SVKM'S NMIMS, MPSTME, Shirpur Campus

INTERIM REPORT

INTERIM REPORT

PROJECT: **STOCK MARKET FORECAST**

Faculty Mentor:
Ms Varsha Nemade
(Assistant Professor)
MPSTME,
NMIMS Shirpur

Submitted By:
Dipanshu Agarwal – N201
Riya Airen – N204
Saurabh Ajit – N205
Campus Course: MBATECH CE
Batch: 2017 – 2022

A report submitted in partial fulfilment of the requirements of 5 Years Integrated MBA (Tech) Program of Mukesh Patel School of Technology Management & Engineering, NMIMS.

ABSTRACT

Predicting the Stock Market has been the bane and goal of investors since its existence. Everyday billions of dollars are traded on the exchange, and behind each dollar is an investor hoping to profit in one way or another. Entire companies rise and fall daily based on the behaviour of the market. Should an investor be able to accurately predict market movements, it offers a tantalizing promises of wealth and influence. In the real world, the stock market predictions can be categorized in 2 parts, Fundamental Analysis and Technical Analysis.

In this undertaking, we will be creating a supervised machine learning model which will help us to somewhat predict the price value of stocks/security of a company i.e. State Bank of India to be specific. The Model will be using Time-Series Analysis, Time series is a set of observations or data points taken at specified time usually at equal intervals and it's used to predict the future values based on the previous observed values.

The stock data for the past 9 years (2011-2019) has been collected and trained using SARIMA model with different parameters. The proposed model was applied to dataset to forecast the future value of Sate bank of India (SBI). The results will be used to analyse the stock prices and their prediction in depth in future research efforts.

INTRODUCTION

OBJECTIVE:

In the past decades, there is an increasing interest in predicting markets among economists, policymakers, academics and market makers. The objective of the proposed work is to study and implement the supervised learning algorithm to predict the stock price.

TECHNICAL OBJECTIVE

The technical objectives will be implemented in Python. The system must be able to access a list of historical prices. It must calculate the estimated price of stock based on the historical data. It must also provide an instantaneous visualization of the market index.

PURPOSE:

The stock market appears in the news every day. Every time it reaches a new high or a new low. The rate of investment and business opportunities in the Stock market can increase if an efficient algorithm could be devised to predict the short term price of an individual stock.

We will see if there is a possibility of devising a model using time series which will predict stock price with a less percentage of error. And if the answer turns to be **YES**, we will also see how reliable and efficient will this model be.

- to add to the academic understanding of stock market prediction
- this project will focus exclusively on predicting the daily trend (price movement) of individual stocks
- the project will also analyse the accuracies of these predictions

MAIN TEXT

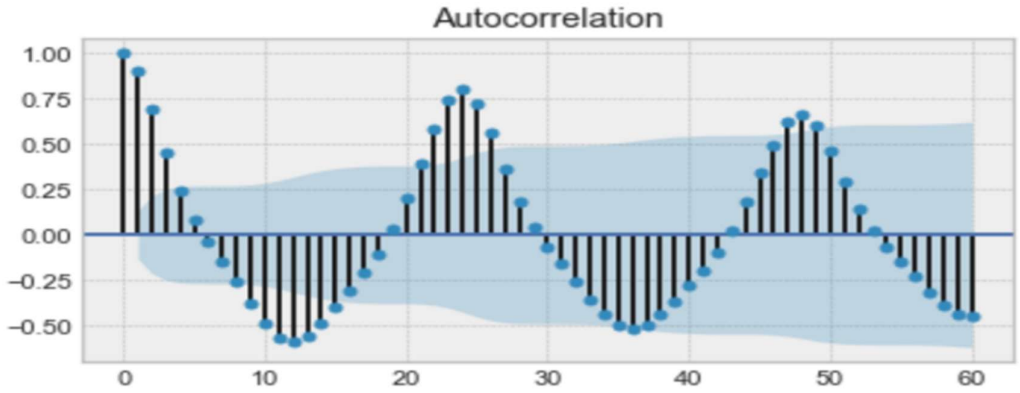
TIME-SERIES

It is simply a series of data points ordered in time. In a time series, time is often the independent variable and the goal is usually to make a forecast for the future.

The Time-Series generated may have any of the 3 properties:-

AUTO-CORRELATION

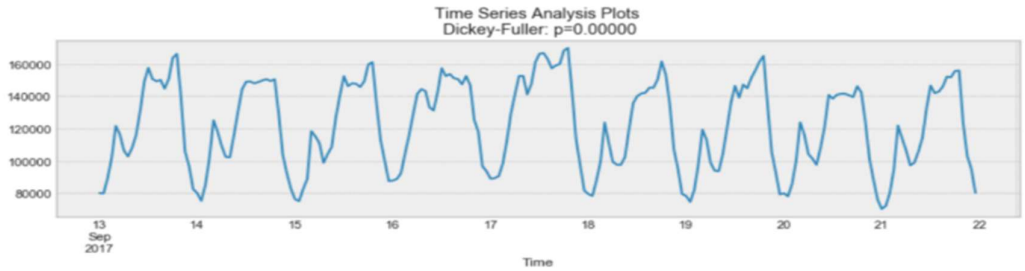
Refers to the similarity between observations as a function of the time lag between them. For example, in the graph below the first and the 24th value have a high autocorrelation similarly for the 12th and 36th value



Example of an autocorrelation plot

SEASONALITY

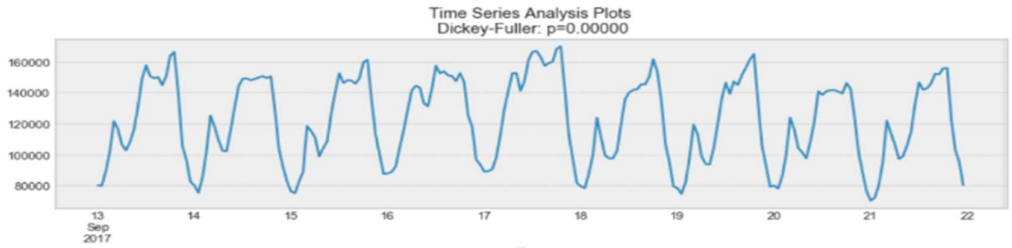
Refers to periodic functions, for example electricity consumption is high during the day and low during night, or online sales increase during Christmas before slowing down again. Seasonality can also be derived from an **Auto-Correlation Plot** if it has a sinusoidal shape.



Example of seasonality

STATIONARY

Refers to an important characteristic of time-series. A time-series is said to be stationary if its statistical properties do not change over time. In other words, it has constant mean and variance, and co-variance is independent of time.



Example of a stationary process

In the above, we see that the process above is stationary. The mean and variance do not vary over time.

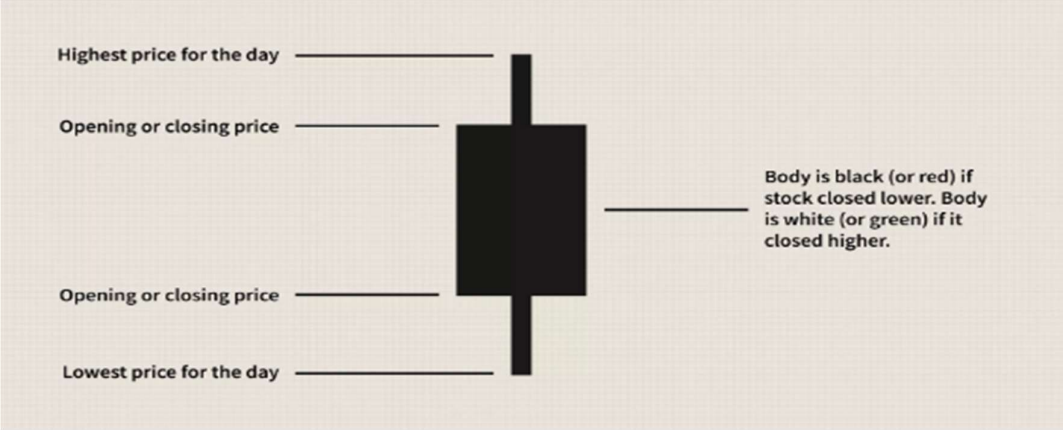
Often **Stock Prices** are not a stationary process, since we might see a growing trend, or its volatility might increase over time i.e. variance is ever-changing.

Ideally, we want to have a stationary time-series for modelling. Of course, not all of them are stationary, but we can make different transformations to make them stationary.

FUNDAMENTALS OF TRADING

In the above graph,

we see a peculiar representation technique being used which is known as a candlestick, a candlestick is a type of price chart used in technical analysis that displays the high, low, open, and closing prices of a security for a specific period.



The Color of the Bar i.e. Red or Green denotes that the stock closed on a Lower price or a Higher Price respectively on that particular day.

There is a third state which is known as a consolidated state, in which the number of buyers and sellers for a security are the same in the market. In this case, it is denoted using a simple horizontal line.

The Main Factors Which Bring About A Huge Change In The Variance And Mean Of The Security Prices Are:-

1. **Buyer and Seller: -** From the colours, we can also conclude that Red or Black means there are more number of buyers in the market as they will try to drive the price of security down so as to buy it at a low cost, and Green or White means that the numbers of seller of that particular security is more as they will try to drive the price higher so as to earn maximum profit on selling the security.
2. **IPO: -** stand for Initial Public Offering. When the news media report that a company is "going public," this **means** that company is making an **initial public offering**. This **means** that the company is offering its shares for sale to the public for the first time.
3. **Pandemics: -** External shocks can derail economic trends and abruptly alter market sentiment. Not all risk is economic policy or monetary

4. **Merger and Acquisitions: -** Mergers and acquisitions are transactions in which the ownership of companies, other business organizations, or their operating units are transferred or consolidated with other entities.
5. **Import and Export: -** An economic condition that occurs when a country is importing more goods than it is exporting is called trade deficit. It is also referred to as net exports. A trade deficit is calculated by deducting value of goods being exported from the goods being imported by a country. It is mentioned in the currency that a particular country uses. An extended trade deficit can adversely impact a country's economy and its stock markets. A country with extended trade deficit means it's into debt. The investors take notice of such financial parameters and also take note of plunge in spending on locally produced goods. This hurts domestic producers and their share prices. This reduces demand in the domestic stock markets and result in decline of market.
6. **Government Issues/Changes to Laws: -** The laws and ordinance passed by the government may stimulate the market or it may even bring a pause on the Foreign Investments happening in the market on the basis of the current situations of the global economy and trade treaties.

DETAILED OUTLINE

TECHNOLOGIES

PYTHON

It has been chosen as the language of choice for this project. This was an easy decision for the multiple reasons.

1. Python as a language has an enormous community behind it. Any problems that might be encountered can be easily solved with a trip to Stack Overflow. Python is among the most popular languages on the site which makes it very likely there will be a direct answer to any query.
2. Python has an abundance of powerful tools ready for scientific computing. Packages such as Numpy, Pandas, and Scikit are freely available and well documented. Packages such as these can dramatically reduce, and simplify the code needed to write a given program. This makes iteration quick.
3. Python as a language is forgiving and allows for programs that look like pseudo code. This is useful when pseudocode given in academic papers needs to be implemented and tested. Using Python, this step is usually reasonably trivial.

However, Python is not without its flaws. The language is dynamically typed and packages are notorious for Duck Typing. This can be frustrating when a package method returns something that, for example, looks like an array rather than being an actual array. Coupled with the fact that standard Python documentation does not explicitly state the return type of a method, this can lead to a lot of trials and error testing that would not otherwise happen in a strongly typed language. This is an issue that makes learning to use a new Python package or library more difficult than it otherwise could be.

NUMPY

It is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays. It is the fundamental package for scientific computing with Python. Besides its obvious scientific uses, Numpy can also be used as an efficient multi-dimensional container of generic data.

PANDAS

It is an open-source library that is built on top of NumPy library. It is a Python package that offers various data structures and operations for manipulating numerical data and time series. It is mainly popular for importing and analyzing data much easier. Pandas is fast and it has high-performance & productivity for users.

MATPLOTLIB

Matplotlib is an amazing visualization library in Python for 2D plots of arrays. Matplotlib is a multi-platform data visualization library built on NumPy arrays and designed to work with the broader SciPy stack. It was introduced by John Hunter in the year 2002.

One of the greatest benefits of visualization is that it allows us visual access to huge amounts of data in easily digestible visuals. Matplotlib consists of several plots like line, bar, scatter, histogram etc.

SCIKIT-LEARN

Scikit-learn is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support vector machine, random forest, gradient boosting, k-means etc. It is mainly designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.

Scikit-learn is largely written in Python, with some core algorithms written in Cython to achieve performance. Support vector machines are implemented by a Cython wrapper around LIBSVM .i.e., logistic regression and linear support vector machines by a similar wrapper around LIBLINEAR.

SCIPY

SciPy refers to several related but distinct entities:

- The *SciPy ecosystem*, a collection of open source software for scientific computing in Python.
- The *community* of people who use and develop this stack.
- Several *conferences* dedicated to scientific computing in Python - SciPy, EuroSciPy, and SciPy.in.
- The SciPy library, one component of the SciPy stack, providing many numerical routines.

STATSMODELS

It is a Python module that provides classes and functions for the estimation of many different statistical models, as well as for conducting statistical tests, and statistical data exploration. An extensive list of result statistics

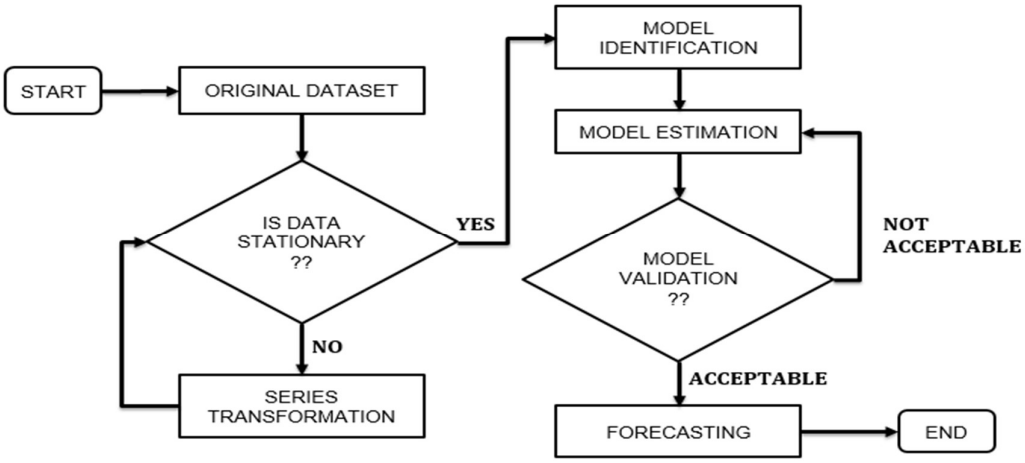
are available for each estimator. The results are tested against existing statistical packages to ensure that they are correct.

ITERTOOLS

It is a module that provides various functions that work on iterators to produce complex iterators. This module works as a fast, memory-efficient tool that is used either by themselves or in combination to form iterator algebra.

METHODOLOGY

We will be using a 6-step approach to build our model then we will predict stock prices with the help of the model.



PROGRESS TILL NOW

DATASET

We have taken historical data of State Bank of India (SBI) that is we have taken collected data from January 2011 to December 2018 and after a successful implementation of our model, we will use the respective model to predict the stock closing prices for the year 2019. The Data collected has been stored in the form of a CSV file; in Comma-Separated Values format.

```
Date,Open,High,Low,Close,Volume
30-04-2020,194,195.75,189.8,190.4,2845074
29-04-2020,185.3,191.7,183.8,190,2679334
28-04-2020,181.6,185.2,181.3,184.3,2078231
27-04-2020,182.9,183.5,180.5,180.85,1826256
24-04-2020,184.5,184.5,179,179.7,4049624
23-04-2020,188.25,189.9,185.25,186.8,3408851
22-04-2020,185.1,189.5,181.4,188.6,3433220
21-04-2020,188.95,188.95,183,184.9,2858972
20-04-2020,194.4,197.3,190.1,192.55,3386549
17-04-2020,195.05,198,186.4,193.3,4596208
16-04-2020,182.3,189.8,180.3,188.6,2923051
15-04-2020,187.45,191.9,180.25,182.3,3427072
13-04-2020,187.65,188.85,183.1,183.55,2181525
09-04-2020,187.75,190,183.75,187.7,2979988
08-04-2020,183.45,194.5,180.8,182.9,3804368
```

SNAPSHOT OF THE SBI DATASET SAMPLE

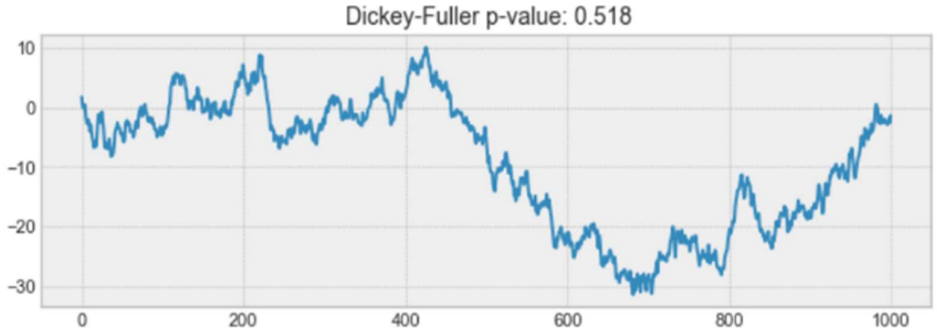
DATASET PRE-PROCESSING

TESTING STATIONARITY

We will using the Dickey-Fuller Hypothesis Test to check whether our dataset which is organized using **TIME-SERIES** is stationary in nature or not. The D-F Hypothesis Test is a null test hypothesis that states that only one root is present.

If it is, then $p > 0$, and the process is not stationary.
Otherwise, $p = 0$, the null hypothesis is rejected, and the process is considered to be stationary.

As an example, the process below is not stationary. Notice how the mean is not constant through time.



Example of a non-stationary process

SERIES TRANSFORMATION

We will be using moving average and Exponential Smoothing method to transform our data into a stationary data.

Moving Average: - The moving average model is probably the most naive approach to time series modelling. This model simply states that the next observation is the mean of all past observations.

Exponential Smoothing: - Exponential smoothing uses a similar logic to moving average, but this time, a different *decreasing weight* is assigned to each observations. In other words, *less importance* is given to observations as we move further from the present.

MODEL SELECTION

We have selected the Seasonal Autoregressive Integrated Moving Average, or SARIMA, Model for our dataset, whereas a basic ARIMA model would have sufficed to train on any dataset with a trend, whereas SARIMA only adds the Seasonality factor into the mix, it opens up a complete new horizon of analysis and ultimately adds on to the ease-of data training and model building.

We needed the extra seasonality factor, because it adds on the ability to work with multivariate time-series that is to even find trends in data which may or may not deviate vastly from the trend at some point due to some external factors.

SARIMA is actually the combination of simpler models to make a complex model that can model time series exhibiting non-stationary properties and seasonality.

- At first, we have the **autoregression model AR (p)**. This is basically a regression of the time series onto itself. Here, we assume that the current value depends on its previous values with some lag. It takes a parameter **p** which represents the maximum lag.
- Then, we add the moving average model **MA (q)**. This takes a parameter q which represents the biggest lag after which other lags are not significant on the autocorrelation plot.
- After, we add the **order of integration I (d)**. The parameter **d** represents the number of differences required to make the series stationary.
- Finally, we add the final component: **seasonality S (P, D, Q, s)**, where **s** is simply the season's length. Furthermore, this component requires the parameters **P** and **Q** which are the same as **p** and **q**, but for the seasonal component. Finally, **D** is the order of seasonal integration representing the number of differences required to remove seasonality from the series.

Combining all, we get the **SARIMA (p, d, q) (P, D, Q, s)** model.

REFERENCES

1. A 6 Step Field Guide for Building Machine Learning Projects by Daniel Bourke
2. Introduction to Time Series Forecasting With Python by Jason Brownlee
3. “Introduction to Time Series Analysis and Forecasting” by Douglas C. Montgomery, Cheryl L. Jennings, and Murat Kulahci
4. “Time Series Analysis: Forecasting and Control” by George E. P. Box, Gwilym M. Jenkins, Gregory C. Reinsel, and Greta M. Ljung
5. Monogan, James. (2015). Time Series Analysis. 10.1007/978-3-319-23446-5_9.
6. Maçaira, Paula & Thomé, Antonio Marcio & Oliveira, Fernando Luiz & Ferrer, Ana Luiza. (2018). Time series analysis with explanatory variables: A systematic literature review. Environmental Modelling & Software. 107. 199-209. 10.1016/j.envsoft.2018.06.004.
7. Adhikari, Ratnadip & Agrawal, R.. (2013). An Introductory Study on Time series Modeling and Forecasting. 10.13140/2.1.2771.8084.
8. Nedeltcheva, Galia. (2015). Forecasting Stock Market Trends. Economic Quality Control. 10.1515/eqc-2015-6003.
9. Vanukuru, Kranthi. (2018). Stock Market Prediction Using Machine Learning. 10.13140/RG.2.2.12300.77448.
10. Sharma, Surbhi & Kaushik, Baij. (2018). Quantitative Analysis of Stock Market Prediction for Accurate Investment Decisions in Future. Journal of Artificial Intelligence. 11. 48-54. 10.3923/jai.2018.48.54.