A COMPARATIVE STUDY ON SELECTED MACRO VARIABLES OF GDP USING EXPLORATORY ANALYSIS, AND EVALUATING DIFFERENT TIME SERIES MODEL

ACKNOWLEDGEMENT

This year has been an extremely informative journey for, my friend and me. We would like to extend our gratitude to **Proff. Jyoti M. Divecha (Head of Department)** for entrusting upon me these invaluable projects. The journey of the study at the department and the projects gave us immense insight into the world of analytics we are very thankful to **Ms. Rupal C. Rabari** our internal project guide for their incomparable affection during my projects works. Documentation is heart of project, so we take opportunity to express our heartfelt thanks to all my dear friends who support and encourage my project partner and me to complete our documentation successfully. These projects have been the outcome of ideas of combination of ideas suggestions and contribution of many people.

We express our gratitude to **Mr. Agniva Das, Dr. Dharmesh Raykundaliya** for their immense support and timely help and for their incomparable affection during our project work.

My project is dedicated to all the people whom we met, took guidance interviewed and something from them. At this occasion, we want to grab this opportunity to acknowledge our sincere thanks to all of them while submitting.

Mr. Akshay Vanjare. (Roll No. 02)

Mr. Lalit Alone. (Roll No. 18)

Place: Anand

Date:

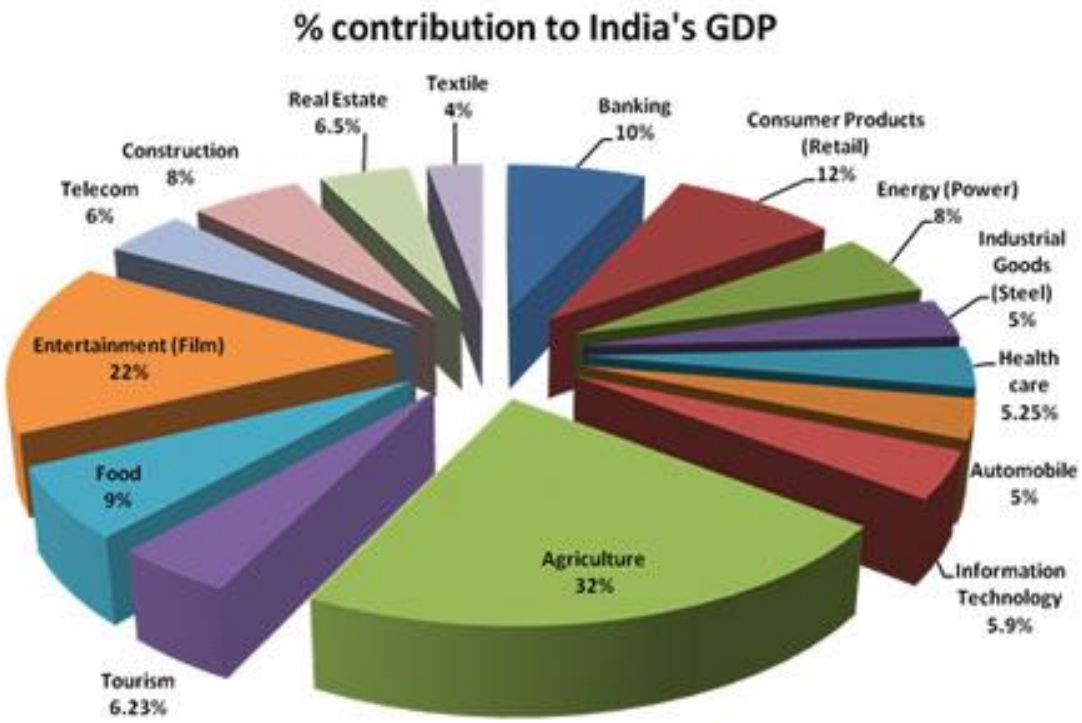


Fig. % Contribution to India's GDP

PROJECT REPORT

(PS04CSTA24)

ON

A COMPARATIVE STUDY ON SELECTED MACRO VARIABLES OF GDP USING EXPLORATORY ANALYSIS, AND EVALUATING DIFFERENT TIME SERIES MODEL

BY

AKSHAY N. VANJARE

LALIT R. ALONE

PROJECT GUIDE

MS. RUPAL C. RABARI

MR. AGNIVA DAS

"MASTERS OF APPLIED STATISTICS"

DEPARTMENT OF STATISTICS

SARDAR PATEL UNIVERSITY

VALLABH VIDYANAGAR

2019 – 2020

CERTIFICATE

This is to certify that Mr. Akshay Nivrutti Vanjare, Exam No. 02, "Master of Science in Applied Statistics", Semester-IV has successfully completed his project work on **"A COMPARATIVE STUDY ON SELECTED MACRO VARIABLES OF GDP USING EXPLORATORY ANALYSIS, AND EVALUATING DIFFERENT TIME SERIES MODEL"** for PS04CSTA24 in term 2019-2020.

Place: Vallabh Vidhyanagar

Date:

Project Guide

(Ms. Rupal C. Rabari)

Head of Department

(Proff. Jyoti M. Divecha)

CERTIFICATE

This is to certify that Mr. Lalit Ramesh Alone, Exam No. 18, "Master of Science in Applied Statistics", Semester-IV has successfully completed his project work on **"A COMPARATIVE STUDY ON SELECTED MACRO VARIABLES OF GDP USING EXPLORATORY ANALYSIS, AND EVALUATING DIFFERENT TIME SERIES MODEL"** for PS04CSTA24 in term 2019-2020.

Place: Vallabh Vidhyanagar

Date:

Project Guide

(Ms. Rupal C. Rabari)

Head of Department

(Proff Jyoti M. Divecha)

INDEX

Sr. No	Contents	Page No.
1	Abstract …….……	8
2	Objective and Hypothesis of the study .……	9
3	Introduction ……...	10
	3.1 What is GDP? ……	10
	3.2 History of GDP ……..	12
	3.3 Source for GDP Data …….	13
4	Theory ……..…….....	14
	4.1 Descriptive Statistics ……..	14
	4.2 Statement of Theory or Hypothesis ……...	15
	4.3 Classical Linear regression ……	18
	4.4 Autocorrelation ……..	21
	4.5 Heteroscedasticity ……..	23
	4.6 Multicollinearity ……	26
	4.7 Principal Component Analysis …….	27
	4.8 ARIMA Model ……..	28
5	Methodology ……...	34
6	Data …….	35
7	Analysis and Result …….....	38
8	Discussion and Interpretation …….	43
9	Conclusion ……..	45
10	References ……..	46
11	Appendix ……	47
	11.1 Coding and Output …….	47

ABSTRACT

Financial Architecture aims sustainability of an economy by ensuring consistent growth rate. GDP is an indicator of the growth of an economy. Higher GDP of an economy reflects robust growth of an economy and vice-versa and as such every country tries to maximize the growth rate of GDP. There are certain macro factors operating in the economic environment that will influence the GDP growth rate. The study makes an attempt to determine the influence of selected economic variables namely Agriculture, Mining & Quarrying, Manufacturing, Electricity Gas & Water Supply, Construction, Trade, Financial Real Estate & Personal Services, Public Administration, Gross National Income, Net National Income, Per Capita Income, Private Final Consumption Expenditure, Government Final Consumption Expenditure, Changes in Stocks, Valuables, Export, Less Import.

The data is collected by using secondary sources relating to the selected Economic variables. The data is collected for a period from 1950-51 to 2018-19 with annual intervals. The scope of the study is confined only to selected economic variables. Correlation and ANOVA are used for analyzing the relationship between the GDP and selected economic variables. The study revealed that Exchange rate, Sensex and Balance of Payment reflected by current and capital account balances are the factors that significantly predict GDP of the economy. Financial architecture broadly refers to the framework and series of measures that are considered necessary to prevent future economic crises and help manage these crises when they occur. It refers to the structures, practices and rules which are designed in order to overcome the influence of crisis on the economy.

OBJECTIVES & HYPOTHESIS OF THE STUDY

Objectives:

The main objectives of the study are:

1. To identify the relationship between selected economic variables and GDP of Indian Economy.
2. To analyze the impact of selected economic variables on GDP of Indian Economy
3. To briefly overview the trends in Indian GDP and its related sector such as Agriculture, Service, Manufacturing, Export, Import etc. in India and study of change in contribution of different sector in Indian GDP and estimate it for future value.

Hypothesis:

1. H0: Null Hypothesis – There is no significant relationship between GDP and selected economic variables of Indian Economy.
   H1: Alternate Hypothesis – There is a significant relationship between GDP and selected economic variables of Indian Economy.
2. H0: Null Hypothesis – GDP is independent of economic variables of Indian economy.
   H1: Alternate Hypothesis – GDP is Independent on economic variables.

INTRODUCTION

What Is GDP?

Gross Domestic Product (GDP) is the total monetary or market value of all the finished goods and services produced within a country's borders in a specific time period. As a broad measure of overall domestic production, it functions as a comprehensive scorecard of the country's economic health.

Though GDP is usually calculated on an annual basis, it can be calculated on a quarterly basis as well. In the United States, for example, the government releases an annualized GDP estimate for each quarter and also for an entire year. Most of the individual data sets will also be given in real terms, meaning that the data is adjusted for price changes, and is, therefore, net of inflation.

KEY TAKEAWAYS

- Gross Domestic Product (GDP) is the monetary value of all finished goods and services made within a country during a specific period.
- GDP provides an economic snapshot of a country, used to estimate the size of an economy and growth rate.
- GDP can be calculated in three ways, using expenditures, production, or incomes. It can be adjusted for inflation and population to provide deeper insights.
- Though it has limitations, GDP is a key tool to guide policymakers, investors, and businesses in strategic decision making.

The Basics of GDP

GDP includes all private and public consumption, government outlays, investments, additions to private inventories, paid-in construction costs, and the foreign balance of trade (exports are added, imports are subtracted).

There are several types of GDP measurements:

- ***Nominal GDP*** is the measurement of the raw data.
- ***Real GDP*** takes into account the impact of inflation and allows comparisons of economic output from one year to the next and other comparisons over periods of time.
- ***GDP growth rate*** is the increase in GDP from quarter to quarter.
- ***GDP per capita*** measures GDP per person in the national populace; it is a useful way to compare GDP data between various countries.

The balance of trade is one of the key components of a country's (GDP) formula. GDP increases when the total value of goods and services that domestic producers sell to foreigners exceeds the total value of foreign goods and services that domestic consumers buy, otherwise known as a trade surplus. If domestic consumers spend more on foreign products than domestic producers sell to foreign consumers - a trade deficit- then GDP decreases.

Calculating GDP

GDP can be determined via three primary methods. All, when correctly calculated, should yield the same figure. These three approaches are often termed the expenditure approach, the output (or production) approach, and the income approach.

GDP Formula Based on Spending

The expenditure approach, also known as spending approach, calculates the spending by the different groups that participate in the economy. This approach can be calculated using the following formula:

$$\mathbf{GDP = C + G + I + (E - I)}$$

Or (consumption + government spending + investment + net exports). All these activities contribute to the GDP of a country. The U.S. GDP is primarily measured based on the expenditure approach.

The **C** is private consumption expenditures or consumer spending. Consumers spend money to buy consumption goods and services, such as groceries and haircuts. Consumer spending is the biggest component of GDP. Consumer confidence, therefore, has a very significant bearing

on economic growth. A high confidence level indicates that consumers are willing to spend, while a low confidence level reflects uncertainty about the future and an unwillingness to spend.

The **G** represents government consumption expenditure and gross investment. Governments spend money on equipment, infrastructure, and payroll. Government spending assumes particular importance as a component of GDP when consumer spending and business investment both decline sharply, as, for instance, after a recession.

The **I** is for private domestic investment or capital expenditures. Businesses spend money to invest in their business activities (buying machinery, for instance). Business investment is a critical component of GDP since it increases productive capacity and boosts employment.

(E-I) is net exports, calculated as total exports minus total imports (**E-I = Exports - Imports**). Goods and services that an economy makes that are exported to other countries, less the imports that are brought in, are net exports. A current account surplus boosts a nation's GDP, while a chronic deficit is a drag on GDP. All expenditures by companies located in the country, even if they are foreign companies, are included in the calculation.

History of GDP

GDP first came to light 1937 in a report to the U.S. Congress in response to the Great Depression, conceived of and presented by an economist at the National Bureau of Economic Research, Simon Kuznets. At the time, the preeminent system of measurement was GNP. After the Bretton Woods conference in 1944, GDP was widely adopted as the standard means for measuring national economies, though ironically the U.S. continued to use GNP as its official measure of economic welfare until 1991, after which it switched to GDP.

Beginning in the 1950s, however, some economists and policymakers began to question GDP. Some observed, for example, a tendency to accept GDP as an absolute indicator of a nation's failure or success, despite its failure to account for health, happiness, (in) equality and other constituent factors of public welfare. In other words, these critics drew attention to a distinction between economic progress and social progress. However, most authorities, like Arthur Okun, an economist for President Kennedy's Council of Economic Advisers, held

firm to the belief that GDP is as an absolute indicator of economic success, claiming that for every increase in GDP there would be a corresponding drop in unemployment.

Sources for GDP Data

The World Bank hosts one of the most reliable web-based databases. It has one of the best and most comprehensive lists of countries for which it tracks GDP data. The International Money Fund (IMF) also provides GDP data through its multiple databases, such as World Economic Outlook and International Financial Statistics.

Another highly reliable source of GDP data is the Organization for Economic Cooperation and Development (OECD). The OECD provides not only historical data but also forecasts for GDP growth. The disadvantage of using the OECD database is that it tracks only OECD member countries and a few nonmember countries.

In the India, the Federal Reserve collects data from multiple sources, including a country's statistical agencies and the World Bank. The only drawback to using a Federal Reserve database is a lack of updating in GDP data and an absence of data for certain countries.

The Bureau of Economic Analysis (BEA), a division of the Indian Department of Commerce, issues its own analysis document with each GDP release, which is a great investor tool for analyzing figures and trends and reading highlights of the very lengthy full release.

THEORY

Descriptive statistics

In this step, we are checking some descriptive statistics of our *GDP* dataset. We are calculating below statistics:

Measure of Central Tendency

A measure of central tendency is a number used to represent the centre or middle of a set of data values. In other words, the measures of central tendency describe a distribution in terms of its most "frequent", "typical" or "average" data value. But there are different ways of representing or expressing the idea of "typicality".

Arithmetic Mean

For a given set of observations, Arithmetic Mean is defined as the sum of all the observations divided by the number of observations. Thus, if a variable x assumes n values $x1$, $x2,x3,\ldots,xn$ then **AM** of x, to be denoted by $\hat{x}$, given by,

$$\hat{x} = \frac{x1,x2,x3,\ldots\ldots xn}{n} = \frac{1}{n}\sum_{i=0}^{n} xi$$

Median

The median can be defined as that point in a distribution above which and below which lie 50% of all the cases or observations in the distribution.

Measure of Dispersion

Measure of dispersion is defined as lack of uniformity in the sizes or quantities of the items of a group or series.

Range

Range is the difference between the smallest value and the largest value of a series.
It is calculated by below formula:

Range = Maximum –Minimum

Standard Deviation and Variance

Standard deviation is calculated as the square root of average of squared deviations taken from actual mean. It is also called root mean square deviation. The square of standard deviation i.e., σ^2 is called 'variance'. Calculation of standard deviation in case of raw data
Formula for Standard Deviations is as below:

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{n}(x_i - \mu)^2}$$

Standard Error

Standard error is defined as standard deviation of sample statistics. Formula for Standard error is as below:

$$\text{SE} = \frac{s}{\sqrt{n}}$$

Statement of Theory or Hypothesis

Keynes stated:

Economic growth can be defined as the increase in the inflation-adjusted market value of the goods and services produced by an economy over time. It is conventionally measured as the percent rate of increase in real gross domestic product, or real GDP

Specification of the Econometric Model of GDP

The purely mathematical model of the GDP function given in Eq. (1) is of limited interest to the econometrician, for it assumes that there is an *exact* or *deterministic* relationship between GPD and economic variables. But relationships between economic variables are generally inexact. For example, Thus, if we were to obtain data on consumption expenditure and disposable (i.e., after-tax) income of a sample of, say, 500 Indian families and plot these data on a graph paper with consumption expenditure on the vertical axis and disposable income on the

horizontal axis, we would not expect all 500 observations to lie exactly on the straight line of Eq. (1) because, in addition to income, other variables affect consumption expenditure. For example, size of family, ages of the members in the family, family religion, etc., are likely to exert some influence on consumption. To allow for the inexact relationships between economic variables, the econometrician would modify the deterministic consumption function in Eq. (1) as follows:

$$Y = \beta 1 + \beta 2X + u \ (1)$$

Where *u*, known as the disturbance**,** or error**,** term**,** is a random(stochastic) variable that has well-defined probabilistic properties.

Obtaining Data

To estimate the econometric model given equation (1.1) that is, to obtain the numerical values of β's, we need data. Although we will have more to say about the crucial importance of data for economic analysis in the project, for now let us look at the data following table, which relate to the Indian economy for the period 1951–2018. The *Y* variable in this table is the gross domestic product its mean GDP and the *X* variable is Agriculture, forestry & fishing, Manufacturing, Electricity, gas, water supply & other, Construction, Trade, hotels, transport, communication, Financial , real estate & prof services, Public Administration, Gross National Income, Net National Income, Per capita income, Private final consumption expenditure, Government final consumption expenditure, Changes in stocks ,Exports of goods and services, Less Imports of goods and services. a measure of GDP in crore rupees and all variables are also measure in rupees accept electricity (electricity measure in Watt). Therefore, the data are in "real" terms; that is, they are measured in constant prices. The data are plotted in Figure (1).For the time being neglect, the line drawn in the figure.

$$\text{GDP (Y)} = \beta_1 + X_2\beta_2 + X_3\beta_3 + \ldots\ldots\ldots\ldots + X_{17} + \mu \quad \ldots\ldots (1.1)$$

The above equation (1.1) in variables dependent and independent variables X2, X3… X17 are labelled below table no. (1) The above model is also known as Classical Linear Regression Model. With respective slope coefficient β's and intercept β_1.

Table No: 8.1 Describe variables

Y	**GDP**
X2	Agriculture
X3	Mining & quarrying
X4	Manufacturing
X5	Electricity Gas & Water Supply
X6	Construction
X7	Trade
X8	Financial Real Estate & Personal Services
X9	Public Administration
X10	Gross National Income
X11	Net National Income
X12	Per Capita Income
X13	Private Final Consumption Expenditure
X14	Government Final Consumption Expenditure
X15	Changes in Stocks
X16	Export
X17	Less Import.
X12	Per Capita Income
X13	Private Final Consumption Expenditure
X14	Government Final Consumption Expenditure
X15	Changes in Stocks
X16	Export
X17	Less Import.

Estimation of the Econometric Model

Now that we have the data, our next task is to estimate the parameters of the GDP function. The numerical estimates of the parameters give empirical content to the GDP function. For now, note that the statistical technique of regression analysis is the main tool used to obtain the estimates. Using this technique and the data given in above table, we obtain the following estimates of β's. By using regression analysis.

Theoretical econometrics is concerned with the development of appropriate methods for measuring economic relationships specified by econometric models. In this aspect, econometrics leans heavily on mathematical statistics. For example, one of the methods used extensively in this book is least squares. Theoretical econometrics must spell out the assumptions of this method, its properties, and what happens to these properties when one or more of the assumptions of the method are not fulfilled.

The modern interpretation of regression is, however, quite different. Broadly speaking, we may say Regression analysis is concerned with the study of the dependence of one variable, the dependent variable, on one or more other variables, the explanatory variables, with a view to estimating and/or predicting the (population) mean or average value of the former in terms of the known or fixed (in repeated sampling) values.

Classical Linear Regression Model:

The Assumptions Underlying the Method of Least Squares, If our objective is to estimate β's only, the method of OLS is to estimate the parameter of regressors. But in regression analysis our objective is not only to obtain$\hat{\beta}$'s but also to draw inferences about the true β*'s.* For example, we would like to know how close $\hat{\beta}$*'s* are to their counterparts in the population or how close $\hat{Y}i$ is to the true $E(Y \mid Xi)$. To that end, we must not only specify the functional form of the model, as in Eq. (1.1), but also make certain assumptions about the manner in which *Yi* are generated. To see why this requirement is needed, look at the PRF: $Yi = \beta 1 + \beta iXi + ui$. It shows that *Yi* depends on both *Xi* and *ui*. Therefore, unless we are specific about how *Xi* and *ui* are created or generated, there is no way we can make any statistical inference about the *Yi* and also, as we shall see, about

β's. Thus, the assumptions made about the *Xi* variable(s) and the error terms are extremely critical to the valid interpretation of the regression estimates.

The Gaussian, Standard, or Classical Linear Regression model (CLRM), which is the cornerstone of most econometric theory, makes 7 assumptions. We first discuss these assumptions in the context of the more than two-variable regression model; and then we extend them to multiple regression models, that is, models in which there is more than one regressors.

Classical Assumptions:

1. Regression linear in parameters.
2. Error term has zero population mean
3. Error term not correlated with x's
4. No serial correlation
5. No heteroscedasticity
6. No perfect multicollinearity
7. Error usually normally distributed

<u>In details</u>

1. Assumption 1.

   Linear Regression Model: The regression model is linear in the parameters, though it may or may not be linear in the variables. That is the regression model as shown in Eq. (1.1):

$$Yi = \beta 1 + \beta i\, Xi + u \qquad\qquad i=2\ldots\ldots\ldots 17$$

2. Assumption 2.

   Fixed X Values or X Values Independent of the Error Term: Values taken by the regressors X may be considered fixed in repeated samples (the case of fixed regressors) or they may be sampled along with the dependent variable Y (the case of stochastic

regressors). In the latter case, it is assumed that the X variable(s) and the error term are independent, that is, cov (Xi, ui) = 0.

3. Assumption 3.

   Zero Mean Value of Disturbance ui: Given the value of Xi, the mean, or expected, value of the random disturbance term ui is zero. Symbolically, we have E(ui |Xi) = 0 Or, if X is nonstochastic, E(ui) = 0

4. Assumption 4.

   Homoscedasticity or Constant Variance of ui: The variance of the error, or disturbance, term is the same regardless of the value of X. Symbolically,

$$\begin{aligned} Var\ (ui) &= E\ [ui - E\ (ui\ |Xi)]\ 2 \\ &= E\ (u2i|Xi), && \text{because of Assumption 3} \\ &= E\ (u2i), && \text{if Xi are nonstochastic} \\ &= \sigma 2 \end{aligned}$$

   Where var stands for variance.

5. Assumption 5.

   No Autocorrelation between the Disturbances: Given any two X values, Xi and X_j (i ≠j), the correlation between any two ui and u_j (i ≠ j) is zero. In short, the observations are sampled independently. Symbolically,

$$Cov\ (ui, uj\ |Xi, Xj) = 0$$

$$Cov\ (ui, uj) = 0, \text{ if X is nonstochastic.}$$

   Where i and j are two different observations and where cov means covariance.

   Fig. (8.1) Patterns of correlation among the

   1) Positive serial Correlation
   2) Negative serial Correlation
   3) Zero Correlation

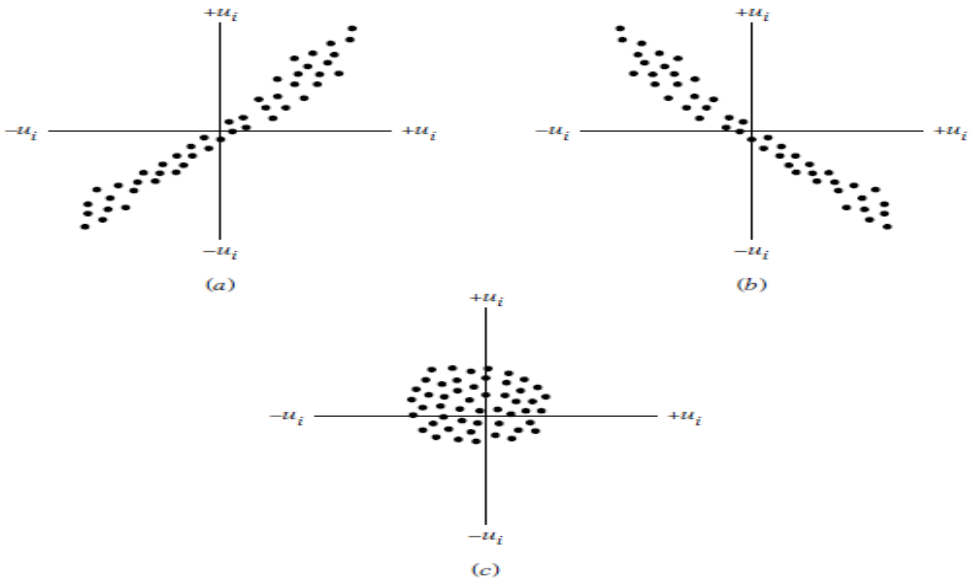


Fig.8.1 Patterns of correlation

6. Assumption 6.

   The Number of Observations n must be Greater than the Number of Parameters to Be Estimated: Alternatively, the number of observations must be greater than the number of explanatory variables.

7. Assumption 7.

   The Nature of X Variables: The X values in a given sample must not all be the same. Technically, var (X) must be a positive number. Furthermore, there can be no outliers in the values of the X variable, that is, values that are very large in relation to the rest of the observations.

Autocorrelation

One of the basic assumptions in linear regression model is that the random error components or disturbances are identically and independently distributed. So in the model it is assumed that

$$Y = X\beta + \mu$$

$$E(\mu_t \mu_{t-s}) = \begin{cases} \sigma_i^2 & if\ s=0 \\ 0 & if\ s\neq 0 \end{cases}$$

i.e., the correlation between the successive disturbances is zero.

In this assumption, when E (μ_t, μ_{t-s}) =σ_μ^2 , s=0 is violated, i.e., the variance of disturbance term does not remain constant, then problem of heteroscedasticity arises. When E(μ_t μ_{t-s}) = σ_μ^2, s≠0 is violated, i.e., the variance of disturbance term remains constant though the successive disturbance terms are correlated, then such problem is termed as problem of autocorrelation.

When autocorrelation is present, some or all off diagonal elements in E (μμ') are nonzero. Sometimes the study and explanatory variables have a natural sequence order over time, i.e., the data is collected with respect to time. Such data is termed as time series data**.** The disturbance terms in time series data are serially correlated.

The auto covariance at lag is defined as $\gamma_{s=}$ E (μ_t μ_{t-s}) s=0, $\pm1, \pm2$

At zero lag, we have constant variance, i.e. $\gamma_{s=}$ E (μ_t^2) = σ^2

The autocorrelation coefficient at lag s is defined as:

$$P_s = \frac{E(\mu t\ \mu t-s)}{\sqrt{var\mu_t}\sqrt{var\mu_{t-s}}} = \frac{\gamma_s}{\gamma_0};\ s = 0, \pm1, \pm2$$

Assume ρ and γ_s are symmetrical in, i.e., these coefficients are constant over time and depend only on length of lag s. The autocorrelation between the successive terms $(\mu_{2\ and}\ \mu_1)$, $(\mu_{3\ and}\ \mu_2)$, $(\mu_{n\ and}\ \mu_{n-1})$ gives the autocorrelation of order one, i.e. ρ_1 similarly, the autocorrelation between the successive terms $(\mu_{3\ and}\ \mu_1)$, $(\mu_{4\ and}\ \mu_2)$, $(\mu_{n\ and}\ \mu_{n-2})$ gives the autocorrelation of order two ρ_2

Source of Autocorrelation

Some of the possible reasons for the introduction of autocorrelation in the data are as follows:

1. Carryover of effect, at least in part, is an important source of autocorrelation. For example, the monthly data on expenditure on household is influenced by the expenditure of preceding month. The autocorrelation is present in cross-section data as well as time series data. In the cross-section data, the neighboring units tend to be similar with respect to the characteristic under

study. In time series data, the time is the factor that produces autocorrelation. Whenever some ordering of sampling units is present, the autocorrelation may arise.

2. Another source of autocorrelation is the effect of deletion of some variables. In regression modelling, it is not possible to include all the variables in the model. There can be various reasons for this, e.g., some variable may be qualitative, sometimes direct observations may not be available on the variable etc. The joint effect of such deleted variables gives rise to autocorrelation in the data.

3. The misspecification of the form of relationship can also introduce autocorrelation in the data. It is assumed that the form of relationship between study and explanatory variables is linear. If there are log or exponential terms present in the model so that the linearity of the model is questionable then this also gives rise to autocorrelation in the data.

4. The difference between the observed and true values of variable is called measurement error or errors–in-variable. The presence of measurement errors on the dependent variable may also introduce the autocorrelation in the data.

Heteroscedasticity

In the multiple regression model $Y= X\beta + \in$. It is assumed that $V(\in) =\sigma_i^2$ and $V(\in^2) =\sigma^2$; $Cov(\in_i \in_j) =\sigma_i^2$ In this case, the diagonal elements of covariance matrix of $\in$ are same indicating that the variance of each $\in_i$ is same and off-diagonal elements of covariance matrix of $\in$ are zero indicating that all disturbances are pair wise uncorrelated. This property of constancy of variance is termed as homoscedasticity and disturbances are called as homoscedasticity disturbances. In many situations, this assumption may not be plausible and the variances may not remain same. The disturbances whose variances are not constant across the observations are called heteroscedastic disturbance and this property is termed as heteroscedasticity. In this case

$$V(\in^2) =\sigma^2 , \qquad i=1,2\ldots n.$$

Examples: Suppose in a simple linear regression model, x denote the income and y denotes the expenditure on food. It is observed that as the income increases, the variation in expenditure on food increases because the choice and varieties in food increase, in general, up to certain extent. So the variance of observations on y. and so the variances of disturbances will not remain constant. In general, it will be increasing as income increases.

In another example, suppose in a simple linear regression model, x denotes the number of hours of practice for typing and y denotes the number of typing errors per page. It is expected that the number of typing mistakes per page decreases as the person practices more. The homoscedastic disturbances assumption implies that the number of errors per page will remain same irrespective of the number of hours of typing practice which may not be true is practice.

Possible reasons for Heteroscedasticity:

There are various reasons due to which the heteroscedasticity is introduced in the data. Some of them are as follows:

1. The nature of phenomenon under study may have an increasing or decreasing trend. For example, the variation in consumption pattern on food increases as income increases, similarly the number of typing mistakes decreases as the number of hours of typing practice increases.
2. The skewness in the distribution of one or more explanatory variables in the model also causes heteroscedasticity in the model.
3. The incorrect data transformations and incorrect functional form of the model can also give rise to the heteroscedasticity problem.

Test of Heteroscedasticity

1. Goldfield & Quant Test

The popular method is applicable if one assumes that σ_i^2 is positively related to one of the explanatory variables in the regression model. Consider two variables in the regression model.

$$Y_i = \beta_1 + \beta_2 X_i + \mu_i;\ i=1,2,\ldots,n$$

Suppose $\sigma_i^2 \alpha\ X_i^2$ i.e. $\sigma_i^2 = \sigma_i^2 X_i^2$

If equation is appropriate it could mean that σ_i^2 would be larger, larger the value of Xi. If that turns out to be test heteroscedasticity is, most likely to present in the model to test this goldfield & Quant have suggested following step

1) Order the observations according to the values of x_i
2) Beginning with the lowest value of x_i In short arrange data in ascending order Omit 'C' central observation where c is specified apriority & divide the remaining (n-1) observation in two group, each of (n-c)/2 observation.
3) Fit separate OLS regression to 1st $\frac{n-c}{2}$ observation && obtain RSS_1 and RSS_2 respectively. These RSS_1 each has ($\frac{n-c}{2}$ - k) df.
4) Compute the ratio: $F = \frac{RSS1/_{dF}}{RSS2/_{DF}} = \frac{RSS2}{RSS1} \sim F(\frac{n-c}{2} - k, \frac{n-c}{2} - k)$
5) If F is greater than

   $F > F(\frac{n-c}{2} - k, \frac{n-c}{2} - k, \alpha)$

   We reject the null hypothesis of homoscedasticity.

2. Breusch Pagan Test

The Gold and Quant test is based on the assumption of heteroscedasticity, is related to a single identifiable variable, which is responsible of heteroscedasticity. We use it for ordering of the data. However, the heteroscedasticity is related several variables that do not move all together. Then, it is not possible to achieve unique order. Breusch & Pagan (1979) have devised a test which does not depend on ordering of data. The test is applicable using the langrangian multiplier principle.

Let's us assume that our model for heteroscedasticity is σ_i^2= h(μ_i)

Where,μ_i= α_0+ $\alpha_1$$Z_{1i}$+$\alpha_2$ Z_{2i}+ ……+ α_5Z5iWhere, Z_1, Z_2,…,Z_5 are the variables from X_1,X_2….X_5 (Which are responsible for heteroscedasticity) & h is a some function does not depends on i.

The null hypothesis is that there is no heteroscedasticity, which is equivalent to:

H_0= α_1= α_2 =……= α_5 = 0

If H0 is accepted then, σ_i^2= h(μ_i) = h(x_0) = constant ∀i

The Breusch & pagan procedures two regressions & summarize as follows.

1) Regress y on X_1,X_2….,X_5 & estimate residual
2) Define $\widehat{\sigma^2}$ =RSS/n =$\sum_{i=1}^{n} e_i^2$
3) Regress =$\frac{ei^2}{\widehat{\sigma^2}}$ on Z_1, Z_2……..Z
4) Define the LM statistic

$$LM = \frac{ESS}{2}$$

Where ESS is expected as obtain in step 3

Under the H_0 2_M ~ $\chi^2_{\alpha(s)}$ & the null hypothesis is rejected is LM>$\chi^2_{\alpha(s)}$

Multicollinearity

Assumption 7 of the classical linear regression model (CLRM) is that there is no Multicollinearity among the regressors included in the regression model. The assumption 7 is violate the problem is multicollinearity.

Multicollinearity refers to a situation in which two or more explanatory variables in a multiple regression model are highly linearly related. We have perfect multicollinearity if, for example as in the equation above, the correlation between two independent variables is equal to 1 or −1. In practice, we rarely face perfect multicollinearity in a data set. More commonly, the issue of multicollinearity arises when there is an approximate linear relationship among two or more independent variables.

Method of detection of Multicollinearity

There are several methods to detect multicollinearity.

1. **Variance-Inflating Factor (VIF)**

Some authors have suggested a formal detection-tolerance or the variance inflation factor (VIF) for multicollinearity:

$$VIF = \frac{1}{(1-R_i^2)}$$

Where R_i^2 is the coefficient of determination of a regression of explanatory *j* on all the other explanators. A tolerance of less than 0.20 or 0.10 and/or a VIF of 5 or 10 and above indicates a multicollinearity problem.

2. **High R^2 but Few Significant *t* Ratios**

Consider the k-variable linear regression model:

$$Yi = \beta_1 + \beta_2 X_{2i} + \beta_3 X_{3i} + \cdots + \beta_k X_{ki} + u_i$$

In cases of high collinearity, it is possible to find, as we have just noted, that one or more of the partial slope coefficients are individually statistically insignificant on the basis of the t test. Yet the R_2 in such situations may be so high, say, in excess of 0.9, that on the basis of the F test one can convincingly reject the hypothesis that $\beta_2 = \beta_3 = \cdots = \beta_k = 0$. Indeed, this is one of the signals of multicollinearity insignificant t values but a high overall R_2 (and a significant F value).

Principal Component Analysis

In order to avoid the problems, we've seen in previous examples regarding multicollinearity and predicting values, we can use a process called principal component analysis. This process is a dimension reduction tool used to reduce a large set of correlated predictor variables to a smaller, less correlated set, called principal components, that still contains most of the information in the larger set. The first principal component contains as much of the variability in the data as possible, and the principal components following the first,

account for remaining variability as much as they possibly can. The analysis is usually performed on a square symmetric matrix, such as the covariance matrix (correlation matrix) which was explained.

Definition: The principal components for a set of vectors are a set of linear combinations of the vectors, chosen so that this captures the most information in a smaller subset of vectors. Even though this method may seem like a fool proof way to handle problems that multicollinearity causes, there is no guarantee that the new dimensions are interpretable after dimension reduction. Sometimes, when a variable is left out, important information and variance of the data is also removed so we aren't able to estimate parameters accurately.

ARIMA Model

Model Building:

Models for time series data can have many forms and represent different stochastic processes. When modelling variations in the level of a process, broad classes of practical importance are Exponential Smoothing models, the autoregressive (AR) models, the integrated (I) models, and the moving average (MA) models. These three classes depend linearly on previous data points. Combinations of these ideas produce autoregressive moving average (ARMA) and autoregressive integrated moving average (ARIMA) models. The autoregressive fractionally integrated moving average (ARFIMA) model generalizes the former three. Extensions of these classes to deal with vector-valued data are available under the heading of multivariate time-series models and sometimes the preceding acronyms are extended by including an initial "V" for "vector", as in VAR for vector autoregression. An additional set of extensions of these models is available for use where the observed time-series is driven by some "forcing" time-series (which may not have a causal effect on the observed series): the distinction from the multivariate case is that the forcing series may be deterministic or under the experimenter's control. For these models, the acronyms are extended with a final "X" for "exogenous".

Non-linear dependence of the level of a series on previous data points is of interest, partly because of the possibility of producing a chaotic time series. However, more importantly, 1empirical investigations can indicate the advantage of using predictions derived from non-linear models, over those from linear models, as for example in nonlinear autoregressive exogenous models. Further references on nonlinear time series analysis: (Kantz and Schreiber), and (Abarbanel)

Among other types of non-linear time series models, there are models to represent the changes of variance over time (heteroscedasticity). These models represent autoregressive conditional heteroscedasticity (ARCH) and the collection comprises a wide variety of representation (GARCH, TARCH, EGARCH, FIGARCH, CGARCH, etc.). Here changes in variability are related to, or predicted by, recent past values of the observed series. This is in contrast to other possible representations of locally varying variability, where the variability might be modelled as being driven by a separate time-varying process, as in a doubly stochastic model.

In recent work on model-free analyses, wavelet transform based methods (for example locally stationary wavelets and wavelet decomposed neural networks) have gained favour. Multiscale (often referred to as multiresolution) techniques decompose a given time series, attempting to illustrate time dependence at multiple scales. See also Markov switching multifractal (MSMF) techniques for modelling volatility evolution.

A Hidden Markov Model (HMM) is a statistical Markov model in which the system being modelled is assumed to be a Markov process with unobserved (hidden) states. An HMM can be considered as the simplest dynamic Bayesian network. HMM models are widely used in speech recognition, for translating a time series of spoken words into text.

For building any forecasting model, below are some key step need to follow:

1) Splitting into train and test:

   For validation mechanism, we are splitting our datasets into training and testing data.

2) Identifying the model performance metrics:

For identification of time series model performance metrics, we are using Scale dependent errors. It is defined as forecast errors are on the same scale as the data. Accuracy measures that are based only on error are therefore scale-dependent and cannot be used to make comparisons between series that involve different units.

The two most commonly used scale-dependent measures are based on the absolute errors or squared errors. We are using below accuracy metrics for our models:

- Mean absolute error (MAE)
- Root mean squared error (RMSE)
- Mean absolute percentage error (MAPE)

The formula for Mean absolute error (MAE) is as follow:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}(|y_i - \hat{y}_i|)$$

The formula for Root mean squared error (RMSE) is as follow:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

When comparing forecast methods applied to a single time series, or to several time series with the same units, the MAE is popular as it is easy to both understand and compute. A forecast method that minimizes the MAE will lead to forecasts of the median, while minimizing the RMSE will lead to forecasts of the mean. Consequently, the RMSE is also widely used, despite being more difficult to interpret.

However, it is widely seen reporting any error in Percentage form. Percentage errors have the advantage of being unit-free, and so are frequently used to compare forecast performances between data sets. The most commonly used measure is Mean absolute percentage error.

Formula for MAPE is as below:

$$\text{MAPE} = \frac{1}{n}\sum_{i=0}^{n}\frac{|Actual - Predicted|}{Actual} * 100$$

3) Exploring different time series model:

In this step, we are exploring below time series models:

- Simple Exponential Smoothing
- Exponential smoothing
- ARIMA Model
- Neural Network Modals

Time Series

A set of ordered observations of a quantitative variable taken at successive points in time is known as 'Time Series'. In other words, arrangement of statistical data in chronological order, i.e., in accordance with occurrence of time, is known as 'Time Series'. Time in terms of years, months, days, or hours, is simply device that enables one to relate all phenomenon's to a set of common, stable reference points. Mathematically, a time series is defined by the functional relationship

$$Y_t = f(t)$$

Components of time series

The various forces at work, affecting the values of a phenomenon in a time series, can be broadly classified into the four categories, commonly known as the components of time series, and they as follow.

1) Secular Trend or Long-term movement
2) Periodic Changes or Short-Term Fluctuations
    a) Seasonal variations
    b) Cyclic variations
3) Random or Irregular Movements

Mathematical Models for Time Series

The Following are the two models commonly used for the decomposition of a time series into Components.

1. Decomposition by Additive Model

$$Yt = Tt + St + Ct + Rt$$

2. Decomposition by Multiplicative Model

$$Yt = Tt * St * Ct * Rt$$

Uses of Time Series

1. It enables us to study the past behaviour of the phenomenon under consideration, i.e., to determine the type and nature of the variations in the data.
2. It enables to predict or estimate or forecast the behaviour of the phenomenon in future which is very essential for business planning.
3. It helps us to compare the changes in the values of different phenomenon at different times or places, etc.

Prediction and forecasting

In statistics, prediction is a part of statistical inference. One particular approach to such inference is known as predictive inference, but the prediction can be undertaken within any of the several approaches to statistical inference. Indeed, one description of statistics is that it provides a means of transferring knowledge about a sample of a population to the whole population, and to other related populations, which is not necessarily the same as prediction over time. When information is transferred across time, often to specific points in time, the process is known as forecasting.

1) Fully formed statistical models for stochastic simulation purposes, so as to generate alternative versions of the time series, representing what might happen over non-specific time-periods in the future
2) Simple or fully formed statistical models to describe the likely outcome of the time series in the immediate future, given knowledge of the most recent outcomes (forecasting).
3) Forecasting on time series is usually done using automated statistical software packages and programming languages, such as Wolfram Mathematic a, R, S, SAS, SPSS, Minitab, pandas (Python) and many others.

4) Forecasting on large scale data is done using Spark which has spark-ts as a third party package.

Forecasting:

Forecasting is the process of making predictions of the future based on past and present data and most commonly by analysis of trends. A commonplace example might be estimation of some variable of interest at some specified future date. Prediction is a similar, but more general term. Both might refer to formal statistical methods employing time series, cross-sectional or longitudinal data, or alternatively to less formal judgmental methods. Usage can differ between areas of application: for example, in hydrology the terms "forecast" and "forecasting" are sometimes reserved for estimates of values at certain specific future times, while the term "prediction" is used for more general estimates, such as the number of times floods will occur over a long period. Risk and uncertainty are central to forecasting and prediction; it is generally considered good practice to indicate the degree of uncertainty attaching to forecasts. In any case, the data must be up to date in order for the forecast to be as accurate as possible. In some cases the data used to predict the variable of interest is itself forecasted.

METHODOLOGY

Research Design:

A research design is the specification of methods and procedures for the needed information. Exploratory research design is adopted in the present study. It basically seeks to extract information about the influence and relationship between GDP and selected economic variables of Indian Economy

Sources of Data:

The data is collected by using secondary sources relating to the selected economic variables. Annual data is collected for a period from 1950-51 to 2018-19. And state wise data is collected for a period from 1980-81 to 2019-20.

Tools used in Analysis:

To analysis the data of GDP, first we check Heteroscedasticity, Autocorrelation & Multicollinearity. We see in data presence of Heteroscedasticity and Multicollinearity. And then remove Heteroscedasticity problem in GDP data, we apply Durbin-Watson d-statistic test, and for Multicollinearity problem, we apply Principal Component Analysis (PCA).The present study attempts to study the relationship between GDP and selected variables of Indian economy by using Coefficient of correlation, Analysis of Variance and impact of economic variables on GDP with the help of Regression Analysis.

DATA

Descriptive Statistics

Predictors Descriptive Statistics			
Predictors	**Mean**	**Std. Deviation**	**N**
Agriculture	491812.94	451560.055	69
Mining & Querrying	67358.58	94148.602	69
Manufacturing	383248.33	566220.172	69
Electricity Gas &Water supply	47512.42	70837.410	69
Construction	183895.38	267778.247	69
Trade	371077.71	601966.204	69
Financial real estate & prof servs	402803.14	692139.586	69
Public Adm	214721.23	405921.083	69
Gross National Income	2421562.74	3374183.134	69
Net National Income	2163615.65	2984706.577	69
Per Capita Income	21243.07	21480.965	69
Private Final Consumption Expenditure	1548223.86	1884287.975	69
Government Final Consumption Expenditure	270681.74	354361.915	69
Changes in Stocks	41124.16	67924.956	69
Export	447319.43	791462.122	69
Less Import	527958.90	913334.905	69
GDP	262015131.78	341648422.963	69

State wise Descriptive Statistics						
States	**N**	**Minimum**	**Maximum**	**Mean**	**Std. Deviation**	**Variance**
Andhra Pradesh	40	0	265140	84881.95	76126.195	5795197561.946
ArunachalPradesh	40	0	139588	22226.18	44076.794	1942763795.276
Assam	40	0	82078	26875.10	24825.567	616308796.656
Bihar	40	0	130171	33452.70	30941.068	957349690.369
Chhattisgarh	40	0	96887	19030.65	31559.336	995991691.977
Goa	40	0	458304	67592.23	135583.766	18382957593.153
Gujarat	40	0	367581	90536.67	84968.966	7219725263.199
Haryana	40	3386	264207	68013.95	73381.558	5384852996.356
HimachalPradesh	40	0	179188	33195.65	52696.209	2776890421.003
Jammu Kashmir	40	0	91882	19693.88	27088.835	733804990.215
Jharkhand	40	0	89491	25258.27	25641.429	657482860.563
Karnataka	40	0	272721	78942.60	72311.265	5228918995.579
Kerala	40	0	204105	59613.67	63150.898	3988035981.199
Madhya Pradesh	40	0	178144	49361.83	40558.141	1644962810.404
Maharashtra	40	0	742042	151033.43	161101.058	25953550762.661
Manipur	40	0	69978	12046.43	21739.522	472606836.558
Meghalaya	40	0	98151	18248.20	30981.075	959827000.267
Mizoram	40	0	201741	26952.23	54307.421	2949295976.897
Odisha	40	3535	125131	38159.05	34495.312	1189926554.049
Punjab	40	0	154996	54057.93	48105.097	2314100341.866
Rajasthan	40	4637	213079	59441.43	50842.578	2584967718.404
Tamilnadu	40	8081	403416	99664.02	88953.196	7912671054.999
Telangana	40	0	228216	56215.78	74116.599	5493270189.563
Tripura	40	0	113102	17469.88	32047.685	1027054115.599
UttarPradesh	40	15554	396309	93071.40	92612.947	8577158004.349
Uttarakhand	40	0	198738	36174.20	58361.191	3406028577.087

States	N	Minimum	Maximum	Mean	Std. Deviation	Variance
West Bengal	40	0	308837	74143.42	73203.399	5358737564.558
AndamanNicobar	40	0	159664	21321.18	46877.324	2197483514.558
Chandigarh	40	0	329209	58652.35	93116.424	8670668434.695
Delhi	40	0	365529	80702.13	105265.719	11080871665.394
Puducherry	40	184	237279	41018.88	74989.662	5623449474.215

ANALYSIS AND RESULT

1. Normality Test (Shapiro.test)

   w = 0.96091, p-value = 0.05548

2. Test for Autocorrelation (Durbin-Watson test)

   DW = 1.6311, p-value = 0.001123

3. Test for Heteroscedasticity (Breusch-Godfrey test)

   LM test = 2.819, df = 1, p-value = 0.9315

4. Test for Homoscedasticity (Goldfeld-Quant test)

   GQ = 0.42874, df1 = 13, df2 = 12, p-value = 0.928

   (For remove autocorrelation, we have use different autoregressive i.e. AR(1), AR(2))

5. Test for Autocorrelation (Durbin-Watson test)

   DW = 2.0103, p-value = 0.1016

6. Test for Multicollinearity (VIF)

V1	V2	V3	V4	V5	V6
1.921787e+03	5.139554e+02	2.080581e+03	1.503807e+03	4.998877e+02	4.717581e+03
V7	V8	V9	V10	V11	V12
1.267128e+03	8.010229e+02	2.520421e+05	3.009495e+05	5.813778e+03	3.478407e+03
V13	V14	V15	V16		
9.893064e+02	3.760357e+00	4.285390e+02	2.234663e+02		

7. Principal Component Analysis

```
> df
   comp round.EV..3. round.var..3. round.CV..3.
1     1       13.999         87.494        87.494
2     2        1.540          9.625        97.119
3     3        0.312          1.950        99.069
4     4        0.097          0.606        99.675
5     5        0.040          0.250        99.925
6     6        0.017          0.106       100.031
7     7        0.014          0.088       100.119
8     8        0.011          0.069       100.188
9     9        0.007          0.044       100.231
10   10        0.004          0.025       100.256
11   11        0.000          0.000       100.256
12   12       -0.003         -0.019       100.238
13   13       -0.005         -0.031       100.206
14   14       -0.009         -0.056       100.150
15   15       -0.011         -0.069       100.081
16   16       -0.013         -0.081       100.000
```

8. Principal Component Regression

```
Call:
lm(formula = Y ~ ., data = data3)

Residuals:
     Min       1Q   Median       3Q      Max
-0.09933 -0.01142  0.00395  0.01905  0.04095

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.962663   0.093246    42.5   <2e-16 ***
X1          -0.229284   0.001049  -218.5   <2e-16 ***
X2          -0.271277   0.003143   -86.3   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.02832 on 56 degrees of freedom
Multiple R-squared:  0.9991,    Adjusted R-squared:  0.999
F-statistic: 2.953e+04 on 2 and 56 DF,  p-value: < 2.2e-16
```

9. ARIMA (Annually)

    1) Stationary (Augmented Dickey-Fuller Test)

       Dickey-Fuller = -3.7564, Lag order = 4, p-value = 0.02684

       Alternative hypothesis: stationary

    2) Forecast (Box-Ljung test)

       X-squared = 21.709, df = 20, p-value = 0.3565

10. ARIMA (Quarterly)

    1) Stationary (Augmented Dickey-Fuller Test)

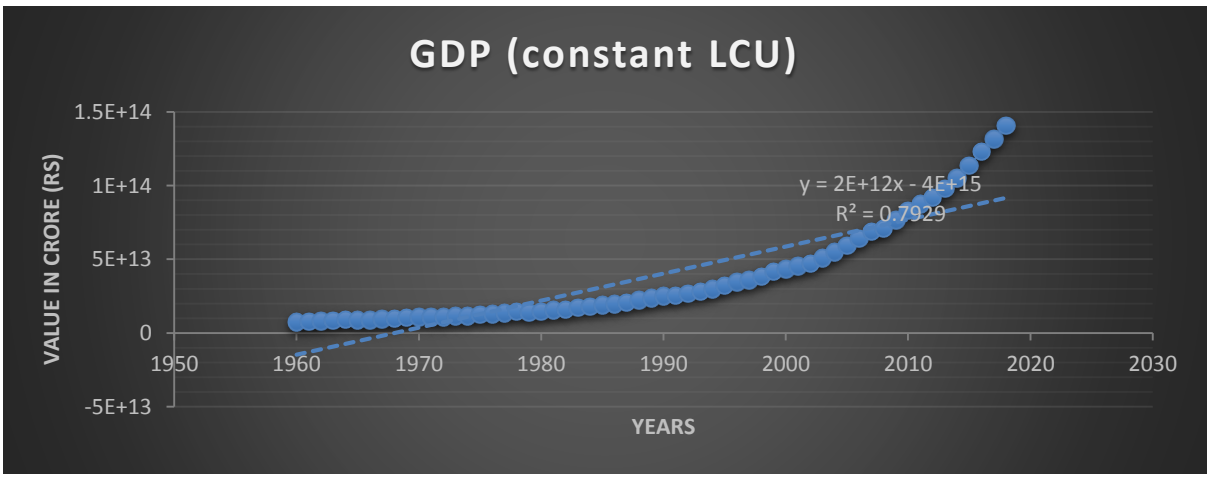
       Dickey-Fuller = -2.4136, Lag order = 4, p-value = 0.4065

       Alternative hypothesis: stationary

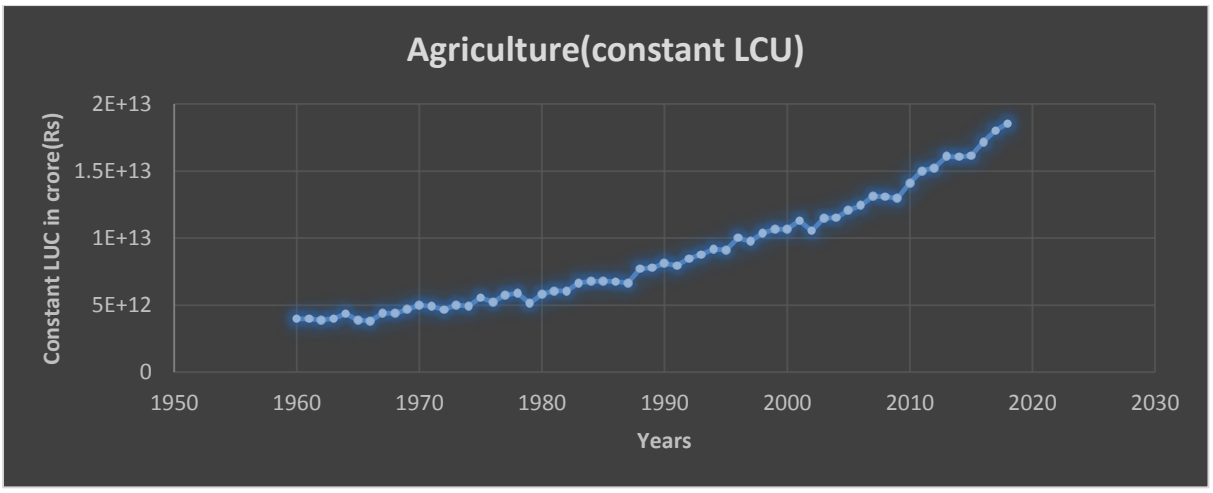
    2) Forecast (Box-Ljung test)

       X-squared = 16.033, df = 18, p-value = 0.5902

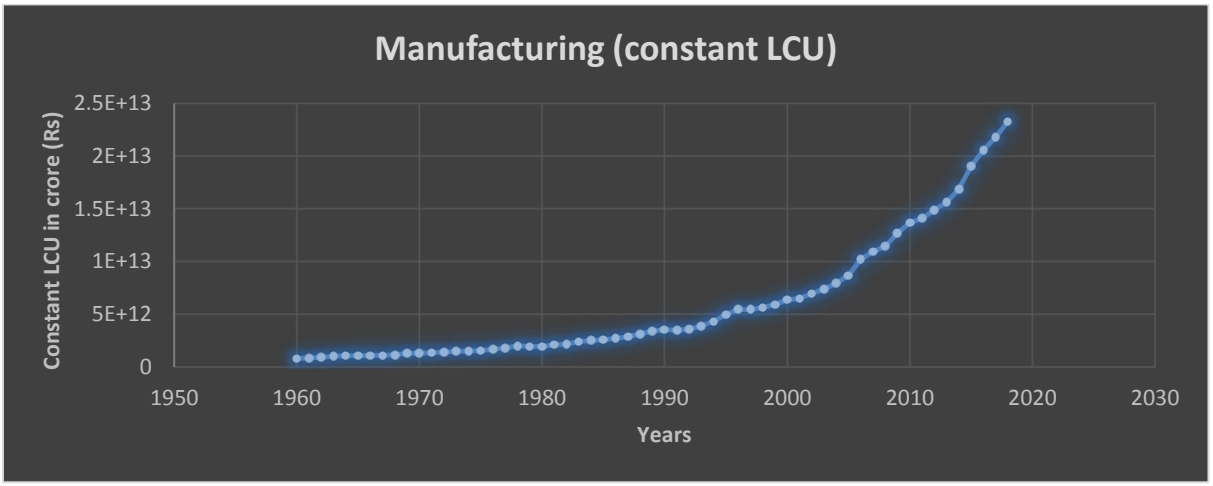
11. GDP (constant LCU) Vs Years (1960 – 2018)



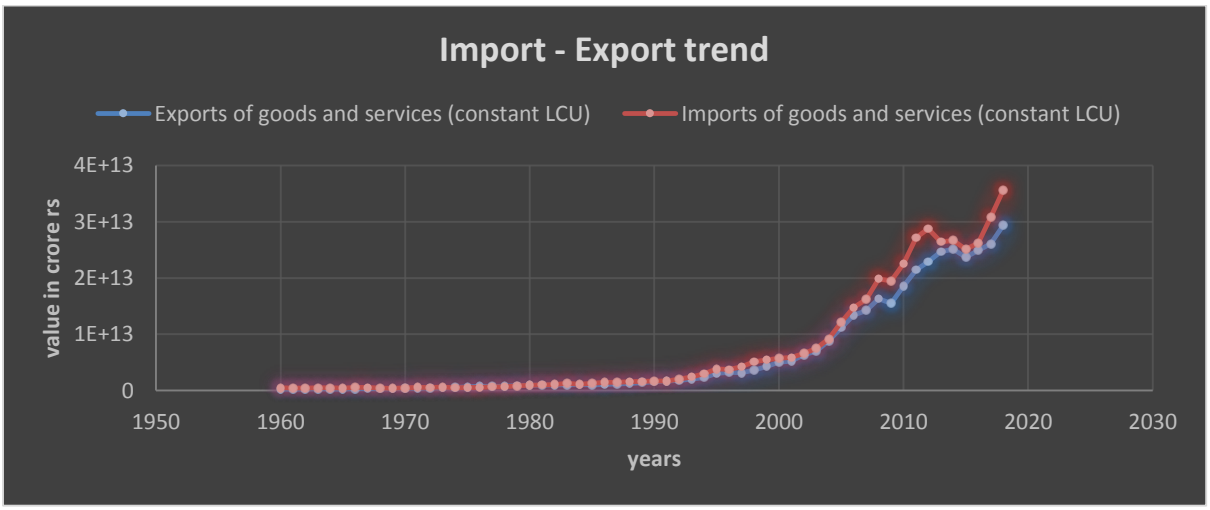
12. Agriculture (constant LCU) Vs Years



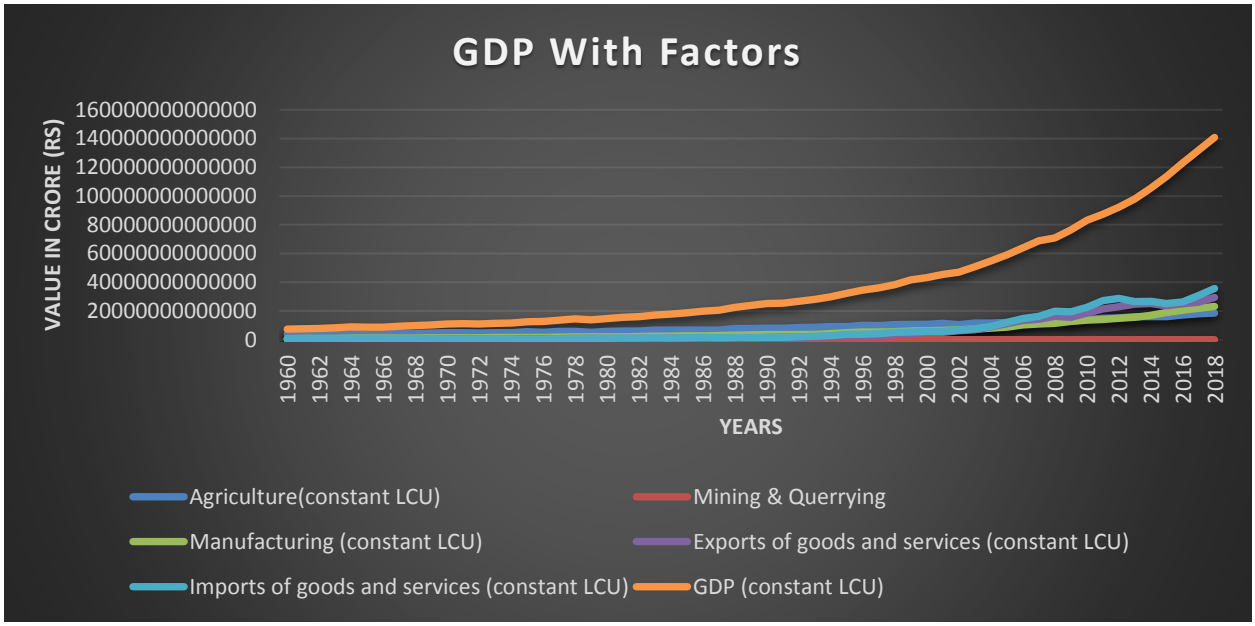
13. Manufacturing, value added (constant LCU)



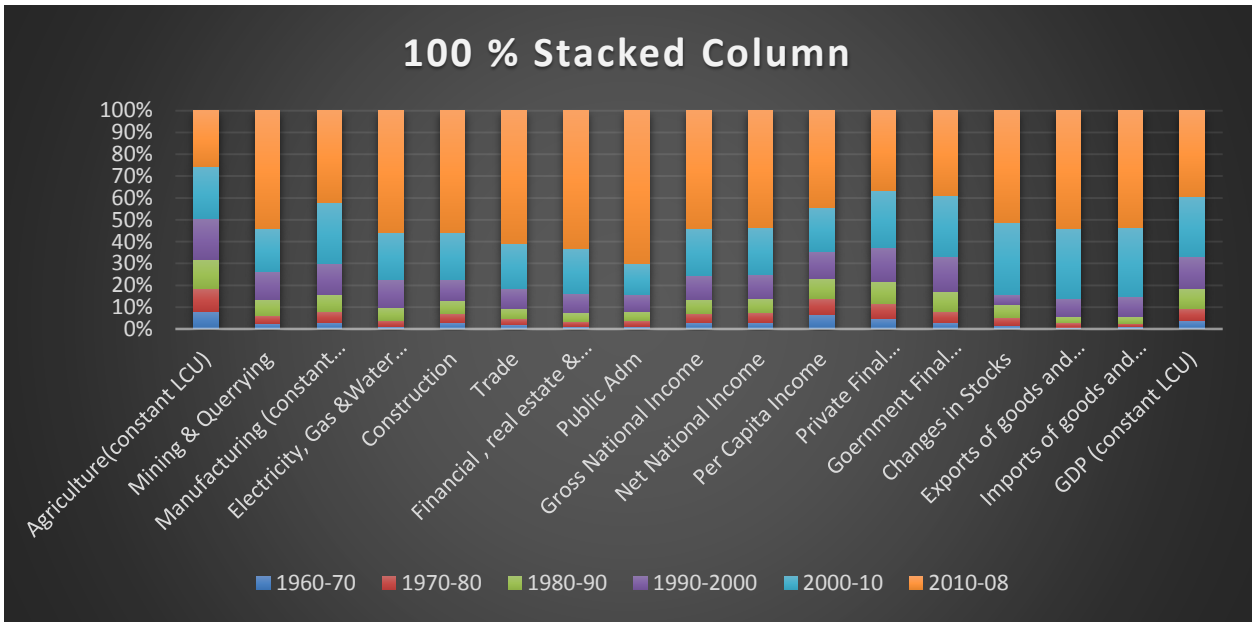
14. Import – Export trend.



15. All factors with GDP



16. 100% Stacked Column



DISCUSSION AND INTERPRETATION

1. Calculated p-value is greater than tabulated p-value 0.05 (alpha). Then our data is follows normal.
2. From the DW test, there is present of autocorrelation in the data.
3. The calculated p-value is greater than tabulated p-value (alpha 0.05) so we can say that data is homoscedastic.
4. For remove Autocorrelation, we can use different AR (1), AR (2) and AR (3). In the last AR (3) iteration we conclude that DW-statistics value "d" is near to 2 therefore there is no autocorrelation in the data.
5. From the calculated VIF, there is present of High multicollinearity in the data.
6. The first component explains 87.494% variation of the data set. Second component explains 9.625 percent of the variation. The first three components explain around 99.069 per cent variation of the data set.
7. From the principal component regression with the original dependent variable GDP the model is good. The entire components are significant effect of the given data.
8. ARIMA (Annually)
    1) Here the p-value displayed as 0.01, assuming significance alpha = 0.01,we reject the null hypothesis and classify this as stationary.
    2) The test statistic of the test is Q = 21.709 and the p-value of the test is 0.3565, which is much larger than 0.05. Thus, we fail to reject the null hypothesis of the test and conclude that the data values are independent and the And the Annual data are the forecast is pretty good up to next 10 years. GDP starting from 1953 to ending 2020. And the ending of the 2020 GDP rate is 4.2. And next 10 Years forecast in smoothly down GDP rate up to next 10 Years.
9. ARIMA (Quarterly)
    1) Here the p-value displayed as 0.4065, assuming significance alpha = 0.4065, we reject the null hypothesis and classify this as not stationary, we have to decompose data.

2) The test statistic of the test is Q = 16.033 and the p-value of the test is 0.5902, which is much larger than 0.05. Thus, we fail to reject the null hypothesis of the test and conclude that the data values are independent.

10. The above GDP graph is the constant value of local currency unit in India. The trend is the monotonically increasing to upward direction, and the linear line is well the accuracy in line is 79% in the linear line with respective parameters.
11. In the above fig, you can see the trend of agriculture is increasing in exponential growth.
12. In the above fig, manufacturing increasing in price to every year up to 2018. There are exponentially increasing trend in price of manufacturing factor.
13. The Import – Export trend is also increasing trend, but mostly we seen the trend of Export is less than Import. That is the Net export trend is also affect on GDP to improve Indian Economy.
14. The GDP and several factors are in the graphs are increasing except mining and querrying.

CONCLUSION

The Above analysis we conclude that the classical linear regression model we can see the some assumption is violated and how its deal systematically first we data transformed for normality and linearity assumption, after we goes no autocorrelation assumption there are also violated, then we tackle the problem of autocorrelation help of AR model after we check the multicollinearity it's also violated then we used to remove multicollinearity in data using Principal component analysis and remove the multicollinearity in the data.

Now all the classical linear regression model satisfy. Last we remove the multicollinearity problem in data using PCA. In PCA first two component is more amount of variation in the original data. Then we extract first two PCs components. And Regress first two PCs component and it's also known as Principal Component regression model. Then we estimate the parameters of original variable help of weighted matrix to multiply my PCs component coefficient and get the original classical linear regression model variables with best accuracy R-squares value is very close to one. The value of F-test found as very high and R square is found with very high variation that is 0.997. It is, therefore, concluded that all the variable are statistically significant at 5% of L.O.C.

Then we goes to Time series analysis and forecast the Annual GDP and Quarterly GDP for future values of GDP.

The conclusive outcome of the Indian Economy study is found as significantly GDP for all GDP variables with positive correlation. The GPD data analysis is to predict the Indian economy through classical linear regression model. We are satisfy all the assumption of linear regression model and finally conclude that our dependent variable is statistically significant effect of predicted GDP variables. And we forecast Annual GDP rate and Quarterly GDP rate using ARIMA Model (autoregressive integrated moving average), and the data are present in the time series data. Annual data are start from 1953 to 2020, the forecast of Annual GDP for next ten years forecasting and we see the forecast values is smoothly down direction. But we see the Quarterly GDP forecast in the start to 2001 to 2020 with respective quarters, here we see the forecast of the next five years means every years four quarters. Quarterly forecast value see the

values are upward direction. So our forecast of quarterly GDP positive direction and its good forecasting for Indian Economy country.

First we main objective how to improve Indian Economy. And the Indian Economy develop means to Improve Indian GDP. There are several sector, components develop our Indian economy. The main sector to develop Indian Economy is Agriculture, Industry and services and there are 15.4%, 23% and 61.5% contributes respectively. The GDP Components are Household consumption (59.1%), Government consumption (11.5%), Investment in fixed capital (28.5%), Investment in inventories (3.9%), Export of goods and services(19.1%) and Imports of goods services (-22%). The our objective of Indian economic GDP is changes when improve the economic development, increase employment, self-sufficient, economic stability, social welfare and services, regional development, comprehensive development, to reduce economic inequalities, social justice and increase in standard of living.

India, a developing country, a country that attracts huge business because of its large population. Economic growth in India has been one of the many positives that have existed since its independence.

Top seven Factors Affecting the Indian Economy, and how to implement on seven factors. There have been any recent blames on government about the economic slowdown but slowdowns like these are just an indication of the major changes that are about to come. We'll let us have a look at some of the factors that affect the Indian economy.

1) Capital flow and stock exchange Market.

India attracts investors. With such a huge population there is a huge chance for a thriving business opportunity. Owing to these factors the capital keeps flowing in India and the foreign exchange rates also help. Even if the market falls, India has less to worry about as the currency will still be overhauled.

2) Political changes.

This is among the major factors that affect the economic growth in India. The new governance brings in new changes and new policies. These policies play a major role in changing the import/export scenario which in turn plays a major part in the economy. The relation between the various foreign ministers also plays a very important role.

3) Global currency trends.

The currency of India is more or less interlinked with other major countries like USA, UK and Japan. If the domination value of these countries falls, then the value of INR is bound to fall. Similarly, if the value rises, then it affects the Indian economy as so much money is dependent on foreign exchange. Thus, foreign exchange is another major factor.

4) Demographic and Poverty Rates.

India has taken out millions of people out of poverty after independence. The result of this reflects on the positive economic growth. With India being such a huge international market, it

cannot afford people staying in poverty. With people out of poverty the value of India enhances internationally. If poverty somehow rises then it is bound to take the economy down.

5) Energy and Oil.

India is among the major oil importing countries in the world. When the price of oil fluctuates and it gets inflated then the INR is bound to get disturbed. It takes an unstable route which is not very good for such a fast growing economy.

6) The RBI banks.

RBI has almost everything to do with Indian economy. A slight change in the assessment ranking of RBI has a major impact on the INR. It can lead to over assessment or under ranking of the rupee.

7) Taxation system.

Taxation system impacts hugely the economy of a country. The easier, simple and stricter tax system is implemented in the country, the better will be the cash flow.

Citizens will be bound to stay corrupt free and honest. All this can only be implemented with the help of an open system. Once the taxation system is smooth, the country's economy will surely grow.

REFERENCES

Software Tools

1. R studio
2. SPSS
3. MS-Word
4. MS-Excel

Website

1. https://www.rbi.org.in/Scripts/AnnualPublications.aspx?head=Handbook%20of%20Statistics%20on%20Indian%20Economy/
2. http://mospi.nic.in/data: secondary data for annually and quarterly GDP data collected.
3. https://data.worldbank.org/indicator/NY.GDP.PCAP.CD?locations=AF&start=1971
4. Basic Econometrics by Damodar N. Gujrati
5. https://uclspp.github.io/PUBL0055/seminar8.html
6. https://www.jstatsoft.org/index
7. http://www.kse.org.ua/uploads/file/library/2003/Demchuk.pdf
8. http://www.diva-portal.org/smash/get/diva2:664110/FULLTEXT01.pdf
9. http://dergipark.gov.tr/download/article-file/364168
10. http://www.business.uwa.edu.au/data/assets/pdf_file/0004/2712244/15.10-Siddique,-A.-THE-IMPACT-OF-EXTERNAL-DEBT-ON-ECONOMIC-GROWTH-EMPIRICAL-EVIDENCE-FROM-HIGHLY-INDEBTED-POOR-COUNTRIES.pdf

APPENDIX

CODING AND OUTPUT

Import Dataset

```
data=read.csv(file.choose(),header = T)
> View(data)
> df=data[,-1]
> dim(df)
```

[1] 59 17

```
> head(df)
```

	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10
1	3.985662e+12	8857	7.900065e+11	2127	26295	36161	31252	12905	408739	385761
2	3.989018e+12	9367	8.574691e+11	2431	27219	38495	32596	13731	420953	396844
3	3.909672e+12	10479	9.198878e+11	2730	28233	40527	33693	15510	429594	404119
4	4.001130e+12	10789	1.006896e+12	3227	31680	43489	34735	17242	451446	424527
5	4.370205e+12	10945	1.076520e+12	3523	34225	46693	35688	19093	485193	456327
6	3.887638e+12	12231	1.086518e+12	3892	36509	46981	36766	19773	467155	436650

	x11	x12	x13	x14	x15	x16	y
1	8889	6.024061e+12	393950147076	6128	293837064043	495391527474	7.370436e+12
2	8938	6.126761e+12	423982005990	5204	293290899649	447061619964	7.644819e+12
3	8901	6.206577e+12	511559938963	4508	291590307387	464538221660	7.868898e+12
4	9149	6.437464e+12	628618037771	3680	316095783962	480295306195	8.340588e+12
5	9627	6.822025e+12	656720472996	6218	295511469528	496528052096	8.962207e+12
6	9003	6.828064e+12	720379534332	4695	254541237910	441630878920	8.725984e+12

```
> str(df)
```

```
> str(df)
'data.frame':   59 obs. of  17 variables:
 $ x1 : num  3.99e+12 3.99e+12 3.91e+12 4.00e+12 4.37e+12 ...
 $ x2 : num  8857 9367 10479 10789 10945 ...
 $ x3 : num  7.90e+11 8.57e+11 9.20e+11 1.01e+12 1.08e+12 ...
 $ x4 : num  2127 2431 2730 3227 3523 ...
 $ x5 : num  26295 27219 28233 31680 34225 ...
 $ x6 : num  36161 38495 40527 43489 46693 ...
 $ x7 : num  31252 32596 33693 34735 35688 ...
 $ x8 : num  12905 13731 15510 17242 19093 ...
 $ x9 : num  408739 420953 429594 451446 485193 ...
 $ x10: num  385761 396844 404119 424527 456327 ...
 $ x11: num  8889 8938 8901 9149 9627 ...
 $ x12: num  6.02e+12 6.13e+12 6.21e+12 6.44e+12 6.82e+12 ...
 $ x13: num  3.94e+11 4.24e+11 5.12e+11 6.29e+11 6.57e+11 ...
 $ x14: num  6128 5204 4508 3680 6218 ...
 $ x15: num  2.94e+11 2.93e+11 2.92e+11 3.16e+11 2.96e+11 ...
 $ x16: num  4.95e+11 4.47e+11 4.65e+11 4.80e+11 4.97e+11 ...
 $ y  : num  7.37e+12 7.64e+12 7.87e+12 8.34e+12 8.96e+12 ...
```

```
> y=df[, 17] # y is dependent variable GDP
> head(y)
```

[1] 7.370436e+12 7.644819e+12 7.868898e+12 8.340588e+12 8.962207e+12 8.725984e+12

```
> x=df[,-17]   # x is independent variable of original data.
```

The above data are transforming in to log transformation all variables.

```
> y=log(y)
> x=cbind(log(x$x1),log(x$x2),log(x$x3),log(x$x4),log(x$x5),log(x$x6),log(x$x7),
+       log(x$x8),log(x$x9),log(x$x10),log(x$x11),log(x$x12),log(x$x13),log(x$x14),
+       log(x$x15),log(x$x16))
```

Warning message:

In log(x$x14) :NaNs produced

```
> head(x) # call data into log transformation
```

```
> head(x)
         [,1]     [,2]     [,3]     [,4]     [,5]     [,6]     [,7]     [,8]
[1,] 29.01372 9.088963 27.39531 7.662468 10.17713 10.49574 10.34984 9.465370
[2,] 29.01457 9.144948 27.47725 7.796058 10.21167 10.55828 10.39194 9.527411
[3,] 28.99447 9.257129 27.54752 7.912057 10.24825 10.60972 10.42505 9.649240
[4,] 29.01760 9.286282 27.63789 8.079308 10.36344 10.68026 10.45550 9.755104
[5,] 29.10583 9.300638 27.70475 8.167068 10.44071 10.75135 10.48257 9.857077
[6,] 28.98882 9.411729 27.71400 8.266678 10.50531 10.75750 10.51233 9.892073
         [,9]    [,10]    [,11]    [,12]    [,13]    [,14]    [,15]    [,16]
[1,] 12.92083 12.86297 9.092570 29.42678 26.69949 8.720624 26.40629 26.92861
[2,] 12.95028 12.89130 9.098067 29.44369 26.77296 8.557183 26.40443 26.82596
[3,] 12.97060 12.90946 9.093919 29.45663 26.96073 8.413609 26.39862 26.86431
[4,] 13.02021 12.95873 9.121400 29.49316 27.16679 8.210668 26.47931 26.89767
[5,] 13.09230 13.03096 9.172327 29.55118 27.21052 8.735204 26.41197 26.93091
[6,] 13.05442 12.98689 9.105313 29.55206 27.30304 8.454253 26.26273 26.81374
```

```
> x=as.data.frame.matrix(x)
> summary(x)
```

```
> summary(x)
       V1              V2              V3              V4        
 Min.   :28.97   Min.   : 9.089   Min.   :27.40   Min.   : 7.662  
 1st Qu.:29.26   1st Qu.: 9.672   1st Qu.:28.06   1st Qu.: 8.980  
 Median :29.69   Median :10.656   Median :28.85   Median :10.138  
 Mean   :29.69   Mean   :10.652   Mean   :28.93   Mean   :10.102  
 3rd Qu.:30.07   3rd Qu.:11.313   3rd Qu.:29.67   3rd Qu.:11.007  
 Max.   :30.55   Max.   :12.823   Max.   :30.78   Max.   :12.565  

       V5              V6              V7              V8        
 Min.   :10.18   Min.   :10.50   Min.   :10.35   Min.   : 9.465  
 1st Qu.:10.76   1st Qu.:11.12   1st Qu.:10.83   1st Qu.:10.436  
 Median :11.35   Median :11.90   Median :11.89   Median :11.336  
 Mean   :11.63   Mean   :12.15   Mean   :12.08   Mean   :11.434  
 3rd Qu.:12.27   3rd Qu.:12.95   3rd Qu.:12.95   3rd Qu.:12.039  
 Max.   :13.87   Max.   :14.72   Max.   :14.86   Max.   :14.343  

       V9             V10             V11             V12        
 Min.   :12.92   Min.   :12.86   Min.   : 9.091   Min.   :29.43  
 1st Qu.:13.39   1st Qu.:13.30   1st Qu.: 9.221   1st Qu.:29.79  
 Median :14.05   Median :13.95   Median : 9.543   Median :30.41  
 Mean   :14.25   Mean   :14.15   Mean   : 9.770   Mean   :30.50  
 3rd Qu.:14.86   3rd Qu.:14.75   3rd Qu.:10.067   3rd Qu.:31.08  
 Max.   :16.45   Max.   :16.33   Max.   :11.436   Max.   :32.02  

      V13             V14             V15             V16        
 Min.   :26.70   Min.   : 6.912   Min.   :26.26   Min.   :26.73  
 1st Qu.:27.71   1st Qu.: 8.896   1st Qu.:27.12   1st Qu.:27.06  
 Median :28.64   Median : 9.776   Median :27.98   Median :28.12  
 Mean   :28.57   Mean   : 9.979   Mean   :28.38   Mean   :28.55  
 3rd Qu.:29.31   3rd Qu.:11.338   3rd Qu.:29.69   3rd Qu.:29.75  
 Max.   :30.34   Max.   :12.524   Max.   :31.01   Max.   :31.20  
                 NA's   :5
```

The above log transformation to some negative values in data are not covert in log transformation, the values convert to nans values in independent variable 14 to remove nans value to cleaning the data.

```
> x$V14=ifelse(is.na(x$V14),median(x$V14,na.rm = T),x$V14)
> summary(x)
```

```
> x$V14=ifelse(is.na(x$V14),median(x$V14,na.rm = T),x$V14)
> summary(x)
       V1              V2              V3              V4
 Min.   :28.97   Min.   : 9.089   Min.   :27.40   Min.   : 7.662
 1st Qu.:29.26   1st Qu.: 9.672   1st Qu.:28.06   1st Qu.: 8.980
 Median :29.69   Median :10.656   Median :28.85   Median :10.138
 Mean   :29.69   Mean   :10.652   Mean   :28.93   Mean   :10.102
 3rd Qu.:30.07   3rd Qu.:11.313   3rd Qu.:29.67   3rd Qu.:11.007
 Max.   :30.55   Max.   :12.823   Max.   :30.78   Max.   :12.565
       V5              V6              V7              V8
 Min.   :10.18   Min.   :10.50   Min.   :10.35   Min.   : 9.465
 1st Qu.:10.76   1st Qu.:11.12   1st Qu.:10.83   1st Qu.:10.436
 Median :11.35   Median :11.90   Median :11.89   Median :11.336
 Mean   :11.63   Mean   :12.15   Mean   :12.08   Mean   :11.434
 3rd Qu.:12.27   3rd Qu.:12.95   3rd Qu.:12.95   3rd Qu.:12.039
 Max.   :13.87   Max.   :14.72   Max.   :14.86   Max.   :14.343
       V9             V10             V11             V12
 Min.   :12.92   Min.   :12.86   Min.   : 9.091   Min.   :29.43
 1st Qu.:13.39   1st Qu.:13.30   1st Qu.: 9.221   1st Qu.:29.79
 Median :14.05   Median :13.95   Median : 9.543   Median :30.41
 Mean   :14.25   Mean   :14.15   Mean   : 9.770   Mean   :30.50
 3rd Qu.:14.86   3rd Qu.:14.75   3rd Qu.:10.067   3rd Qu.:31.08
 Max.   :16.45   Max.   :16.33   Max.   :11.436   Max.   :32.02
      V13             V14             V15             V16
 Min.   :26.70   Min.   : 6.912   Min.   :26.26   Min.   :26.73
 1st Qu.:27.71   1st Qu.: 8.929   1st Qu.:27.12   1st Qu.:27.06
 Median :28.64   Median : 9.776   Median :27.98   Median :28.12
 Mean   :28.57   Mean   : 9.962   Mean   :28.38   Mean   :28.55
 3rd Qu.:29.31   3rd Qu.:11.084   3rd Qu.:29.69   3rd Qu.:29.75
 Max.   :30.34   Max.   :12.524   Max.   :31.01   Max.   :31.20
> sum(is.na(x))
[1] 0
```

```
> sum(is.na(x)) # there is no missing values in the above data.
```

[1] 0

The data frame complete y and x then we regress y on x variables.

```
> df1=data.frame(y,x)  # df1 is data frame to y(GDP) and x ( GDP factors)
> model=lm(y~.,data = df1)
> summary(model)
```

```
> model=lm(y~.,data = df1)
> summary(model)

Call:
lm(formula = y ~ ., data = df1)

Residuals:
      Min        1Q    Median        3Q       Max
-0.0160260 -0.0057336  0.0002305  0.0050085  0.0264007

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.932268   2.600900   1.127 0.265971
V1           0.238026   0.101139   2.353 0.023356 *
V2          -0.013942   0.033502  -0.416 0.679421
V3           0.057369   0.071840   0.799 0.429039
V4          -0.056473   0.044593  -1.266 0.212347
V5           0.046575   0.036678   1.270 0.211134
V6           0.172876   0.088793   1.947 0.058249 .
V7           0.186637   0.040083   4.656 3.22e-05 ***
V8          -0.193376   0.068071  -2.841 0.006912 **
V9          -0.197695   0.749596  -0.264 0.793273
V10          0.311880   0.958190   0.325 0.746428
V11         -0.046873   0.212910  -0.220 0.826817
V12          0.328581   0.080497   4.082 0.000196 ***
V13          0.230715   0.047211   4.887 1.54e-05 ***
V14          0.001913   0.002278   0.840 0.405749
V15          0.047546   0.022437   2.119 0.040039 *
V16         -0.064755   0.012422  -5.213 5.32e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.009465 on 42 degrees of freedom
Multiple R-squared:  0.9999,    Adjusted R-squared:  0.9999
F-statistic: 3.12e+04 on 16 and 42 DF,  p-value: < 2.2e-16
```

The above data we regress regression model GDP on all selected GDP factors. The give regression model, we seen the model are good, but actually it's not good. Because the R^2 is very high in the data but few significant t-ratio in the model, there is problem of multicollinearity in the data. There we check the one by one assumption of Classical Linear Regression Model.

Check the Normality

Data is normal or not, the check the normality for the data to use test of shapiro.test in R. package install library (lmtest).

```
> shapiro.test(residuals(model))# H0: data is normal
```

```
> shapiro.test(residuals(model))# H0: data is normal

        Shapiro-Wilk normality test

data:  residuals(model)
W = 0.96091, p-value = 0.05548
```

Interpretation: The above the check the p-value is greater than 0.05 (alpha). The conclusion is the data is normal follow.

The plot the GPD vs Year

```
> plot(data[,1],data[,18],ylab="GDP",xlab = "Year")
```

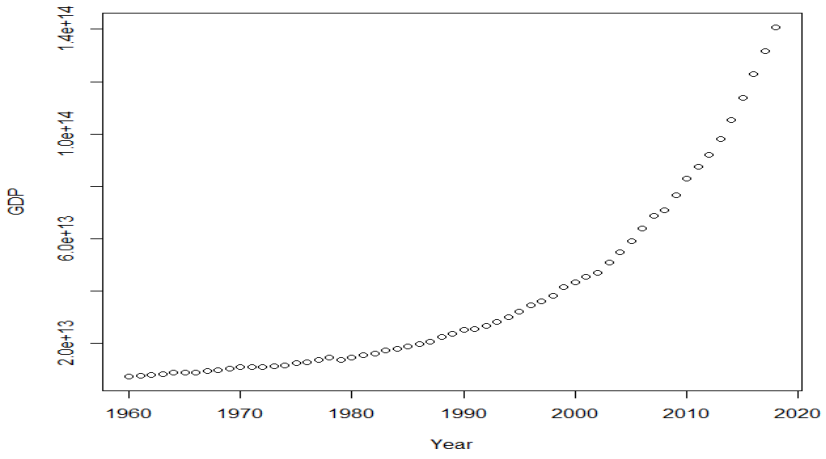


Fig. Scatter plot GDP vs. Year

The above graph we can see the GDP are exponentially increasing in the period of time. The GDP is measure in crore in rupees. The GDP data are plot 1960-2018 period of time GDP in the above GDP graph data.

```
> hist(y)  # Histogram of GDP.
```

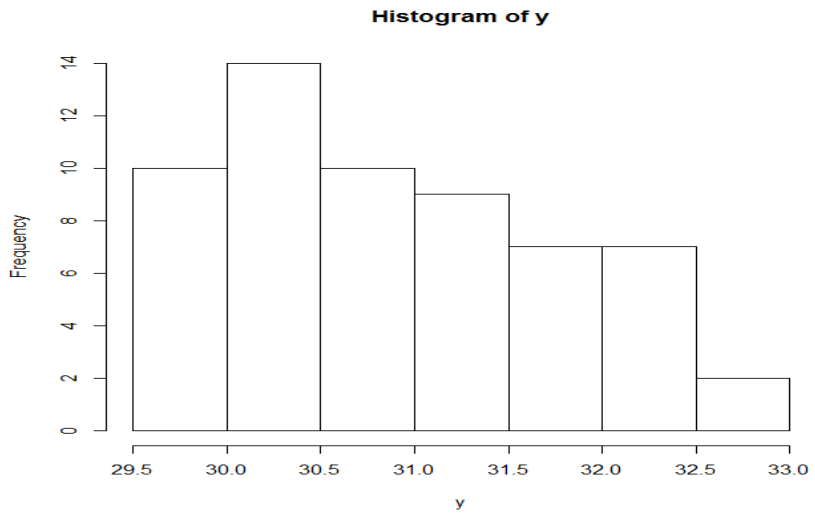


Fig Histogram for Frequency vs. y(GDP)

We can see the Histogram is approximately normal shape in histogram plot of GDP.

```
>acf(y) # ACF is known as Auto Correlation Function.
```

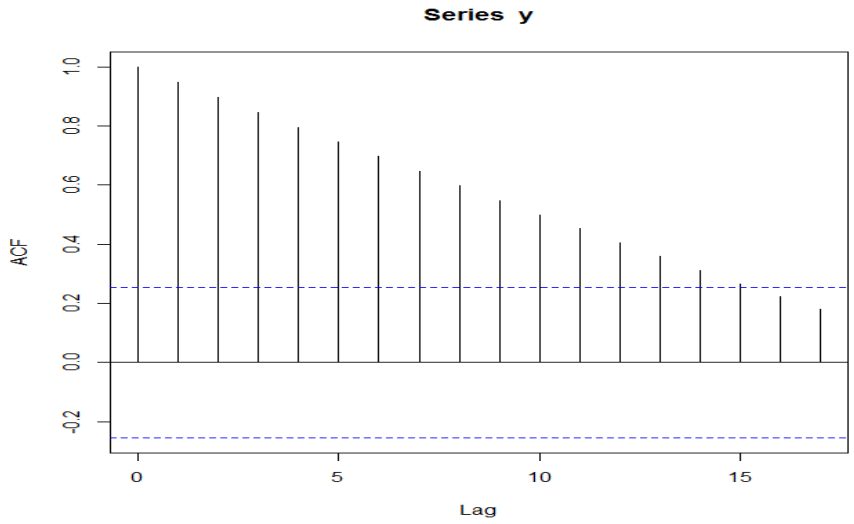


Fig ACF plot for Series y(GDP)

The ACF plot is GDP. The ACF plot we seen the line of center of ACF is above all the line plotted in the plot. The upper boundary of ACF is cross some line in the graph. The total no. of line cross is 16. That is the above in lag value is 16 in GDP data.

To check Autocorrelation

Some test is to test of data in presence of autocorrelation or not.

Durbin-Watson test (DW test)

```
> library(zoo)
> library(lmtest)
>dwtest(model) # Ho: data is no autocorrelation
```

```
> dwtest(model) # Ho: data is no autocorrelation

        Durbin-Watson test

data:  model
DW = 1.6311, p-value = 0.001123
alternative hypothesis: true autocorrelation is greater than 0
```

Interpretation: The DW test we can see and conclude the given data are presence of autocorrelation is the data.

11.4 To check the Heteroscedasticity.

1. By using Breusch-pagan test. To check data is homoscedasticity.
2. Goldfeld-Quandt test

```
>bgtest(model) # H₀: the data is homoscedasticity or there is constant variance.
```

```
> bgtest(model)

        Breusch-Godfrey test for serial correlation of order up to 1

data:  model
LM test = 2.819, df = 1, p-value = 0.09315
```

Interpretation: The conclude the above test the p value is greater than alpha (0.05). That is data is homoscedasticity.

```
>gqtest(model2) # H0 : data is homoscedasticity.
```

```
> gqtest(model2)

        Goldfeld-Quandt test

data:  model2
GQ = 0.42874, df1 = 13, df2 = 12, p-value = 0.928
alternative hypothesis: variance increases from segment 1 to 2
```

Interpretation: The above data is Homoscedastic.

The assumption of autocorrelation is violated in the above assumption. That is data in presence of autocorrelation. Then we remove the presence of autocorrelation in data. By using the AR (1) with first difference of autoregressive that is lag 1 first iteration.

```
> e=model$residuals
>e=as.vector(e):e
```

```
> e=as.vector(e);e
 [1] -0.0003454586 -0.0001594235 -0.0020026815 -0.0124960798  0.0077913564
 [6] -0.0057579570 -0.0015038902  0.0056898221  0.0057701704  0.0027961667
[11] -0.0007312502  0.0039025365  0.0029682326  0.0068336828  0.0207029984
[16]  0.0126672123 -0.0050059478 -0.0009330813 -0.0091018124 -0.0025434773
[21] -0.0084446962 -0.0093482161 -0.0017450636 -0.0048960909 -0.0160260097
[26] -0.0116239890  0.0014766119  0.0048431303  0.0036898111  0.0011666817
[31] -0.0030840331 -0.0094747600  0.0083199028 -0.0074767762  0.0048351459
[36]  0.0079182745 -0.0059670851  0.0051739642 -0.0028380975  0.0054827200
[41]  0.0069282047 -0.0125238554  0.0015638955  0.0009645678  0.0264007274
[46]  0.0085310904  0.0043566902 -0.0088182036 -0.0126445966  0.0083430573
[51] -0.0098961971  0.0002304768  0.0021791883 -0.0104412427 -0.0037233498
[56] -0.0057093353  0.0035250507  0.0056542303  0.0045570577
```

```
> e1=e[1:58]
> e2=e[2:59]
> d=(sum(e1^2)+sum(e2^2)-(2*sum(e1*e2)))/(sum(e^2))
> d
```

[1] 1.631067

```
> rho=sum(e1*e2)/sum(e^2)
> rho
```

[1] 0.181691

```
> x=as.vector(x)
> class(x)
```

[1] "data.frame"

```
>yone=y[1]*sqrt(1-rho^2)
> y_1=y[2:59]-rho*y[1:58]
> y_1=append(y_1,yone,after = 0)
> end(y_1)
```

[1] 59 1

We check the first AR(1) model lag difference in ACF plot.

```
>acf(y_1)
```

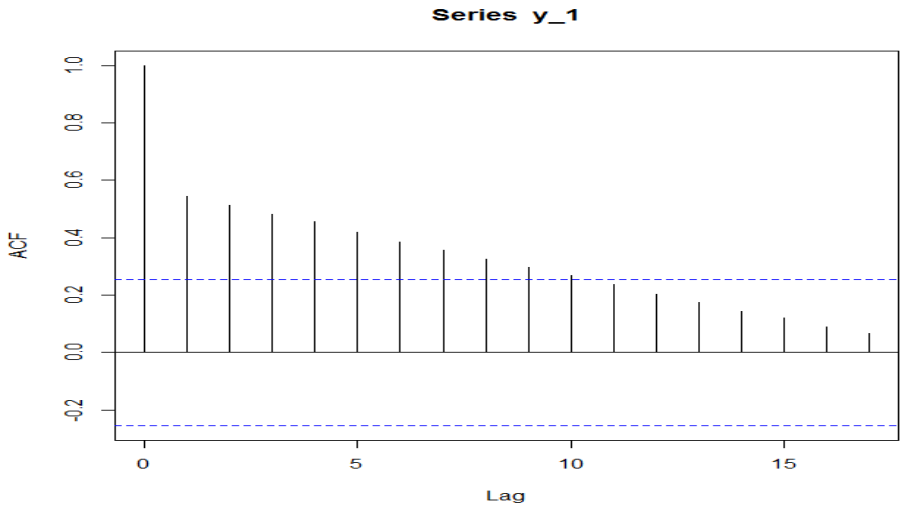


Fig ACF plot for Series y_1

The above ACF plot we can see the lag value is decrease by 5. The above ACF plot is presence of autocorrelation in the data, because the line cross the some line.

```
> z=rho*x[1:58,1:16]
> w=x[2:59,1:16]
> x_11=x[1,1:16]*sqrt(1-rho^2)
> x_12=w-z
```

```
> x_1=rbind(x_11,x_12)
> head(x_1)
```

```
> head(x_1)
        V1       V2       V3       V4        V5        V6        V7       V8
1 28.53081 8.937684 26.93933 7.534931 10.007742 10.321042 10.177572 9.307825
2 23.74303 7.493565 22.49977 6.403856  8.362576  8.651302  8.511472 7.807638
3 23.72279 7.595573 22.55515 6.495583  8.392878  8.691378  8.536922 7.918195
4 23.74956 7.604345 22.63276 6.641758  8.501426  8.752572  8.561366 8.001923
5 23.83359 7.613404 22.68320 6.699130  8.557767  8.810841  8.582898 8.084662
6 23.70055 7.721886 22.68029 6.782795  8.608330  8.804075  8.607740 8.101130
        V9      V10      V11      V12      V13      V14      V15      V16
1 12.70577 12.64888 8.941230 28.93699 26.25509 8.575475 25.96678 26.48041
2 10.60268 10.55421 7.446029 24.09710 21.92190 6.972724 21.60664 21.93327
3 10.61765 10.56723 7.440882 24.10698 22.09632 6.858845 21.60117 21.99027
4 10.66357 10.61320 7.469116 24.14115 22.26827 6.681991 21.68292 22.01666
5 10.72665 10.67648 7.515050 24.19254 22.27456 7.243399 21.60092 22.04384
6 10.67566 10.61928 7.438783 24.18288 22.35914 6.867145 21.46391 21.92064
```

```
> dim(x_1)
```

[1] 59 16

Again model fit to new data set of y_1 and x_1 of first iteration of AR(1). We check the model1 with dataset data1.

```
> data1=data.frame(y_1,x_1)
> model1=lm(y_1~.,data = data1)
> summary(model1)
```

```
> model1=lm(y_1~.,data = data1)
> summary(model1)

Call:
lm(formula = y_1 ~ ., data = data1)

Residuals:
       Min         1Q     Median         3Q        Max
-0.0166482 -0.0060177 -0.0003647  0.0052934  0.0247282

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.023739   0.071663   0.331 0.742093
V1           0.312450   0.058929   5.302 3.97e-06 ***
V2           0.004188   0.033619   0.125 0.901446
V3           0.124774   0.055024   2.268 0.028559 *
V4          -0.081000   0.046711  -1.734 0.090245 .
V5           0.069720   0.032932   2.117 0.040219 *
V6           0.165498   0.089546   1.848 0.071625 .
V7           0.188220   0.040604   4.636 3.44e-05 ***
V8          -0.118624   0.034725  -3.416 0.001421 **
V9          -0.080842   0.751778  -0.108 0.914877
V10         -0.095147   0.839526  -0.113 0.910306
V11          0.049537   0.163589   0.303 0.763524
V12          0.384841   0.071545   5.379 3.09e-06 ***
V13          0.187372   0.039379   4.758 2.32e-05 ***
V14          0.002347   0.002170   1.082 0.285519
V15          0.054495   0.019743   2.760 0.008524 **
V16         -0.057096   0.013568  -4.208 0.000133 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.009355 on 42 degrees of freedom
Multiple R-squared:  0.9999,    Adjusted R-squared:  0.9999
F-statistic: 3.101e+04 on 16 and 42 DF,  p-value: < 2.2e-16
```

Same as the above model, the R^2 is high but few significant t-ratio. Again check the autocorrelation by using DW test.

```
>dwtest(model1) # To check the model1 to data in presence of autocorrelation or not.
```

```
> dwtest(model1)

        Durbin-Watson test

data:  model1
DW = 1.8205, p-value = 0.01682
alternative hypothesis: true autocorrelation is greater than 0
```

Interpretation: The above DW-test in p value is less than alpha that is presence of autocorrelation in data.

Again second iteration to AR(2) autoregressive model to remove the autocorrelation.

```
> e_1=model1$residuals;e_1
```

```
> e_1=model1$residuals;e_1
            1             2             3             4             5
 0.0002288402 -0.0013130392 -0.0030241504 -0.0098815432  0.0082669852
            6             7             8             9            10
-0.0049807497 -0.0039971881  0.0056501350  0.0069680329  0.0031284950
           11            12            13            14            15
-0.0004147739  0.0055062150  0.0025399810  0.0053343328  0.0184678678
           16            17            18            19            20
 0.0132197654 -0.0051095732 -0.0006075532 -0.0087915551 -0.0010257976
           21            22            23            24            25
-0.0092041570 -0.0105283392 -0.0009383554 -0.0077684361 -0.0166482220
           26            27            28            29            30
-0.0074524873  0.0047026811  0.0049282741 -0.0003646896 -0.0023139527
           31            32            33            34            35
-0.0065973423 -0.0086839344  0.0102099623 -0.0054379686  0.0071840703
           36            37            38            39            40
 0.0062002883 -0.0075102132  0.0076635304 -0.0032600954  0.0066856945
           41            42            43            44            45
 0.0052523946 -0.0130257599  0.0068646476  0.0025673097  0.0247281620
           46            47            48            49            50
 0.0041637045  0.0024980111 -0.0067065919 -0.0127863821  0.0101754756
           51            52            53            54            55
-0.0108781416  0.0016869430  0.0006137210 -0.0128422452 -0.0005622780
           56            57            58            59
-0.0024934927  0.0041556985  0.0040768888  0.0014809005
```

```
> sum(is.na(e_1))
```

[1] 0

```
> e_1=as.vector (e_1);e_1
```

```
> e_1=as.vector(e_1);e_1
 [1]  0.0002288402 -0.0013130392 -0.0030241504 -0.0098815432  0.0082669852
 [6] -0.0049807497 -0.0039971881  0.0056501350  0.0069680329  0.0031284950
[11] -0.0004147739  0.0055062150  0.0025399810  0.0053343328  0.0184678678
[16]  0.0132197654 -0.0051095732 -0.0006075532 -0.0087915551 -0.0010257976
[21] -0.0092041570 -0.0105283392 -0.0009383554 -0.0077684361 -0.0166482220
[26] -0.0074524873  0.0047026811  0.0049282741 -0.0003646896 -0.0023139527
[31] -0.0065973423 -0.0086839344  0.0102099623 -0.0054379686  0.0071840703
[36]  0.0062002883 -0.0075102132  0.0076635304 -0.0032600954  0.0066856945
[41]  0.0052523946 -0.0130257599  0.0068646476  0.0025673097  0.0247281620
[46]  0.0041637045  0.0024980111 -0.0067065919 -0.0127863821  0.0101754756
[51] -0.0108781416  0.0016869430  0.0006137210 -0.0128422452 -0.0005622780
[56] -0.0024934927  0.0041556985  0.0040768888  0.0014809005
```

```
> e11=e_1[1:58];e11
```

```
> e11=e_1[1:58];e11
 [1]  0.0002288402 -0.0013130392 -0.0030241504 -0.0098815432  0.0082669852
 [6] -0.0049807497 -0.0039971881  0.0056501350  0.0069680329  0.0031284950
[11] -0.0004147739  0.0055062150  0.0025399810  0.0053343328  0.0184678678
[16]  0.0132197654 -0.0051095732 -0.0006075532 -0.0087915551 -0.0010257976
[21] -0.0092041570 -0.0105283392 -0.0009383554 -0.0077684361 -0.0166482220
[26] -0.0074524873  0.0047026811  0.0049282741 -0.0003646896 -0.0023139527
[31] -0.0065973423 -0.0086839344  0.0102099623 -0.0054379686  0.0071840703
[36]  0.0062002883 -0.0075102132  0.0076635304 -0.0032600954  0.0066856945
[41]  0.0052523946 -0.0130257599  0.0068646476  0.0025673097  0.0247281620
[46]  0.0041637045  0.0024980111 -0.0067065919 -0.0127863821  0.0101754756
[51] -0.0108781416  0.0016869430  0.0006137210 -0.0128422452 -0.0005622780
[56] -0.0024934927  0.0041556985  0.0040768888
```

```
> e21=e_1[2:59];e21
```

```
> e21=e_1[2:59];e21
 [1] -0.0013130392 -0.0030241504 -0.0098815432  0.0082669852 -0.0049807497
 [6] -0.0039971881  0.0056501350  0.0069680329  0.0031284950 -0.0004147739
[11]  0.0055062150  0.0025399810  0.0053343328  0.0184678678  0.0132197654
[16] -0.0051095732 -0.0006075532 -0.0087915551 -0.0010257976 -0.0092041570
[21] -0.0105283392 -0.0009383554 -0.0077684361 -0.0166482220 -0.0074524873
[26]  0.0047026811  0.0049282741 -0.0003646896 -0.0023139527 -0.0065973423
[31] -0.0086839344  0.0102099623 -0.0054379686  0.0071840703  0.0062002883
[36] -0.0075102132  0.0076635304 -0.0032600954  0.0066856945  0.0052523946
[41] -0.0130257599  0.0068646476  0.0025673097  0.0247281620  0.0041637045
[46]  0.0024980111 -0.0067065919 -0.0127863821  0.0101754756 -0.0108781416
[51]  0.0016869430  0.0006137210 -0.0128422452 -0.0005622780 -0.0024934927
[56]  0.0041556985  0.0040768888  0.0014809005
```

```
> d=(sum(e11^2)+sum(e21^2)-(2*sum(e11*e21)))/(sum(e_1^2))
> d
```

[1] 1.820518

```
> rho1=sum(e11*e21)/sum(e_1^2)
> rho1
```

[1] 0.08943536

```
> y_1one=y_1[1]*sqrt(1-rho1^1)
> y_11one=y_1[2:59]-rho1*y_1[1:58]
> y_11=append(y_11one,y_1one,after = 0)
> end(y_11)
```

[1] 59 1

```
>acf(y_11) # the third acf plot in the second AR(2) model .
```

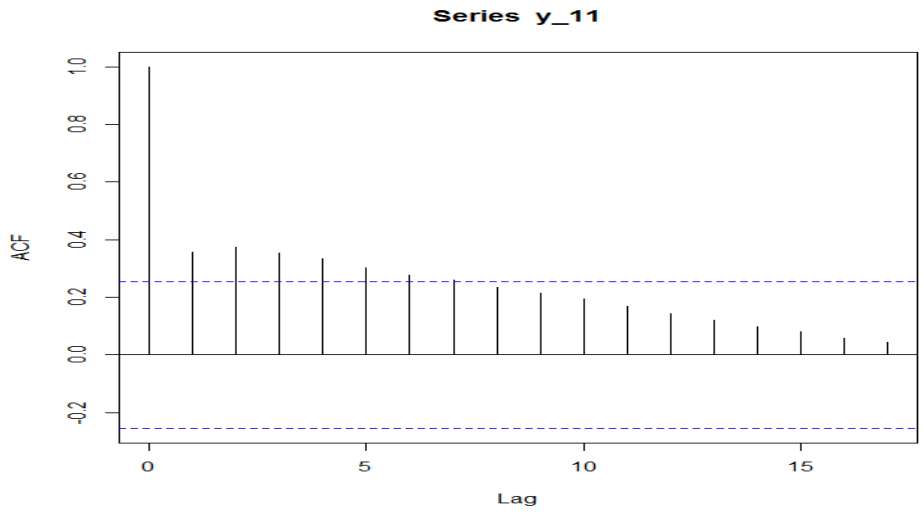


Fig 11.5 ACF plot for Series y_11

The second iteration we check the lag value decrease the above two ACF plot.

```
>x_one=x_1[1,1:16]*sqrt(1-rho1^2)
> z1=rho1*x_1[1:58,1:16]
> w1=x_1[2:59,1:16]
> x_1one=w1-z1
> dim(x_1one)
```

[1] 58 16

```
> x_11=rbind(x_one,x_1one)
> dim(x_11)
```

[1] 59 16

```
> data2=data.frame(y_11,x_11)
> dim(data2)
```

[1] 59 17

We again fit the regression model. For dataset data2 to the second AR(2) model

```
> model2=lm(y_11~.,data = data2)
> summary(model2)
```

```
> model2=lm(y_11~.,data = data2)
> summary(model2)

Call:
lm(formula = y_11 ~ ., data = data2)

Residuals:
      Min        1Q    Median        3Q       Max
-0.033810 -0.006402 -0.001714  0.004655  0.030075

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.658363   0.060670  60.300  < 2e-16 ***
V1           0.199759   0.078530   2.544  0.01474 *
V2           0.011335   0.045074   0.251  0.80268
V3           0.073164   0.073112   1.001  0.32270
V4          -0.040954   0.063517  -0.645  0.52258
V5           0.062005   0.044533   1.392  0.17114
V6           0.235133   0.119056   1.975  0.05487 .
V7           0.103163   0.055530   1.858  0.07022 .
V8          -0.248347   0.046270  -5.367 3.21e-06 ***
V9           0.077047   0.992487   0.078  0.93849
V10          0.094782   1.098882   0.086  0.93167
V11         -0.003449   0.215505  -0.016  0.98731
V12          0.163510   0.096899   1.687  0.09894 .
V13          0.340179   0.052534   6.475 8.22e-08 ***
V14          0.001157   0.002767   0.418  0.67783
V15          0.028733   0.025676   1.119  0.26947
V16         -0.059350   0.018437  -3.219  0.00248 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.01223 on 42 degrees of freedom
Multiple R-squared:  0.9999,    Adjusted R-squared:  0.9998
F-statistic: 1.983e+04 on 16 and 42 DF,  p-value: < 2.2e-16
```

Same as the above three model condition.

Again check autocorrelation by using DW-test.

```
>dwtest(model2)
```

```
> dwtest(model2)

        Durbin-Watson test

data:  model2
DW = 2.0103, p-value = 0.1016
alternative hypothesis: true autocorrelation is greater than 0
```

Interpretation: The all the iteration we can conclude the second iteration of AR (2) model to remove the autocorrelation in the model. Because the p- value is greater than alpha (0.05). and also, second criterial the DW-statistics value "d" is near to 2 then the conclude the there is no autocorrelation in data.

```
> library(lmtest)
```

To check Multicollinearity

```
> library("faraway")
>faraway::vif(model2) # The VIF > 5. i.e there is multicollinearity
```

```
V1          V2          V3          V4          V5          V6
1.921787e+03 5.139554e+02 2.080581e+03 1.503807e+03 4.998877e+02 4.717581e+03
V7          V8          V9          V10         V11         V12
1.267128e+03 8.010229e+02 2.520421e+05 3.009495e+05 5.813778e+03 3.478407e+03
V13         V14         V15         V16
9.893064e+02 3.760357e+00 4.285390e+02 2.234663e+02
```

Interpretation: From the above VIF, there is presence of multicollinearity in the data.

Principal Component Analysis

```
# Data 2 loaded for PCA .
> str(data2)
```

```
> str(data2)
'data.frame':   59 obs. of  17 variables:
 $ y_11: num  27.8 21.7 22.1 22.2 22.2 ...
 $ V1  : num  28.4 21.2 21.6 21.6 21.7 ...
 $ V2  : num  8.9 6.69 6.93 6.93 6.93 ...
 $ V3  : num  26.8 20.1 20.5 20.6 20.7 ...
 $ V4  : num  7.5 5.73 5.92 6.06 6.11 ...
 $ V5  : num  9.97 7.47 7.64 7.75 7.8 ...
 $ V6  : num  10.28 7.73 7.92 7.98 8.03 ...
 $ V7  : num  10.14 7.6 7.78 7.8 7.82 ...
 $ V8  : num  9.27 6.98 7.22 7.29 7.37 ...
 $ V9  : num  12.65 9.47 9.67 9.71 9.77 ...
 $ V10 : num  12.6 9.42 9.62 9.67 9.73 ...
 $ V11 : num  8.91 6.65 6.77 6.8 6.85 ...
 $ V12 : num  28.8 21.5 22 22 22 ...
 $ V13 : num  26.1 19.6 20.1 20.3 20.3 ...
 $ V14 : num  8.54 6.21 6.24 6.07 6.65 ...
 $ V15 : num  25.9 19.3 19.7 19.8 19.7 ...
 $ V16 : num  26.4 19.6 20 20 20.1 ...
```

```
> R=round(cor(data2[,-1]),2) # only independent variable call.
> R
```

```
      V1   V2   V3   V4   V5   V6   V7   V8   V9  V10  V11  V12  V13  V14  V15  V16
V1  1.00 0.53 0.92 0.39 0.57 0.54 0.52 0.47 0.67 0.67 0.69 0.98 0.90 0.44 0.78 0.80
V2  0.53 1.00 0.81 0.98 0.98 0.99 0.99 0.99 0.98 0.98 0.95 0.69 0.83 0.80 0.92 0.90
V3  0.92 0.81 1.00 0.72 0.84 0.82 0.81 0.77 0.90 0.90 0.90 0.98 1.00 0.68 0.96 0.97
V4  0.39 0.98 0.72 1.00 0.95 0.97 0.98 0.97 0.93 0.92 0.87 0.57 0.74 0.79 0.86 0.83
V5  0.57 0.98 0.84 0.95 1.00 1.00 0.99 0.98 0.99 0.99 0.98 0.73 0.84 0.82 0.94 0.93
V6  0.54 0.99 0.82 0.97 1.00 1.00 1.00 0.99 0.99 0.99 0.96 0.71 0.83 0.82 0.93 0.92
V7  0.52 0.99 0.81 0.98 0.99 1.00 1.00 0.99 0.98 0.98 0.95 0.69 0.82 0.82 0.93 0.91
V8  0.47 0.99 0.77 0.97 0.98 0.99 0.99 1.00 0.97 0.96 0.95 0.64 0.78 0.80 0.89 0.87
V9  0.67 0.98 0.90 0.93 0.99 0.99 0.98 0.97 1.00 1.00 0.99 0.81 0.90 0.81 0.97 0.96
V10 0.67 0.98 0.90 0.92 0.99 0.99 0.98 0.96 1.00 1.00 0.99 0.81 0.91 0.80 0.97 0.96
V11 0.69 0.95 0.90 0.87 0.98 0.96 0.95 0.95 0.99 0.99 1.00 0.82 0.89 0.78 0.95 0.95
V12 0.98 0.69 0.98 0.57 0.73 0.71 0.69 0.64 0.81 0.81 0.82 1.00 0.97 0.57 0.89 0.91
V13 0.90 0.83 1.00 0.74 0.84 0.83 0.82 0.78 0.90 0.91 0.89 0.97 1.00 0.68 0.96 0.97
V14 0.44 0.80 0.68 0.79 0.82 0.82 0.82 0.80 0.81 0.80 0.78 0.57 0.68 1.00 0.78 0.77
V15 0.78 0.92 0.96 0.86 0.94 0.93 0.93 0.89 0.97 0.97 0.95 0.89 0.96 0.78 1.00 0.99
V16 0.80 0.90 0.97 0.83 0.93 0.92 0.91 0.87 0.96 0.96 0.95 0.91 0.97 0.77 0.99 1.00
```

```
> a=as.matrix((data2[,-1]))
> head(a)
```

```
        V1       V2       V3       V4       V5        V6        V7       V8
1 28.41648 8.901867 26.83137 7.504736 9.967638 10.279682 10.136787 9.270525
2 21.19137 6.694220 20.09044 5.729967 7.467530  7.728236  7.601237 6.975190
3 21.59932 6.925384 20.54287 5.922852 7.644968  7.917646  7.775695 7.219916
4 21.62791 6.925032 20.61553 6.060823 7.750806  7.975255  7.797863 7.293756
5 21.70954 6.933306 20.65903 6.105122 7.797439  8.028052  7.817210 7.369007
6 21.56899 7.040979 20.65161 6.183656 7.842963  8.016074  7.840125 7.378075
         V9       V10      V11      V12      V13      V14      V15      V16
1 12.654857 12.598189 8.905399 28.82103 26.14988 8.541110 25.86272 26.37429
2  9.466332  9.422955 6.646366 21.50911 19.57376 6.205773 19.28430 19.56499
3  9.669392  9.623311 6.774943 21.95184 20.13573 6.235237 19.66877 20.02866
4  9.713977  9.668113 6.803638 21.98513 20.29207 6.068567 19.75101 20.04995
5  9.772946  9.727284 6.847047 22.03346 20.28299 6.645792 19.66170 20.07477
6  9.716321  9.664423 6.766672 22.01921 20.36700 6.219329 19.53202 19.94914
```

```
> dim(a)
```

[1] 59 16

Principal Component command “princomp”

```
> pc<-princomp(a)
> summary(pc)
```

```
Importance of components:
                          Comp.1     Comp.2     Comp.3      Comp.4     Comp.5
Standard deviation     3.597098 1.17756593 0.60614159 0.255427817 0.22028116
Proportion of Variance 0.872458 0.09349975 0.02477354 0.004399227 0.00327186
Cumulative Proportion  0.872458 0.96595777 0.99073131 0.995130541 0.99840240
                           Comp.6       Comp.7       Comp.8       Comp.9
Standard deviation     0.1041100962 0.0655955129 0.0562893554 0.0489055408
Proportion of Variance 0.0007308456 0.0002901271 0.0002136449 0.0001612709
Cumulative Proportion  0.9991332460 0.9994233730 0.9996370179 0.9997982888
                          Comp.10      Comp.11      Comp.12      Comp.13
Standard deviation     3.633910e-02 3.022681e-02 1.851103e-02 1.699034e-02
Proportion of Variance 8.904065e-05 6.160622e-05 2.310474e-05 1.946454e-05
Cumulative Proportion  9.998873e-01 9.999489e-01 9.999720e-01 9.999915e-01
                          Comp.14      Comp.15      Comp.16
Standard deviation     9.620437e-03 5.679942e-03 1.082555e-03
Proportion of Variance 6.240647e-06 2.175343e-06 7.902055e-08
Cumulative Proportion  9.999977e-01 9.999999e-01 1.000000e+00
```

```
>screeplot(pc,type = "line") # screeplot of principal component, and select PC’s
```

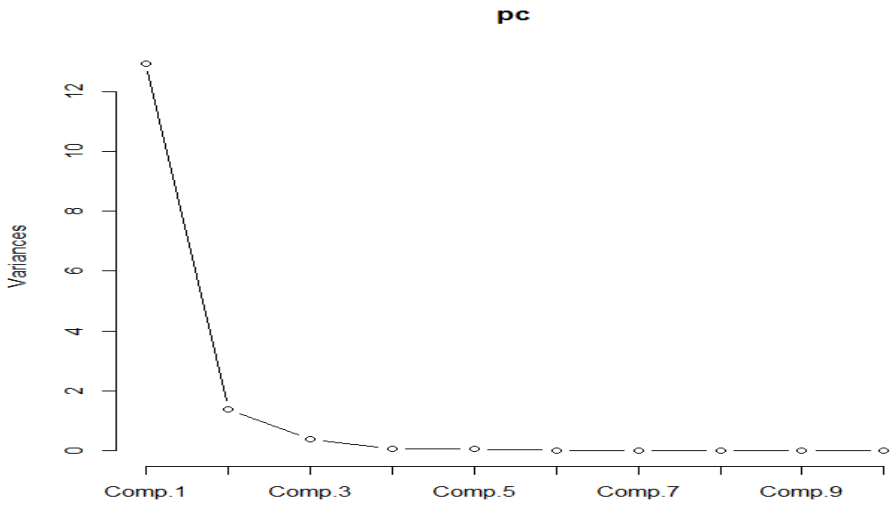


Fig Scree plot for pc

We see the Scree plot the first three component select. Because the straight line start the third line.

Call the eigen value by using R (correlation matrix)

```
> E=eigen(R)
> head(E)
```

```
> head(E)
$values
 [1] 13.9987996549  1.5403982482  0.3119358327  0.0971847981  0.0397756212
 [6]  0.0165550834  0.0141048632  0.0110459094  0.0068984524  0.0038822155
[11] -0.0004635094 -0.0031041970 -0.0045205603 -0.0085686519 -0.0107496968
[16] -0.0131740636

$vectors
             [,1]        [,2]          [,3]        [,4]         [,5]         [,6]
 [1,] -0.1921632 -0.55720355 -0.0396417912 -0.07858384  0.303898265  0.303214781
 [2,] -0.2574918  0.19198691  0.1400755057  0.12746051  0.396361611 -0.212794132
 [3,] -0.2491186 -0.29124168 -0.0128930092  0.12854694  0.027481212 -0.189156128
 [4,] -0.2427372  0.30308825  0.1213999577  0.54209218  0.176875644  0.222516967
 [5,] -0.2610823  0.15218781  0.0648879816 -0.23368118 -0.278817812  0.182078724
 [6,] -0.2599578  0.18328960  0.0839547666 -0.04670208 -0.139524379  0.357493212
 [7,] -0.2582351  0.20063109  0.0812136362  0.09558472 -0.140914447  0.213859839
 [8,] -0.2523807  0.24531022  0.1394296089 -0.15291845  0.290220537 -0.274204178
 [9,] -0.2663439  0.05852127  0.0672258416 -0.14621670  0.109878171  0.208384890
[10,] -0.2660131  0.05200004  0.0898604592 -0.14986887  0.001118251 -0.003941390
[11,] -0.2621290  0.02083094  0.0918567130 -0.61047405  0.046600373 -0.272463762
[12,] -0.2268438 -0.42616878  0.0008608973 -0.04724351  0.047025720  0.329081473
[13,] -0.2499500 -0.27052544  0.0160865204  0.32405962  0.142112193 -0.400809619
[14,] -0.2178543  0.18560549 -0.9507978762 -0.02962061  0.107615351  0.002385519
[15,] -0.2633290 -0.08829330 -0.0154959458  0.20862203 -0.382835261 -0.033880565
[16,] -0.2617573 -0.12322040 -0.0325570816  0.09856973 -0.567815630 -0.328382010
```

```
             [,7]         [,8]         [,9]        [,10]        [,11]
 [1,] -0.17107008  0.079689232  0.039609928  0.158152208  0.074975748
 [2,]  0.15549765  0.345700093  0.043949327  0.484623676  0.172837471
 [3,] -0.16307997 -0.270975629  0.342528507 -0.272320131  0.020267890
 [4,] -0.08555259 -0.216803457  0.342948812  0.048696813  0.127148946
 [5,]  0.07993500 -0.100842494  0.547888961 -0.073458264 -0.184975659
 [6,]  0.31663416 -0.123962600 -0.218545097 -0.008190975 -0.344078530
 [7,] -0.13271933  0.033605015 -0.340270260 -0.371656212  0.634421564
 [8,] -0.35171960 -0.327171132 -0.405624488 -0.124906350 -0.321667545
 [9,] -0.11222233 -0.033548021 -0.057590211  0.317248322 -0.211626857
[10,]  0.42613361  0.458144380 -0.020046803 -0.273296904 -0.032970409
[11,] -0.10991576 -0.001078605  0.211039774 -0.035139801  0.307100802
[12,]  0.13354986 -0.160517338 -0.234586892  0.023934311  0.068699090
[13,]  0.34506038  0.001185226 -0.043591586 -0.314843251 -0.193892288
[14,]  0.02298889  0.002040845 -0.006815051 -0.009371329  0.008907204
[15,] -0.54424967  0.551035407 -0.037173508 -0.009884403 -0.271905893
[16,]  0.15738389 -0.282079861 -0.163354377  0.477438840  0.181250187
             [,12]        [,13]        [,14]        [,15]        [,16]
 [1,] -0.119294947  0.56767137  0.145554147  0.186702224 -0.01461777
 [2,] -0.240053109 -0.02290646 -0.223950033 -0.326055455 -0.18335566
 [3,]  0.350690483  0.08962624 -0.476874517 -0.378521214 -0.07918588
 [4,]  0.048038308 -0.09362593  0.011540213  0.435333743  0.26195081
 [5,] -0.438776126  0.06946567  0.288182577 -0.319038173  0.02068231
 [6,]  0.022378077  0.17104152 -0.405116979  0.232639885 -0.46096512
 [7,] -0.008724555  0.10611446  0.186871890 -0.221046046 -0.20364510
 [8,] -0.223346020  0.15971628 -0.003127668 -0.054657017  0.28486125
 [9,]  0.633766348 -0.21079608  0.425063270 -0.216608410 -0.08988299
[10,]  0.263583196  0.17878753 -0.080215264  0.017530404  0.56580166
[11,]  0.042529130 -0.23826044 -0.069392120  0.474564254 -0.18044594
[12,] -0.267312287 -0.61196367 -0.168851344 -0.101712486  0.25582596
[13,] -0.078515989 -0.12600275  0.432571335  0.136128105 -0.30470364
[14,] -0.018120421 -0.01597948 -0.008796797  0.007143505  0.01228742
[15,] -0.086630403 -0.15701305 -0.096241433  0.113449586 -0.04285852
[16,]  0.038320864  0.20278783  0.046053981  0.063600683  0.18626395
```

```
> end(E)
```

[1] 2 1

```
> EV=round(E$values,3)
> EV
```

[1] 13.999 1.540 0.312 0.097 0.040 0.017 0.014 0.011 0.007 0.004 0.000

[12] -0.003 -0.005 -0.009 -0.011 -0.013

```
>end(EV)
```

[1] 16 1

Percentage of Variance

```
>for(i in 1:3)
+ {
+   var=(EV/(sum(EV)))*100
+ }
> var
```

[1] 87.49375 9.62500 1.95000 0.60625 0.25000 0.10625 0.08750 0.06875
[9] 0.04375 0.02500 0.00000 -0.01875 -0.03125 -0.05625 -0.06875 -0.08125

Cumulative Variance

```
> CV=cumsum(var)
> CV
```

[1] 87.49375 97.11875 99.06875 99.67500 99.92500 100.03125 100.11875
[8] 100.18750 100.23125 100.25625 100.25625 100.23750 100.20625 100.15000
[15] 100.08125 100.00000

Wight or eigenvector

```
>wt=E$vectors
> head(wt)
```

```
> head(wt)
           [,1]       [,2]        [,3]        [,4]       [,5]       [,6]
[1,] -0.1921632 -0.5572035 -0.03964179 -0.07858384  0.30389826  0.3032148
[2,] -0.2574918  0.1919869  0.14007551  0.12746051  0.39636161 -0.2127941
[3,] -0.2491186 -0.2912417 -0.01289301  0.12854694  0.02748121 -0.1891561
[4,] -0.2427372  0.3030882  0.12139996  0.54209218  0.17687564  0.2225170
[5,] -0.2610823  0.1521878  0.06488798 -0.23368118 -0.27881781  0.1820787
[6,] -0.2599578  0.1832896  0.08395477 -0.04670208 -0.13952438  0.3574932
            [,7]        [,8]        [,9]        [,10]       [,11]       [,12]
[1,] -0.17107008  0.07968923  0.03960993  0.158152208  0.07497575 -0.11929495
[2,]  0.15549765  0.34570009  0.04394933  0.484623676  0.17283747 -0.24005311
[3,] -0.16307997 -0.27097563  0.34252851 -0.272320131  0.02026789  0.35069048
[4,] -0.08555259 -0.21680346  0.34294881  0.048696813  0.12714895  0.04803831
[5,]  0.07993500 -0.10084249  0.54788896 -0.073458264 -0.18497566 -0.43877613
[6,]  0.31663416 -0.12396260 -0.21854510 -0.008190975 -0.34407853  0.02237808
           [,13]       [,14]      [,15]       [,16]
[1,]  0.56767137  0.14555415  0.1867022 -0.01461777
[2,] -0.02290646 -0.22395003 -0.3260555 -0.18335566
[3,]  0.08962624 -0.47687452 -0.3785212 -0.07918588
[4,] -0.09362593  0.01154021  0.4353337  0.26195081
[5,]  0.06946567  0.28818258 -0.3190382  0.02068231
[6,]  0.17104152 -0.40511698  0.2326399 -0.46096512
```

```
> dim(wt)
```

[1] 16 16

Create the component array.

```
> comp=c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16)
> comp
```

[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Create the Data Frame.

```
> df=data.frame(comp,round(EV,3),round(var,3),round(CV,3))
> df
```

```
> df
   comp round.EV..3. round.var..3. round.CV..3.
1     1       13.999         87.494        87.494
2     2        1.540          9.625        97.119
3     3        0.312          1.950        99.069
4     4        0.097          0.606        99.675
5     5        0.040          0.250        99.925
6     6        0.017          0.106       100.031
7     7        0.014          0.088       100.119
8     8        0.011          0.069       100.188
9     9        0.007          0.044       100.231
10   10        0.004          0.025       100.256
11   11        0.000          0.000       100.256
12   12       -0.003         -0.019       100.238
13   13       -0.005         -0.031       100.206
14   14       -0.009         -0.056       100.150
15   15       -0.011         -0.069       100.081
16   16       -0.013         -0.081       100.000
```

Interpretation: It is seen that the first component explains 87.494 percent variation of the data set. Second component explains 9.625 percent of the variation. The first three components explain around 99.069 per cent variation of the data set.

Principal Component Regression Model

```
>wt=data.frame(wt)
> head(wt)
```

```
> head(wt)
           [,1]       [,2]        [,3]        [,4]       [,5]       [,6]
[1,] -0.1921632 -0.5572035 -0.03964179 -0.07858384  0.30389826  0.3032148
[2,] -0.2574918  0.1919869  0.14007551  0.12746051  0.39636161 -0.2127941
[3,] -0.2491186 -0.2912417 -0.01289301  0.12854694  0.02748121 -0.1891561
[4,] -0.2427372  0.3030882  0.12139996  0.54209218  0.17687564  0.2225170
[5,] -0.2610823  0.1521878  0.06488798 -0.23368118 -0.27881781  0.1820787
[6,] -0.2599578  0.1832896  0.08395477 -0.04670208 -0.13952438  0.3574932
            [,7]        [,8]        [,9]        [,10]       [,11]       [,12]
[1,] -0.17107008  0.07968923  0.03960993  0.158152208  0.07497575 -0.11929495
[2,]  0.15549765  0.34570009  0.04394933  0.484623676  0.17283747 -0.24005311
[3,] -0.16307997 -0.27097563  0.34252851 -0.272320131  0.02026789  0.35069048
[4,] -0.08555259 -0.21680346  0.34294881  0.048696813  0.12714895  0.04803831
[5,]  0.07993500 -0.10084249  0.54788896 -0.073458264 -0.18497566 -0.43877613
[6,]  0.31663416 -0.12396260 -0.21854510 -0.008190975 -0.34407853  0.02237808
           [,13]       [,14]      [,15]       [,16]
[1,]  0.56767137  0.14555415  0.1867022 -0.01461777
[2,] -0.02290646 -0.22395003 -0.3260555 -0.18335566
[3,]  0.08962624 -0.47687452 -0.3785212 -0.07918588
[4,] -0.09362593  0.01154021  0.4353337  0.26195081
[5,]  0.06946567  0.28818258 -0.3190382  0.02068231
[6,]  0.17104152 -0.40511698  0.2326399 -0.46096512
```

```
> dim(a)
```

[1] 59 16

```
> Z=a%*%as.matrix(wt[1:16,1:2])
> head(Z)
```

```
> head(z)
         X1        X2
1 -64.19568 -33.67923
2 -47.96681 -25.08903
3 -49.07764 -25.58985
4 -49.27047 -25.62772
5 -49.52001 -25.53049
6 -49.35987 -25.47112
```

```
> dim(Z)
```

[1] 59 2

```
> Y=data2[,1]
> end(Y)
```

[1] 59 1

```
> data3=data.frame(Y,Z)
> str(data3)
```

'data.frame': 59 obs. of 3 variables:

$ Y : num 27.8 21.7 22.1 22.2 22.2 ...

$ X1: num -64.2 -48 -49.1 -49.3 -49.5 ...

$ X2: num -33.7 -25.1 -25.6 -25.6 -25.5 ...

The generate the data y of original variable of GDP and x1 and x2 are the z of the two components.

```
> head(data3)
```

```
> head(data3)
         Y         X1         X2
1 27.80198 -64.19568 -33.67923
2 21.67609 -47.96681 -25.08903
3 22.13241 -49.07764 -25.58985
4 22.18339 -49.27047 -25.62772
5 22.23996 -49.52001 -25.53049
6 22.19470 -49.35987 -25.47112
```

Fit the model with the given dataset data3 with respective components.

```
>lm.fit=lm(Y~.,data=data3)
>lm.fit
```

Call:

lm(formula = Y ~ ., data = data3)

Coefficients:

(Intercept) X1 X2

3.9627 -0.2293 -0.2713

Summary model

```
> sum=summary(lm.fit)   ## from here we can see the significance of the PC's
```

```
> sum
```

```
Call:
lm(formula = Y ~ ., data = data3)

Residuals:
     Min       1Q   Median       3Q      Max
-0.09933 -0.01142  0.00395  0.01905  0.04095

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.962663   0.093246    42.5   <2e-16 ***
X1          -0.229284   0.001049  -218.5   <2e-16 ***
X2          -0.271277   0.003143   -86.3   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.02832 on 56 degrees of freedom
Multiple R-squared:  0.9991,    Adjusted R-squared:  0.999
F-statistic: 2.953e+04 on 2 and 56 DF,  p-value: < 2.2e-16
```

Interpretation: The above principal component regress with the original dependent variable GDP the model is good. The entire components are significant effect of the given data.

Now, we estimate the coefficient of betas of original 16 variable of GDP factors to help of principal component regression model.

```
> betas=t(as.vector(sum$coefficients[,1]))
> intercept=betas[1];intercept
```

[1] 3.962663

```
> beta=betas[-1];beta
```

[1] -0.2292841 -0.2712765

```
> beta=as.matrix(beta)
>betan=Wt[,1:2]%*%(beta)
>betan
```

[,1]

[1,] 0.1952162068

[2,] 0.0069572380

[3,] 0.1361259687

[4,] -0.0265649391

[5,] 0.0185770422

[6,] 0.0098820345

[7,] 0.0047826913

[8,] -0.0086800145

[9,] 0.0451929684

[10,] 0.0468861877

[11,] 0.0544510661

[12,] 0.1676212602

[13,] 0.1306967470

[14,] -0.0003998758

[15,] 0.0843290563

[16,] 0.0934435963

```
> betan1=append(intercept,betan);betan1
```

```
> betan1=append(intercept,betan);betan1
 [1]  3.9626630602  0.1952162068  0.0069572380  0.1361259687 -0.0265649391
 [6]  0.0185770422  0.0098820345  0.0047826913 -0.0086800145  0.0451929684
[11]  0.0468861877  0.0544510661  0.1676212602  0.1306967470 -0.0003998758
[16]  0.0843290563  0.0934435963
```

The Our Model is given by-

$\widehat{GDP} = \beta_1 + X_2\beta_2 + X_3\beta_3 + \ldots\ldots\ldots + X_{17} + \mu$

$\widehat{GDP}$ = 3.9626 + (0.1952) Agriculture + (0.006957) Mining & quarrying +……

……. + (0.08432) Export + (0.09344) Less Import.

The model to complete with coefficient of original variables x's are above the fit the model in the linear form of classical linear regression model. GDP of GDP factors.

ARIMA (Annually)

Forecasting Annual GDP

Annual GDP is the average amount of total GDP that a place generally receives. Then we say annual GDP of India in crore rupees.

1. **Import the data set**

```
> ## GDP Annual forecast ###
>data=read.csv(file.choose(),header = T)
>head(data)
```

```
    Year GDP.Growth
1 1952-53        2.3
2 1953-54        2.8
3 1954-55        6.1
4 1955-56        4.2
5 1956-57        2.6
6 1957-58        5.7
```

We can see the above data extract head part and one variable is year and with respective GDP growth.

2. **Data Structure**

```
>str(data)
```

```
'data.frame':          68 obs. of  2 variables:
 $ Year      : chr  "1952-53" "1953-54" "1954-55" "1955-56" ...
 $ GDP.Growth: num  2.3 2.8 6.1 4.2 2.6 5.7 1.2 7.6 2.2 7.1 ...
```

```
>gdp=data[,2];gdp
```

```
 [1]  2.3  2.8  6.1  4.2  2.6  5.7  1.2  7.6  2.2  7.1  3.1  2.1  5.1  7.6  3.7  1.0  8.1
[18]  2.6  6.5  5.0  1.0  0.3  4.6  1.2  9.0  1.2  7.5  5.5  5.3  6.0  3.5  7.5  3.8  5.3
```

```
[35]  4.8  4.0  9.6  5.9  5.5  1.1  5.5  4.8  6.7  7.6  7.6  4.1  6.2  8.5  4.0  4.9  3.9
[52]  7.9  7.8  9.3  9.3  9.8  3.9  8.5 10.3  6.6  5.5  6.4  7.4  8.0  8.3  7.0  6.1  4.2
```

3. To convert GDP rate in Time series data

```
>tsdata=ts(gdp,frequency = 1,start = c(1953))
>tsdata
```

```
Time Series:
Start = 1953
End = 2020
Frequency = 1
 [1]  2.3  2.8  6.1  4.2  2.6  5.7  1.2  7.6  2.2  7.1  3.1  2.1  5.1  7.6  3.7  1.0  8.1
[18]  2.6  6.5  5.0  1.0  0.3  4.6  1.2  9.0  1.2  7.5  5.5  5.3  6.0  3.5  7.5  3.8  5.3
[35]  4.8  4.0  9.6  5.9  5.5  1.1  5.5  4.8  6.7  7.6  7.6  4.1  6.2  8.5  4.0  4.9  3.9
[52]  7.9  7.8  9.3  9.3  9.8  3.9  8.5 10.3  6.6  5.5  6.4  7.4  8.0  8.3  7.0  6.1  4.2
```

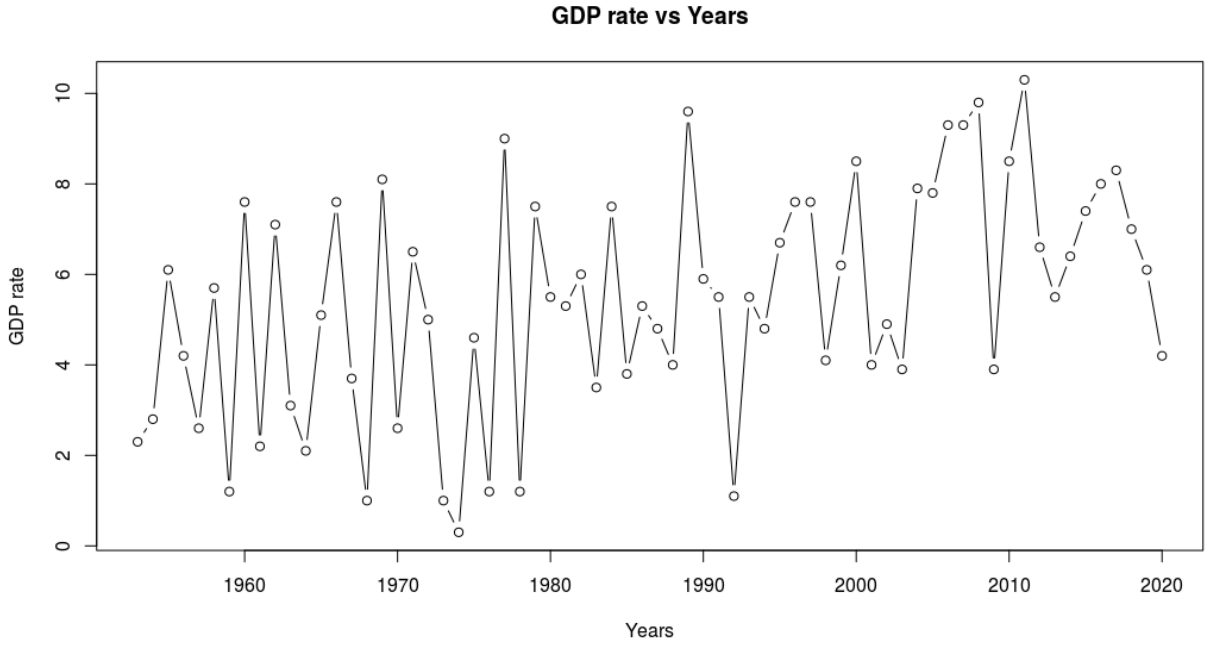
```
>attributes(tsdata)
```

```
$tsp
[1] 1953 2020    1

$class
[1] "ts"
```

The time series data starting from 1953 to ending year is 2020 with respective GDP rate.

4. Plot the Annual GDP rate again respective years

```
>plot(tsdata,type = 'b',xlab='Years',ylab='GDP rate',main='GDP rate vs Years')
```



GDP growth again Years plot, In the plot are look like stationary data. And we check it stationary or not.

5. Box-Ljung test

The Ljung–Box test may be defined as:

H_0: The data are independently distributed (i.e. the correlations in the population from which the sample is taken are 0, so that any observed correlations in the data result from randomness of the sampling process).

H_a: The data are not independently distributed; they exhibit serial correlation.

```
>Box.test(tsdata,type = 'Ljung-Box')
```

Box-Ljung test

data: tsdata
X-squared = 0.27949, df = 1, p-value = 0.597

Conclusion: The above test result the data are no serial correlation.

6. ADF test

```
>adf.test(tsdata)
```

```
        Augmented Dickey-Fuller Test

data:  tsdata
Dickey-Fuller = -3.7546, Lag order = 4, p-value = 0.02684
alternative hypothesis: stationary
```

Conclusion: The above ADF test result, the data is stationary.

7. ARIMA Model

The ARIMA Model is Autoregressive Integrative Moving Average. There are three parameters. One is p and p for AR, second is d , and d for differentiation and last third parameter is q and q for moving average.

```
> #ARIMA model
>library(forecast)
>model=auto.arima(gdp)
>summary(model)
```

```
Series: gdp
ARIMA(1,1,1)

Coefficients:
         ar1      ma1
      -0.2115  -0.8550
s.e.   0.1292   0.0622

sigma^2 estimated as 5.484:  log likelihood=-151.91
AIC=309.82   AICc=310.2   BIC=316.43

Training set error measures:
                    ME     RMSE      MAE       MPE     MAPE      MASE        ACF1
Training set 0.4128692 2.289473 1.863669 -42.60053 72.93265 0.6812101 -0.03340509
```

Here to see the ARIMA model with run with respective parameters ARIMA(1,1,1). The summary of the ARIMA model is given by with respective accuracy.

```
>attributes(model)
```

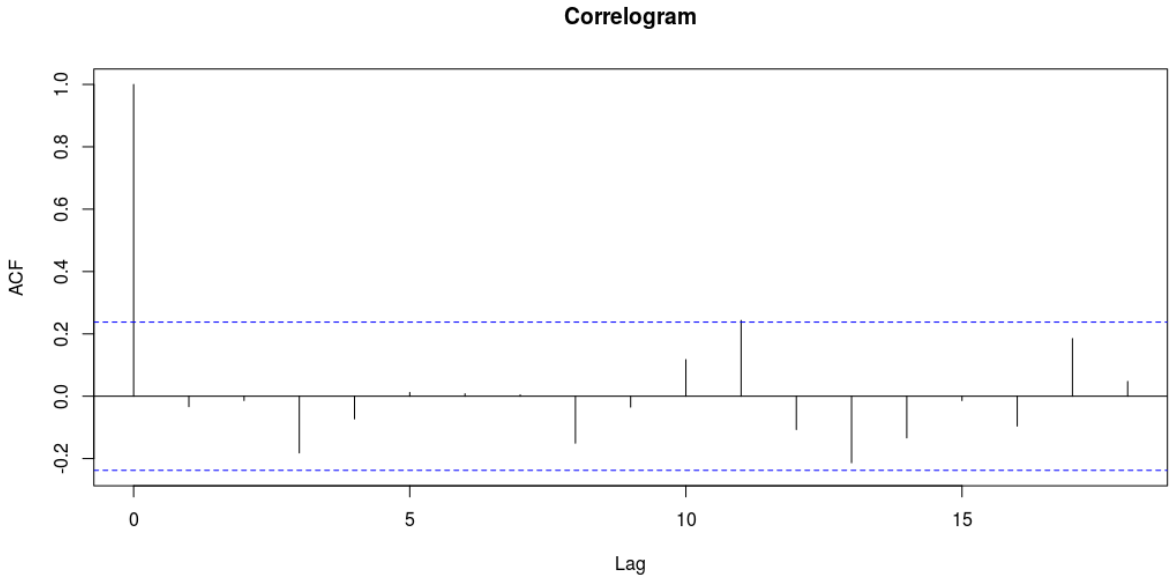
```
$names
 [1] "coef"      "sigma2"    "var.coef"  "mask"      "loglik"    "aic"       "arma"
 [8] "residuals" "call"      "series"    "code"      "n.cond"    "nobs"      "model"
[15] "bic"       "aicc"      "x"         "fitted"

$class
```

[1] "forecast_ARIMA" "ARIMA" "Arima"

There are several attributes and classes.

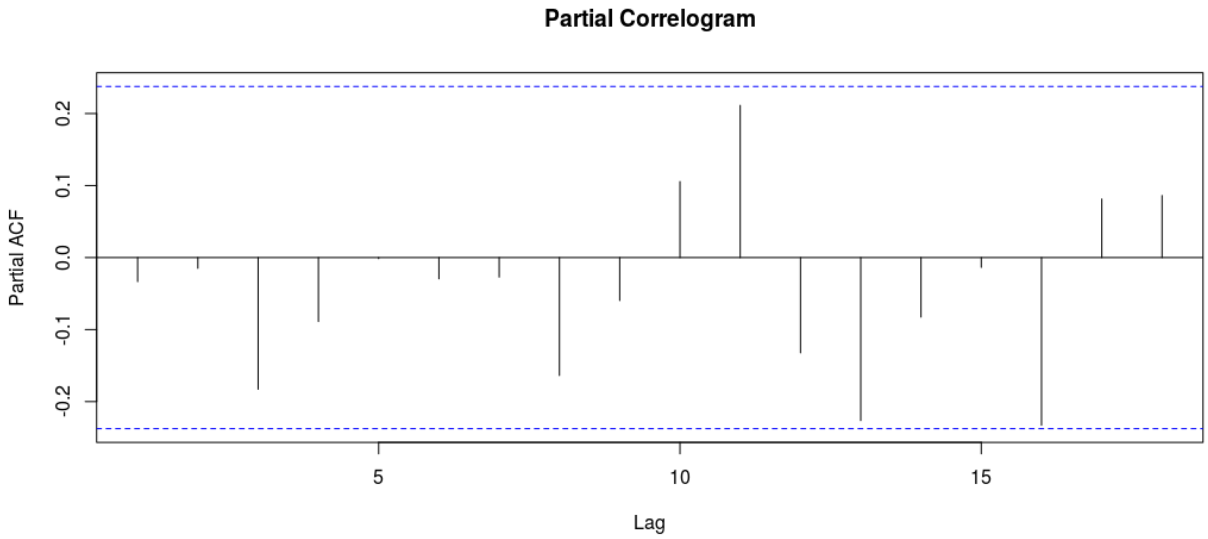
8. ACF & PACF plots

```
> #ACF and PACF plot
>acf(model$residuals,main='Correlogram')
```



ACF plot we can see the first lag are outside the line. And other remaining under the line. That is the lag value is one in ACF plot.

```
>pacf(model$residuals, main='Partial Correlogram')
```



The Partial Auto correlation Function plot in all the line between the lines. Its good.

```
> #Ljung-Box text
>Box.test(model$residuals,lag=20,type='Ljung-Box')
```

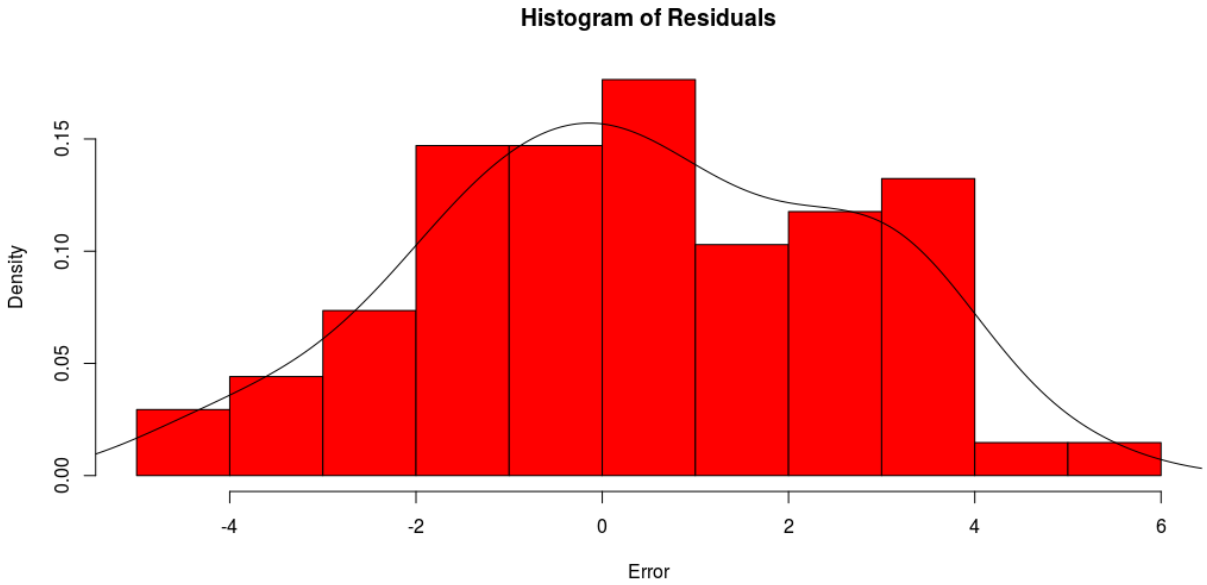
Box-Ljung test

data: model$residuals
X-squared = 21.709, df = 20, p-value = 0.3565

Conclusion: The above residual Ljung-Box test we conclude there is no serial correlation.

9. Residual plot

```
> #Residual Plot
>hist(model$residuals,
+      col = 'red',
+      xlab='Error',
+      main = 'Histogram of Residuals',
+      freq = F)
>lines(density(model$residuals))
```



Histogram of residuals plot its look like the normal shape with approximately mean zero value with bell shape.

10. Forecast

```
> #Forecast
>Annual_Forecast=forecast(model,10)
>Annual_Forecast
```

	Point Forecast	Lo 80	Hi 80	Lo 95	Hi 95
69	7.338658	4.337635	10.339681	2.748990	11.92833
70	6.674716	3.667053	9.682378	2.074893	11.27454
71	6.815164	3.769858	9.860470	2.157771	11.47256
72	6.785454	3.721874	9.849034	2.100113	11.47080
73	6.791739	3.706562	9.876916	2.073368	11.51011
74	6.790410	3.684531	9.896288	2.040378	11.54044
75	6.790691	3.664092	9.917290	2.008970	11.57241
76	6.790631	3.643481	9.937781	1.977481	11.60378
77	6.790644	3.623069	9.958218	1.946257	11.63503
78	6.790641	3.602774	9.978508	1.915220	11.66606

```
>tail(data)
```

	Year	GDP.Growth
63	2014-15	7.4
64	2015-16	8.0
65	2016-17	8.3
66	2017-18	7.0
67	2018-19	6.1

68 2019-20 4.2

The above figure forecast of annual GDP growth for the next ten years we can see the output to next 2021 to 2030 years forecast.

Years	Annual GDP Forecast next 10 Years
2021	7.3387
2022	6.6747
2023	6.8152
2024	6.7855
2025	6.7917
2026	6.7904
2027	6.7907
2028	6.7906
2029	6.7906
2030	6.7906

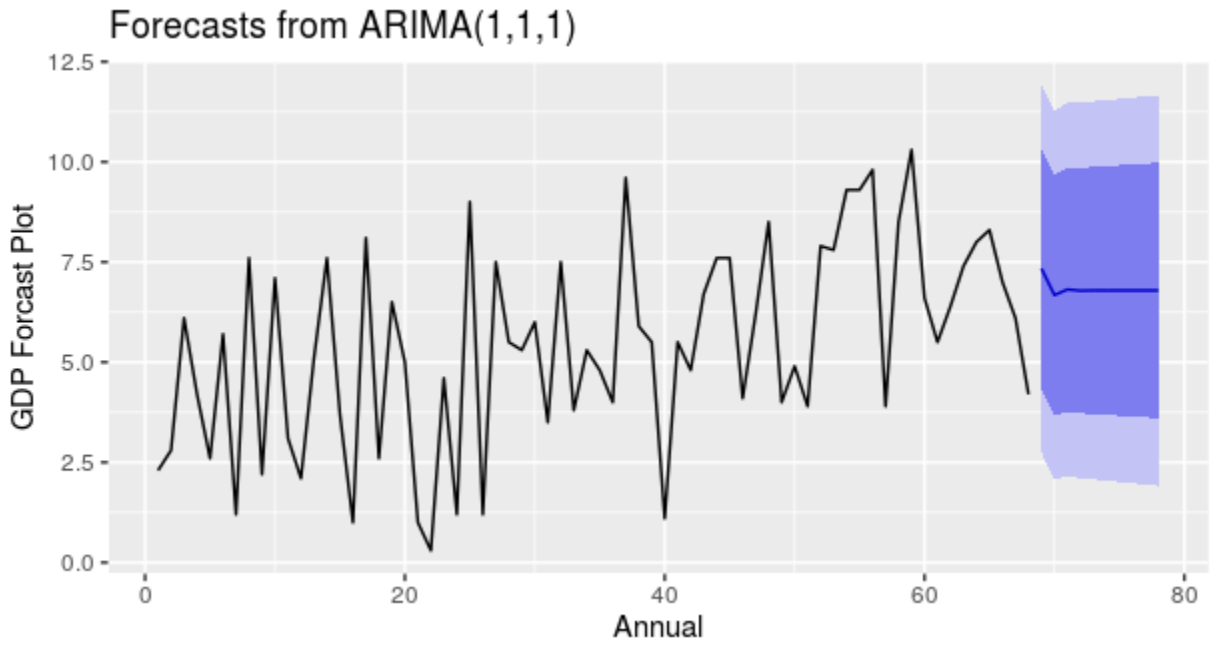
Here we can see the GDP growth slowly down direction. So we can see the forecast graphically how to look like forecast plot.

```
>library(ggplot2)
```

10.1 Forecast Plot

```
>autoplot(Annual_Forecast,
+         type='b',
+         xlab='Annual',
+         ylab='GDP Forcast Plot',
+         col='blue',
+         las=2)
```

Forecast Plot form ARIMA Model with parameters(1,1,1)



The above Forecast GDP plot its annual GDP forecast is slowly downward direction for next ten years forecast values.

11. Model Accuracy

```
>accuracy(Annual_Forecast)
```

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
Training set	0.4128692	2.289473	1.863669	-42.60053	72.93265	0.6812101	-0.03340509

Interpretation: The above time series analysis for GDP forecasting for Annual data. We can see the data is already stationary. And the Annual data are the forecast is pretty good up to next 10 years. GDP starting from 1953 to ending 2020. And the ending of the 2020 GDP rate is 4.2. And next 10 Years forecast in smoothly down GDP rate up to next 10 Years

11.9 ARIMA (Quarterly)

1. **Import the data**

```
> ## GDP quarterly forecast ###
>data=read.csv(file.choose(),header = T)
>head(data)
```

```
    Year Quarter GDP_growth
1 2000-01      Q1        5.1
2            Q2        6.7
3            Q3        4.4
4            Q4        1.8
5 2001-02      Q1        4.6
     6              Q2        5.3
```

2. Data Structure

```
>str(data)
```

```
'data.frame':          80 obs. of  3 variables:
 $ Year      : chr  "2000-01" "" "" "" ...
 $ Quarter   : chr  "Q1" "Q2" "Q3" "Q4" ...
 $ GDP_growth: num  5.1 6.7 4.4 1.8 4.6 5.3 6.8 6.4 5.1 5.4 ...
```

```
>gdp=data[,3]
```

3. To convert GDP rate in to time series data

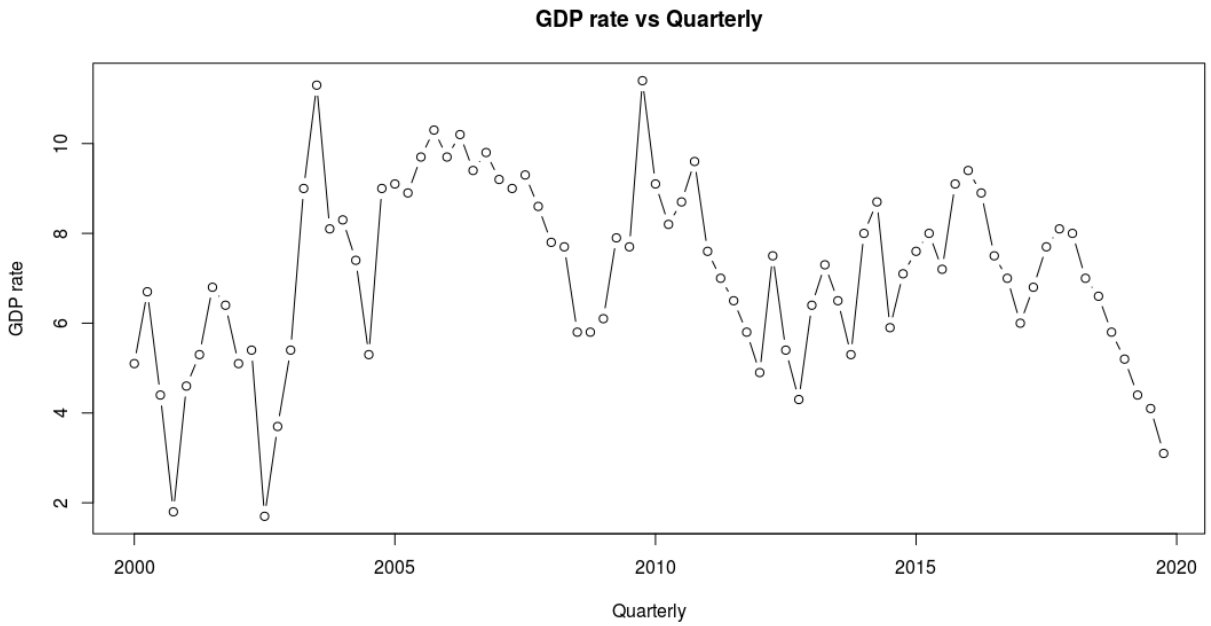
```
>tsdata=ts(gdp,frequency = 4,start = c(2000,1))
>tsdata
```

```
     Qtr1 Qtr2 Qtr3 Qtr4
2001  5.1  6.7  4.4  1.8
2002  4.6  5.3  6.8  6.4
2003  5.1  5.4  1.7  3.7
2004  5.4  9.0 11.3  8.1
2005  8.3  7.4  5.3  9.0
2006  9.1  8.9  9.7 10.3
2007  9.7 10.2  9.4  9.8
2008  9.2  9.0  9.3  8.6
2009  7.8  7.7  5.8  5.8
2010  6.1  7.9  7.7 11.4
2011  9.1  8.2  8.7  9.6
2012  7.6  7.0  6.5  5.8
2013  4.9  7.5  5.4  4.3
2014  6.4  7.3  6.5  5.3
2015  8.0  8.7  5.9  7.1
2016  7.6  8.0  7.2  9.1
2017  9.4  8.9  7.5  7.0
2018  6.0  6.8  7.7  8.1
2019  8.0  7.0  6.6  5.8
2020  5.2  4.4  4.1  3.1
```

Hear we extract the quarterly data up to 20 years with respective quartiles.

4. Plot GDP rate again Quarterly time

```
>plot(tsdata,type = 'b',xlab='Quarterly',ylab='GDP rate',main='GDP rate vs Quarterly')
```



The Quartile GDP rate plot we can see there is no pattern follow, and the above data is stationary or not. To check by using Box-test.

5. Ljung-Box test

```
>Box.test(tsdata,type = 'Ljung-Box')
```

Box-Ljung test

data: tsdata
X-squared = 38.509, df = 1, p-value = 5.449e-10

6. ADF text

```
>adf.test(tsdata)# Null hypothesis is data is non stationary
```

Augmented Dickey-Fuller Test

data: tsdata
Dickey-Fuller = -2.4136, Lag order = 4, p-value = 0.4065
alternative hypothesis: stationary

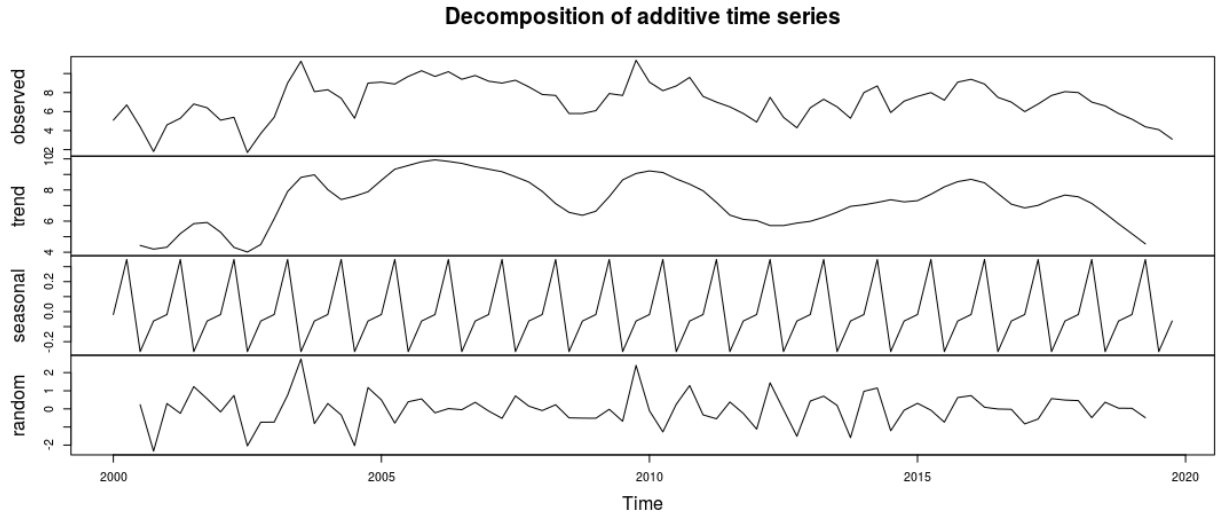
The Above data is non-stationary. To convert stationary data and to decompose the data.

7. Decompose data

```
>decomp=decompose(tsdata)
>decomp$figure
```

[1] -0.01907895 0.34802632 -0.26644737 -0.06250000

```
>plot(decomp$figure,
+     type='b',
+     xlab='Quarterly',
+     ylab='Seasonally Index',
+     col='blue',
+     las=2)
>plot(decomp)
```



Here we decompose of the time series data with additive type, there is present the seasonality and randomness in the data.

8. ARIMA Model

```
> #ARIMA model
>library(forecast)
>model=auto.arima(gdp);model
```

```
Series: gdp
ARIMA(1,0,0) with non-zero mean

Coefficients:
ar1    mean
0.7187  6.9610
s.e.0.0803  0.5509

sigma^2 estimated as 2.065:  log likelihood=-141.87
AIC=289.74   AICc=290.05   BIC=296.88
```

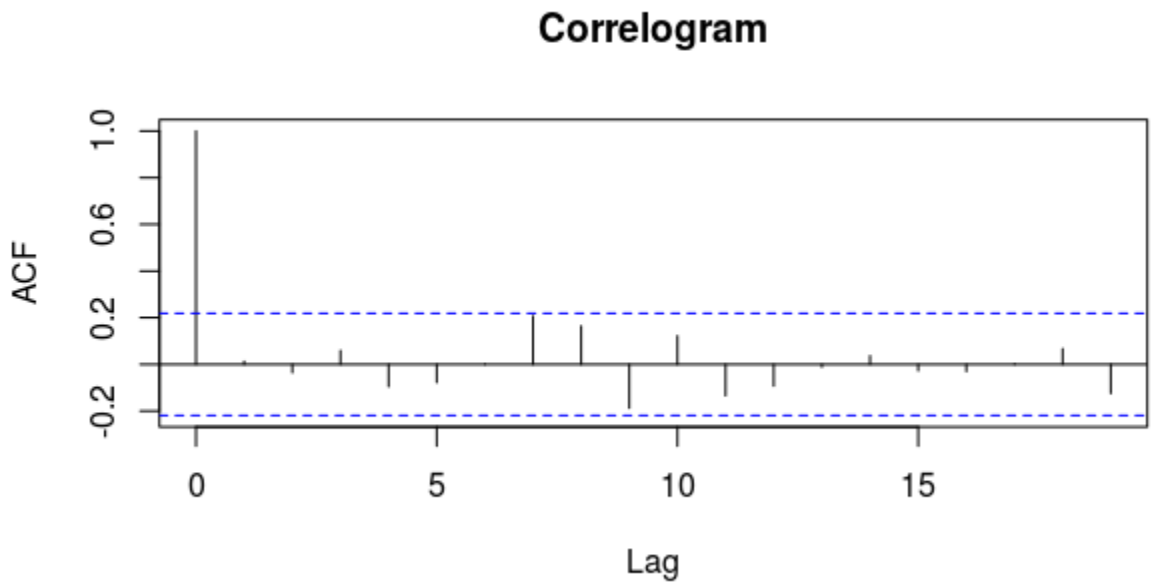
```
>attributes(model)
```

```
$names
 [1] "coef"      "sigma2"    "var.coef"  "mask"      "loglik"    "aic"       "arma"
 [8] "residuals" "call"      "series"    "code"      "n.cond"    "nobs"      "model"
[15] "bic"       "aicc"      "x"         "fitted"

$class
[1] "forecast_ARIMA" "ARIMA"          "Arima"
```

ARIMA model is Auto regressive Integrative Moving Average, and the model coefficient ar1 and ma1 is auto regressive and ma1 is moving average and the model run the parameters is ARIMA(1,0,0) with standard error. And to check the AIC for accuracy.

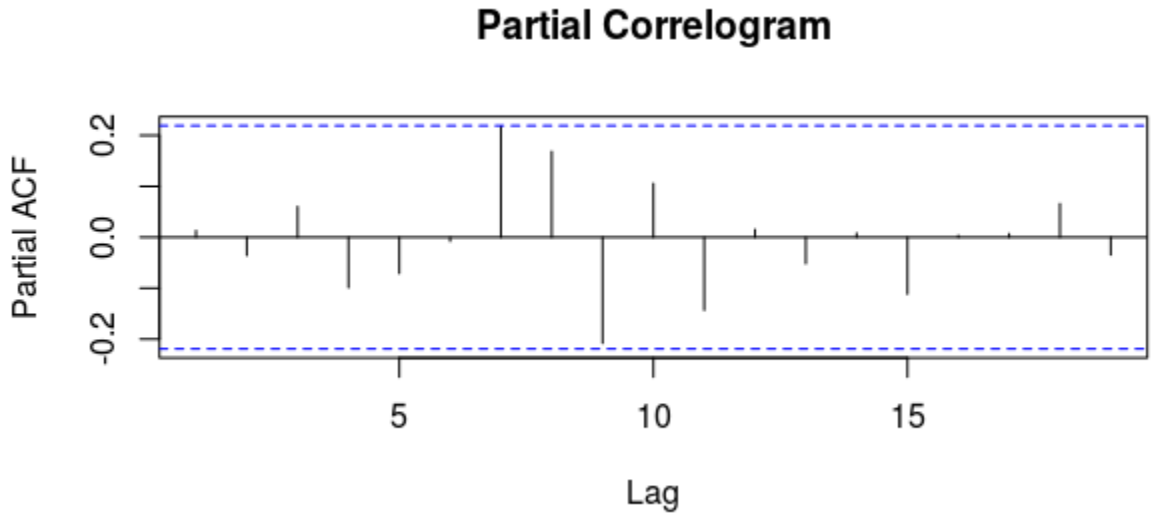
9. ACF & PACF plot

```
> #ACF and PACF plot
>acf(model$residuals,main='Correlogram')
```



The above ACF plot we can see the only one lag the above ACF plot because there only on line cross the line.

```
>pacf(model$residuals ,main='Partial Correlogram')
```




```
> #Ljung-Box text
>Box.test(model$residuals,lag=18,type='Ljung-Box')
```

Box-Ljung test

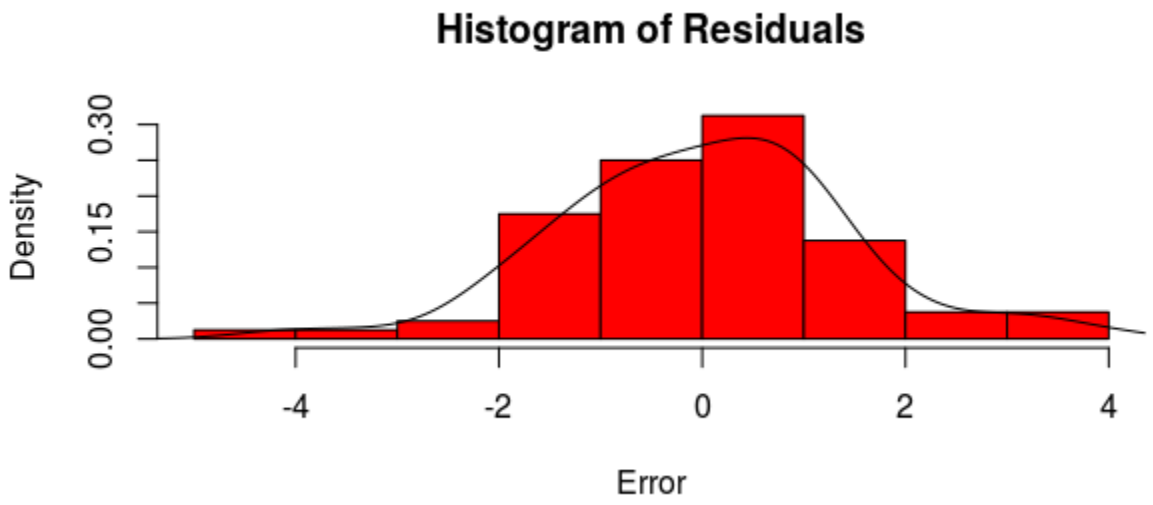
data: model$residuals

X-squared = 16.033, df = 18, p-value = 0.5902

Ljung-Box text we can see the p val is greater than 0.05. That is there is no serial correlation.

10. Residuals Histogram plot

```
> #Residual Plot
>hist(model$residuals,
+      col = 'red',
+      xlab='Error',
+      main = 'Histogram of Residuals',
+      freq = F)
>lines(density(model$residuals))
```



The above Histogram of Residual plot is look like normally shape.

11. Forecasting On Quartile GDP

```
> #Forecast
>Qurterly_Forecast=forecast(model,20)
>Qurterly_Forecast
```

	Point Forecast	Lo 80	Hi 80	Lo 95	Hi 95
81	4.186270	2.344715	6.027825	1.369855	7.002685
82	4.966928	2.699144	7.234712	1.498652	8.435205

83	5.527955	3.068798	7.987112	1.766999	9.288911
84	5.931142	3.378759	8.483525	2.027610	9.834675
85	6.220896	3.621674	8.820118	2.245729	10.196063
86	6.429130	3.806045	9.052216	2.417467	10.440794
87	6.578780	3.943454	9.214106	2.548397	10.609163
88	6.686327	4.044702	9.327952	2.646310	10.726344
89	6.763616	4.118744	9.408489	2.718632	10.808601
90	6.819161	4.172613	9.465710	2.771614	10.866708
91	6.859079	4.211665	9.506493	2.810209	10.907949
92	6.887766	4.239906	9.535627	2.838213	10.937319
93	6.908383	4.260292	9.556474	2.858477	10.958289
94	6.923199	4.274989	9.571409	2.873111	10.973287
95	6.933847	4.285575	9.582118	2.883665	10.984029
96	6.941499	4.293195	9.589802	2.891268	10.991729
97	6.946998	4.298678	9.595318	2.896742	10.997254
98	6.950950	4.302622	9.599279	2.900682	11.001219
99	6.953790	4.305458	9.602123	2.903515	11.004066
100	6.955832	4.307497	9.604167	2.905553	11.006110

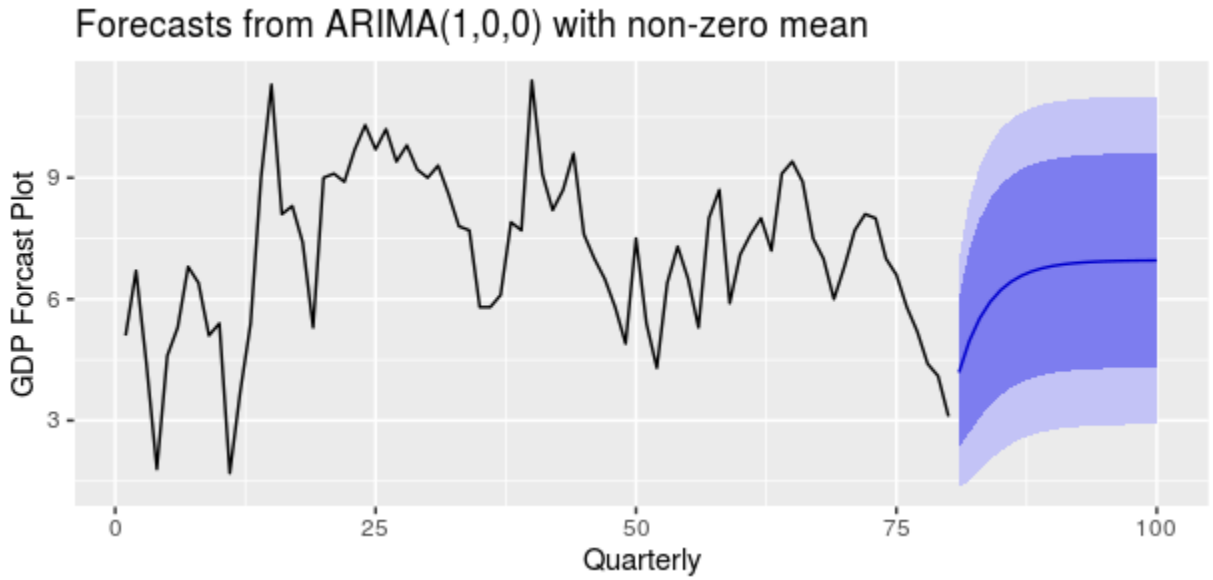
Years	Quarter	Qurtile GDP forecast
2020-21	**Q1**	4.186
	Q2	4.967
	Q3	5.528
	Q4	5.931
2021-22	**Q1**	6.221
	Q2	6.429
	Q3	6.579
	Q4	6.686
2022-23	**Q1**	6.764
	Q2	6.819
	Q3	6.859
	Q4	6.888
2023-24	**Q1**	6.908
	Q2	6.923
	Q3	6.934
	Q4	6.941
2024-2025	**Q1**	6.947
	Q2	6.951
	Q3	6.954
	Q4	6.956

```
>tail(data)
```

```
    Year Quarter GDP_growth
75           Q3        6.6
76           Q4        5.8
77 2019-20   Q1        5.2
78           Q2        4.4
79           Q3        4.1
80           Q4        3.1
```

The above Quartile GDP forecasting is increasing order, it good to known.

```
>library(ggplot2)
>autoplot(Qurterly_Forecast,
+         type='b',
+         xlab='Quarterly',
+         ylab='GDP Forcast Plot',
+         col='blue',
+         las=2)
```



Plot of Quartile GDP Forecasting with respective quartiles.

12. Model Accuracy

```
>accuracy(Qurterly_Forecast)
```

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
Training set	0.02380438	1.418897	1.106555	-6.815382	20.41545	0.9359513	0.01296374

Interpretation: Finally we see the above Quartile GDP forecasting for the next five years with respective quartiles. And look at the forecast plot the value of forecast is increasing direction. So the quartile forecast is best forecast in the above ARIMA Model (1, 0, 0) with non-zero mean.