A PROJECT REPORT ON

"INFOSYS STOCK PRICE PREDICTION USING ARIMA MODEL"

SUBMITED TO

RASHTRASANT TUKADOJI MAHARAJ NAGPUR UNIVERSITY

P.G.T.D. OF STATISTICS

SUBMITTED BY

Ms. Bhagyashri Lakhadive, Ms. Riya Dehariya, Mr. Swapnil Wankhede

Guide	Co-Guide
Ms. MANJIRI GAJARLAWAR	Mrs. VISHAKHA JAISWAL

Post Graduate Teaching Department of Statistics

RTMNU, Nagpur-440033

(2022 – 2023)

POST GRADUATE TEACHING DEPARTMENT OF STATISTICS

R.T.M NAGPUR UNIVERSITY, NAGPUR

CERTIFICATE

This is to certify that the project report entitled **“Infosys Stock Price Prediction using ARIMA Model”** is a bonafide work carried out by **Ms. Bhagyashri Lakhadive, Ms. Riya Dehariya** and **Mr. Swapnil Wankhede** is partial fulfillment for the award of degree of Master of Science (M.Sc.) in Statistics Examination of the Rashtrasant Tukadoji Maharaj University, Nagpur during the year 2022-2023. It has been found to be satisfactory and hereby approved for the submission.

Date :-

Place :- Nagpur

Guide	**Co-Guide**
Ms. Manjiri Gajarlawar	**Mrs. Vishakha Jaiswal**

Forwarded by, **Dr Rajesh Singh,** Professor and Head, Department of Statistics, R.T.M.N.U. Nagpur

DECLARATION

We hereby declare that the work reported in the project entitled **“Infosys Stock Price Prediction using ARIMA Model**” has been carried out under the co- guidance and guidance of **Mrs. Vishakha Jaiswal** and **Ms. Manjiri Gajarlawar,** Assistant Professor, Department of Statistics, RTMNU, Nagpur. The work has not been submitted as a whole or in part of any other University or Institute for the award of degree or diploma or Certificate.

Ms. Bhagyashri Lakhadive

Ms. Riya Dehariya

Mr. Swapnil Wankhede

Date :- Ms. Manjiri Gajarlawar

Place : Nagpur Guide

ACKNOWLEDGMENT

It is our pleasure and privilege to express our sincere gratitude towards our Co-guide Ms. Vishakha Jaiswal and guide Ms. Manjiri Gajarlawar for their continuous and invaluable guidance, helpful suggestions, encouragement and faith in us during the entire course of our project.

We also express our sincere thanks to Dr. Rajesh Singh, Professor and Head of Department of Statistics, RTMNU, Nagpur for their inspiration.

We are thankful to all the teaching and non-teaching staff of Department of Statistics for their timely help and support during the course of our project.

Last but not the least, we would like to thank all those who had helped directly or indirectly towards the completion of this project.

M.Sc. Semester IV

Post Graduate Teaching Department of Statistics,

R.T.M.N.U. Nagpur

CONTENT

Chapter	Particulars	Page No.
1.	**INTRODUCTION**	**1-10**
1.1	Objective	2
1.2	Data Information	2
1.3	Statistical Tools and Software	2
1.4	Company Information	3
1.5	Stock market	5
2.	**REVIEW OF LITERATURE**	**11-14**
2.1	Introduction	12
2.2	Basic research on stock prediction	12
3.	**DATA COLLECTION AND ANALYSIS**	**15-21**
3.1	Introduction	16
3.2	Data Collection	16
3.3	Exploratory Data Analysis (EDA)	17
3.4	Interpretation of Decreasing Trend in Closing Price	19
4.	**TIME SERIES ANALYSIS**	**22-68**
4.1	Introduction	23
4.2	Purpose	23
4.3	Components of time series	24
4.4	Mathematical Models for Time Series	26

4.5	Moving Average Method	30
4.6	Weighted Moving Average	34
4.7	Exponentially weighted moving average	36
4.8	Double Exponentially weighted moving average	38
4.9	Stationarity in Time series	39
4.10	Augmented dickey-fuller test or unit root test	46
4.11	Interpretation of stationarity	52
4.12	Implementation of ARIMA Model	54
4.13	Useful Metrics for study	64
4.14	Interpretation of ARIMA	67
4.15	Diagnosis of ARIMA model	69
5	**Conclusion**	**74**
6	**Limitation of Study**	**76**
7	**Future Scope**	**77**
8	**Challenges**	**78**
	Bibliography	**79**
	Annexure	**81**

ABSTRACT

The main objective is to predict the stock closing price of the January-2023 month of Infosys company with the help of past data of 1 year by using the ARIMA model. The given Bombay Stock Exchange dataset is collected on daily basis from the money control app. Only the closing price of the stock has been considered. It is found that the ARIMA model is able to predict 99% accurate stock closing price of January – 2023 month. Exploratory Data Analysis to know the dataset, Moving Averages to check the smoothness of the trend has been used. Decomposition of Time Series Analysis, Augmented Dickey-Fuller test, Autocorrelation, Partial Autocorrelation, and ARIMA are statistical techniques that have been used with the help of Python language.

Keywords: Stock prediction, ARIMA model, Moving averages, Time series analysis, Python, etc.

Chapter 1

INTRODUCTION

1.1 Objective

The main objective is to predict the stock closing price of the January-2023 month of Infosys Company with the help of past data of 1 year by using the ARIMA model.

1.2 Data Information

Many more things have been explored and analysed with the help of the given dataset which contains 269 observations and 5 variables by considering the time as a variable. The data consist of numerical values except Date. The variables do not include any missing and duplicate values. These variables are Closing price, Opening price, Highest price and Lowest price of stock. All the variables are studied in the data analysis however for the time series analysis, Closing price of the stock is considered only as per the objective of the project. All the detail information are in upcoming chapter.

1.3 Statistical Tools and Software

Python and Excel are mostly used tool while performing exploratory data analysis and time series analysis. Pandas, NumPy, Matplotlib, Seaborn, Sklearn, Scipy are libraries are used to explore, analyse and implement the model.

Histogram, Box Whisker plot, line plot, etc are used to visualize the data for better understanding of the data. After knowing the data well, Closing price of the stock is studied, as the objective is to predict the stock closing price of the January month.

In time series analysis, Moving average, Weighted moving average, Exponentially weighted moving average, Double exponentially weighted moving average is applied, Decomposition of Time Series, Examined Stationarity, Found that series is not stationary, Augmented Dickey fuller test is used to remove the non-stationarity, To know the parameter of model, plotted the Autocorrelation and Partial Autocorrelation graph, Implemented ARIMA model after knowing the data well.

1.4 Company Information

Infosys Limited is an Indian Multinational Company. It deals with Information technology, Consulting and Outsourcing. The company is Headquartered in Bangalore, Karnataka in India. The company was incorporated on 2nd July 1981. It has its operations worldwide. Infosys has its major presence in India, United States, China, Australia, Japan, Middle East and Europe. The company started its journey with a capital of US$250, and today it is a US$ 77.88 billion company. It is the first IT Company from India to be listed on NASDAQ. Employee count stood at 335K with 39.3% women. It is listed on BSE, NSE and NYSE.

Type: Private

Traded as: INFY

Exchange: NSE and BSE

Industry: INDUSTRY- IT Consulting & Software

Founders: Narayan Murthy, Nandan Nilekani, S. Gopalkrishnan, S.D. Shibulal, K. Dinesh, N.S. Raghavan, Ashok Arora

Area served: Worldwide

Key people: Nandan Nilekani(chairman),Salil Parekh (MD & CEO)

1.4.1 Geographical presence:

Infosys has 82 sales and marketing offices and 123 development centres across the world as of 31 March 2018, with major presence in India, United States, China, Australia, Japan, Middle East and Europe. In 2019, 60%, 24%, and 3% of its revenues were derived from projects in North America, Europe, and India, respectively. The remaining 13% of revenues were derived from the rest of the world. In 2022, Infosys's presence in Russia came under scrutiny. Infosys issued a clarification stating that they don't have active relationships with Russian firms. By November, 2022; the only people working there were administrative staff helping with transferring the existing contracts to other contractors.

1.4.2 Services:

The main services include Application Development & Maintenance, Business Process Management, Consulting Services, Incubating Emerging offerings. The main industries served by Infosys include Aerospace & Defence, Agriculture, Agriculture, Automotive, Healthcare, Insurance, Logistics, Media & Entertainment, Mining, Oil & Gas, Travel &Hospitality, Utilities, Waste Management, Financial Services, Education, Communication Services & Chemical Manufacturing.

1.5 Stock market

A stock market is a marketplace where buyers and sellers meet to trade i.e. buy and sell shares of publicly listed companies. A stock market is fondly known as a share market, equity market or share bazaar. It is a platform where you can invest in various financial instruments, including shares, bonds, futures and derivatives.

1.5.1 Stock Exchange in India

A stock exchange is a regulated market for trading. If a company wishes to sell its shares, it should be registered in the stock exchange. Once registered, it can list its shares and sell them at a price to the investor. Investors and traders can connect to exchanges via brokers, who place buy or sell orders on the exchange. Traders can buy and share sells of different companies. The stock exchange offers high liquidity, as the process is transparent and fast. A dividend is paid to investors based on the company’s growth if the company earns profit, the dividend increases. If the company is growing, it attracts more investors, and the company issues more shares. As the demand of shares increases, the price of share also increases. A stock exchange also evaluates the price of the share. Stock exchanges help companies to raise funds. Therefore the company needs to list themselves in the stock exchange. Shares listed on the stock exchange are known as equity and these shareholders are known as Equity Shareholders.

1.5.2 Regulation of Stock Markets in India

The Securities and Exchange Board of India (SEBI) regulates the stock market, the stock exchanges and the Depositories Participants in India. It was constituted in 1992 under the SEBI Act. Along with the overall administrative control of stock markets, SEBI is also entrusted with the role of conducting inspections and formulating rules for the transparent functioning of the stock markets. There are two major stock exchanges in India:

- Bombay Stock Exchange (BSE).
- National Stock Exchange (NSE).

1.5.2.1 BSE (Bombay Stock Exchange)

Bombay Stock Exchange (BSE) is the largest and first securities exchange market in India. It was founded by Premchand Roychand as the Native Shares and Stock Brokers' Association and is now managed by Sethurathnam Ravi. Based in Mumbai, the Bombay Stock Exchange has close to 6,000 companies listed on it and is comparable to stock exchanges in New York, London, Tokyo, and Shanghai.

1.5.2.2 NSE (National Stock Exchange)

NSE is the leading stock exchange and was the first stock exchange that offered a screen-based system for trading. It is the fourth largest stock exchange in the world in terms of equity trading volume as per the World Federation of Exchanges (WFE). It brought transparency to Indian market trading with a fully integrated business model that provides high-quality data and services. NSE has a high trading volume than other stock exchanges. NSE is a good option for investors who take high risks.

The stock market in India operates during a specific time window. The regular market trading hours are from 09:15 AM and close at 03:30 PM. There’s a pre-opening session before 09:15 AM and a post-closing session after 03:30 PM.

Pre-open session : The pre-opening session starts at 09:00 AM and goes on till 09:15 AM. One can start placing the order for any transactions during this period. Pre-open order matching starts immediately after close of pre-open order entry, which means these orders are given preference as soon as the market hours starts as they are cleared of in the beginning.

Regular trading session: This is also known as the continuous trading session, and it runs from 09:15 AM to 03:30 PM. During this session, you can trade freely, place orders to buy or sell stocks, and modify or cancel your buy or sell orders without any limitations. During this window, a bilateral order matching system is followed. This means that each sell order is matched with a buy order that has been placed at the same stock price, and each buy order is matched with a sell order that has been placed at the same stock price.

The Post Closing session: This session begins when the regular trading session comes to a close at 03:30 PM. During this period, you can bid for the following day’s trade as this is post market closing session. If there are adequate number of buyers and sellers, bids placed during this period are confirmed.

1.5.3 Types of Share markets

There are two types of share markets in the country:

1.5.3.1 Primary share market:

This is where companies or businesses register themselves and list for the first time. Companies enter the primary share market to raise funds by offering their stocks to the general public. When a company lists itself in the primary share market and offers to sell its shares for the first time, it is known as Initial Public Offering (IPO). Here, you must understand that shares are a physical representation of a small value of the company, and owning the shares means that you are a part-owner of the company in the proportion of the shares you hold.

1.5.3.2 Secondary share market

After the company lists in the primary market, the actual trading of a company's shares occurs in the secondary share market. After a company's shares are listed on a stock exchange, investors can trade, i.e., sell or purchase the shares through a broker. One can easily open a Demat Account and a Trading Account, following which you can effectively trade in stock markets via broking platforms.

1.5.4 Initial Public Offering

IPO means Initial Public Offering. It is a process by which a privately held company becomes a publicly-traded company by offering its shares to the public for the first time. A private company that has a handful of shareholders shares the ownership by going public by trading its shares. Through the IPO, the company gets its name listed on the stock exchange.

Chapter 2

REVIEW OF LITERATURE

2.1 Introduction

In this chapter, we present a brief review of the work done by various researchers on similar objectives and statistical tools. The ARIMA (Autoregressive Integrated moving Average) models has been widely studied in the financial literature. ARIMA models are a class of time series models that can be used to predict future values based on past observations.

2.2 Basic research on stock prediction

One of the earliest studies on stock prediction using ARIMA models was conducted by Box and Jenkins in the 1970s. The ARIMA model and demonstrated its usefulness in modeling and forecasting various types of time series data were introduced. Since then, Many researchers have applied ARIMA models to stock price prediction and have made several improvements and modifications to the original model.

1. Zhang et al. (2019) compared the performance of ARIMA, LSTM (Long Short-Term Memory), and hybrid models for predicting the stock prices of Chinese companies and found that ARIMA had the lowest prediction accuracy among the three models. It is observed that ARIMA models have been extensively used in the literature for stock price prediction, and they have shown

promising results in some cases. However, the accuracy of the model may depend on various factors, such as the quality and quantity of the data, the specific stock being analyzed, and the modeling approach used.

2. "Forecasting Stock Prices using ARIMA and LSTM Models" by Farhad Akhbari and others (2020) - This paper compares the performance of ARIMA and LSTM (Long Short-Term Memory) models for stock price prediction. The authors find that the LSTM model outperforms the ARIMA model in terms of accuracy.

3. "Predicting Stock Prices using ARIMA and Prophet Models" by Muhammad Nabeel Aslam and others (2020) - This paper compares the performance of ARIMA and Prophet models for predicting stock prices. The authors find that the Prophet model outperforms the ARIMA model in terms of accuracy and robustness.

4. "An Empirical Study on ARIMA Model for Stock Price Prediction" by Fengdong Qi and others (2020) - This paper evaluates the performance of an ARIMA model for predicting the stock price of a Chinese company. The authors find that the ARIMA model provides accurate predictions in the short term,

but its performance deteriorates as the prediction horizon increases.

5. "Stock Price Prediction using ARIMA Model" by Anirban Bhattacharya and others (2021) - This paper presents an ARIMA model for predicting stock prices using historical data. The authors use a sliding window approach to train and test the model, and evaluate its performance using root mean square error (RMSE) and mean absolute percentage error (MAPE).

Overall, these studies suggest that ARIMA models can be useful for stock market prediction, but their performance can vary depending on the specific application and the characteristics of the data being used.

Chapter 3

DATA COLLECTION AND ANALYSIS

3.1 Introduction

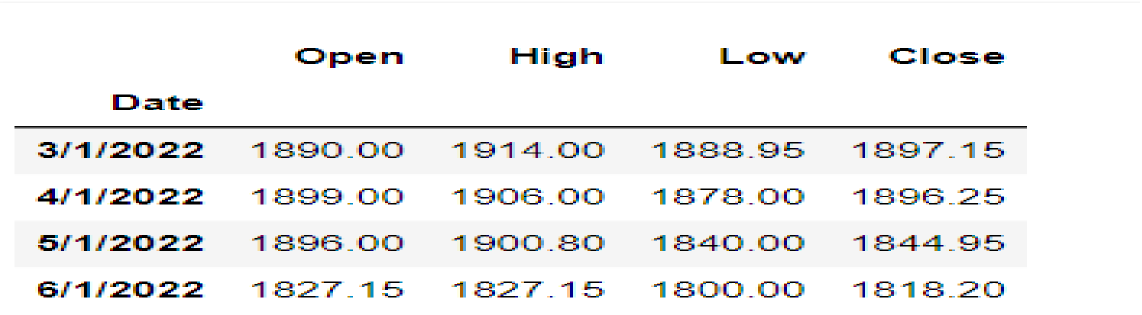
In this chapter, the basics results obtained in the study are presented. Those results are helpful to know the data well. This chapter includes how we have collected the secondary data and what results are obtained from data analysis.

3.2 Data Collection

The data from 1st January 2022 to 31st January 2023 of Bombay stock exchange (BSE) has been collected from the money control app which provides the business and finance related news. The collected dataset consist of 269 observations with 5 variables including time period. These variables are_

Date	The date of particular day
Open	The opening price of stock of the particular day
Close	The trading price at the end of the day
High	The Highest price at which a stock traded during the period.
Low	The Lowest price at which a stock traded during the period

Since, the Stock Exchange is open on the weekdays and is closed on weekend. The Data does not include observations of weekends and holidays declared by the Exchange in advance. We have piece of observations from our data that starts from 1st January and ended on 31st December present in the form of data frames.

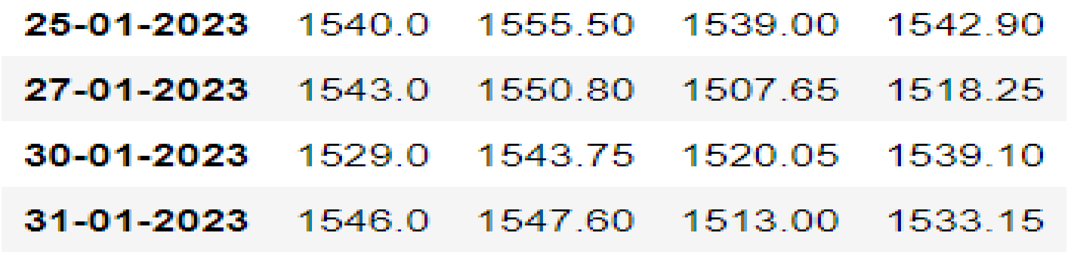


	Open	High	Low	Close
Date				
3/1/2022	1890.00	1914.00	1888.95	1897.15
4/1/2022	1899.00	1906.00	1878.00	1896.25
5/1/2022	1896.00	1900.80	1840.00	1844.95
6/1/2022	1827.15	1827.15	1800.00	1818.20

………………

………………

………………



25-01-2023	1540.0	1555.50	1539.00	1542.90
27-01-2023	1543.0	1550.80	1507.65	1518.25
30-01-2023	1529.0	1543.75	1520.05	1539.10
31-01-2023	1546.0	1547.60	1513.00	1533.15

3.3 Exploratory Data Analysis (EDA)

EDA is an approach or philosophy for data analysis that employs a variety of techniques to uncover underline structure, extract important variables, detect outliers or anomalies, test underline assumptions, develop parsimonious model and determine optimal factor settings.

First, a directorial path for the dataset is created using pandas read function. Data has a shape (269, 5) it means 269 row labels and 5 features. It is found that data do not contain duplicate, missing values. The Date column is converted to a proper Datetime datatype and set as index as we need to track variation in stock price on daily basis.

3.3.1 Univariate Analysis:

This is the simplest form of data analysis, where the data is being analysed consists of just one variable. Since it is single variable, it doesn’t deal with causes and relationships. The main purpose of univariate analysis is to describe the data and find patterns that exist within it. Outlies detection is additionally part of analysis. The characteristics of population distribution include:

1. Central tendency: The central tendency or location of distribution has got to do with typical or middle values. The commonly useful measures of central tendency are statistics called mean, median, and sometimes mode during which the foremost common is mean. For skewed distribution or when there’s concern about outliers, the median may be preferred.

2. Spread: Spread is an indicator of what proportion distant from the middle we are to seek out the find the info values. the quality deviation and variance are two useful measures of spread. The variance is that the mean of the square of the individual deviations and therefore the variance is the root of the variance.

3. Skewness and kurtosis: Two more useful univariates descriptors are the skewness and kurtosis of the distribution. Skewness is that the measure of asymmetry and kurtosis may be a more subtle measure of tailed Ness compared to a normal distribution

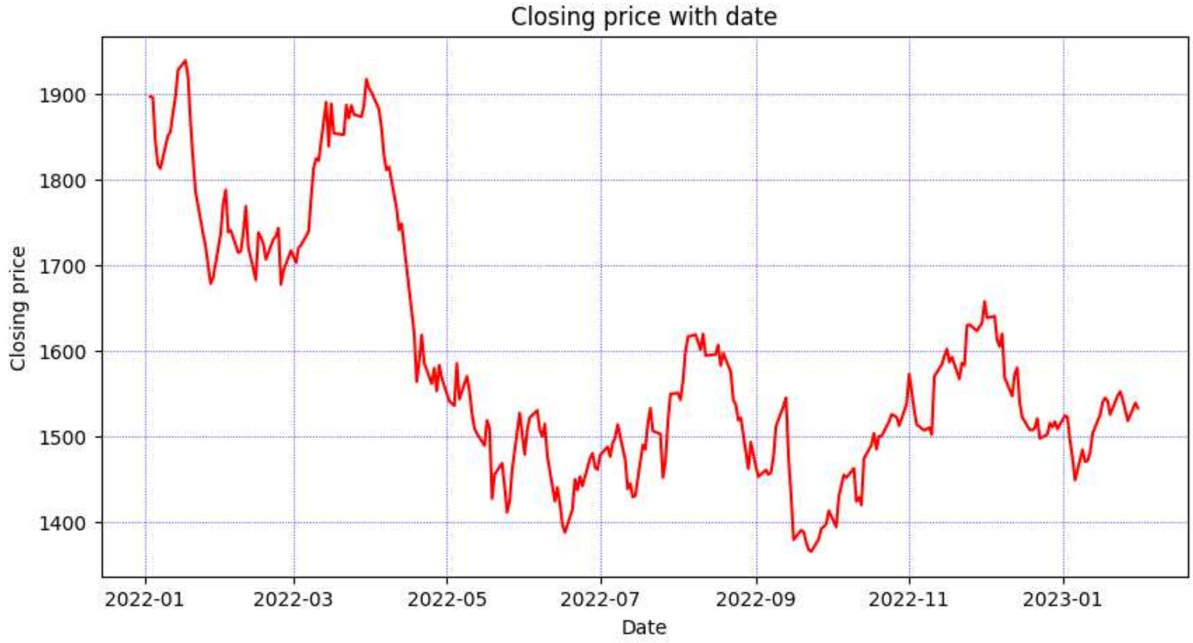
After performing the univariate analysis on the variables, It is found that variables are not normally distributed as it varies with time. Histogram and Box-whisker plots are used to know the data.

3.3.2 Bivariate Analysis:

To know the relationship between two variables, we perform the bivariate analysis. It is observed that all the variables except time are highly correlated. Scatter plot and heatmap matrix are used to understand the data.

3.4 Interpretation of Decreasing Trend in Closing Price

The objective of this project is to predict the stock's closing price of the January month-2023, so the closing price of stock is considered for time series analysis whereas, rest of the features have been studied in exploratory data analysis. It is observed that all the variables are highly correlated. This is the reason we are mentioning visualization of the closing price-



The year 2022 experienced a sudden changes in the stock prices of various tech companies from Indian Market. Infosys India too experienced a sudden decreasing trend in its stock prices. One of the most important reasons for the declining stock prices is due to Global Recession. Recessions happen every few years after the economy reaches its peak. Technology companies across various industries have been laying off employees at a rapid rate. This downturn was attributed to a variety of factors, including the disruption caused by the Covid-19 pandemic, Russia's invasion of Ukraine, and rising inflation leading to increased interest rates, which all caused investor's concerns and this led to a massive change in the stock prices.

Infosys experienced a major downward trend from March 2022, due to consecutive top-level resignations. On 11 March 2023, the company announced the resignation of its President, Mohit Joshi, who spent over 22 years at the company. There were several cases of top profile people, who had spent decades at the company, leaving in a short span of time. The reason for the drop is that investor sentiment was hurt when top management began to flee from the company as if it was being hit by a tsunami. Investor sentiment declined, resulting in dumping of shares and decreasing its value.

Infosys share fell in the last week of August, after it was reported that Infosys has reduced the variable pay out for all its employees to about 70% for Q1FY23. Variable pay is compensation given to an employee based on the results of their performance. This news led a falloff in stock prices of the company.

Infosys share price gained after the Company announced to consider a proposal for a share buyback on Thursday, 13 October. Share buyback, or share repurchase, is when a company buys back its own shares from investors. There has also been a huge fall in the valuation of IT shares since last November, resulting in a fall off in the stock prices. Hence, investors are staying away from the Infosys share and other IT stocks. The stock has also been under pressure in the past few sessions amid concerns about the impact of the global banking crisis on the growth of IT companies.

Chapter 4

TIME SERIES ANALYSIS

4.1 Introduction

A time series is a set of ordered observations of a quantitative variables being recorded at specific interval of time. In other words, it is the arrangement of statistical data in chronological order. We perform time series analysis, when we have to predict Future values based on past data. Time series is time dependent. Examples, price and dividends of shares in stock market analysis, interest rates, National Income and Forex Reserves, etc.

Mathematically, time series is defined by functional relationship

$$y_t = \text{f(t)}$$

Where, y_t - value of variable under consideration at time t

4.2 Purpose

1. The purpose of Time series analysis in stock analysis is to forecast the future stock prices on the basis of past trends and patterns in the data. Time series analysis is particularly well suited for stock analysis because the stock prices are often influenced by the no of factors that change over time, factors including economic indicators, market trends, company specific events, etc.

2. Time series analysis also helps to quantify the level of risk associated with a particular stock, which is an important consideration for

investors. By analysing historical stock prices, time series models can help to identify trends, patterns, and fluctuations in the stock market that can be used to make better and informed decisions.

3. In time series analysis, statistical models are used to examine and quantify the relationship between past stock prices and other relevant economic and financial variables. These models take into account various factors such as economic indicators, market sentiment, news events, and trends in financial markets to generate predictions about future stock prices.

Overall, the purpose of time series analysis in stock price prediction is to provide valuable insights into the behaviour of stock prices, enabling investors to make informed investment decisions and maximize their returns.

4.3 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 following four categories, commonly known as the components of a time series. Some of which are present (in a given time series) in varying degrees.

A) Secular Trend or Long-term Movement

B) Periodic Changes or Short-term Fluctuations

1) Seasonal variations, and 2) Cyclic variations.

C) Random or Irregular Movements

4.3.1 Trend

Trend is the general tendency of the data to increase or decrease over period of time, it persists over long period of time. Trend is general, smooth, long-term, average tendency. It is not necessary that the increase or decrease should be in same direction throughout the given period. Example: population growth over the years.

4.3.2 Periodic Changes

It would be observed that in many economic phenomena, apart from the growth factor in a time series there are forces at work which prevent the smooth flow of the series in a particular direction and tend to repeat themselves manner. The resultant effect of such forces may be classified as:

4.3.2.1 Seasonal Variations

These are the regular pattern of up and down fluctuations which operates in a regular and periodic manner over span of less than a year. It is short-term variations occurring due to seasonal factors. The seasonality will be there if the data are recorded quarterly, monthly, weekly, daily, hourly, and so on.

4.3.2.2 Cyclic variations

It is similar to seasonality. It is medium-term variation caused by circumstances, which repeat in irregular interval. The duration can vary and gap between the two cycles is much longer compared to seasonality.

4.3.3. Irregular (or Random) Component:

It refers to variations which occur due to unpredictable factors and also do not repeat in particular patterns. Also called as random or residual fluctuations, which are not accounted for by trend and seasonal and cyclic variations. In some cases, the importance of irregular fluctuations may not be significant while in some these may be very effective and might give rise to cyclic movements.

4.4. Mathematical Models for Time Series

The main problem in the analysis of time series are:

1) To identify the forces or components at work, the net effect of whose interaction is exhibited by the movement of a time series, and

2) To isolate, study, analyse and measure them independently, i.e by holding other things constant.

Two models are commonly used for decomposition of a time series into its components.

4.4.1 Additive Model

Additive models assume that the components of the time series can be added together to produce the final result. In this type of model, the effects of individual components are additive, and the final value of the time series at any given point in time is equal to the sum of the values of these components. For example, an additive model for a time series might be represented as:

$$y_t = \text{Trend(t) + Seasonality(t) +cyclicity(t)+ Residual(t)}$$

where y(t) is the value of the time series at time t, Trend(t) is the trend component, Seasonality(t) is the seasonal component, and Residual(t) is the residual or error component.

4.4.2 Multiplicative Model

Multiplicative models, on the other hand, assume that the components of the time series multiply to produce the final result. In this type of model, the effects of individual components are multiplicative, and the final value of the time series at any given point in time is equal to the product of the values of these components. For example, a multiplicative model for a time series might be represented as:

y_t= Trend(t) x Seasonality(t) x cyclicity(t) x Residual(t)

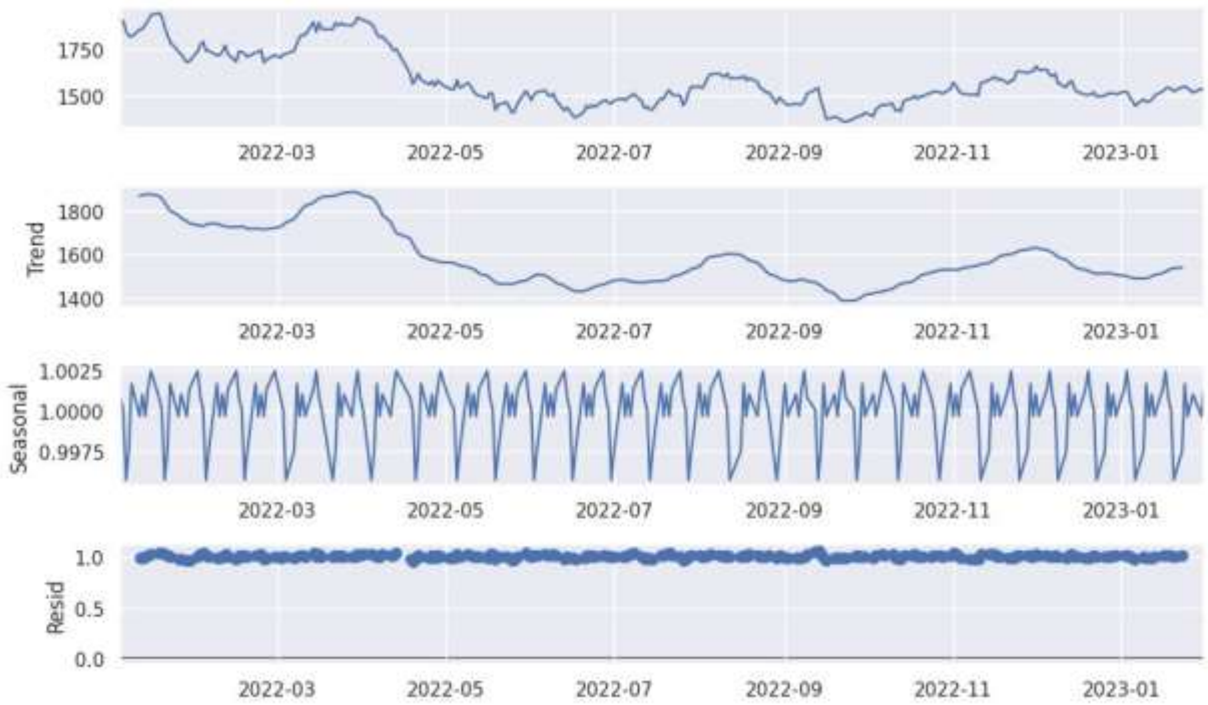
It may be pointed out that the multiplicative decomposition of time series is as the additive decomposition of logarithmic values of original time series, i.e.,

logy(t)= log(trend(t)) + log(seasonality(t)) + log(cyclicity(t)) + log(Residual(t))

The choice between additive and multiplicative models depends on the nature of the time series data. In general, additive models are appropriate for time series with a constant rate of change, while multiplicative models are appropriate for time series with a variable rate of change.

4.4.3 Interpretation of Decomposition of Time Series

Time series data can exhibit a variety of patterns, and it is often helpful to split a time series into several components, each representing an underlying pattern category. Time series decomposition is one of the best ways to understand how a time series behaves. The Decomposition Plot graphs the observed values, the underlying trend, cyclic patterns, and randomness of the data.



The above graph shows a multiplicative decomposition of the data. Here we have used period of 10. Since it is a multiplicative model, we note that cyclic patterns and residuals are both centered at one (instead of zero). The multiplicative model better explains the variations of the original time series.

The trend shows the overall movement in the series. From the above graph, we can see very clearly that the data have an overall decreasing trend. A cyclic pattern exists when data exhibit rises and falls that are not of fixed period. The stock market tends to cycle between periods of high and low values, but there is no set amount of time between those fluctuations. The three components are shown separately in the bottom three panels of graph. These components can be added together to reconstruct the data shown in the top panel. The residual component shown in the bottom panel is what is left over when the cyclic and trend components have been subtracted from the data.

4.5 Moving Average Method

In statistics, a moving average is a calculation to analyze data points by creating a series of averages of different subsets of the full data set. A moving average is commonly used with time series data to smooth out short-term fluctuations and highlight longer-term trends or cycle. The technique represents taking an average of a set of numbers in a given range while moving the range.

Moving average is a smoothing process which is widely used in financial analysis, signal processing, and other fields where it is useful to smooth noisy data and extract underlying trends or patterns.

The formula for a Moving average is :

$$\hat{y}_t = \frac{y_t + y_{t-1} + \cdots + y_{t-N+1}}{N}$$

4.5.1 Objective to use moving Average

Moving averages have the property to reduce the amount of variation present in the data. In the case of time series, this property is used to eliminate fluctuations, and the process is called smoothing of time series. It is important in time series analyses for several reasons :

1. **Smoothing :** Moving averages can help to smooth out noisy or erratic data over time, making it easier to identify trends and patterns in the data. By taking the average of a series of data points over a specific time window, a moving average can reduce the impact of short-term fluctuations in the data and highlight longer-term trends.

2. **Highlighting Seasonality :** In some time series data, there may be seasonal patterns that repeat over a fixed period of time. Moving averages can be used to identify these patterns and remove the seasonal effects, making it easier to identify other patterns and trends in the data.

3. **Forecasting :** Moving averages can be used to forecast future values in a time series.

4. **Identifying turning points :** Moving averages can be used to identify turning points in a time series.

5. **Data visualization :** Moving averages can be used to create charts and graphs that make it easier to visualize trends and patterns in a time series.

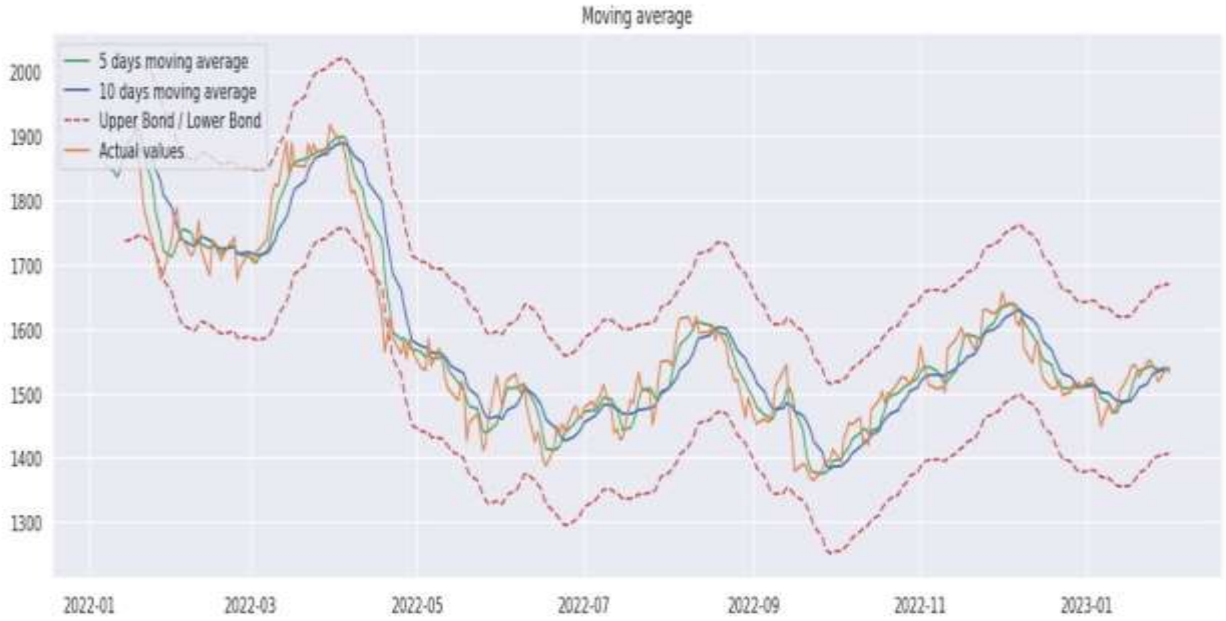
6. **Removing outliers :** Moving averages can be used to remove outliers and anomalies in the data, making it easier to analyze and model the underlying process.

In simple terms, a moving average takes the average of several different points in the data set and then plots it over time. A longer-term moving average will give more emphasis to older data points, while a shorter-term one will look more closely at recent values.

In case of stock price prediction, by examining how the line moves from period to period, investors can get a sense of where prices may be headed in the near future. For example, if prices were generally increasing with each new period up until now, then investors may expect prices to continue rising at least until there is clear evidence suggesting otherwise. On the other hand, if prices began dropping off sharply after some time period and continued to do so until present day, then this could indicate that downwards trend could continue.

4.5.2 Interpretation to check the smoothness:

We have a time series of daily stock prices over the 1 year. To calculate a 5-day moving average, we would take the average of the stock prices for 1-5, then the average of the stock prices for days 2-6, and so on, until we reach the end of the time series. In the same way, we are calculating a 10-day moving average. The resulting sequence of averages would show how the stock prices have changed over time, with the noise and short-term fluctuations smoothed out as shown in graph_



It is observed that 10 days moving average is showing more smoothing as compared to 5 days moving average. Moving averages smooth out periodic variations, if the period of the moving average is same as the period of the variations. But if the period of moving averages is less or more than the period of variation, the periodic effect still remains in the moving averages figures. we can observe that actual data has more fluctuations. After applying the moving average for the period of 10, we get estimated values which is equal to 1538.5099 at some level of significance.

4.6 Weighted Moving Average

A weighted moving average (WMA) is a type of moving average that assigns different weights to each data point in the time series. Unlike a simple moving average, which assigns equal weight to all data points, a WMA places greater emphasis on recent data points or other data points that are deemed more important or relevant.

4.6.1 Steps involving for Calculation of Weighted Moving Average

Calculating the weighted moving average involves taking recent data points and assigning a greater weighting compared to past data points. When summed up, the total value of the weights should be equal to 100% or 1. On the other hand, the WMA considers the importance of each data point, which is reflected in the 'weight' assigned to it. We first select a specific time frame and assign a weight to each data point within that time frame. The sum of the weights should equal 1.0. The weighted moving average is then calculated by multiplying each data point by its assigned weight, adding up the results, and dividing by the sum of the weights. The weighted Moving Average Formula :-

The WMA formula is expressed as follows:

$$\hat{y}_t = \frac{\sum_{t=1}^{n} W_t * y_t}{\sum_{t=1}^{n} W_t}$$

Where,

y = Actual value

W = Weighting factor

n = Number of periods in a weighting group

WMA helps in determining trend direction**.** When the price is above its weighted MA line, it's usually a signal that on average, the asset is trading higher than it has over the period being analysed. This in turn confirms an uptrend. Alternatively, when the price is below the WMA line, then it confirms a downtrend. WMA determines trend faster then moving average. A rising weighted MA can indicate support for price action. While a falling WMA can indicate resistance to price action over a given period. Traders tend to use this as a strategy to place buy orders when the price is near the rising WMA or place sell orders when the price is near the falling weighted MA. Weighted moving averages can be used to detect signals or changes in the data that may be of interest. For example, if the weighted moving average crosses a certain threshold or breaks a particular trend line, this may indicate a change in the underlying pattern of the data.

4.6.2 Interpretation for WMA:

Suppose we are calculating a 10-day weighted moving average for a given dataset, and 0.3,0.2, 0.2,0.1, 0.05,0.05,0.025,0.025,0.025,0.025 are assigned weights to the most recent 10 data points. To calculate the WMA for the most recent day, you would multiply the most recent data point by the weight of 0.3, the second most recent data point by the weight of 0.2, and so on, until you have multiplied each data point by its assigned weight. You would then add up the results and divide by the sum of the weights to get the 10-day WMA. After applying the moving average for the period of 10, we get estimated values which is equal to 1534.50125 at some level of significance.

4.7 Exponentially weighted moving average:

An exponential weighted moving average (EWMA) is a type of moving average (MA) that places a greater weight and significance on the most recent data points. An exponentially weighted moving average reacts more significantly to recent price changes than a simple moving average, which applies an equal weight to all observations in the period.

Exponentially weighted moving average is often applied to time-ordered sequence of random variables. It computes a weighted average of the sequence by applying weights that decreases geometrically with the age of the observations. Exponentially weighted moving average (EWMA) is a commonly used method in time series analysis for smoothing out data and identifying trends. In a time series, data is collected at different points in time, and EWMA is used to give more weight to recent observations while still considering older ones. This makes it easier to identify trends and changes in the underlying pattern of the data.

EWMA is used by assigning weights to each observation, with the weights decreasing exponentially as the observations become older. The smoothing factor, which is typically a value between 0 and 1, determines the rate at which the weights decrease. The higher the smoothing factor, the more weight is given to recent observations.

By smoothing out the data using EWMA, it becomes easier to identify trends and underlying patterns in the time series. This can be helpful in making predictions and identifying anomalies or unusual events in the data. For example, in finance, EWMA can be used to analyse stock prices, identify trends in sales data, or forecast demand for a product. In engineering, EWMA can be used to monitor system performance over time and identify changes in patterns that may indicate potential issues or problems.

Overall, EWMA is a powerful tool in time series analysis that can help to reduce noise in the data, identify trends, and highlight changes in patterns over time.

The formula for calculating an EWMA is:

$$EWMA_t = (1 - \alpha) * y_t + \alpha * EMA_{t-1}$$

Where:

- $EWMA_t$is the exponentially weighted moving average at time t
- y_t is the value of the time series at time t
- EMA_{t-1} is the exponentially weighted moving average at time t-1
- α is the smoothing factor, which determines the weight given to the current observation. It is a number between 0 and 1, with larger values giving more weight to recent observations.

4.8 Double Exponentially weighted moving average

The double exponential weighted moving average (DEWMA) is devised to reduce the lag in the results produced by a traditional moving average. Technical traders use it to lessen the amount of "noise" that can distort the movements on a price chart.

The name double comes from the fact that the value of an EMA (Exponential Moving Average) is doubled. To keep it in line with the actual data and to remove the lag the value "*EMA of EMA*" is subtracted from the previously doubled ema.

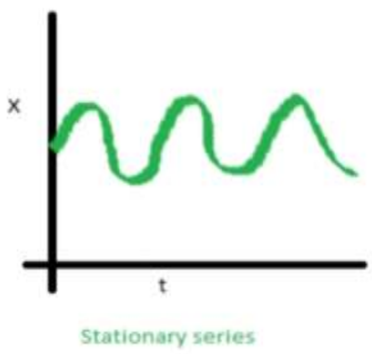
DEWMA = 2* EMA -EMA(EMA)

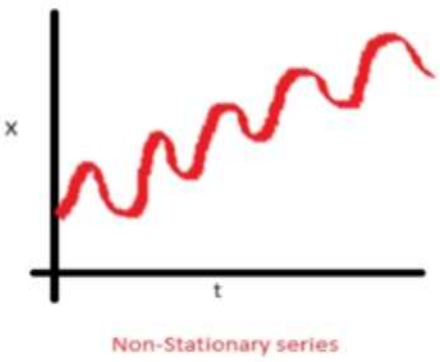
As shown in the formula it reduces the weight on the recent values and by calculating ema of the ema we are trying to remove the weight on the long slower part of the average that has built up over time. It significantly helps make quicker decisions than the simple MA crossovers.

4.9 Stationarity in Time series

A stationary time series is one whose statistical properties such as mean, variance and covariance are all constant over time. Most statistical forecasting methods are based on the assumption that the time series can be rendered approximately stationary through the use of mathematical transformations. A stationary series is relatively easy to predict. We simply predict that its statistical properties will be the same in the future as they have been in the past.

A time series whose statistical properties change over time is called a non-stationary time series. Non-stationary data are unpredictable and cannot be modeled or forecasted. Thus a time series with a trend or seasonality is non-stationary in nature. This is because the presence of trend or seasonality will affect the mean, variance and other properties at any given point in time.

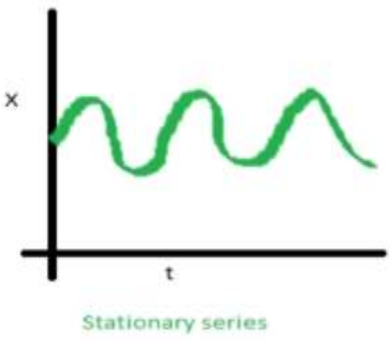


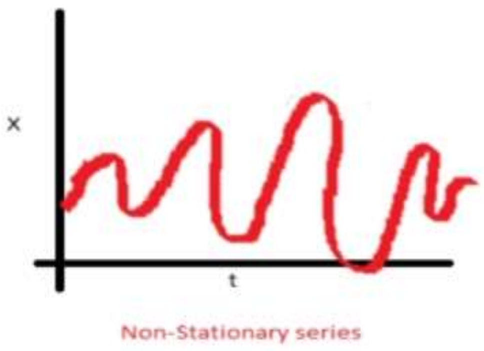


Fig(A) Fig(B)

Here Fig(A) is stationary, In this case, the mean, variance and covariance are constant with time.

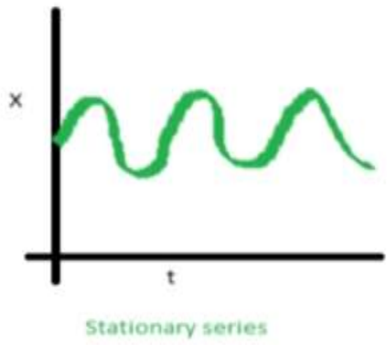
Fig(B) is not stationary because we can clearly see that the mean varies (increases) with time which results in an upward trend. Thus, this is a non-stationary series. For a series to be classified as stationary, it should not exhibit a trend.



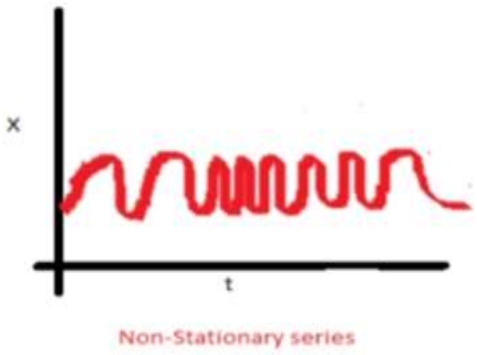


Fig(C) Fig(D)

In Fig(D), we certainly do not see a trend in the series, but the variance of the series is a function of time. A stationary series must have a constant variance.



Fig(E)



Fig(F)

In Fig(F), we see that the spread becomes closer as time increases. Hence, the covariance is not constant with time.

4.9.1 Types of Stationary :

A) Strict Stationary

A strict stationary series satisfies the mathematical definition of a stationary process. For a strict stationary series, the mean, variance and covariance are not the function of time. The aim is to convert a non-stationary series into a strict stationary series for making predictions.

B) Trend Stationary:

A series that has no unit root but exhibits a trend is referred to as a trend stationary series. Once the trend is removed, the resulting series will be strict stationary. The KPSS test classifies a series as stationary on the absence of unit root. This means that the series can be strict stationary or trend stationary.

C) Difference Stationary:

A time series that can be made strict stationary by differencing falls under difference stationary. ADF test is also known as a difference stationarity test.

The time series which have **trends** or with **seasonality**, are not stationary. Because trends will have a change in the movement of data concerning time which will cause the change in mean over time. Whereas seasonality occurs when the pattern in time series shows a variation for a regular time interval which will cause the variance to change over time.

Cyclic behaviour and **white noise** in time series are stationary. The cyclic behaviour of time series will be stationary because the cycles are not of a fixed length, so before we observe the series we cannot be sure where the peaks and troughs of the cycles will be.

4.9.2 Importance of Stationarity:

It is easy to make predictions on a stationary series since we can assume that the future statistical properties will not be different from those currently observed. Most of the time-series models, in one way or the other, try to predict those properties (mean or variance, for example). Future predictions would be wrong if the original series were not stationary.

A Stationary time series help us to obtain meaningful sample statistics such as means, variances, and correlations with other variables. Such statistics are useful as descriptors of future behavior only if the series is stationary. For example, if the series is consistently increasing over time, the sample mean and variance will grow with the size of the sample, and they will always underestimate the mean and variance in future periods. And if the mean and variance of a series are not well-defined, then neither are its correlations with other variables.

4.9.3 Transformation non-stationary series into stationary series:

Most time series models work under the assumption that the underlying data is stationary, that is the mean, variance, and covariance are not time-dependent. Non-stationary data are unpredictable and cannot be modeled or forecasted. The results obtained by using non-stationary time series may be spurious in that they may indicate a relationship between two variables where one does not exist. In order to receive consistent, reliable results, the non-stationary data needs to be transformed into stationary data. More likely, if the time series is non-stationary, which means that we have to identify the trends present in the series and manipulate the data to become stationary. After the trends are removed we can apply advanced modeling techniques while maintaining the valuable knowledge of the separated trends.

If the time series is not stationary, we can often transform it to a stationary with one of the following techniques:

1. Differencing : A non-stationary time series can be converted to a stationary time series through a technique called differencing. Differencing series is the change between consecutive data points in the series.

$$y_t' = y_t - y_{t-1}$$

This is called first order differencing. In some cases, just differencing once will still yield a non-stationary time series. In that case a second order differencing is required. Second order differencing is the change between two consecutive data points In a first order differenced time series. To generalize, differencing of order **d** is used to convert non-stationary time series to stationary time series. The differenced data will contain one less point than the original data. Although you can difference the data more than once, one difference is usually sufficient.

2. Rolling Mean : When the mean is time dependent , we can subtract the rolling mean from a time series.. A rolling mean is the mean of the previous *x* number of observations in the series, where the time between each observation is consistent.

3. If the data contain a trend, we can fit some type of curve to the data and then model the residuals from that fit. Since the purpose of the fit is to simply remove the long term trend, a simple fit such as a straight line, is typically used.

4. For non-constant variance, taking the logarithm or square root of the series may stabilize the variance. For negative data, you can add a suitable constant to make all the data positive before applying the transformation. This constant can then be subtracted from the model to obtain predicted values and forecasts for future points

4.10 Augmented dickey-fuller test (ADF) or unit root test

4.10.1 Importance of test

Various financial market variables such as future contracts, stock prices, dividends, spot and exchange rates, etc. show non-stationary behaviour because of presence of unit root. Different types of non-stationarity have different economic implications. Identifying the nature of non-stationarity is important. These tests help in determining whether the trend is stochastic, through the presence of a unit root, or deterministic, through the presence of a unit root, or deterministic, through the presence of a polynomial time trend.

4.10.2 Dickey-Fuller test (DF test): -

It is crucial to specify the null and alternative hypothesis properly keeping in view the trend properties of the data. The trend properties of the data under the alternative hypothesis will determine the form of the regression used. Type of deterministic terms in the test regression will influence the asymptotic distributions of the unit root test statistics.

Case I: Without drift and trend

Consider the Autoregressive (AR (1)) process

$$\mathrm{y(t)} = \varphi \mathrm{y}_{\mathrm{t-1}} + \epsilon_t, \ \epsilon_t \sim \mathrm{WN}(0, \sigma_{\mathrm{u}}^2) \quad \text{---------------------(1)}$$

$$\varphi = 1 \colon \Delta\, \mathrm{y(t)} = \epsilon \quad \text{--------------------(2)}$$

Or $\Delta\, \mathrm{y(t)} = \gamma \mathrm{y}_{\mathrm{t-1}} + \epsilon_t \gamma = \varphi - 1$ -------------------(3)

$$\mathrm{H}_0 : \gamma = 0 \text{ vs } \mathrm{H}_1 : \gamma < 0$$

The test statistic t is

$$\mathrm{t} = \frac{\hat{\gamma}}{\mathrm{SE}\hat{\gamma}} \quad \text{----------------(4)}$$

Asymptotic distribution $\hat{\varphi} \sim \mathrm{N}\left(\varphi, \frac{1}{\mathrm{n}}(1 - \varphi^2)\right)$ -------(5)

Case2: constants only

Consider the process

$$\mathrm{y_t} = \delta_0 + \varepsilon \mathrm{t}$$

$$\epsilon \mathrm{t} = \varphi \epsilon \mathrm{t} - 1 + \epsilon_{\mathrm{t}}$$

When$|\varphi| < 1$, we can write the process as

$$\mathrm{y_t} = \delta_0 (1 - \varphi) + \varphi \mathrm{y}_{\mathrm{t-1}} + \epsilon_t$$

$$= \mathrm{c} + \varphi \mathrm{y}_{\mathrm{t-1}} + \epsilon_t,$$

$$\mathrm{c} = \delta_0 (1 - \varphi)$$

When $\varphi = 1$, the process becomes

$$\mathrm{y_t} = \mathrm{y}_{\mathrm{t-1}} + \epsilon_t$$

Unit root hypothesis

$$\mathrm{H0: \varphi \ = \ 1, Model\, yt \ = \ yt - 1 \ + \ } \epsilon_t \Rightarrow \mathrm{\ I(1) without\ drift}$$

$$\mathrm{H1: \varphi \ < \ 1, Model\, yt \ = \ c \ + \ \varphi yt - 1 \ + \ } \epsilon_t \Rightarrow \mathrm{\ I(0)\ with\ constant}$$

$$\mathrm{LS\ estimator\ of\ \varphi: } \hat{\varphi} \ = \frac{\sum(\mathrm{y_t} - \bar{\mathrm{y}})(\mathrm{y_{t-1}} - \bar{\mathrm{y}})}{\sum(\mathrm{y_{t-1}} - \bar{\mathrm{y}})^2}$$

Test Statistic: t =$\frac{\hat{\varphi}-1}{\mathrm{SE}\ (\hat{\varphi})}$

The asymptotic distributions of

$\mathrm{n}\ (\hat{\varphi} \ - 1)$ and $\mathrm{t}_{\varphi=1}$ are not normal

and different from (6) and (7).

Case III: Constant and Linear Time trend

Consider the process

$$\mathrm{y_t} \ = c \ + \ \varphi\delta_1 \ + \ \delta_1(\ 1 \ - \ \varphi\ \mathrm{t}) \ + \ \varphi \mathrm{y_{t-1}} \ + \ \epsilon_t$$

$$= \ \alpha \ + \ \beta \mathrm{t} \ + \ \varphi \mathrm{yt} - 1 \ + \ \epsilon_t$$

$$\mathrm{where}\ \alpha \ = c + \ \varphi\delta_1, \beta \ = \delta_1(\ 1 \ - \ \varphi\ \mathrm{t})$$

For $\varphi \ = \ 1$, the process is

$$\mathrm{y_t} \ = \ \delta_1 \ + \ \mathrm{y}_{\ \mathrm{t-1}} + \ \epsilon_t$$

Under unit root hypothesis

$$\mathrm{H0: \varphi \ = \ 1, Model\ \ yt} \ = \ \delta_1 \ + \ \mathrm{y}_{\ \mathrm{t-1}} + \ \epsilon_t$$

$\Longrightarrow$ I(1) with drift

Under H1: $|\varphi| < 1$, Model

$y_t = \alpha + \beta t + \varphi y_t - 1 + \epsilon_t \Longrightarrow$ I(0) with

constant and linear trend

We obtain least squares estimator of φ, say, $\widehat{\varphi}$

$$t = \frac{\varphi\hat{} - 1}{SE\ (\widehat{\varphi})} = \frac{\widehat{\gamma}}{SE\widehat{\gamma}}$$

The asymptotic distributions of

$n\ (\widehat{\varphi} - 1)$and $t_{\varphi=1}$ are not normal.

Augmented Dickey-fuller test:-

The Augmented dickey-fuller test is used to determine whether the time series data is stationary or not. It is based on Dickey-fuller test, which is used to test for the presence of a unit root in time series. It is an extension to DF test that takes into account the possible presence of autocorrelation. Autocorrelation refers to the correlation between the values of a time series at different time points. It uses a regression model to test whether the first difference of the time series is stationary. Here are the various cases of the test options

Consider AR (p) process

$$y_t = \varphi_1 y_{t-1} + \cdots + \varphi_p y_{t-1} + \epsilon_t \text{-----------------(1)}$$

The model can be reparametrized as

$y_t = \varphi y_{t-1} + \theta_1 \Delta y_{t-1} + \cdots + \theta_{p-1} \Delta y_{t-p+1} + \epsilon_t$------------(2)

$$\varphi = \sum_{j=1}^{p} \varphi_j, \ \theta_i = -\sum_{j=i+1}^{p} \varphi_j$$

Hence, we can write (2) as

$y_t = \varphi y_t - 1 + \sum_{j=1}^{p-1} \theta_j \Delta y_{t-j} + \epsilon_t$-----------------------(3)

We can write

$$\theta_i - \theta_{i-1} = -\sum_{j=i+1}^{p} \varphi_j + \sum_{j=i}^{p} \varphi_j = \varphi_i; i = 2, \ldots, p-1$$

$$\varphi + \theta_1 = \varphi_1$$

$$\theta_{p-1} = -\phi_p$$

Hence

$$\begin{aligned} &\varphi_1 y_{t-1} + \varphi_2 y_{t-2} \ldots + \varphi_p y_{t-p} \\ &\quad = (\varphi + \theta_1) y_{t-1} + (\theta_2 - \theta_1) y_{t-2} + (\theta_3 - \theta_2) y_{t-3} \ldots \\ &\quad + (\theta_{p-1} - \theta_{p-2}) y_{t-p+1} + (-\theta_{p-1}) y_{t-p} \\ &\quad = \varphi y_{t-1} + \theta_1 \Delta y_{t-1} + \cdots + \theta_{p-1} \Delta y_{t-p+1} \end{aligned}$$

$\sum_{j=1}^{p-1} \theta_j \Delta y_{t-j}$ = Augmentation term

Δy_{t-j} capture serial correlation

AR(p) process with deterministic linear trend:

$$\Delta y_t = [\delta_0(1-\varphi) + \varphi\delta_1] + \delta_1(1-\varphi)t$$

$$+\gamma y_{t-1} + \sum_{j=1}^{k} \theta_j \Delta y_{t-j} + \ \epsilon_t \text{----------------------(4)}$$

$\sum_{j=1}^{k} \theta_j \Delta y_{t-j}$ = Augmentation term of order k

We consider model

$$\Delta y_t \ = \ \beta' D_t + \gamma y_{t-1} + \sum_{j=1}^{k} \theta_j \Delta y_{t-j} + \ \epsilon_t \text{ -------------------(5)}$$

D_t: vector of deterministic terms (intercept, trend etc.)

K is chosen so that the residual follows purely random process. We apply OLS to (5). The test statistic is

$$\mathrm{ADF_t} = \frac{\hat{\gamma}}{\mathrm{SE}(\hat{\gamma})}$$

We can also normalise bias statistic

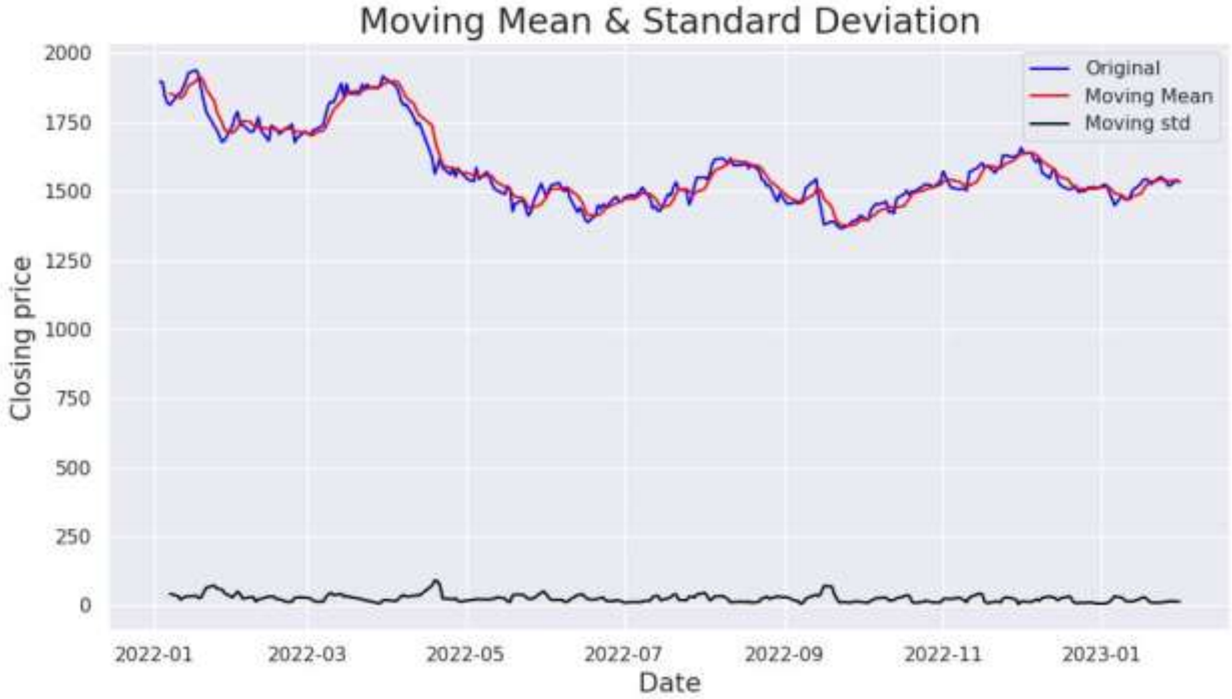
$$\mathrm{ADF_{nb}} = \frac{\hat{\gamma}}{1 - \widehat{\theta_1} - \cdots - \hat{\theta}_k}$$

We use the same critical values as in DF test. We can apply ADF test if residuals are generated by invertible MA or stationary and invertible ARMA process as these process can be approximated by high order AR process. Quality of a unit root test depends upon whether the test is performed within appropriate model.

4.11 Interpretation of stationarity

4.11.1 Graphical Interpretation with the help of moving mean and standard deviation:

To analyse stationarity we have taken rolling mean and rolling standard deviation. we used window of 10 for rolling mean and standard deviation using pandas series function.



We can conclude from the graph that the rolling standard deviation is very much constant for this at 0, but the rolling mean does display a trend over time and it is highly correlated to the original time series thus, it means that the mean is not constant over the period of time. Thus, a visual inspection concludes that our data is not stationary.

4.11.2 With the help of statistical Augmented Dickey-Fuller Test or Unit root test

To check the given series is stationary or not by Augmented Dickey-fuller test or Unit root test we set up the hypothesis,

H_0 : The data is non-stationary or There is a presence of unit root.

(i.e The data needs to be differenced to make it stationary).

H_1 : The data is stationary or There is absence of unit root.

(i.e The data is stationary and doesn’t need to be differenced).

Results of Dickey-Fuller Test:

Test Statistic	-2.289897
P-value	0.175247
Number of observations used	268
Critical value (5%)	-2.87236

Looking at the test statistic value, we can see that it is greater than a 5% critical value score of -2.87386 and the p values is also greater than 0.05 (significance level) Therefore, we cannot reject the null-hypothesis and can conclude that this data is non-stationary.

4.12 Implementation of ARIMA Model

ARIMA is a statistical model used for forecasting time series data. The ARIMA equation is a regression type equation in which the independent variables are lags of the dependent variable and lags of the forecast errors.

y'_t = c $+ \varphi_1 y'_{t-1} + \cdots + \varphi_p y'_{t-p} + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t$

There are three terms in the equation :

Auto Regression (AR) : The time series is regressed with its previous values i.e. y_{t-1}, y_{t-2} etc. The order of the lag is denoted as **p.**

Integration (I) : The time series uses differencing to make it stationary. The order of the difference is denoted as **d.**

Moving Average (MA) : The time series is regressed with residuals of the past observations i.e. error ε_{t-1}, error ε_{t-2} etc. The order of the error lag is denoted as **q.**

4.12.1 Differencing :

A Non-Stationary series can be made stationary through differencing. The first difference of a time series is the series of changes from one period to the next. If y_t denotes the value of the time series y at period t, then the first difference of Y at period t is equal to $y_t - y_{t-1}$

The number of times that the original series must be differenced in order to achieve stationarity is called the order of integration, denoted by d.

Backshift notation :

The backward shift operator denoted by B is convenient for describing the process of differencing. It is a useful notational device when working with time series lags :

$$By_t = y_{t-1}$$

In other words, B operating on y_t, has the effect of shifting the data back to one period.

Two applications of B to y_t, shifts the data back to two periods:

$$B(By_t) = B^2 y_t = y_{t-2}$$

A first difference can be written as

$$y'_t = y_t - y_{t-1} = y_t - By_t = (1-B)y_t$$

Similarly, if second-order differences have to be computed, then:

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

In general, a d^{th} -order difference can be written as

$$B^d y_t = (1-B)^d y_t$$

4.12.2 Autoregressive models

In a multiple regression model, we forecast the variable of interest using a linear combination of predictors. In an autoregression model, we forecast the variable of interest using a linear combination of past values of the variable. The term auto regression indicates that it is a regression of the variable against itself.

Thus, an autoregressive model of order p can be written as

$$y_t = \mathrm{c} + \varphi_1 y_{t-1} + \varphi_2 y_{t-2} + \cdots + \varphi_p y_{t-p} + \varepsilon_t$$

where ε_t is white noise. This is like a multiple regression but with lagged *values* of y_t as predictors. We refer to this as an AR**(**p**)** model**,** an autoregressive model of order p. Autoregressive models are remarkably flexible at handling a wide range of different time series patterns. Changing the parameters $\varphi_1, \ldots\ldots, \varphi_p$ results in different time series patterns. The variance of the error term ε_t will only change the scale of the series, not the patterns.

For an AR(1) model:

- when φ_1= 0, y_t is equivalent to white noise;
- when φ_1 = 1 and c = 0, y_t is equivalent to a random walk;
- when φ_1 = 1and c ≠ 0, y_t is equivalent to a random walk with drift;
- when φ_1 < 0, y_t tends to oscillate around the mean.

We normally restrict autoregressive models to stationary data, in which case some constraints on the values of the parameters are required.

For an AR(1) model: $-1 < \varphi_1 < 1$.

For an AR(2)model: $-1 < \varphi_2 < 1$, $\varphi_1 + \varphi_2 < 1$, $\varphi_2 - \varphi_1 < 1$.

The order of an AR model can be find out using Partial Autocorrelation Function (PACF).

4.12.3 Moving average models :

Rather than using past values of the forecast variable in a regression, a moving average model uses past forecast errors in a regression-like model.

$$y_t = \mathrm{c} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q}$$

where ε_t is white noise. We refer to this as an MA(q) model, a moving average model of order q.

Each value of y_t can be thought of as a weighted moving average of the past few forecast errors.

However, moving average *models* should not be confused with the moving average smoothing. A moving average model is used for forecasting future values, while moving average smoothing is used for estimating the trend-cycle of past values.

Changing the parameters $\theta_1, \ldots., \theta_q$ results in different time series patterns. As with autoregressive models, the variance of the error term ε_t will only change the scale of the series, not the patterns.

It is possible to write any stationary AR(p) model as an MA(∞) model. For example, using repeated substitution, we can demonstrate this for an AR(1) model:

$$\begin{aligned} y_t &= \varphi_1 y_{t-1} + \varepsilon_t \\ &= \varphi_1(\varphi_1 y_{t-2} + \varepsilon_{t-1}) + \varepsilon_t \\ &= (\varphi_{21} y_{t-2} + \varphi_1 \varepsilon_{t-1}) + \varepsilon_t \\ &= \varphi_{31} y_{t-3} + \varphi_{21}\varepsilon_{t-2} + \varphi_1 \varepsilon_{t-1} + \varepsilon_t \end{aligned}$$

Provided −1< φ_1<1 the value of φ_{k1} will get smaller as k gets larger. So eventually we obtain

$$y_t = \varepsilon_t + \varphi_1 \varepsilon_{t-1} + \varphi_{21}\varepsilon_{t-2} + \varphi_{31}\varepsilon_{t-3} + \cdots$$

an MA(∞) process.

The order of an MA model can be find out using Autocorrelation Function (ACF).

4.12.4 ARIMA model:

If we combine differencing with auto regression and a moving average model, we obtain a ARIMA model. ARIMA is an acronym for Autoregressive Integrated Moving Average.

The full model can be written as

$$y'_t = c + \varphi_1 y'_{t-1} + \cdots + \varphi_p y'_{t-p} + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t \qquad (1)$$

where y'_t is the differenced series (it may have been differenced more than once).

The “predictors” on the right hand side include both lagged values of y_t and lagged errors.

We call this an **ARIMA(p, d, q) model**, where

p = order of the autoregressive part;

d = degree of first differencing involved;

q = order of the moving average part;

The same stationarity and invertibility conditions that are used for autoregressive and moving average models also apply to an ARIMA model.

Special cases of ARIMA models.	
White noise	ARIMA(0,0,0)
Random walk	ARIMA(0,1,0) with no constant
Random walk with drift	ARIMA(0,1,0) with a constant
Auto regression	ARIMA(p,0,0)
Moving average	ARIMA(0,0,q)

Once we start combining components in this way to form more complicated models, it is much easier to work with the backshift notation.

Equation (1) can be written in backshift notation as

$$\left(1 - \varphi_1 B - \cdots - \varphi_p B_p\right)(1 - B) d y_t$$

$$= \; c + (1 + \theta_1 B + \cdots + \theta_q B_q)\varepsilon_t$$

4.12.5 Autocorrelation analysis

Autocorrelation analysis is an important step in the Exploratory Data Analysis of time series forecasting. The autocorrelation analysis helps detect patterns and check for randomness. It’s especially important when you intend to use an autoregressive–moving-average (ARMA) model for forecasting because it helps to determine its parameters. The analysis involves looking at the Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) plots.

To figure out the **order of an AR model**, we need to **look at the PACF.** To figure out the **order of an MA model**, we need to **look at the ACF**. Both ACF and PACF assume stationarity of the underlying time series**.**

4.12.5.1 Autocorrelation Function (ACF)

Autocorrelation is the correlation between a time series with a lagged version of itself. The ACF starts at a lag of 0, which is the correlation of the time series with itself and therefore results in a correlation of 1.

From the ACF, you can assess the randomness and stationarity of a time series. You can also determine whether trends and seasonal patterns are present. For random data, autocorrelations should be near zero for all lags. The autocorrelation function declines to near zero rapidly for a stationary time series. the ACF drops slowly for a non-stationary time series.

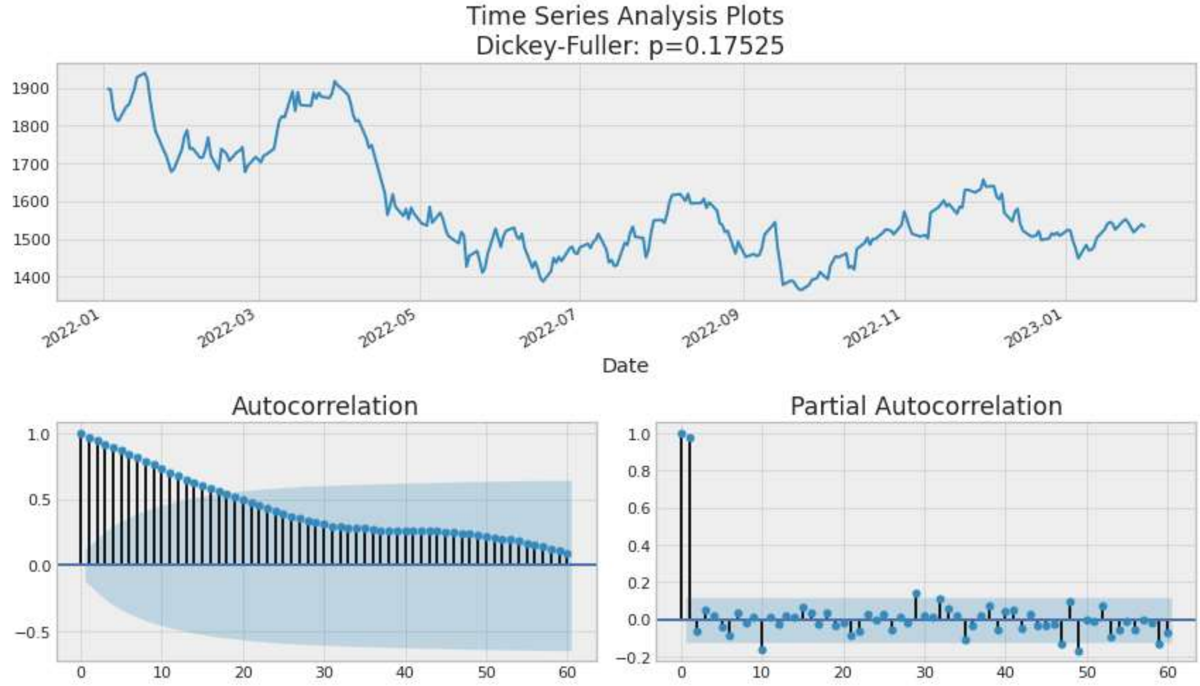
4.12.5.2 Partial autocorrelation function

The partial autocorrelation function is similar to the ACF except that it displays only the correlation between two observations that the shorter lags between those observations do not explain. The autocorrelation function helps assess the properties of a time series. In contrast, the partial autocorrelation function (PACF) is more useful during the specification process for an autoregressive model.

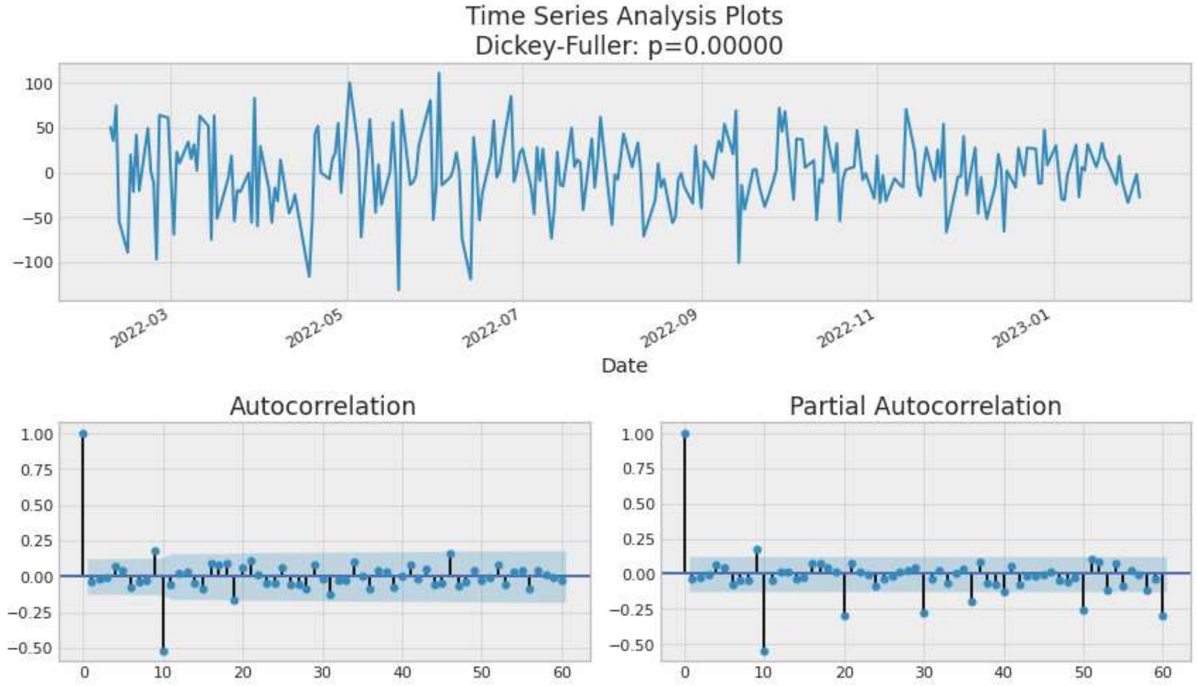
The partial autocorrelation, at lag k is the autocorrelation between y_t and y_{t+k} after eliminating the effect of y_{t+1}, … , y_{t+k-1}. that is not accounted for by lags 1 through k−1

4.12.5.3 Interpretation of ACF and PACF :

A stationary time series is one whose statistical properties such as mean, variance and covariance are all constant over time. By ADF test performing we can see that our data set has p value of 0.17525, which is greater than the level of significance 0.05 (i.e. p = 0.17525 > 0.05). Thus in such case, we accept our null hypothesis and conclude that our data set is non-stationary.



From above graph, the dataset is showing a decreasing trend and is not stationary. Additionally, we can also see a blue area in the ACF and PACF plots. This blue area depicts the 95% confidence interval and is an indicator of the significance threshold. It means that the points lying within this blue area in the ACF and PACF plots is said to be statistically in control and the points lying outside the blue area in the ACF and PACF plots are said to be statistically out of control. Finally we can conclude that our data is non-stationary and we need to perform differencing to make it stationary.



A non-stationary time series can be made stationary after applying differencing. Differencing can help stabilize the mean of a time series by removing changes in the level of a time series, and therefore eliminating trend and seasonality. Here in the above graph , we can see that after applying differencing our data set has now become stationary. It is now showing a constant mean, variance and covariance. Also the p value is less than 0.05, so we can conclude that, now our data is stationary.

Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) are often used to decide the number of Autoregressive (AR) and Moving Average (MA) lags for the ARIMA models.

The Partial Autocorrelation Function (PACF) plot can be used to draw a correlation between the time series and its lag. From the above Partial Autocorrelation plot, we can see that the 9th lag is significantly out of the limit so we can select the order of the p (AR) as 9.

To find out the value of q we can use the Autocorrelation Function (ACF) plot, which will tell us how much moving average is required to remove the autocorrelation from the stationary time series. Here, in the above auto correlation plot, we can see that 9th lag is out of the significance limit so we can say that the optimal value of q (MA) is 9.

4.13 Useful Metrics for study

Here we have included those metrics which are used in study for analyzing the performance of the models

4.13.1 Mean absolute percentage error (MAPE)

The mean absolute percentage error (MAPE), also known as mean absolute percentage deviation (MAPD), measures the accuracy of a method for constructing fitted time series values in statistics. The

MAPE is the sum of the individual absolute forecast errors, divided by the actual values for each period. It's an accuracy measure based on the relative percentage of errors. The closer the MAPE value is to zero, the better is the predictions.

The mean absolute percentage error (MAPE) is defined as follows:

$$MAPE = \frac{100}{N} \times \sum_{i=1}^{N} \left| \frac{x_i - \widehat{x_i}}{x_i} \right|$$

Where :

- $\{x_i\}$ is the actual observations time series
- $\{\widehat{x_i}\}$is the estimated or forecasted time series
- N is the number of non-missing data points

4.13.2 Akaike's Information Criteria

The Akaike Information Critera (AIC) is a widely used measure of a statistical model. When comparing two models, the one with the lower AIC is generally "better". Akaike's Information Criterion (AIC), which is useful in selecting predictors for regression, is also useful for determining the order of an ARIMA model. It can be written as

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

where L is the likelihood of the data,

$$k = 1 \ \ if \ \ c \neq 0 \ \ and \ \ k = 0 \ \ if \ \ c = 0.$$

Note that the last term in parentheses is the number of parameters in the model (including σ^2, the variance of the residuals).

For ARIMA models, the corrected AIC can be written as

$$AIC_c = AIC + \frac{2(p+q+k+1)(p+q+k+2)}{T-p-q-k-2}$$

and the Bayesian Information Criterion can be written as

$$BIC = AIC + [\log(T)-2]\ (p+q+k+1)$$

Good models are obtained by minimising the AIC, AIC_cor BIC.

It is important to note that these information criteria tend not to be good guides to selecting the appropriate order of differencing *(d)* of a model, but only for selecting the values of p and q. This is because the differencing changes the data on which the likelihood is computed, making the AIC values between models with different orders of differencing not comparable.

4.13.3 Bayesian Information Criterion (BIC)

Bayesian information criterion (BIC) is a criterion for model selection among a finite set of models. It is based on the likelihood function, and it is closely related to Akaike information criterion (AIC). When fitting models, it is possible to increase the likelihood by adding parameters, but doing so may result in overfitting. The BIC resolves this problem by introducing a penalty term for the number of parameters in the model. BIC has been widely used for model identification in time series and linear regression.

Mathematically BIC can be defined as-

$$BIC = ln(n)k - 2ln(\hat{L})$$

Where, $\hat{L}$ is the maximized value of the likelihood function of the model. n is the number of data points. K is the number of free parameters to be estimated.

The only difference between AIC and BIC is that, BIC considers the number of observations in the formula, whereas AIC does not. Though BIC is always higher than AIC, lower the value of these two measures, better the model.

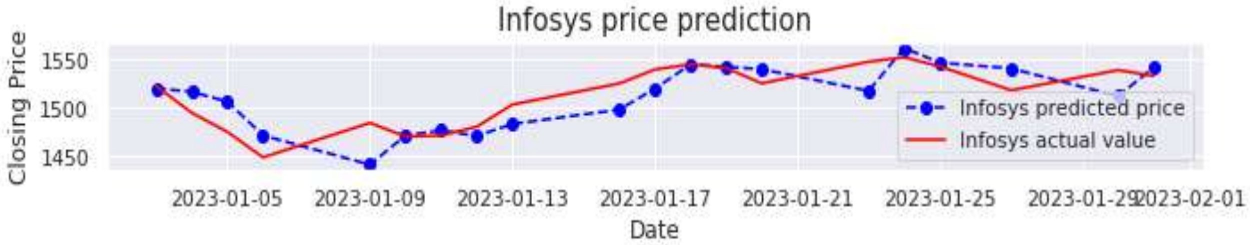
4.14 Interpretation of ARIMA

In autoregression model (regression of the time series onto itself). The basic assumption is that the current series values depend on its previous values with some lag (or several lags). The maximum lag in the model is referred to as p. To determine the initial p, need to look at the PACF plot and It is found that the value of p is 9. In moving average model, the error depends on the previous with some lag, which is referred to as q. The initial value can be found on the ACF plot is 9. Order of integration is simply the number of nonseasonal differences needed to make the series stationary. In our case, it’s just 1 because we used first differences. It is denoted by d. The parameter of ARIMA model is (p, d, q) which is equal to (9,1,9) in accordance with given dataset.

The objective of this project is to predict the stock closing price of the January-2023 month of Infosys company with the help of past data of 1 year only by using ARIMA model. First, the objective is to train the past 1 year dataset so that we can predict the future values of January month. After getting the order of ARIMA model which is (9,1,9), model is fitted to training dataset. In background, just we get the parameter of ARIMA model (9,1,9), in search of the best model among possible combination of this parameters, we use the Akaike's information criteria, we select the best model whose AIC information is less among all to predict the future values. We Perform stepwise search to minimize AIC mentioning some results as follows

```
ARIMA(1,1,1)(0,0,0)[0] intercept   : AIC=2358.597, Time=0.35 sec
ARIMA(0,1,0)(0,0,0)[0] intercept   : AIC=2358.301, Time=0.03 sec
ARIMA(1,1,0)(0,0,0)[0] intercept   : AIC=2359.440, Time=0.09 sec
ARIMA(0,1,1)(0,0,0)[0] intercept   : AIC=2359.412, Time=0.13 sec
ARIMA(0,1,0)(0,0,0)[0]             : AIC=2357.022, Time=0.02 sec
```

It Is observed that the Best model with ARIMA model parameter (0,1,0) whose AIC value is 2357.022 that is low among all combinations. It includes 249 observation to be trained. These observation are helpful to predict future values. These are shown in following graph_



It is observed that around 1.0 percentage MAPE (mean absolute percentage Error) implies the model is about 99% accurate in predicting the test set observations. The red line is showing the Infosys actual value whereas the blue line is showing Infosys predicted price. Above graph is showing how the Infosys predicted price and Infosys actual price is matching.

4.15 Diagnosis of ARIMA Model

ARIMA model and identifying any problems or issues with the model's assumptions or specifications. The aim of diagnostics is to ensure that the ARIMA model is a good fit for the data and that the model's assumptions should be satisfied. Conducting various diagnostic tests is an important step in time series modelling.

Each observation in a time series can be forecast using all previous observations. The “residuals” in a time series model are what is left over after fitting a model. For many (but not all) time series models, the residuals are equal to the difference between the observations and the corresponding fitted values:

$$e_t = y_t - \widehat{y_t}$$

Residuals are useful in checking whether a model has adequately captured the information in the data. A good forecasting method will yield residuals with the following properties:

1. The residuals are uncorrelated. If there are correlations between residuals, then there is information left in the residuals which should be used in computing forecasts.
2. The residuals have zero mean. If the residuals have a mean other than zero, then the forecasts are biased.

Any forecasting method that does not satisfy these properties can be improved. It is possible to have several different forecasting methods for the same data set, all of which satisfy these properties. Checking these properties is important in order to see whether a method is using all of the available information.

If either of these properties is not satisfied, then the forecasting method can be modified to give better forecasts.

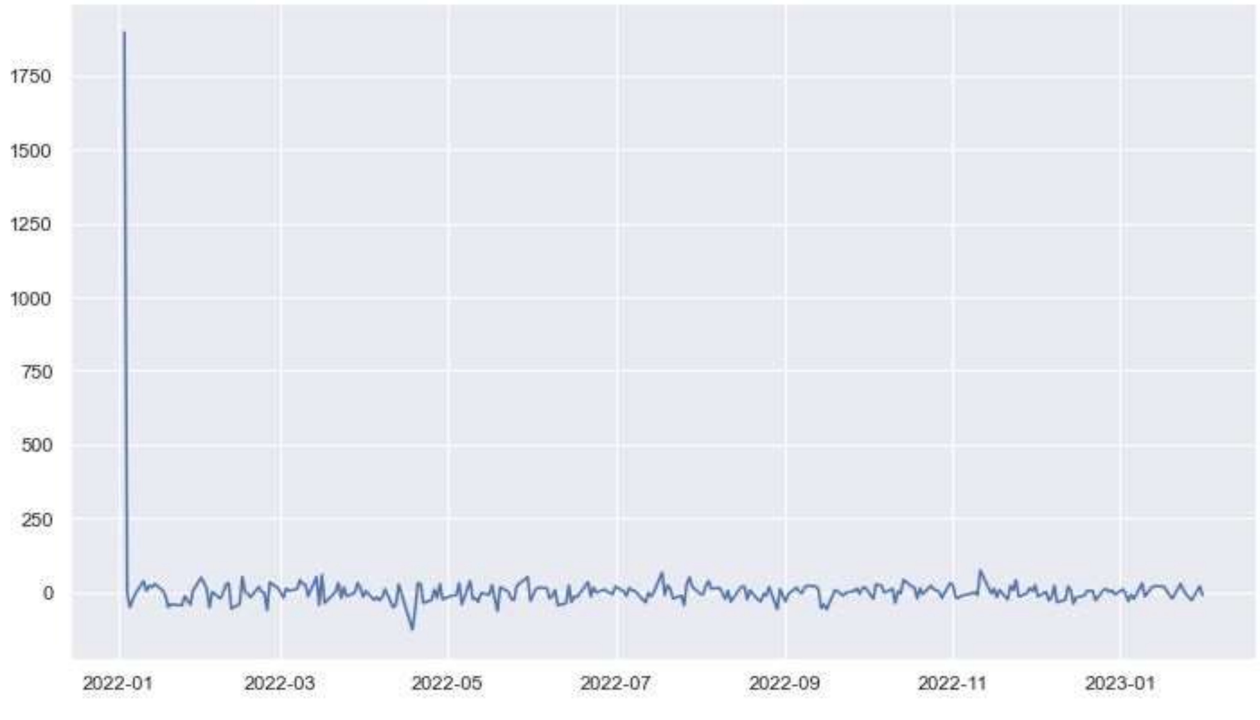
In addition to these essential properties, it is useful (but not necessary) for the residuals to also have the following two properties.

1. The residuals have constant variance.
2. The residuals are normally distributed.

The above two properties make the calculation of prediction intervals easier.

4.15.1 Residual analysis

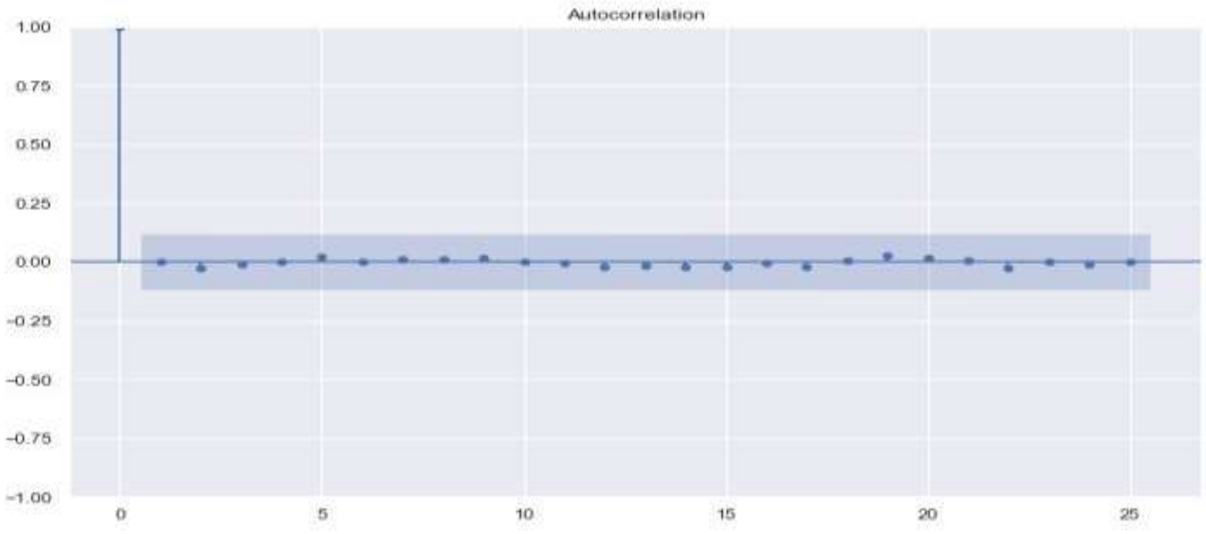
This involves analyzing the residuals (i.e., the difference between the observed values and the predicted values) of the ARIMA model. The residuals should exhibit no systematic pattern, no significant autocorrelation, and no significant heteroscedasticity.

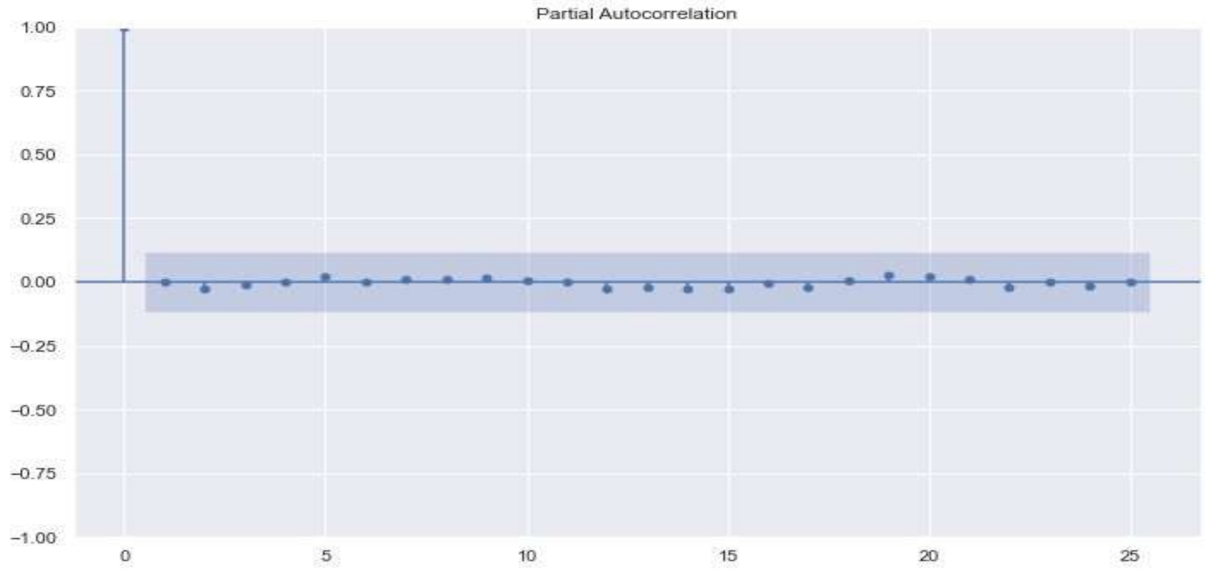


The above graph is showing that the residuals are stationary with mean zero and constant variance. This means that ARIMA model is good fit for the data, and residuals which are random error does not show any kind of trend or pattern in the data.

4.15.2 ACF and PACF plots:

The autocorrelation function (ACF) and partial autocorrelation function (PACF) plots are used to assess the presence of autocorrelation in the residuals. The ACF plot should show no significant autocorrelation, while the PACF plot should show a sharp cut off at lag p in an AR (p) model.





From the ACF and PACF plot of residuals, we can see that all residuals are lying within the blue area of ACF and PACF plots. The mean of the residuals is close to zero and there is no significant correlation in the residuals series. Thus we can say that all the residuals are statistically in control and the residuals are stationary.

4.15.3 With the help of Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC)

These are information criteria that can be used to compare different ARIMA models based on their goodness of fit and complexity. The model with the lowest AIC and BIC value is generally considered to be the best-fitting model which are equal to calculated value 2357.022 and 2357.022 respectively.

5. Conclusion

The Infosys dataset was used. Exploratory Data Analysis, Decomposition of time series, Removing stationarity, Use of augmented dickey fuller test and Implementation of ARIMA are done and came to conclusion that-

1. The Closing stock price of Infosys company is showing decreasing trend. As a result, given series is not stationary as It may be affected by many factors.

2. Augmented Dickey fuller test is used to check given series is stationary or not. It is observed the statistic value (-2.289897) is greater than a critical value (-2.87236) at 5% level of significance or p values (0.175247) is greater than 0.05 (level of Significance). Therefore, we accept the null hypothesis and concluded that the given series is non-stationary.

3. To make the series stationary, differencing is used. It is observed that (p, d, q) = (9,1,9) is found parameter from the given data. The value of p and q is observed from the ACF and PACF graph. Model is fitted with this parameter, observed that, Around 1.0 percentage MAPE (mean absolute percentage Error) implies the model is about 99% accurate in predicting the test set observations.

4. In diagnosis of ARIMA model, It is observed residuals are stationary with mean zero and constant variance. This means that ARIMA model

is good fit for the data, and residuals does not show any kind of trend or pattern in the data. The mean of the residuals is close to zero and there is no significant correlation in the residuals series. All the residuals are statistically in control and the residuals are stationary by observing ACF and PACF plots. The model with the lowest AIC and BIC value is generally considered to be the best-fitting model.

6. Limitation of Study

Like other empirical studies, This study has its limitations. It may be limitation as mentioned below_

1. Only one year of Infosys stock’s dataset is considered for the study purpose due to time circumstance. The study can be strengthen by increasing size of given dataset.
2. we have studied stock data, It is observed that observation of weekends are not available as stock market is closed in weekends.
3. Other affecting factors on stock prices are not considered deeply while interpreting data visualization.

In simple words, ARIMA models are based on historical data and assume that the future will follow the same pattern as the past. However, stock prices are influenced by a wide range of unpredictable factors such as economic news, geopolitical events, and corporate earnings reports. ARIMA models may not be able to capture these external factors, leading to limited predictive power.

7. Future Scope:

The objective of this project is to predict the stock closing price of the January-2023 month of Infosys company with the help of past data of 1 year only by using ARIMA model. In this project, Time series analysis is used as predicative model. However other predicative can be used for predicating Infosys closing price of January. We have taken only 1 year dataset to predict the future values due to some circumstance, however, By taking the large dataset and by considering the other affecting factors, Study can be done for predicating Infosys closing price to reduce the error.

8. Challenges:

Some were the challenges that we faced while doing this project we learnt lots of things to tackle the challenges that made us more effective learners.

1. Interpreting the decreasing trend of Infosys stock was the challenges however we tried to study for getting into proper interpretation by considering some affecting factors like social news, company information etc as much as we could.

2. Use of proper test to remove the stationarity, Implementing the ARIMA model and Study of Time series analysis and Python language while performing simultaneously were the challenge. To overcome these challenges we divided the work in teammates, understanding the concepts from one another.

Bibliography

1. S.C. Gupta & V.K. Kapoor, (2007), Fundamentals of Applied Statistics; Sultan Chand & Sons Educational Publishers, Volume 4, chapter 2: Analysis of Time Series, 2.1 – 2.80.
2. Medhi.j. (July 1984), stochastic processes, published by Mohinder Singh Sejwal for wiley Eastern limited, (223-240)
3. Brockwell, P.J. and Davis, R. A. (2003). Introduction to Time Series Analysis, Springer
4. Chatfield, C. (2001). Time Series Forecasting, Chapmann & Hall, London
5. Fuller, W. A. (1996). Introduction to Statistical Time Series, 2nd Ed. Wiley.
6. Lutkepohl, H. and Kratzing, M. (Ed.) (2004). Applied Time Series Econometrics, Cambridge University Press.
7. Shumway, R. H.and Stoffer D. S. (2010). Time Series Analysis & Its Applications, Springer.
8. Tsay, R. S. (2010). Analysis of Financial Time Series, Wiley.
9. Montgomery, D.C & Johnson, L.A (1977): Forecasting and Time Series Analysis, McGraw Hill
10. Exploratory Data Analysis (EDA) - Types and Tools - GeeksforGeeks
11. Time Series Forecast with Excel. The Forecast Sheet predicts using the… | by @imVivRan |Analytics Vidhya | Medium
12. Data Analysis with Excel: A Complete Guide for Beginners - Udemy Blog
13. https://en.wikipedia.org/wiki/Movingaverage_model#:~:text=In%20time%20series%20analysis%2C%20the,identical%20to%20itself%20random%2Dvariable.
14. https://allthingsstatistics.com/miscellaneous/advantages-disadvantages-moving-averages-method/
15. https://groww.in/p/moving-averages
16. https://corporatefinanceinstitute.com/resources/capital-markets/weighted-moving average-wma/
17. https://www.thebalancemoney.com/simple-exponential-and-weighted-moving-averages-1031196
18. https://towardsdatascience.com/stationarity-in-time-series-analysis-90c94f27322

19. https://www.statisticshowto.com/stationarity/
20. https://people.duke.edu/~rnau/411diff.htm
21. https://www.itl.nist.gov/div898/handbook/pmc/section4/pmc442.htm
22. Stationarity and detrending (ADF/KPSS) — statsmodels
23. https://otexts.com/fpp2/arima.html
24. Interpreting ACF and PACF Plots for Time Series Forecasting | by Leonie Monigatti | Towards Data Science
25. https://otexts.com/fpp2/residuals.html
26. https://medium.com/@analyttica/what-is-bayesian-information-criterion-bic-b3396a894be6
27. https://analyzingalpha.com/interpret-arima-results

Annexure

Project Name: “Infosys Stock Price Prediction using ARIMA Model”

# Importing Libraries

import numpy as np # (mathematical operations); import pandas as pd #(Data frame); import matplotlib.pyplot as plt #(data Visualizations); import seaborn as sns # (Advanced data visualizations):import warnings # (`do not disturb` mode);warnings.filterwarnings('ignore');import numpy as np # (vectors and matrices);import pandas as pd#(tables and data manipulations);import matplotlib.pyplot as plt # (plots);%matplotlib inline;import seaborn as sns more plots from dateutil.relativedelta import relativedelta # (working with dates with style);from scipy.optimize import minimize# (for function minimization);import statsmodels.formula.api as smf # (statistics and econometrics);import statsmodels.tsa.api as smt;import statsmodels.api as sm;import scipy.stats as scs;from itertools import product; # (some useful functions);from tqdm import tqdm_notebook;from statsmodels.tsa.seasonal import seasonal_decompose;%matplotlib inline; from sklearn.metrics import r2_score, median_absolute_error, mean_absolute_error; from sklearn.metrics import median_absolute_error, mean_squared_error, mean_squared_log_error; from sklearn.metrics import r2_score, median_absolute_error, mean_absolute_error; from sklearn.metrics import median_absolute_error, mean_squared_error, mean_squared_log_error;

# Dataset Loading

```
from google.colab import drive

drive.mount('/content/drive')

path = '/content/drive/MyDrive/Colab Notebooks/capstone_project_02/infosys 1 year data.xlsx'

dataset = pd.read_excel(path)
```

# Dataset First look; dataset; dataset.shape; Dataset.info() # Dataset Duplicate Value Count dataset.duplicated().value_counts()# Missing Values/Null Values Count; dataset.isnull().sum() # Understanding Variables dataset.head(); dataset.describe()

Data visualization of different variables through line graph, scatter plots, Histogram with bell shaped curve, Box-Whisker plots, correlation heatmap, pair plots.

Time Series Analysis

Plot dependent variable closing price:

plt.rcParams['figure.figsize']=(10,5); plt.plot(dataset['Close'], color= 'r'); plt.title('Closing price with date'); plt.xlabel('Date'); plt.ylabel('Closing price'); plt.grid(linestyle=':', linewidth = '0.5', color = 'b') # to display and customize gridlines on a plot.; plt.show()

Moving Average:

```
def mean_absolute_percentage_error(y_true, y_pred):

    return np.mean(np.abs((y_true - y_pred) / y_true)) * 100

# moving Average

def moving_average(series, n):

  # calculate average of last n observations

  return np.average(series[-n:])

moving_average(df1, 5)

#moving_average(df1, 1)
```

```
def plotMovingAverage(series, window1, window2, plot_intervals=False, scale=1.96, plot_anomalies=False):
  """series - dataframe with timeseries; window - rolling window size ;plot_intervals - show confidence
intervals; plot_anomalies - show anomalies """
    rolling_mean1 = series.rolling(window=window1).mean()
    rolling_mean2=series.rolling(window=window2).mean()
    plt.figure(figsize=(15,5))
    plt.title("Moving average")
    plt.plot(rolling_mean1, "g", label="5 days moving average")
    plt.plot(rolling_mean2, label="10 days moving average")
# Plot confidence intervals for smoothed values
    if plot_intervals:
        mae = mean_absolute_error(series[window2:], rolling_mean2[window2:])
        deviation = np.std(series[window2:] - rolling_mean2[window2:])
        lower_bond = rolling_mean2 - (mae + scale * deviation)
        upper_bond = rolling_mean2 + (mae + scale * deviation)
        plt.plot(upper_bond, "r--", label="Upper Bond / Lower Bond")
        plt.plot(lower_bond, "r--")
        # Having the intervals, find abnormal values
        if plot_anomalies:
            anomalies = pd.DataFrame(index=series.index, columns=series.columns)
            anomalies[series<lower_bond] = series[series<lower_bond]
            anomalies[series>upper_bond] = series[series>upper_bond]
            plt.plot(anomalies, "ro", markersize=10)
    plt.plot(series[window2:], label="Actual values")
    plt.legend(loc="upper left")
    plt.grid(True)
plotMovingAverage(df1,5,10, plot_intervals=True)
```

Decomposition of time Series Data:

```
plt.rcParams['figure.figsize']=(10,6); df1_mul_decompose = seasonal_decompose(df1, model="multiplicative",
period = 10); df1_mul_decompose.plot(); plt.show()
```

Weighted Average

```
def weighted_average(series, weights):
    """ Calculate weighted average on the series Assuming weights are sorted in descending order
        (larger weights are assigned to more recent observations). """
    result = 0.0
    for n in range(len(weights)):
        result += series.iloc[-n-1] * weights[n]
    return float(result)
weighted_average(df1, [0.3,0.2, 0.2,0.1, 0.05,0.05,0.025,0.025,0.025,0.025 ] )
```

Exponential smoothing:

```
def mean_absolute_percentage_error(y_true, y_pred):
    return np.mean(np.abs((y_true - y_pred) / y_true)) * 100
# Exponential smoothing
def exponential_smoothing(series, alpha):
    """series - dataset with timestamps
        alpha - float [0.0, 1.0], smoothing paramete """
    result = [series[0]] # first value is same as series
    for n in range(1, len(series)):
        result.append(alpha * series[n] + (1 - alpha) * result[n-1])
    return result
def plotExponentialSmoothing(series, alphas):
    """Plots exponential smoothing with different alpha   series - dataset with timestamps alphas - list of floats,
smoothing parameters """
with plt.style.context('seaborn-white'):
        plt.figure(figsize=(15, 7))
        for alpha in alphas:
            plt.plot(exponential_smoothing(series, alpha), label="Alpha {}".format(alpha))
        plt.plot(series.values, "c", label = "Actual")
        plt.legend(loc="best")
        plt.axis('tight')
        plt.title("Exponential Smoothing")
        plt.grid(True);
plotExponentialSmoothing(df1['Close'], [0.4,0.3, 0.2,0.05,0.05])
```

Double exponential Smoothing (Holt’s linear trend method):

```
def double_exponential_smoothing(series, alpha, beta):
    ""series - dataset with timeseries alpha - float [0.0, 1.0], smoothing parameter for level beta - float [0.0, 1.0],
smoothing parameter for trend"""
    # first value is same as series
    result = [series[0]]
    for n in range(1, len(series)+1):
        if n == 1:
            level, trend = series[0], series[1] - series[0]
        if n >= len(series): # forecasting
            value = result[-1]
        else:
            value = series[n]
        last_level, level = level, alpha*value + (1-alpha)*(level+trend)
        trend = beta*(level-last_level) + (1-beta)*trend
        result.append(level+trend)
    return result
def plotDoubleExponentialSmoothing(series, alphas, betas):
    """Plots double exponential smoothing with different alphas and bet  series - dataset with timestamps
alphas - list of floats, smoothing parameters for level betas - list of floats, smoothing parameters for trend"""
 with plt.style.context('seaborn-white'):
        plt.figure(figsize=(20, 8))
        for alpha in alphas:
            for beta in betas:
                plt.plot(double_exponential_smoothing(series, alpha, beta), label="Alpha {}, beta {}".format(alpha,
beta))
        plt.plot(series.values, label = "Actual")
        plt.legend(loc="best")
        plt.axis('tight')
        plt.title("Double Exponential Smoothing")
        plt.grid(True)
plotDoubleExponentialSmoothing(df1['Close'], alphas=[0.5, 0.02], betas=[0.5, 0.02])
```

To check the Stationarity:

```
# Dickey fuller test: # null Hypo: There is no stationarity. # Augmented Dickey Fuller Test test.
from statsmodels.tsa.stattools import adfuller
def test_stationarity(timeseries):
  #Determing rolling statistics
  rolmean = pd.Series(timeseries).rolling(window=10).mean()
  rolstd = pd.Series(timeseries).rolling(window=10).std()
 # plot rolling statistics
```

```
  # timeseries
  orig=plt.plot(timeseries, color='blue', label='Original')
  # mean
  mean = plt.plot(rolmean, color='red', label = 'Moving Mean')
  # Std-dev
  std = plt.plot(rolstd, color = 'black', label='Moving std')
  plt.legend(loc='best')
  plt.title('Moving Mean & Standard Deviation', size = 20)
  plt.xlabel('Date', size=15)
  plt.ylabel('Closing price', size=15)
  plt.show()
  # perform Dickey-Fuller test:
  print('Results of Dickey-Fuller Test:')
  dftest = adfuller(timeseries, autolag='AIC')
  dfoutput=pd.Series(dftest[0:4], index=['Test Statistic', 'P-value', '#lags used','Number of observations used'])
  for key, value in dftest[4].items():
    dfoutput["Critical Value (%s)"%key]=value
  print(dfoutput)
test_stationarity(df1['Close'])
```

Getting rid of non-stationarity

```
def tsplot(y, lags=None, figsize=(12, 7), style='bmh'):
    if not isinstance(y, pd.Series):
        y = pd.Series(y)
with plt.style.context(style):
        fig = plt.figure(figsize=figsize)
 layout = (2, 2)
        ts_ax = plt.subplot2grid(layout, (0, 0), colspan=2)
        acf_ax = plt.subplot2grid(layout, (1, 0))
        pacf_ax = plt.subplot2grid(layout, (1, 1))
         y.plot(ax=ts_ax)
        p_value = sm.tsa.stattools.adfuller(y)[1]
        ts_ax.set_title('Time Series Analysis Plots\n Dickey-Fuller: p={0:.5f}'.format(p_value))
        smt.graphics.plot_acf(y, lags=lags, ax=acf_ax)
        smt.graphics.plot_pacf(y, lags=lags, ax=pacf_ax)

        plt.tight_layout()

tsplot(df1['Close'], lags=60); close_diff = df1['Close'] - df1['Close'].shift(10); tsplot(close_diff[10:], lags=60);
close_diff = close_diff - close_diff.shift(1); tsplot(close_diff[24+1:], lags=60)
```

5. ARIMA Modelling

```
from statsmodels.tsa.arima_model import ARIMA ;from sklearn.metrics import mean_squared_error,
mean_absolute_error # train test split; to_row= int(len(df1['Close'])*0.928); training_data =
list(df1['Close'][0:to_row]); testing_data=list(df1['Close'][to_row:]); #split data into train and training set
plt.figure(figsize=(20,6)); plt.grid(True); plt.xlabel('Dates'); plt.ylabel('Closing prices');
plt.plot(df1['Close'][0:to_row],'green', label="Train data"); plt.plot(df1['Close'][to_row:], 'blue', label='Test
data'); model_predictions = []; n_test_obser=len(testing_data)# n_test_obser

for i in range(n_test_obser):

  model = ARIMA(training_data[:], order = (9,1,9))

  model_fit = model.fit()

  output = model_fit.forecast()

  yhat=list(output[0])[0]

 model_predictions.append(yhat)

  actual_test_value = testing_data[i]

  training_data.append(actual_test_value)

 #print(output) #break

print(model_fit.summary())

plt.figure(figsize=(10,2))

plt.grid(True)

data_range = df1['Close'][to_row:].index

plt.plot(data_range, model_predictions, color = 'blue', marker ='o', linestyle= 'dashed', label = 'Infosys
predicted price')

plt.plot(data_range, testing_data, color = 'red', label= 'Infosys actual value')

plt.title('Infosys price prediction', size = 15)

plt.xlabel('Date', size = 12)

plt.ylabel('Closing Price', size = 12)

plt.legend()

plt.show()

# report performance

mape = np.mean(np.abs(np.array(model_predictions[:])- np.array(testing_data))/np.abs(testing_data))

print("MAPE: " + str(mape))# mean absolute percentage error

# Around 1.0 percentage MAPE (mean absolute percentage Error) implies the model is about 99% accurate in
predicting the test set observations
```