-014 ***

But whether the specification of ARIMA model is really a good one, depends on the residual analysis which requires look at the diagnostic checking.

3.12 Diagnostics for the Five Models

The ARIMA models will be reliable if the residual of the regression is white noise. To test this, we use the correlogram structure of the residual. This is shown in Figures and tables in the next pages which show at any lag the there is no correlation among the residuals. Therefore the estimated regression models are reliable.

Model-1: ARIMA (12,24;2;1,11)

Here for the Auto regressive term 12^{th} and 24^{th} lags are selected. For the moving average term first and 11^{th} lags are selected.

The residual autocorrelation function for this model is given below.

Table 3.10 Residual autocorrelation function for model-1

LAG	ACF	PACF	Q-stat. [p-value]
1	-0.0326	-0.0326	0.1479 [0.701]
2	-0.0672	-0.0684	0.7809 [0.677]
3	0.0411	0.0367	1.0191 [0.797]
4	-0.0999	-0.1026	2.4380 [0.656]
5	-0.0351	-0.0367	2.6150 [0.759]
6	0.0991	0.0829	4.0334 [0.672]
7	-0.0378	-0.0307	4.2410 [0.752]
8	0.0201	0.0234	4.3002 [0.829]
9	-0.0560	-0.0742	4.7631 [0.854]
10	-0.0746	-0.0582	5.5923 [0.848]
11	-0.0593	-0.0757	6.1207 [0.865]
12	0.0963	0.0842	7.5237 [0.821]
13	-0.0888	-0.0978	8.7270 [0.793]
14	0.0394	0.0315	8.9653 [0.833]
15	0.0380	0.0162	9.1894 [0.867]
16	0.0077	0.0409	9.1986 [0.905]
17	-0.0360	-0.0380	9.4034 [0.927]
18	-0.0536	-0.0789	9.8611 [0.936]
19	-0.0455	-0.0347	10.1931 [0.948]
20	-0.0177	-0.0524	10.2437 [0.964]
21	-0.0267	-0.0299	10.3597 [0.974]

The residual ACF and PACF function shows that at all the lags the p-values are not statistically significant which indicates that the model is reasonable. ACF and PACF plot can also show it clearly.

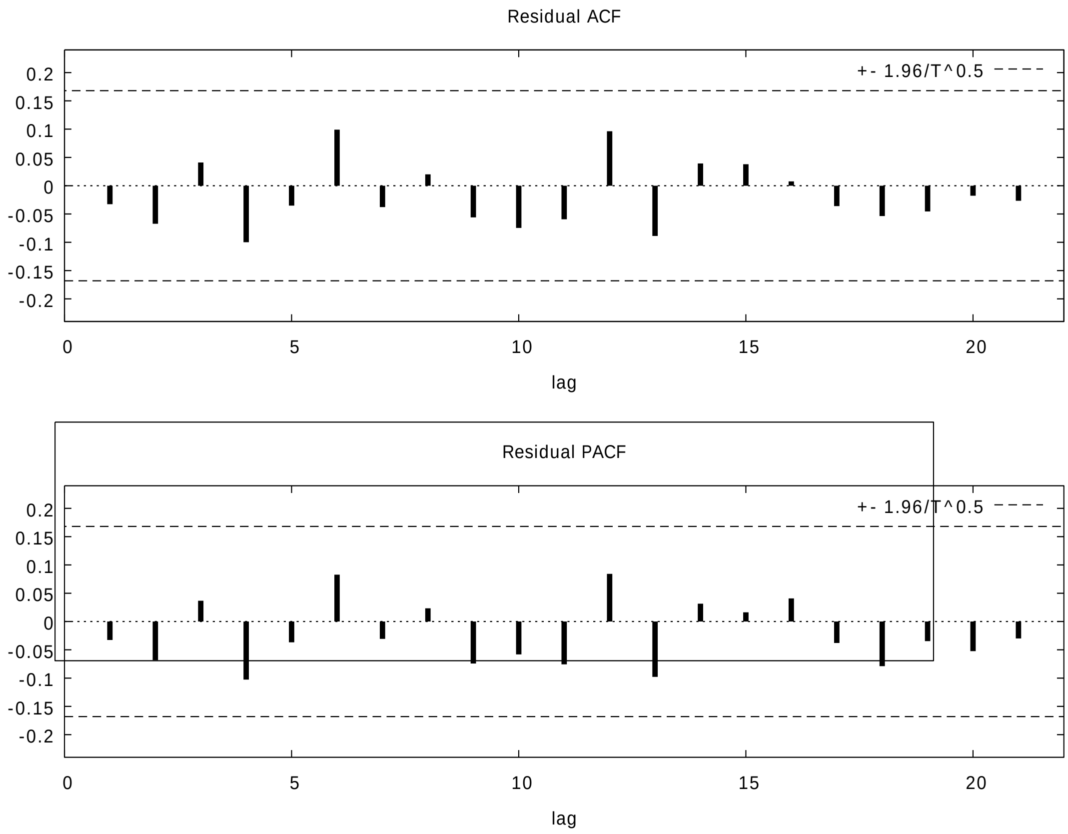


Figure 3.9 ACF and PACF plot of residual for model-1

Figure 3.8 indicates that the residual correlogram does not cross the limit of 5% level of significance.

Model-2: SARIMA (0;2;1, 11)(0;0;1)$_{12}$

Here for the Auto regressive term no lag is selected. For the moving average term first and 11th lags are selected. For the seasonal term only the first lag of moving average is taken with no differences.

The residual autocorrelation function for this model is given in the next page.

Table 3.11 Residual autocorrelation function for model-2

LAG	ACF	PACF	Q-stat. [p-value]
1	-0.0310	-0.0310	0.1335 [0.715]
2	-0.0858	-0.0869	1.1651 [0.558]
3	0.0215	0.0161	1.2304 [0.746]
4	-0.0986	-0.1058	2.6134 [0.624]
5	-0.0320	-0.0359	2.7604 [0.737]
6	0.0968	0.0775	4.1123 [0.661]
7	-0.0588	-0.0576	4.6157 [0.707]
8	0.0154	0.0188	4.6507 [0.794]
9	-0.0503	-0.0705	5.0242 [0.832]
10	-0.0634	-0.0478	5.6220 [0.846]
11	-0.0582	-0.0806	6.1302 [0.865]
12	0.1070	0.0895	7.8615 [0.796]
13	-0.0951	-0.1072	9.2412 [0.754]
14	0.0154	0.0105	9.2777 [0.813]
15	0.0190	-0.0078	9.3338 [0.859]
16	0.0175	0.0405	9.3817 [0.897]
17	-0.0409	-0.0494	9.6460 [0.918]
18	-0.0466	-0.0766	9.9907 [0.932]
19	-0.0532	-0.0440	10.4451 [0.941]
20	-0.0259	-0.0660	10.5534 [0.957]
21	-0.0175	-0.0265	10.6035 [0.970]

The residual ACF and PACF function shows that at all the lags the p-values are not statistically significant which indicates that the model is reasonable. ACF and PACF plot can also show it clearly.

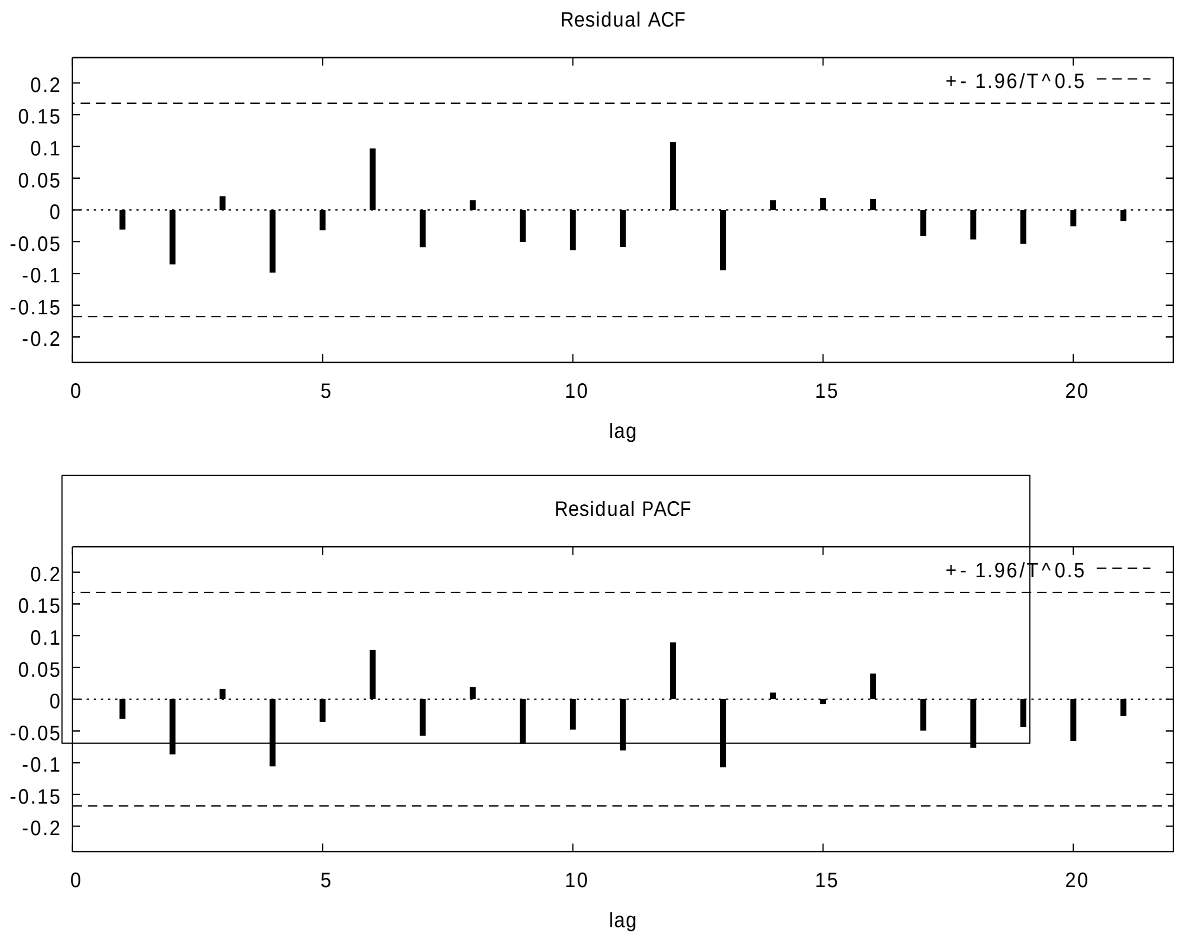


Figure 3.10 ACF and PACF plot of residual for model-2

Figure 3.10 indicates that the residual correlogram does not cross the limit of 5% level of significance.

Model-3: SARIMA(24;2;1,11)(1;0;0)$_{12}$

Here for the Auto regressive term 24th lag is selected. For the moving average term first and 11th lags are selected. For the seasonal term only the first lag of auto regression is taken with no differences.

The residual autocorrelation function for this model is given in the next page.

Table-3.12 Residual autocorrelation function for model-3

LAG	ACF	PACF	Q-stat. [p-value]
1	-0.0199	-0.0199	0.0550 [0.815]
2	-0.0923	-0.0928	1.2488 [0.536]
3	0.0171	0.0134	1.2903 [0.731]
4	-0.0854	-0.0942	2.3270 [0.676]
5	-0.0243	-0.0255	2.4114 [0.790]
6	0.0946	0.0775	3.7044 [0.717]
7	-0.0705	-0.0711	4.4276 [0.729]
8	-0.0004	0.0067	4.4276 [0.817]
9	-0.0318	-0.0527	4.5770 [0.870]
10	-0.0462	-0.0325	4.8953 [0.898]
11	-0.0457	-0.0645	5.2084 [0.921]
12	0.0662	0.0485	5.8722 [0.922]
13	-0.0878	-0.0948	7.0485 [0.900]
14	0.0321	0.0288	7.2073 [0.926]
15	0.0384	0.0167	7.4365 [0.944]
16	0.0108	0.0266	7.4547 [0.963]
17	-0.0485	-0.0546	7.8259 [0.970]
18	-0.0828	-0.1036	8.9166 [0.962]
19	-0.0402	-0.0296	9.1761 [0.970]
20	-0.0095	-0.0516	9.1907 [0.981]
21	-0.0086	-0.0223	9.2029 [0.988]

The residual ACF and PACF function shows that at all the lags the p-values are not statistically significant which indicates that the model is reasonable. ACF and PACF plot can also show it clearly.

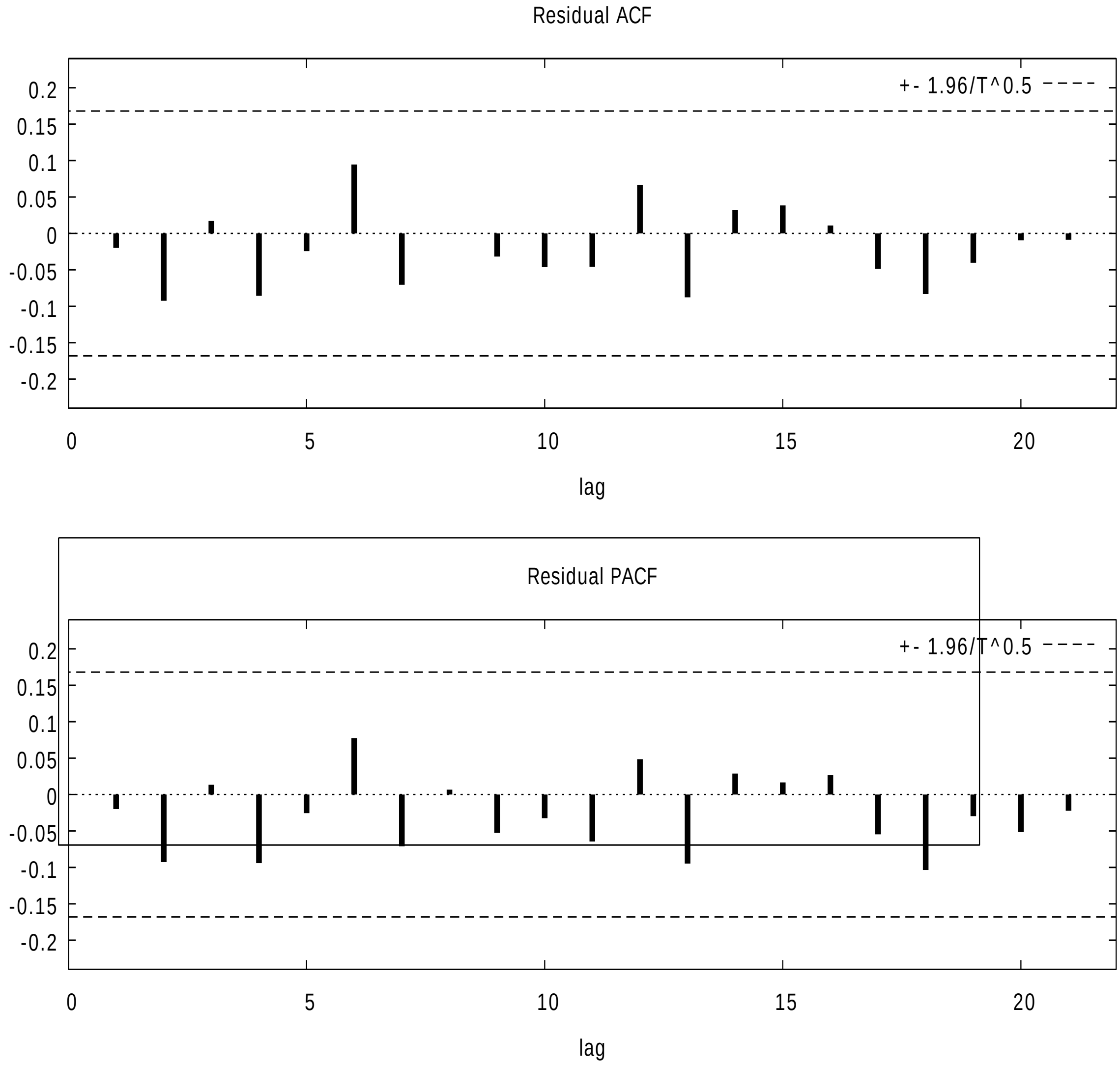


Figure 3.11 ACF and PACF plot of residual for model-3

Figure 3.11 indicates that the residual correlogram does not cross the limit of 5% level of significance.

Model-4: ARIMA (1;2;11,12)

Here for the Auto regressive term only the first lag is selected. For the moving average term 11^{th} and 12^{th} lags are selected.

The residual autocorrelation function for this model is given on the next page.

Table 3.13 Residual autocorrelation function for model-4

LAG	ACF	PACF	Q-stat. [p-value]
1	-0.0634	-0.0634	0.5590 [0.455]
2	-0.1036	-0.1081	2.0633 [0.356]
3	0.0298	0.0159	2.1886 [0.534]
4	-0.1019	-0.1117	3.6652 [0.453]
5	-0.0381	-0.0490	3.8733 [0.568]
6	0.1114	0.0840	5.6653 [0.462]
7	-0.0347	-0.0280	5.8400 [0.559]
8	0.0265	0.0352	5.9431 [0.654]
9	-0.0226	-0.0379	6.0184 [0.738]
10	-0.0434	-0.0223	6.2984 [0.790]
11	-0.1373	-0.1531 *	9.1294 [0.610]
12	0.0594	0.0301	9.6636 [0.645]
13	-0.1273	-0.1611 *	12.1364 [0.516]
14	0.0316	0.0143	12.2902 [0.583]
15	0.0408	-0.0180	12.5482 [0.637]
16	0.0207	0.0374	12.6155 [0.701]
17	-0.0307	-0.0273	12.7638 [0.752]
18	-0.0386	-0.0605	13.0013 [0.791]
19	-0.0557	-0.0355	13.4993 [0.812]
20	-0.0091	-0.0556	13.5128 [0.854]
21	-0.0034	-0.0223	13.5147 [0.890]

The residual ACF and PACF function shows that at all the lags the p-values are not statistically significant which indicates that the model is reasonable. ACF and PACF plot can also show it clearly.

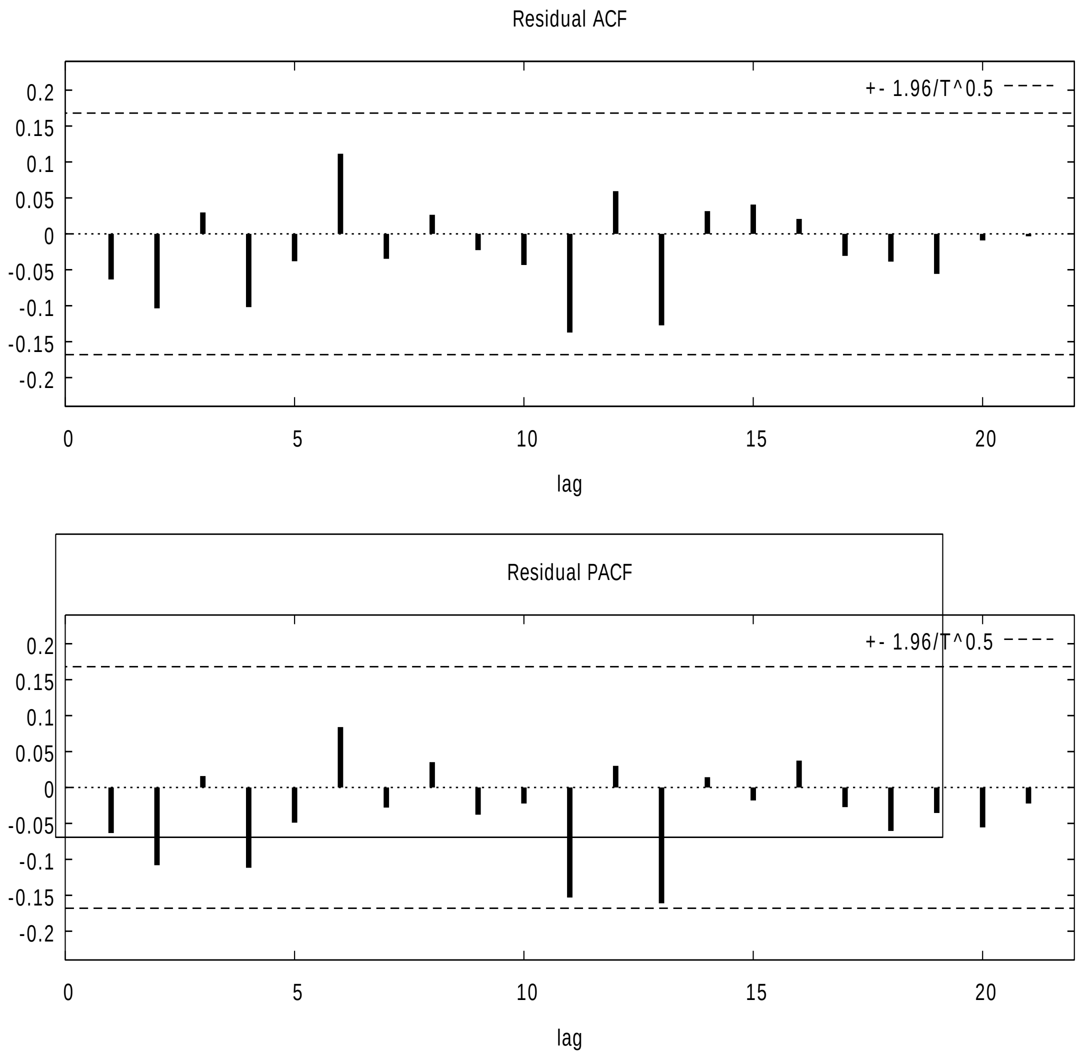


Figure 3.12 ACF and PACF plot of residual for model-4

Figure 3.12 indicates that the residual correlogram does not cross the limit of 5% level of significance.

Model-5 SARIMA (11;2;1)(0;0;1)$_{12}$

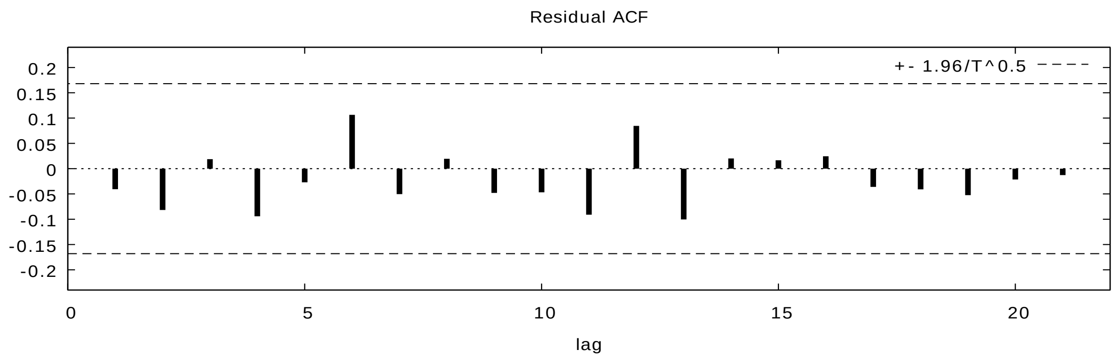
Here for the Auto regressive term 11th lag is selected. For the moving average term first lag is selected. For the seasonal term only the first lag of moving average is taken with no differences.

The residual autocorrelation function for this model is given in the next page.

Table 3.14 Residual autocorrelation function for Model-5

LAG	ACF	PACF	Q-stat. [p-value]
1	-0.0406	-0.0406	0.2297 [0.632]
2	-0.0817	-0.0835	1.1644 [0.559]
3	0.0188	0.0119	1.2140 [0.750]
4	-0.0942	-0.1006	2.4760 [0.649]
5	-0.0270	-0.0333	2.5805 [0.764]
6	0.1064	0.0886	4.2157 [0.648]
7	-0.0504	-0.0460	4.5848 [0.710]
8	0.0195	0.0241	4.6405 [0.795]
9	-0.0479	-0.0634	4.9794 [0.836]
10	-0.0468	-0.0300	5.3052 [0.870]
11	-0.0910	-0.1105	6.5483 [0.834]
12	0.0845	0.0658	7.6302 [0.813]
13	-0.1004	-0.1175	9.1684 [0.760]
14	0.0202	0.0134	9.2310 [0.816]
15	0.0166	-0.0136	9.2737 [0.863]
16	0.0244	0.0421	9.3668 [0.898]
17	-0.0361	-0.0371	9.5719 [0.921]
18	-0.0409	-0.0678	9.8385 [0.937]
19	-0.0524	-0.0392	10.2785 [0.946]
20	-0.0214	-0.0628	10.3524 [0.961]
21	-0.0127	-0.0218	10.3787 [0.974]

The residual ACF and PACF function shows that at all the lags the p-values are not statistically significant which indicates that the model is reasonable. ACF and PACF plot can also show it clearly. Figure 3.13 indicates that the residual correlogram does not cross the limit of 5% level of significance.



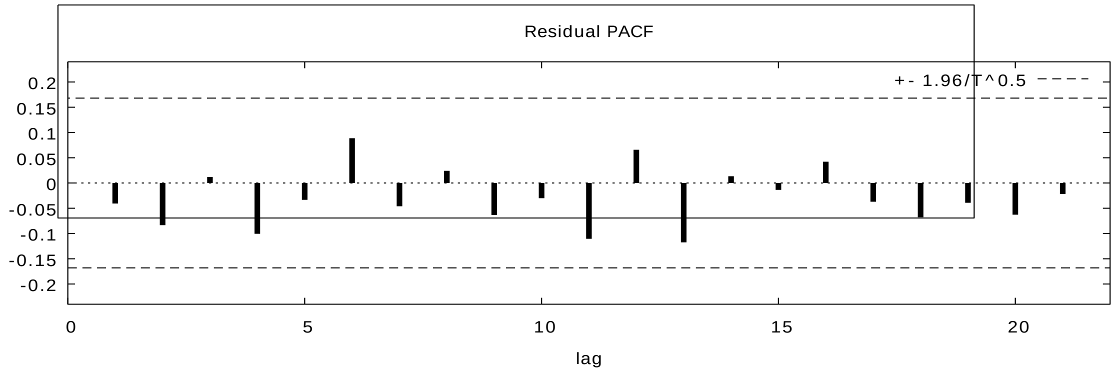


Figure 3.13 ACF and PACF plot of residual for model-5

CHAPTER 4

FORECASTING INFLATION BY ANN MODEL

4.1 Introduction

Detecting trends and patterns in economic and financial data are of great interest to the business world to support the decision-making process. A new generation of methodologies, including neural networks, knowledge-based systems and genetic algorithms, has attracted attention for analysis of trends and patterns.

In particular, neural networks are being used extensively for economic and financial forecasting with stock markets, foreign exchange rate, commodity price, inflation, future trading and bond yields. The application of neural networks in time series forecasting is based on the ability of neural networks to approximate nonlinear functions. In fact, neural networks offer a novel technique that doesn't require a pre-specification during the modeling process because they independently learn the relationship inherent in the variables.

4.2 Background and Methodology

Neural network theory grew out of Artificial Intelligence research, or the research in designing machines with cognitive ability. An artificial neural network is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements, called neurons, working in unison to solve specific problems. ANN learns by experience like people. An ANN is configured for a specific application, such as pattern recognition and time series forecasting, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons. The basic building block of a brain and the neural network is the neuron. The human neuron is shown in Figure 4.1.

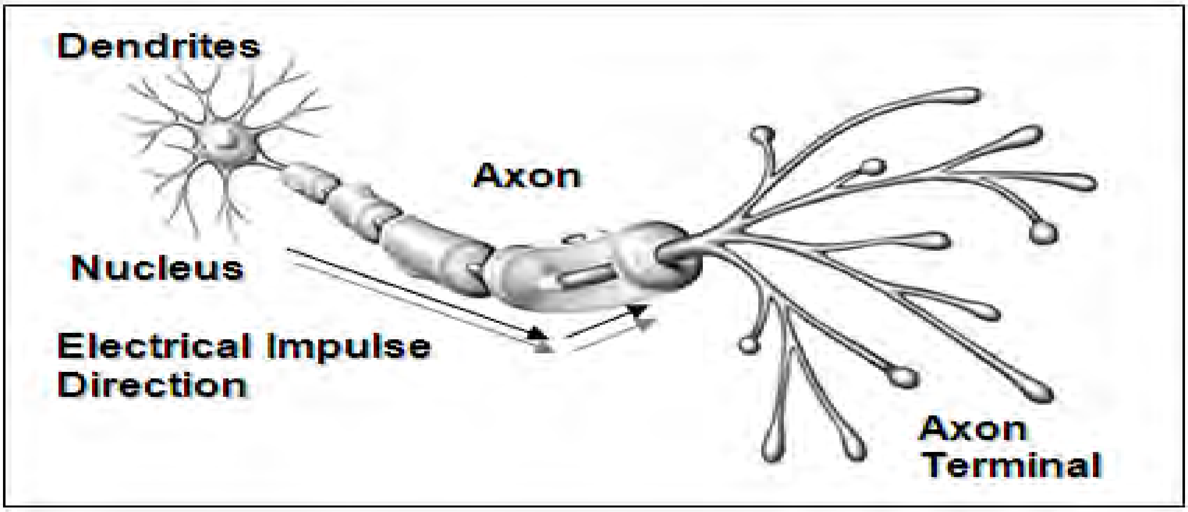


Figure 4.1 Biological Model of Human Neuron (Beale and Jackson (1990))

A neural network is a massively parallel distributed processor that has a natural propensity for storing experiential knowledge and making it available for use. It resembles the brain in two respects: Knowledge is acquired by the network through a learning process and interneuron connection strengths known as synaptic weights are used to store the knowledge (see, for instance, Hykin, 1994).

As described by Beal and Jackson (1990), all inputs to the cell body of the neuron arrive along dendrites. Dendrites can also act as outputs interconnecting inter-neurons. Mathematically, the dendrite's function can be approximated as a summation. Axons, on the other hand, are found only on output cells. It has an electrical potential. If excited, past a threshold, it will transmit an electrical signal. Axons terminate at synapses that connect it to the dendrite of another neuron. The neuron sends out spikes of electrical activity through a long axon, which splits into thousands of branches, see, Figure 4.2. At the end of each branch, a structure called a synapse converts the activity from the axon into electrical effects that inhibit or excite activity from the axon into electrical effects that inhibit or excite activity in the connected neurons. When a neuron receives excitatory input that is sufficiently large compared with its inhibitory input, it sends a spike of electrical activity down its axon. Learning occurs by changing the effectiveness of the synapses so that the influence of one neuron on another changes. The human brain contains approximately 10 billion interconnected neurons creating its massively parallel computational capability.

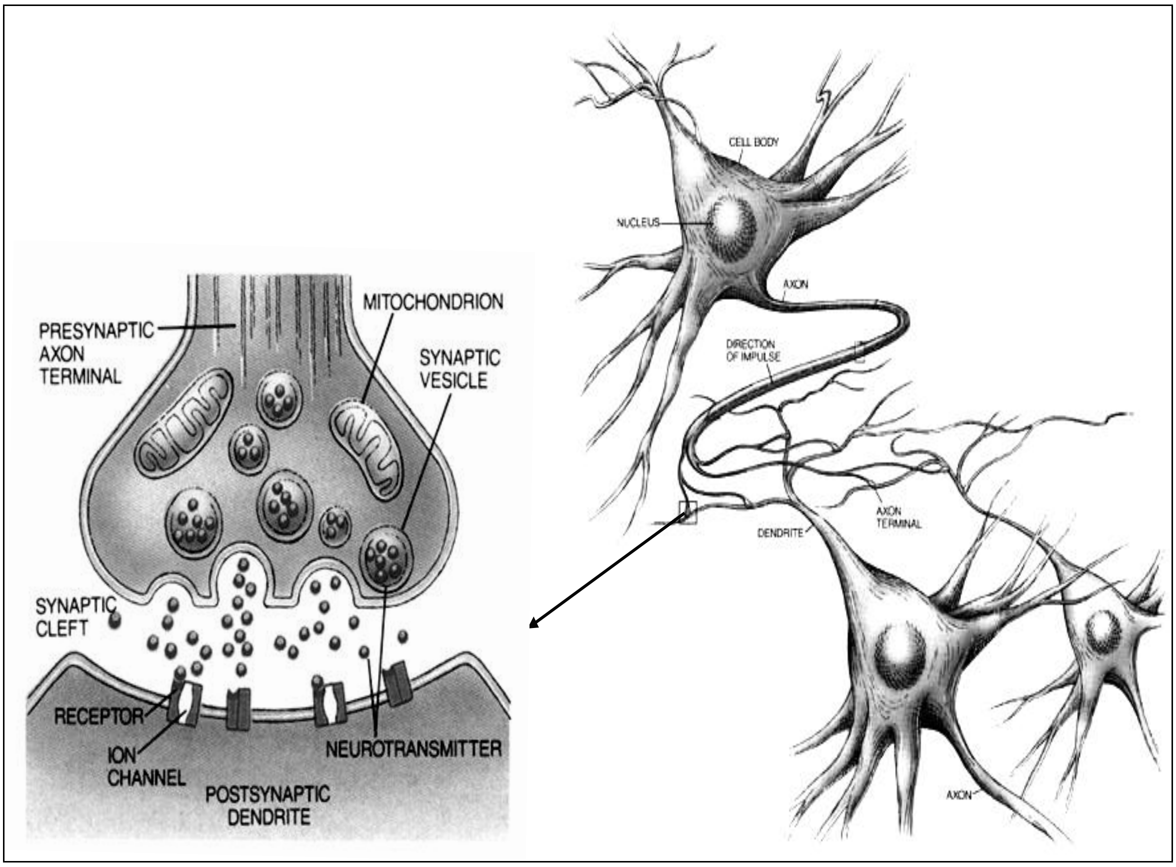


Figure 4.2 Neural Signal Transmission (Kartalopoulos (1996) and Haykin (1994))

4.3 Artificial Neuron

The artificial neuron was developed in an effort to model the human neuron. The artificial neuron depicted in Figure 4.3. Inputs enter the neuron and are multiplied by their respective weights. For analytical purposes, a neuron may be broken down into three parts:

- input connections
- summing and activation functions
- output connections

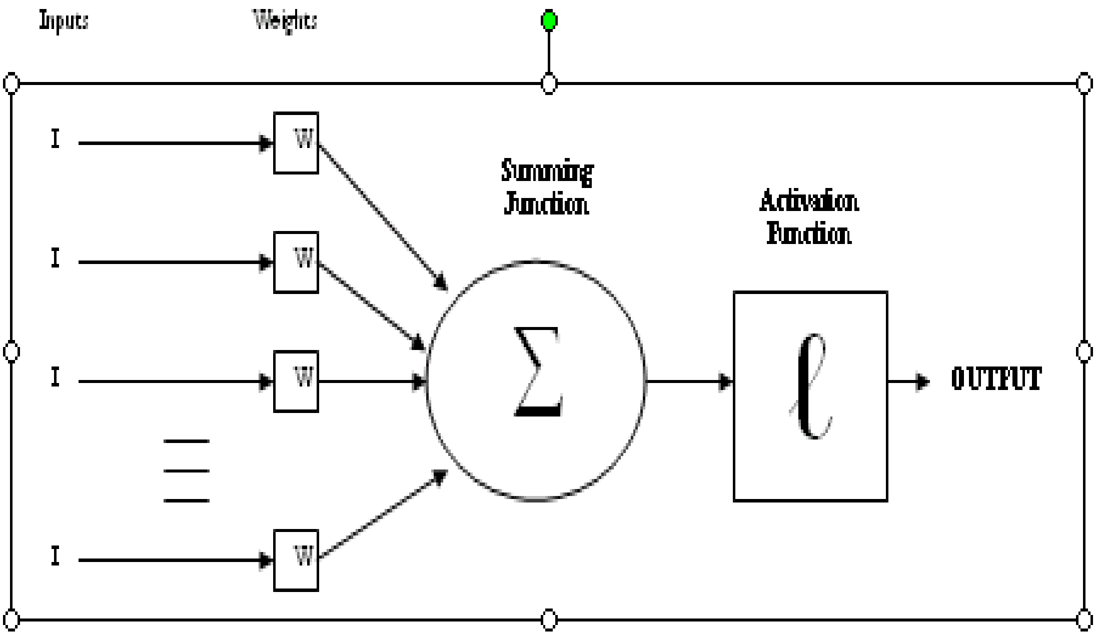


Figure 4.3 Artificial Neuron Model

4.2.1 Input Connections

In artificial neural network, a neuron is connected to other neurons and depends on them to receive the information that it processes. There is no limit to the amount of connections a neuron may receive information from. The information that a neuron receives from others is regulated through the use of weights. When a neuron receives information from other neurons, each piece of information is multiplied by a weight with a value between –1 and +1, which allows the neuron to judge how important the information it receives from its input neurons is. These weights are integral to the way a network works and is trained: specifically, training a network means modifying all the weights regulating information flow to ensure output follows the given criteria, e.g., minimization of RMSE or MAE.

4.2.2 Summing and Activation Functions

The second portion of a neuron is the summing and activation functions. The information sent to the neuron and multiplied by corresponding weights is added together and used as a parameter within an activation function. (In a biological context, a neuron becomes activated when it detects electrical signals from the neurons it is connected (see, Beale and Jackson, 1990). If these signals are sufficient, the neuron will become "activated" – it will send electrical signals to the neurons connected to it.)

Numerous activation functions exist in ANN literature, but we will discuss below the one which we used and that is hyperbolic tangent function: a continuous function with a domain of (–∞, ∞) and a range of (–1, 1):

$$\tanh(x) = \frac{e^{2x}-1}{e^{2x}+1} \qquad \text{…….(4.1)}$$

By providing a function with a limitless domain and a range of (–1, 1), it is perfect for predicting whether or not inflation will rise (tanh (x) = 1) or fall (tanh (x) = –1).

4.3.3 Output Connections

Finally, once the activation function returns a corresponding value for the summed inputs, these values are sent to the neurons that treat the current neuron as an input. The process repeats again, with the current neuron's output being summed with others, and more activation functions accepting the sum of these inputs. The only time this may be ignored is if the current neuron is an output neuron. In this case, the summed inputs and normalized sum is sent as an output and not processed again.

4.4 Neural Network Architecture

While each neuron is, in and of itself, a computational unit, neurons may be combined into layers to create complex but efficient groups that can learn to distinguish between patterns within a set of given inputs. Indeed, by combining multiple layers of such groups, it is theoretically possible to learn any pattern. There are many combinations of neurons that allow one to create different types of neural networks, but the simplest type is a single-layer feedforward network. In this case, a network is composed of three parts: a layer of input nodes, a layer of hidden neurons, and a layer of output nodes, as is shown in the Figure 4.4.

A multilayer feedforward network is similar to a single-layer one. The main difference is that instead of having a hidden layer pass its calculated values to an output layer, it passes them on to another hidden layer. Both types of networks are typically implemented by fully connecting each layer's neurons with the preceding layer's neurons. Thus, if Layer A has k neurons and sends its information to Layer B, with n neurons, each neuron in Layer A has n connections for its calculated output, while each neuron in Layer B has k input connections.

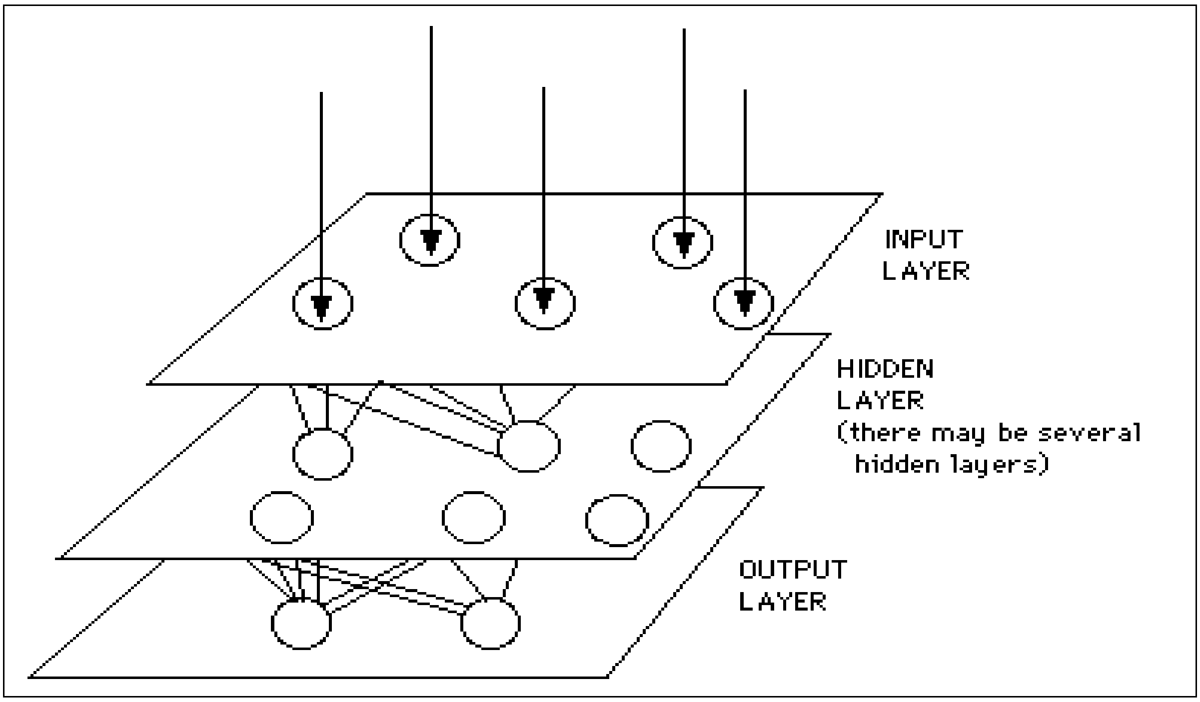


Figure 4.4 Layers of Artificial Neural Networks

Interestingly, such a network can be represented mathematically in a simple manner. Supposing there are k neurons in *Layer A*, let a represent a vector, where a_i is the i^{th} neuron's activation function output. Let b represent the input values to neurons in *Layer B*, with b_j be the j^{th} neuron. Let W be a n by k matrix where w_{ji} represents the weight affecting the connection from a_i to b_j. Keeping this in mind, we can see that for a single-layer feedforward network, we can mathematically represent the flow of information by,

$$Wa = b \qquad \text{.......(4.2)}$$

and the learning thus becomes a modification of each w_{ji} in W. A similar mathematical analogy applies to multilayer feedforward networks, but in this case, there is a W for every layer and 'b' is used as the value for 'a' when moving to subsequent layers. The most popular type of learning within a single-layer feedforward network is the Delta Rule, while multilayer feedforward networks implement the Backpropagation algorithm, which is a generalization of the Delta Rule (*see*, Beale and Jackson, 1990).

The Delta Rule may be summarized with the following equation:

$$\Delta w_{ij} = - \in \delta_j x_i \qquad \text{.......(4.3)}$$

In this case, Δw_{ij} represents the change of the weight connecting the i^{th} neuron with the j^{th} output neuron, x_i is the output value of the i^{th} neuron, ε is the learning rate, and δj is the error term in the output layer, defined as:

$$\delta_k = -(t_k - o_k) \quad \text{.......(4.4)}$$

where t_k is the expected output, while o_k is the actual output. While this rule works well when there is only one hidden layer In case of multiple layers we use generalized delta rule described below.

$$\delta_j = \left(\sum_k \delta_k w_{jk} \right) \left(h_j \left(1 - h_j \right) \right) \quad \text{.......(4.5)}$$

In this case, one uses the same equation for Δw_{ij} but uses the term above instead, with k representing the neurons receiving information from the current neuron being modified. $\delta_k w_{jk}$ is the error term of the k^{th} neuron in the receiving layer, with w_{jk} being the connecting weight. The activation functions of all the neurons in a network implementing backpropagation must be differentiable, because:

$$h_j = \sigma'(z_j) \quad \text{.......(4.5)}$$

with z_j being the net input for the neuron.

Finally, if biases are present, they are treated like regular neurons, but with their output (x) values equal to 1:

$$\Delta B_j = - \in \delta_j \quad \text{.......(4.6)}$$

When implemented, this algorithm has two phases. The first deals with having the network evaluate the inputs with its current weights. Once this is done, and all the neuron and output values are recorded, phase two begins. The algorithm begins this phase by applying the original Delta Rule to the output neurons, modifying weights as necessary. Then the generalized Delta Rule is implemented, with the previous δ values sent to hidden neurons, and their weights changing as required.

The neuron receives inputs from one or more inputs. The output of this neuron depends upon the 'activation function' or 'transfer function' of the neuron. The basic transfer functions that we will be talking about in the coming chapters are the sigmoid (or the log-sigmoid transfer function) and the tan based (tan- sigmoid) transfer function. They are described as follows:

4.5 Transfer Functions

4.5.1 Linear Transfer Function

The linear transfer function calculates the neuron's output by simply returning the value passed to it.

The linear transfer function is shown below.

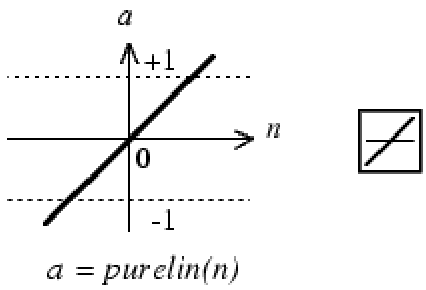


Figure 4.5 Linear Transfer Function

4.5.2 Log sigmoid transfer function

This network can receive inputs from negative infinity to positive infinity and always generates an output between 0 and 1.

The log sigmoid transfer function is shown in the next page.

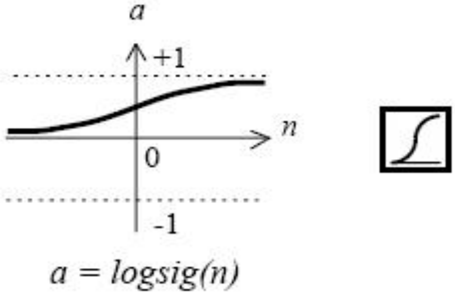


Figure 4.6 Log sigmoid transfer function.

4.5.3Tan sigmoid transfer function

This transfer function receives inputs from negative infinity to positive infinity and gives an output between -1 and 1.

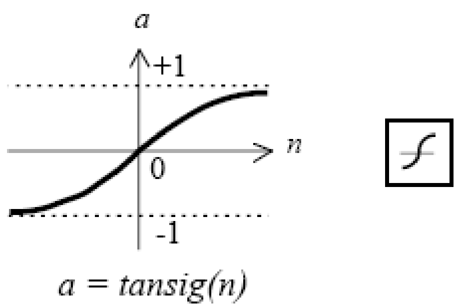


Figure 4.7 Tan sigmoid transfer function.

The arrangement of these neurons leads to different types of neural networks.
Broadly they can be classified as static and dynamic. Static networks do not have any feedback loops (outputs of a neuron fed back to some previous neuron) or taps (delay lines that feed the network with past values of inputs). Dynamic networks may have one of these two. Dynamic networks are preferable for time series as they have memory in the form of loops or delay lines.

4.6 The tapped delay line (TDL)

The taps or the delay line is used to feed the network with the past values of inputs. In figure 4.7 we see that input enters from the left and goes through N-1 delay elements to generate a vector of N outputs

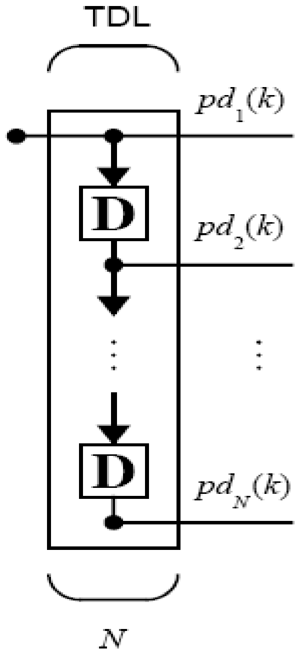


Figure 4.8 Tapped delay line.

Here The Levenberg–Marquardt algorithm is used to train the network which is described below.

4.7 Levenberg-Marquardt backpropagation

The Levenberg–Marquardt algorithm [L44,M63], which was independently developed by Kenneth Levenberg and Donald Marquardt, provides a numerical solution to the problem of minimizing a nonlinear function. It is fast and has stable convergence. In the artificial neural-networks field, this algorithm is suitable for training small- and medium-sized problems.

trainlm is a network training function that updates weight and bias values according to Levenberg-Marquardt optimization.

trainlm is often the fastest backpropagation algorithm in the toolbox, and is highly recommended as a first-choice supervised algorithm, although it does require more memory than other algorithms.

The backpropagation and feedforward algorithms are often used together. Just like many other types of neural networks, the feedforward neural network begins with an input layer. The input layer may be connected to a hidden layer or directly to the output layer. If it is connected to a hidden layer, the hidden layer can then be connected to another hidden layer or directly to the output layer.

4.8 Selection of neural network type

A very simple neural network is estimated for inflation based on‘feedforward with backpropagation’ architecture. “feedforward” term describes how this neural network processes and recalls patterns. In a feedforward neural network, neurons are only connected foreword. Each layer of the neural network contains connections to the next layer (for example, from the input to the hidden layer), but there are no connections back. The term “backpropagation” describes how this type of neural network is trained. Backpropagation is a form of supervised training. When using a supervised training method, the network must be provided with both sample inputs and anticipated outputs. The anticipated outputs are compared against the actual outputs for given input. Using the anticipated outputs, the backpropagation training algorithm then takes a calculated error and adjusts the weights of the various layers backwards from the output layer to the input layer.

Neural networks can in general be divided into two categories – static and dynamic. Static networks have no feedback elements and no delays. The output is calculated directly from the current inputs. Such networks assume that the data is concurrent and no sense of time can be encoded. These networks can thus lead to instantaneous behavior.

Dynamic networks may be difficult to train but are more powerful than static networks. As they have memory in form of delays or recurrent loops, they can be trained to learn sequential or time varying patterns. This makes them networks of choice for various applications like financial predictions, channel equalization, sorting, speech recognition, fault detection etc.

Since we are dealing with a time series it is necessary to use dynamic networks. Dynamic networks can be of two types ones with feed forward connections and taps and those with feedback or recurrent networks.

At this stage the NARX (Nonlinear Autoregressive Neural Network) was considered. NARX networks use taps to set up delays across the inputs and also incorporates the past values of the output. The networks have been employed in dynamic applications.

4.9 NARX network

The nonlinear autoregressive network with exogenous inputs (NARX) is a recurrent dynamic network, with feedback connections enclosing several layers of the network. The NARX model is based on the linear ARX model, which is commonly used in time-series model. All the specific dynamic networks have either been focused networks, with the dynamics only at the input layer, or feedforward networks.

The defining equation for the NARX model is

$$y(t) = f(y(t-1), y(t-2), \dots \dots, y(t-n_y), u(t-1), u(t-2), \dots \dots, u(t-n_u)) \quad (4.7)$$

It can also be used for nonlinear filtering, in which the target output is a noise-free version of the input signal. The use of the NARX network is shown in another important application, the modeling of nonlinear dynamic systems.

The defining equation for the NARX model is where the next value of the dependent output signal y(t) is regressed on previous values of the output signal and previous values of an independent (exogenous) input signal. One can implement the NARX model by using a feedforward neural network to approximate the function f. A diagram of the resulting network is shown below, where a two-layer feedforward network is used for the approximation. This implementation also allows for a vector ARX model, where the input and output can be multidimensional.

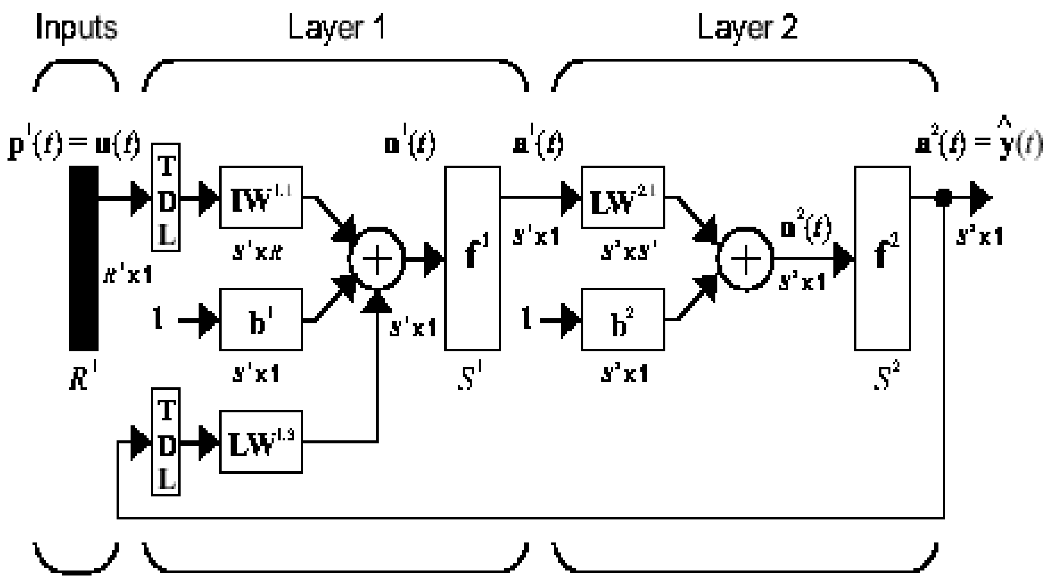


Figure 4.9 NARX network

There are many applications for the NARX network. It can be used as a predictor, to predict the next value of the input signal. It can also be used for nonlinear filtering, in which the target output is a noise-free version of the input signal.

Before showing the training of the NARX network, an important configuration that is useful in training needs explanation. One can consider the output of the NARX network to be an estimate of the output of some nonlinear dynamic system that one is trying to model. The output is fed back to the input of the feedforward neural network as part of the standard NARX architecture, as shown in the left figure below. Because the true output is available during the training of the network, you could create a series-parallel architecture [NaPa91], in which the true output is used instead of feeding back the estimated output, as shown in the right figure below. This has two advantages. The first is that the input to the feedforward network is more accurate. The second is that the resulting network has a purely feedforward architecture, and static backpropagation can be used for training.

The NARX network uses the past values of the actual time series to be predicted and past values of other inputs (like currencies of other nations and technical indicators in our case) to make predictions about the future value of the target series. These networks are again classified as series and parallel architecture

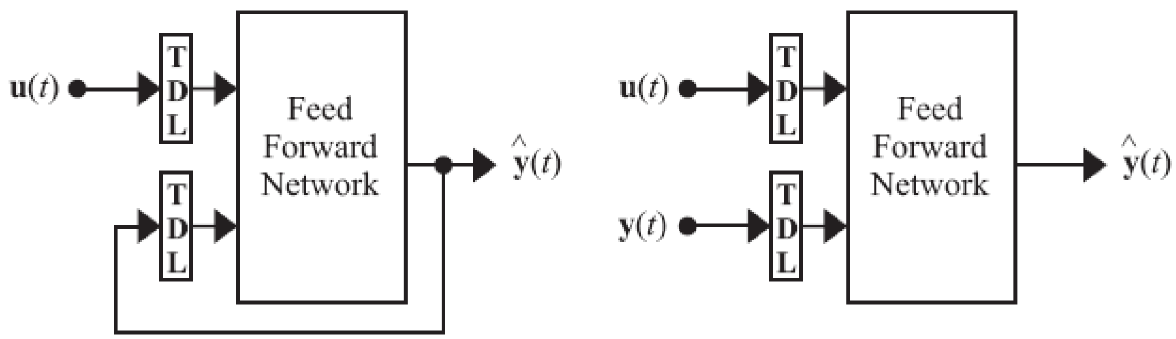


Figure 4.10 Parallel and Series architectures of NARX networks

In Figure 4.10 u(t) represents the past exogenous values (currencies of other nations and technical indicators in our case) y(t) represents the past values of the actual series to be predicted. ŷ(t) indicates the predicted values. If past values of actual series are not being recorded, they will not be available to the system. In such situations the networks uses its past predicted values. In our case we will have the actual past values; hence we prefer to use them instead of our predictions.

Thus it was able to base the model on actual values which are more reliable than the predictions.

The basic NARX network is used for multi step predictions. It is assumed that actual past values of target are not available and the predictions themselves are fed back to the network. Since there will have access to the actual past values it will provide those values instead of our past predictions. This helps the system train on actual values rather than predictions. This is achieved by using the series-parallel version of the NARX network which is described above. Thus a series parallel NARX dynamic network will be used as a basis of system here.

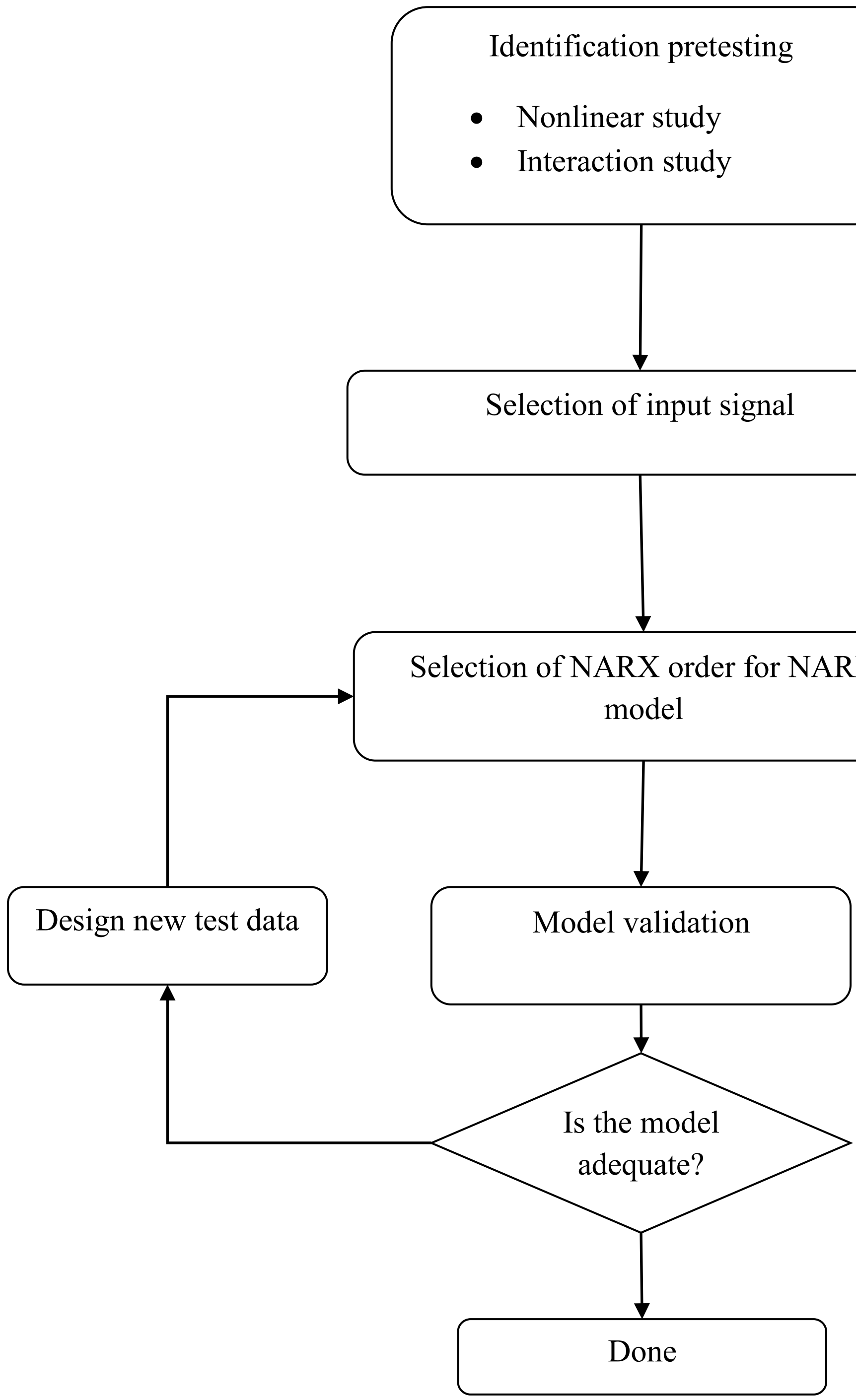


Figure 4.11 NARX network approach

4.11 Training and data preprocessing

Different researchers have different take about the method of preprocessing the data and training a neural network for such application. Here it will assume that it can feeded the inflation rate time series and then tried to predict the inflation rate of the next several months. So there is to just map the targets within the neuron range and train the network using an algorithm that is suited for such an application. The

MATLAB® Neural Network Toolbox provides a detailed survey of algorithms appropriate for various applications. The results therein state that Levenberg-Marquardt (LM) is a good algorithm for this case.

4.12 Network architectures to be considered

The NARX networks will have a Tansig input layer of neurons (default by MATLAB®) for the hidden and the output layers we will use the linear neurons. Tansig neurons and sigmoid neurons have been extensively used in most of the neural network applications. Both of them are similar in behavior and have a similar looking transfer function except for the range of outputs that they can generate. A sigmoid layer has a range from 0 to 1 whereas the range for Tansig transfer function ranges from -1 to 1. Klimasauskas(1993) suggests that sigmoid neurons be preferred to determine average behavior and Tan based layers to find deviations from normal. The application at hand also prompts us to use the Tansig layer. Additionally our development environment the MATLAB® Neural Network Toolbox also recommends Tansig layers for pattern recognition problems and provides it as the default layer. The Tansig activation function has been described previously.

4.13 Number of layers

We have a linear layer of linear neurons at the input; the number of neurons in this layer will be equal to the number of inputs that we have provided to the networks. Using linear neurons at the input is a standard practice and is used merely as an interface between the inputs and the hidden layers. In fact MATLAB® Neural Network Toolbox does not count it as an independent layer. There is no certain figure for the number of hidden layers to be used. Cybenko Hornik et al. (1989), (1991) show how a single layer of hidden neurons is capable of adapting to complex functions. Survey papers on this field, L. Chen-Hua (2001), also reveal how a single hidden layer of neurons is the most preferred option. Using additional layers adds up complexities to the model and increases the time required for training and simulation. The framework described earlier has capabilities to generate, simulate networks with multiple hidden layers. It can be seen from the code that the framework can generate the number of layers specified by the user, and use user specified number of layers for each layer. However the limited processing resources available did not permit us to perform tests with several layers, hence it is decided to use single layer of hidden neurons for this research.

4.14 Number of Taps and Hidden Neurons

The tapped delay line of the NARX network allows passing of past values to the network. They make the data sequential unlike the original concurrent dataset. Thus these taps set up a sense of time and correlation of past values. The numbers of neurons are supposed to be related to the complexity of the application at hand as each neuron in the hidden layer contributes weights and flexibility to the network. There is no standard method to determine the number of neurons to be used and several thumb rules are used. Mehta (1995) has stated how the architecture depends on these thumb rules and it is not necessary that they work well.

4.15 Policies for network selection

Here the proposed system is an adaptive system where the system will first train the NARX networks on the training set, then evaluate performance on the evaluation set. Then the results is used to obtained on the evaluation set to identify networks that are likely to predict the inflation on the test set.

It is believed that a network that behaves satisfactorily on this evaluation data will be a good candidate to make predictions for the test set (future). Now we need to decide which performance parameter of the evaluation set will be actually use to select a network. Popular performance parameters like mean square error, hit rate or realized value can be used. A network is selected that has minimum error on the evaluation set to make predictions on the test set (future).

4.16 Model evaluation performance

The evaluation of performance is essential with the purpose of finding the best neural network architecture, which gives the most reliable and accurate predictions. Based on previous researches, there are some performance function can be used to control the performance of network. Some might prefer the performance tool of the back-propagation algorithm is Mean Square Error (MSE) of training and testing. Moreover, the selection of MSE is supported by MATLAB software, which also the default indicator in training the network. The neural network model with the smallest MSE value is considered to be the best neural network model. Another performance tool is the regression R values. Regression R values show the correlation between outputs and targets value. An R value of 1 means close relationship, 0 means random relationship.

4.17 Implementation

The nonlinear autoregressive network with exogenous inputs (NARX) is a recurrent dynamic network, with feedback connections enclosing several layers of the network. The NARX model is based on the linear ARX model, which is commonly used in time-series modeling. Figure 5.1 illustrates the NARX network that is used in this thesis.

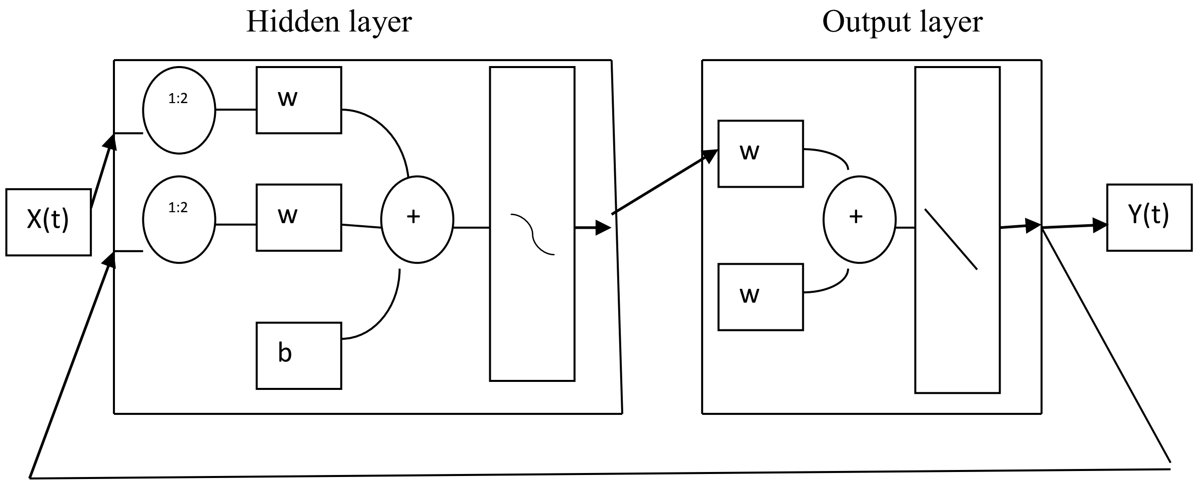


Figure 4.12 NARX network(used in this thesis)

The standard NARX network used here is a two-layer feedforward network, with a sigmoid transfer function in the hidden layer and a linear transfer function in the output layer. This network also uses tapped delay lines (d) to store previous values of the input, x(t) and output, y(t) sequences. First, the training data is loaded and use tapped delay lines with two delays, so training begins with the third data point. There are two inputs to the series-parallel network, the x (t) sequence and the y(t) sequence. For the x (t) sequence the exchange rate is used and for y(t) sequence inflation rate is used. Input and target series are divided in two groups of data. 1st group: used to train the network, 2nd group: this is the new data used for simulation. Input Series is used for predicting new targets. Target Series is used for network validation after prediction. The application randomly divides input vectors and target vectors into three sets as follows:

- 70% are used for training.
- 15% are used to validate that the network is generalizing and to stop training before overfitting.
- The last 15% are used as a completely independent test of network generalization.

y(t) is the output of the NARX network and also feedback to the input of the network and tapped delay lines (d) that store the previous values of x(t) and y(t) sequences. It also has been reported that gradient descent learning can be more effective in NARX networks than in other recurrent architecture. The standard Lavenberg-marquardt backpropagation algorithm is used to train the network with learning rate equal to 0.001. The method regularization has been used which consist of 1000 epoch and regularization parameter used is 1.00e-05. Training automatically stops when generalization stops improving, as indicated by an increase in the Mean Square Error (MSE) of the validation samples. The numbers of neurons in the hidden layer were found by trial and error method and finally 5 hidden neurons were chosen for the suggested network. The proposed network can be represented as 2-5-1, i.e. the proposed ANN model consists of 2 inputs, 5 hidden neurons and 1 hidden layer.

4.18 Summary of ANN model

Object model :	forecasting inflation rate
Input neuron :	Inflation, Exchange rate
Output neuron :	Inflation Rate
Network structure	
Network type :	Feed-forward back propagation
Transfer function :	Tansig/ Purelin
Training function :	Trainlm
Learning function :	Learngdm
Learning conditions	
Learning scheme :	Supervised learning
Learning rule :	Gradient descent rule

Input neuron :	Two
Output neuron :	One
Sample pattern vector :	96(for training), 40 (for testing)
Number of hidden layer :	1 (one)
Neurons in hidden layer :	5
Learning rate :	0.001
Performance goal/Error goal :	0.0001
Maximum epochs (cycles) set :	10,000
MSE at the end of training :	0.0036

CHAPTER 5

RESULT ANALYSIS

5.1 Forecast Evaluation and Forecast Accuracy Criteria of ARIMA Models

Evaluation criteria supported by different statistics shows the forecasting ability or predicting power of the models. There was an effort to forecast within the sample and out of sample inflation data. The purpose of forecasting within the sample is to test for the predictability power of the model. If the magnitude of the difference between the forecasted and actual values is low then the model has a good forecasting power. Due to this objectivity in computation, five models have been proposed which are highly supported by ARIMA model evaluation and selection criteria.

5.2 Performance Comparison of ARIMA Models Based on Evaluation statistics

The accuracy of each model can be checked to determine how the model performed in terms of in-sample forecast. For this purpose different forecast evaluation statistics such as Mean Absolute Error, Root Mean Squared Error and Theil's U statistics have been used.

Empirically taking, there have been examined that Table 5.1 reports the various measures of forecasting errors, namely the root mean squared error (RMSE), mean absolute error (MAE), and mean absolute percentage error (MAPE) and Theil's U and other selection criteria for different models. The first two forecast error statistics depend on the scale of the dependent variable. These are used as relative measures to compare forecasts for the same series across different models, the smaller the error, the better the forecasting ability of that model accordingly. The remaining two statistics are scale invariant. The Theil inequality coefficient always lies between zero and one, where zero indicates a perfect fit. To measure forecasting ability we have estimated within sample and out of sample forecasts. The estimated model is then used to obtain the future forecasts. According to the forecast evaluation criteria Model 1 has least mean error and mean percentage error, model 3 has smallest mean absolute error and mean absolute percentage error, model 4 has the smallest mean squared error and root mean squared error and model 5 has the smallest theil's U statistics.

Table 5.1 Values of different forecast evaluation statistics of the models

Forecast evaluation statistics	Model-1	Model-2	Model-3	Model-4	Model-5
Mean Error	0.0020017	0.0023038	0.002045	0.0024631	0.0023524
Mean Squared Error	0.0099611	0.0096298	0.009767	0.0095889	0.0096709
Root Mean Squared Error	0.099805	0.098131	0.098832	0.097923	0.098341
Mean Absolute Error	0.067985	0.069696	0.067839	0.068184	0.069228
Mean Percentage Error	0.15441	0.15985	0.1559	0.16613	0.16486
Mean Absolute Percentage Error	1.2595	1.2811	1.2573	1.2616	1.2708
Theil's U	0.65021	0.6417	0.64754	0.64258	0.64052

5.3 Performance Comparison of the ARIMA Models based on forecasting accuracy

Here both the in sample an out of sample prediction was done. The purpose of forecasting within the sample is to test for the predictability power of the model which depends on the difference between the actual and forecasted value. If the magnitude of the difference between the forecasted and actual values is low then it can be said that the model has a good forecasting power. In this case Model-2 and Model-5 has shown best results as evident from the Table 5.2. One can observe from the figures that the forecast series of model 2 and model 5 are much closer to the actual series. As the predicted value closely follow/capture both past and future inflation trend, so it can be concluded from the findings that the prediction power of the two models are better and suitable for even twelve periods ahead forecasting. Five models have been proposed which are highly supported by ARIMA model selection criteria. As the predicted value closely follow/capture both past and future inflation trend, it can be concluded from the findings that the prediction power of the model 2 and model 5 is better and suitable for forecasting even upto twelve periods ahead.

Table 5.2 Inflation forecast by various ARIMA models

Months	Actual value	Forecasted byModel 1:ARIMA (12,24;2;1,11)12	Forecasted by Model2:SARIMA(0;2;1,11)(0;0;1)12	Forecasted by Model3: SARIMA (24;2;1,11)(1;0;0)12	Forecasted by Model 4:ARIMA (1;2;11,12)	Forecasted by Model5: SARIMA (11;2;1)(0;0;1)12
2011:12	10.71	10.81	10.76	10.81	10.76	10.76
2012:06	10.62	10.59	10.65	10.69	10.65	10.66
2012:12	8.74	8.73	8.68	8.69	8.67	8.68
2013:01	8.40	8.50	8.45	8.48	8.46	8.45
2013:02	8.19	8.36	8.26	8.31	8.27	8.26
2013:03	8.00	8.27	8.09	8.15	8.11	8.09
2013:04	7.85	8.19	7.96	8.02	7.96	7.94
2013:05	N/A	8.15	7.87	7.95	7.86	7.85
2013:06	N/A	8.14	7.82	7.85	7.78	7.79
2013:07	N/A	8.17	7.79	7.82	7.75	7.75
2013:08	N/A	8.24	7.82	7.84	7.75	7.76
2013:09	N/A	8.38	7.89	7.91	7.81	7.82
2013:10	N/A	8.49	7.97	8.00	7.87	7.88
2013:11	N/A	8.61	8.03	8.10	7.89	7.93
2013:12	N/A	8.68	8.05	8.16	7.85	7.92

5.4 The Best ARIMA Model

As the predicted value closely capture both past and future inflation trend, it can be concluded from the findings that the prediction power of the model 2 and model 5 is better and suitable for forecasting even upto twelve periods ahead. The best model is proposed among the five estimated models on the basis of model diagnostic checking, forecast evaluation and forecast accuracy as presented in Table 5.1 and Table 5.2. On the basis of in sample and out of sample forecast and forecast evaluation statistics two candid models which have sufficient predictive powers and the findings are well compared to the other models is proposed. In this case Seasonal ARIMA (0; 2; 1, 11) (0; 0; $1)_{12}$ and ARIMA (11; 2; 1) (0; 0; $1)_{12}$ outperforms the other ARIMA models.

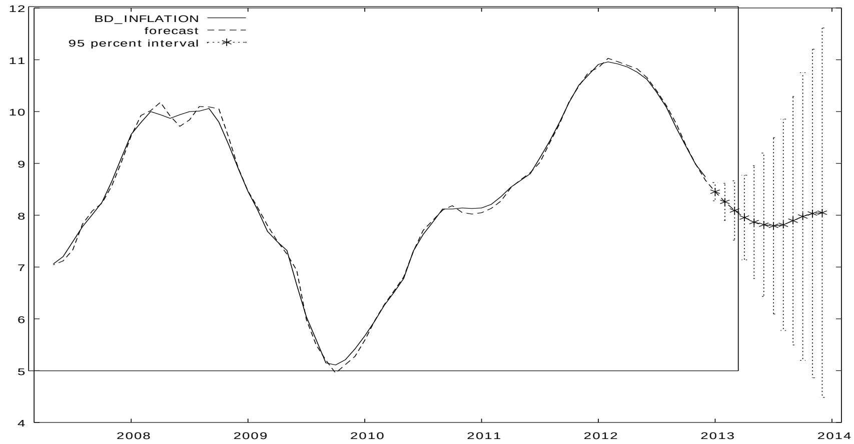


Figure 5.1 Actual Vs forecasted inflation by proposed SARIMA (0; 2; 1, 11) (0; 0; $1)_{12}$ model.

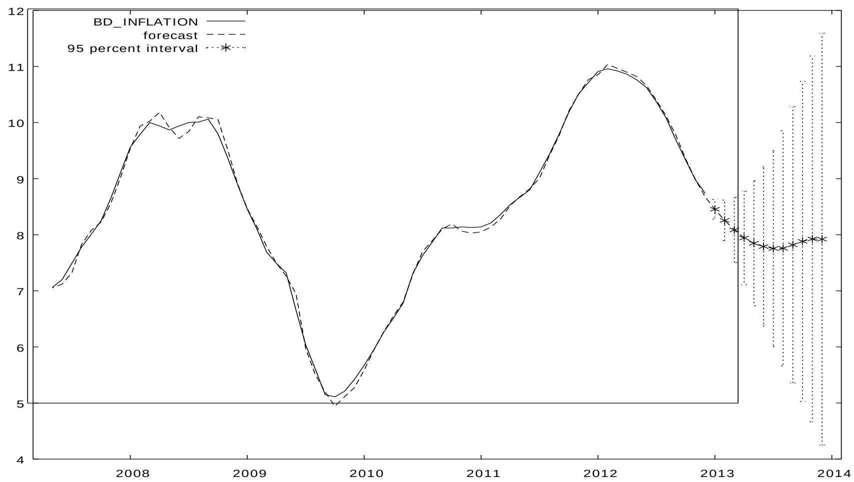


Figure 5.2 Actual Vs forecasted inflation by proposed SARIMA (11; 2;1) $(0;0;1)_{12}$ model.

5.5 Performance evaluation of NARX Network

The coefficient of determination (R^2) represents the precision of data which is significantly close to the fitted line. The value of R^2 varies between 0 and 1. If correlation coefficient, R=0.99951 then R^2=0.99, which means that 99% of the total variation in network prediction can be explained by the linear relationship between experimental values and network predicted values. The other 1% of the total variation in network prediction remains unexplained. The R^2 for different network topography is reported in Table 5.3. From Table 5.3, it is shown that the value of R^2 does not change significantly by increasing the number of neurons from 5 to 10. The network architecture consisting of 1 hidden layer and 5 hidden neurons, shows the best values of R^2 for both training and testing stages of the network.

Table 5.3 Performance evaluation of NARX Network

	Hidden layer	Hidden Neuron	R^2
Training performance	1	5	0.99922
Testing performance			0.99951
Training performance	1	8	0.99901
Testing performance			0.99878
Training performance	1	10	0.98901
Testing performance			0.97681
Training performance	1	10	0.9886
Testing performance			0.9865

Therefore, the network consisting of 5 hidden neurons was selected as the optimum one in this research work. The summary of the proposed network architecture has been presented here.

5.6 Performance Comparison Between ANN and ARIMA Based Models

In order to compare the out-of-sample forecast performance of ANN with ARIMA based models we find the out-of-sample forecast for November 2012 to May 2013 from both of these models based on data for July 2001 to October 2012. Results based on ANN methodology as well as ARIMA methodologies are presented in Table 5.4. Forecasting performance is evaluated on the basis of RMSE criteria. It is observed that RMSE of ANN based forecasts is less than the RMSE of forecasts based on ARIMA model. At least by this criterion forecast based on ANN are more precise.

Table 5.4 Performance Comparison Based on RMSE

Months	Actual	Forecast by ANN	Forecast by SARIMA $(0;2;1,11)(0;0;1)_{12}$	Forecasted by SARIMA $(11;2;1)(0;0;1)_{12}$
2012:11	8.98	8.97	10.76	10.76
2012:12	8.74	8.74	10.65	10.66
2013:01	8.40	8.38	8.68	8.68
2013:02	8.19	8.17	8.45	8.45
2013:03	8.00	8.00	8.26	8.26
2013:04	7.85	7.96	8.09	8.09
2013:05	N/A	7.79	7.96	7.94
RMSE		0.06	0.098131	0.098341

Now it is necessary to discuss why the ANN based model shows the better performance than ARIMA estimated models. The possible reasons are stated in the next page.

First, as opposed to the ARIMA, the traditional model-based methods, ANNs are data-driven self- adaptive methods in that there are few a priori assumptions about the models for problems under study. They learn from examples and capture subtle functional relationships among the data even if the underlying relationships are unknown or hard to describe. Thus ANNs are well suited for problems whose solutions require knowledge that is difficult to specify but for which there are enough data or observations. This modeling approach with the ability to learn from experience is very useful for the problem of solving the Bangladesh's inflation since it is often easier to have data than to have good theoretical guesses about the underlying laws governing the systems from which data are generated.

Second, ANNs can generalize. After learning the data presented to them (a sample), ANNs can often correctly infer the unseen part of a population even if the sample data contain noisy information. As forecasting is performed via prediction of future behavior (the unseen part) from examples of past behavior, it is an ideal application area for neural networks, at least in principle.

Third, ANNs are universal functional approximators. It has been shown that a network can approximate any continuous function to any desired accuracy. ANNs have more general and flexible functional forms which can effectively deal with than the traditional statistical methods like ARIMA. Any forecasting model assumes that there exists an underlying (known or unknown) relationship between the inputs (the past values of the time series and / or other relevant variables) and the outputs (the future values). Frequently, traditional statistical forecasting models have limitations in estimating this underlying function due to the complexity of the real system. ANNs can be a good alternative method to identify this function.

Finally, ANNs are nonlinear. Forecasting has long been the domain of linear statistics. The traditional approaches to time series prediction, such as the Box-Jenkins or ARIMA method, assume that the time series under study are generated from linear processes. Linear models have advantages in that they can be understood and analyzed in great detail, and they are easy to explain and implement. However, they may be totally inappropriate if the underlying mechanism is nonlinear. It is unreasonable to assume a priori that a particular realization of a given time series is generated by a linear process. In fact, real world systems are often nonlinear.

CHAPTER 6

CONCLUSIONS AND RECOMMENDATIONS

6.1 Conclusions

In this study, the most appropriate method for obtaining out-of-sample forecast for Bangladesh's monthly inflation series, using the different ARIMA forecasting models were found out. The out-of-sample forecast accuracies of five ARIMA models were assessed using accuracy measure statistics: MAE, RMSE, MAPE, MASE and Theil's U. Modeling and forecasting inflation using ARIMA models assumes a linear relationship between the inputs and the output. This approach has the disadvantage that the data analyzed often exhibit some nonlinearity that cannot be captured. This work employs neural networks to forecast the inflation. Using root mean square error (RMSE) as a measure of forecasting performance for the case Monthly inflation of Bangladesh, it is proven that neural networks are superior to ARIMA models. An ANN model has been developed for the multistep ahead forecasting of inflation as a function of previous inflation and exchange rate (which was considered as exogenous variable). The model was proved to be successful in terms of agreement with actual values for the inflation. The back-propagation learning algorithm was used for the development of feed-forward single hidden layer network. Tansigmoid function and purelin function were used as the transfer function in the hidden and output layer, respectively. Gradient descent learning was used and there was tapped delay line of two. Training of the network was performed using Lavenberg-marquardt backpropagation algorithm. The single layer feed forward network consisting of one input variable with an exogenous variable, 5 hidden neurons (tangent sigmoid neurons) and one output was found to be the optimum network for the model developed in this study. A good performance of the neural network was achieved with coefficient of determination (R^2) between the model prediction and actual values were 0.99. It can be concluded that ANN model performs accurately to forecast the inflation of Bangladesh.

6.2 Recommendations for Future Work

In this research the best forecasting method for Bangladesh's inflation was driven by the ANN which can have some limitations. ANNs are black-box methods. There is no explicit form to explain and analyze the relationship between inputs and outputs. Also no formal statistical testing methods can be used for ANNs. This causes difficulty in interpreting results from the networks. For overcoming this disadvantage, in future the researcher can combine the rule based modeling with the ANN.

ANNs are prone to have over fitting problems due to their typical, large parameter set to be estimated. There are no structured methods today to identify what network structure can best approximate the function, mapping the inputs to outputs. Hence, the tedious experiments and trial-and-error procedures are often used. Since they are data-driven and model-free, ANNs are quite general but can suffer high variance in the estimation, that is, they may be too dependent on the particular samples observed. Because of this the prediction of the long term future inflation was not possible.

In future, the researchers can also study as to how the anticipated or unanticipated inflation is affecting the other macroeconomic variables (like interest rate, exchange rate, money supply, GDP, unemployment rate etc.) in Bangladesh where the importance of the present study lies.

REFERENCES

Adya, M. and Collopy, F. How effective are neural networks at forecasting and prediction? A review and evaluation. Journal of Forecasting, Vol.17, pp. 481–495. 1998

Binner, J.M., Gazely, A.M. and Chen, S. Financial innovation and divisia money in Taiwan: A neural network approach. The European Journal of Finance, Vol. 8, pp. 238–247. 2002

Binner, J.M., Bissoondeeal, R.K., Thomas, E., Gazely, A.M. and Mullineux, A.W. A comparison of linear forecasting models and neural networks: An application to Euro inflation and Euro Divisia. Applied Economics, Vol. 37(6), pp. 665–680. 2005

Box, G. and G. Jenkins, "Time Series Analysis: Forecasting and Control", Holden-Day, San Francisco. 1976

CC Klimasauskas. "Applying neural networks. Neural Networks in Finance and Investing", Chapter 3, pp.47-72, Probus Publishing Company. 1993

Christiano, L J, " P* Not the inflation forecaster's holy grail,"Federal Reserve Bank of Minneapolis Quarterly Review, pp. 3-18. 1989

Choudhary, M. A. and A. Haider, Neural Network models for Inflation Forecasting: An Appraisal. Discussion Paper No. 0808,Department of Economics, University of Surrey, UK. 2008

Davis RG. "Inflation: Measurement And Policy Issues". Federal *Reserve Bank of New York Quarterly Review*. Vol. 16. pp. 13-24. 1991

Fischer S and Modigliani F. "Towards an Understanding of the Real Effects and Costs of Inflation". *Weltwirtschaftliches Archiv*e. Vol. 114. pp. 810-833.1978

Forsells M and Kenny G. The Rationality of Consumer Inflation Expectations: Survey-Based Evidence for the Euro Area. Working Paper No. 163. European Central Bank. 2002

Francisco Nadal-De Simone,"Forecasting Inflation in Chile Using State-Space and Regime-Switching Models". WP/01/82 – IMF. 2000

Fadalla, L. Chen-Hua, "An Analysis of the Applications of Neural Networks in Finance", Interfaces – Vol. 31, Issue 4 ISSN:0092-2102. July 2001

Gomez J. Wage Indexation, Inflation Inertia and the Cost of Disinflation. Banco de la República. 2002

Granger, C.W.J. and P. Newbold , "Forecasting Economic Time Series", Academic Press:New York.1976

Greg Tkacz and Sarah Hu, Forecasting GDP growth using Artificial Neural Networks. Working Paper 99-3, Bank of Canada. 1999

Gazely, A.M. and Binner, J.M. Better forecasts of future inflation? The neural network approach. Working Paper, department of finance and business information systems. Nottingham Business School. 1998

G. Cybenko. "Approximation by superpositions of a sigmoidal function". Mathematics of Control, Signals, and Systems (MCSS) – Springer. 1989

Jan G. De Gooijer, Rob J. Hyndman, "25 years of time series forecasting" , *International Journal of Forecasting* Volume: 22, pp. 443 – 473. 2006

Jae J. Lee, "Forecasting Time Series under Box and Jenkins' ARIMA Model Using Bayesian Inference" The Journal of American Academy of Business, Cambridge. Vol. 16 Num. 2 . 2011

Kanchan Datta, "ARIMA Forecasting of Inflationin the Bangladesh Economy" The IUP Journal of Bank Management, Vol. X, No. 4,pp.7-15, 2011

K. Hornik. “Approximation capabilities of multilayer feedforward networks”, Neural Networks - portal.acm.org. 1991

K. Hornik, Stinchcombe et al. “Multilayer feedforward networks are universal approximators”, ACM. 1989

Meyler, A, G. Kenny and T. Quinn "Forecasting Irish Inflation using ARIMA models", Central Bank of Ireland Technical Paper 3/RT/98, December. 1998

Marek Hlavacek, Michael Konak and Josef Cada, The application of structured feedforward neural networks to the modeling of daily series of currency in circulation. Working paper series 11, Czech National Bank. 2005

M. Mehta, “Foreign Exchange Markets”, Neural Networks in the Capital Markets, John Wiley, Chichester, pp. 176-198. 1995.

Nakamura, Emi, Inflation forecasting using a neural network Economics Letters, Vol. 86(3), pp. 373-378. 2006

Nag, A.K. and Mitra, A. Forecasting daily foreign exchange rates using genetically optimized neural networks. Journal of Forecasting, Vol.21, pp. 501–511. 2002

Peter Kennedy, “A guide to econometrics”, MIT Press. 1985

Sims, Christopher A., “Macroeconomics and Reality,” *Econometrica*, Vol. 48, pp. 1-48. 1980

Svensson, L. E. O. “Inflation Forecast Targeting: Implementing and Monitoring Inflation Targets”, CEPR Discussion Paper 1511. 1996

Serju Prudence, “Monetary Conditions & Core Inflation: An Application of Neural Networks”, Working Paper, Bank Of Jamaica. 2002

Shazly, M.R.E. and Shazly, H.E.E. Forecasting currency prices using genetically evolved neural network architecture. International Review of Financial Analysis, Vol.8, pp. 67–82. 1999

Toshitaka Sekine, Modeling and Forecasting Inflation in Japan. WP/01/82 – IMF. 2001

Tkacz, G. Neural network forecasting of canadian gdp growth. International Journal of Forecasting, Vol. 17, pp. 57–69. 2001

Yule, G. U. (1927). "On a Method of Investigating Periodicities in Disturbed Series, with Special Reference to Wolfer's Sunspot Numbers". Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, Vol. 226 pp. 636–646. 1927

Zarnowitz, V. and P. Braun, "Twenty-two Years of the NBER-ASA Quarterly Economic Outlook Surveys: Aspects and Comparisons of Forecasting Performance," in J.H. Stock and M.W. Watson (eds) Business Cycles, Indicators, and Forecasting, NBER Studies in Business Cycles, Vol. 28, University of Chicago Press: Chicago. 1993