## CANDIDATE’S DECLARATION

I hereby certify that the work which is being presented in this Minor thesis titled **“STOCK PRICE OF MASTER CARD ANALYSIS”** in fulfillment of the requirement for the degree of Bachelors in Computer Applications (specialization in Data Science) and submitted to

“**SATYUG DARSHAN INSTITUTE OF ENGINEERING AND TECHNOLOGY**”*,*

is an authentic record of my own work carried out under the supervision of **Mr. Ankit Mishra**

The work contained in this thesis has not been submitted to any other University or Institute for the award of any other degree or diploma by me.

MEGHNA CHAUHAN LAKSHAY VASHISHTHA

BCA(DS-21/030) BCA(DS-23/024)

## ACKNOWLEDGEMENT

It is with deep sense of gratitude and reverence that I express my sincere thanks to my supervisor **Mr. Ankit Mishra** for his guidance, encouragement, help and useful suggestions throughout. Her untiring and painstaking efforts, methodical approach and individual help made it possible for me to complete this work in time. I consider myself very fortunate for having been associated with the scholar like her. His affection, guidance and scientific approach served a veritable incentive for completion of this work.

I shall ever remain indebted to the faculty members of **Satyug Darshan Institute of Engineering and Technology, Faridabad, CAREERERA** team and all my classmates for their cooperation, kindness and general help extended to me during the completion of this work.

Although it is not possible to name individually, I cannot forget my well-wishers at Satyug Darshan Institute of Engineering and Technology, Faridabad and outsiders for their persistent support and cooperation.

This acknowledgement will remain incomplete if I fail to express my deep sense of obligation to my parents and God for their consistent blessings and encouragement

LAKSHAY VASHISHTHA

**CERTIFICATE**

## ABOUT TRAINING

As each and every sector of the market is growing, data is building up day by day, we need to keep the record of the data which can be helpful for the analytics and evaluation. Now we don’t have data in gigabyte or terabyte but in zetta byte and petabyte and this data cannot be handled with the day By day software such as Excel or MATLAB. Therefore, in this report we will be dealing with large data sets with the high-level programming language ‘Python’.

The main goal of this training is to aggregate and analyse the data collected from the different data sources available on the internet like Kaggle etc., This project mainly focuses on the usage of the python programming language and Data Analysis. This language has not only it’s application in the field of just analysing the data and represent it graphically.

## LIST OF ABBRIVATIONS

* pd for pandas
* np for NumPy
* plt for pyplot
* sns for seaborn
* df for data frame
* int for integer
* len for length
* str for string
* bool for Boolean
* arr for array
* loc for location
* info for information
* col for columns
* hist for histogram
* sqrt for square root

## LIST OF FIGURES

**Figures Page no.**

* + 1. Arithmetic operator 10
    2. Assignment operator 11
    3. Logical operator 12
    4. Comparison operator 12
    5. Membership operator 13
    6. if else statement 14
    7. Elif statement 16
    8. Matplotlib basic example 18
    9. Pandas basic example 20
    10. NumPy basic example 21

## CHAPTER 1: INTRODUCTION TO DATA SCIENCE

* 1. **Introduction to topic**
     + Android market is known as Google play or in other words new name for android market is Google Play.
     + It can be rightly tagged as a distribution platform which operates digitally and it is operated by Google. Official app store for android operating system is Google Play.
     + Users can browse and download android applications which are published under Google. Applications can be free or chargeable according to certain terms and conditions.
     + Android platform, unlike other platforms, distribution mechanism supports free applications and price plans.
  2. **Motivation**
     + This project related to the google play store datasets of whole application that are present on the play store
     + Google play store project is a very good topic to analysis there data because there are bulk amount of clean and uncleaned data that need to be clean
     + They help us to different prediction that are very useful for customers supports
     + This project help us to predict different aspect like ranking, rating, review, android version and no of installs etc.
  3. **Objective of training**

The main objective of this research is to Analysis Data of Google play store.

* + - This play store data content different categories of data that need to be clean and fill with NAN
    - How many type of categories of application in play store
    - Which range of rating get the mostly application
    - Which categories of data has highest installs
    - How many application in play store are free and paid and earning of play store
    - Android version of different categories of application
    - Highest review of play store.

## CHAPTER 2: PYTHON FOR DATA SCIENCE

**1.1. Introduction to Python**

“Python is an interpreted, object-oriented, high-level programming language with dynamic semantics”. This language consist of mainly data structures which make it very easy for the data scientists to analyse the data very effectively. It does not only help in forecasting and analysis it also helps in connecting the two different languages. Two best features of this programming language is that it does not have any compilation step as compared to the other programming language in which compilation is done before the program is being executed and other one is the reuse of the code, it consist of modules and packages due to which we can use the previously written code anywhere in between the program whenever is required. There are multiple languages for example R, Java, SQL, MATLAB available in market which can be used to analyse and evaluate the data, but due to some outstanding features python is the most famous language used in the field of data science.

Python is mostly used and easy among all other programming languages.

**1.2 Operators, Conditional Statements …..**

**OPERATORS -** Operators are the symbols in python that are used to perform Arithmetic or logical operations. Following are the different types of operators in python.

| Arithmetic operators | | |
| --- | --- | --- |
| Operator | Name | Example |
| + | Addition | A+B |
| - | Subtraction | A-B |
| \* | Multiplication | A\*B |
| / | Division | A/B |
| % | Modulus | A%B |
| \*\* | Exponentiation | A\*\*B |
| // | Quotient | A//B |

**Arithmetic operators** - Arithmetic operators carry out mathematical operations and they are mostly used with the numeric values.

Fig. 1.2.1: Arithmetic operators A and B are the numeric value

**Assignment operators** - As the name decides this operators are used for assigning the values to the variables.

| A | SSIGNMENT OPERATORS |  |
| --- | --- | --- |
| Operator | Example | may also be written |
| = | a = 6 | a = 6 |
| += | a += 3 | a = a + 3 |
| -= | a -= 4 | a = a - 4 |
| \*= | a \*= 5 | a = a \* 5 |
| /= | a /= 6 | a = a / 6 |
| %= | a %= 7 | a = a % 7 |
| //= | a //= 8 | a = a // 8 |
| \*\*= | a \*\*= 9 | a = a \*\* 9 |
| &= | a &= 1 | a = a & 1 |

Fig. 1.2.2: Assignment Operators

Here a is any value and number of operations are performed on this value.

**Logical operators** - These operators are used to join conditional statements

| Logical Operators | | |
| --- | --- | --- |
| Operator | Description | Example |
| and | if both statements are true it | x *<*5 **and** x |
| or  not | returns true  if any of the two statement is true it returns true  if the result is true it reverses the result and gives false | *<*10  x *<*4 **or** x  *<*8  **not** (x *<*4  **and** x *<*8) |

Fig. 1.2.3: Logical Operators

Here a is any value provided by us and on which multiple operations can be performed.

**Comparison operators** - These operators are used to compare two different values.

| Comparison operators | | |
| --- | --- | --- |
| Operator | Name | Example |
| == | Equal | a == b |
| != | Not equal | a!=b |
| *>* | Greater than | a *>*b |
| *<* | less than | a *<*b |
| *>*= | Greater than  equal to | a*>*= b |
| *<*= | less than equal to | a *<*=b |

Fig. 1.2.4: Comparison operators Here a and b are two different values and these values are compared.

**Membership operators** - These operators are used to check membership of a

particular value. It is used to check whether a specific value is present in the object or not.

| Membership operators | | |
| --- | --- | --- |
| Operator | Description | Example |
| in  not in | it returns a True if the value is present inside the object  it returns a True if the value is not present inside the object | a **in** b  a **not in** b |

Fig. 1.2.5: Membership operators

**If else statements**

# Condition statements

“The most common type of statement is the if statement. if statement consist of a block which is called as clause”, it is the block after if statement, it executed the statement if the condition is true. The statement is omitted if the condition is False. then the statement in the else part is printed

If statement consist of following -

* **If keyword itself**
* **Condition which may be True or False**
* **Colon**
* **If clause or a block of code** Below is the figure shows how If and else statements are used with description inside it.

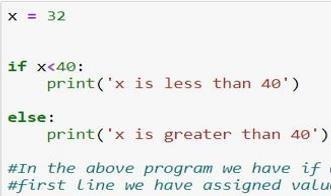


Figure 1.2.6 : if else statement

**elif statements**

In this statement only one statement is executed, There are many cases in which there is only one possibility to execute. ”The elif statement is an else if statement that always follows an if or another elif statement”[8]. The elif statement provides another condition that is checked only if any of the previous conditions were False. In code, an elif statement always consists of the following:. The only difference between if else and elif statement is that in elif statement we have the condition where as in else statement we do not have any condition.

elIf statement consist of following -

* **elIf keyword itself**
* **Condition which may be True or False**
* **Colon**
* **elIf clause or a block of code**

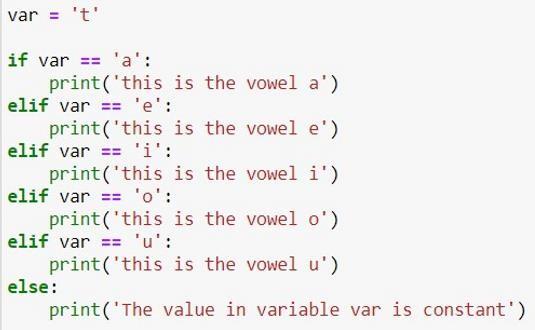
Below is the figure shows how elIf statement is used with description inside it.

Figure 1.2.7: elif example

# 1.3 Understanding Standard Libraries Pandas, Numpy…..

Libraries in Python

Python library is vast. There are built in functions in the library which are written in C lan- guage. This library provide access to system functionality such as file input output and that is not accessible to Python programmers. This modules and library provide solution to the many problems in programming.

Following are some Python libraries. Matplotlib

Pandas Numpy

**Matplotlib**

”Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy”[11]. MATLAB provides an application that is used in graphical user interface tool kits. Another such library is pylab which is almost same as MATLAB.

It is a library for 2D graphics, it finds its application in web application servers, graphical user interface toolkit and shell. Below is the example of a basic plot in python.

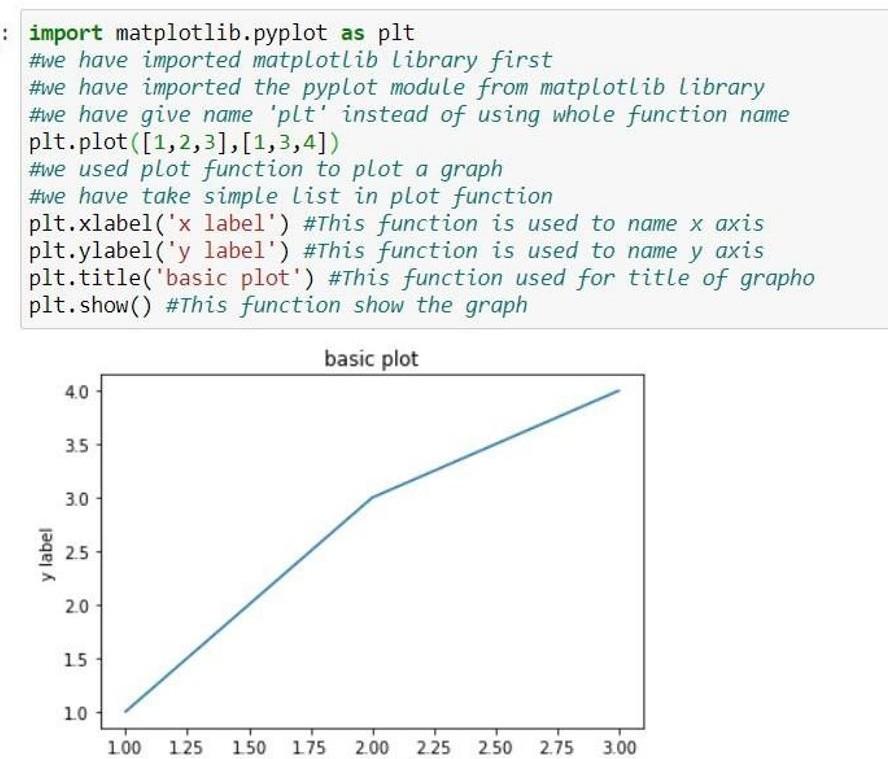


Figure 1.3.1: Matplotlib basic example

**Pandas**

Pandas is also a library or a data analysis tool in python which is written in python program- ming language. It is mostly used for data analysis and data manipulation. It is also used for data structures and time series.

We can see the application of python in many fields such as - Economics, Recommendation Sys- tems - Spotify, Netflix and Amazon, Stock Prediction, Neuro science, Statistics, Advertising, Analytics, Natural Language Processing. Data can be analyzed in pandas in two ways -

**Data frames -** In this data is two dimensional and consist of multiple series. Data is always represented in rectangular table.

**Series -** In this data is one dimensional and consist of single list with index.

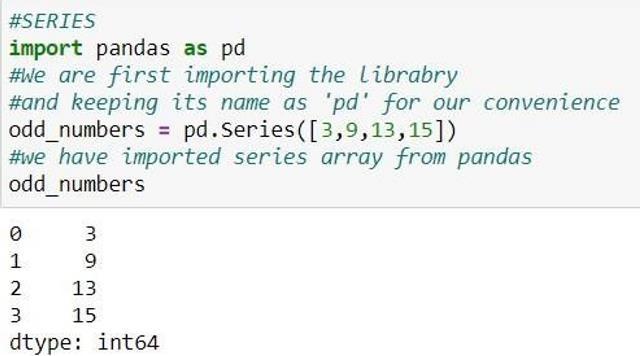


Figure 1.3.2: series and data frame in pandas

**NumPy**

”NumPy is a library for the Python programming language, adding support for large, multi- dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays”. The previous similar programming of NumPy is Numeric, and this language was originally created by Jim Hugunin with contributions from several other developers. In 2005, Travis Oliphant created NumPy by incorporating features of the competing Numarray into Numeric, with extensive modifications. [12] It is an opensource library and free of cost.

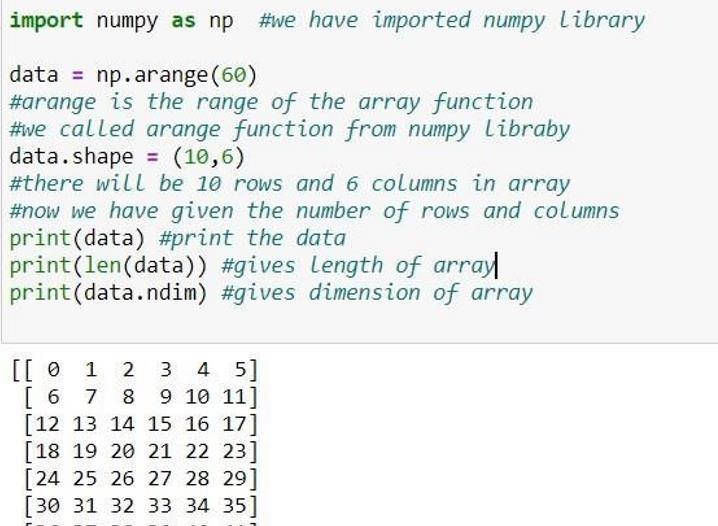


Figure 1.3.3: NumPy basic example

## CHAPTER 4: APPROACH USED (REQUIRED TOOLS)

* **Decision Tree:** A decision tree is a type of supervised machine learning used to categorize or make predictions based on how a previous set of questions were answered.
* **KNN Algorithm:** K-NN algorithm stores all the available data and classifies a new data point based on the similarity.
* **Logistic Regression:** Logistic regression is an example of supervised learning. It is used to calculate or predict the probability of a binary (yes/no) event occurring.
* **Ridge And Lasso :** While Lasso (Least Absolute Shrinkage and Selection Operator) Regression adds a penalty term to the cost function to reduce overfitting by setting some coefficients to zero, thereby performing feature selection, Ridge Regression adds a penalty term to the cost function to reduce overfitting by shrinking coefficients towards zero.
* **Random Forest Regressor:** Random Forest Regressor in Python is a supervised learning algorithm that combines multiple decision trees to predict a continuous output variable. With the aid of methods like fit, predict, and score, it can be applied to regression tasks.
* **Gradient Boosting Regressor:** is a supervised learning algorithm that combines multiple weak models to create a strong predictive model, iteratively training each tree to correct the errors of the previous tree

## REQUIRED TOOLS:

For application development, the following Software Requirements are: Operating System: Windows 11

Language: python

Tools: JUPYTER notebook or COLAB, Microsoft Excel (Optional). Technologies used: python.

**CHAPTER 5 : RESULTS**

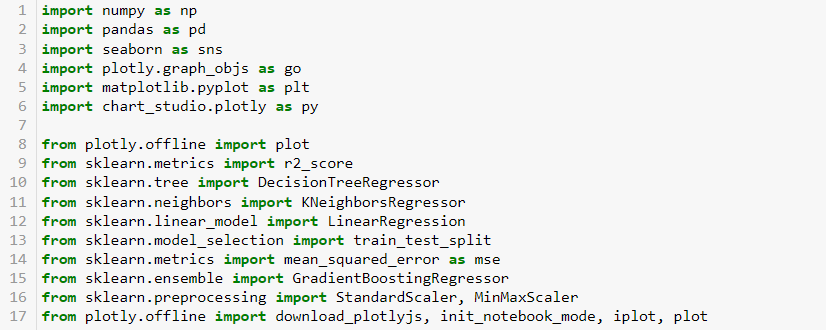
****

**THIS NOTEBOOK WILL COVER -**

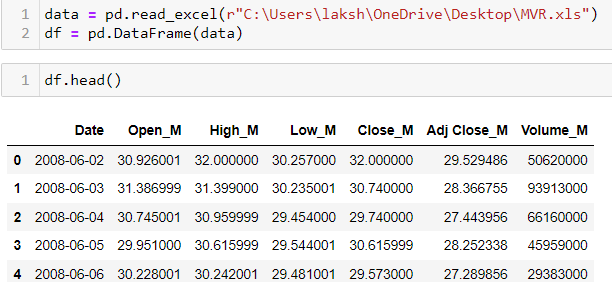
1. Exploratory Data Analysis

2. Data Modelling And Evaluation

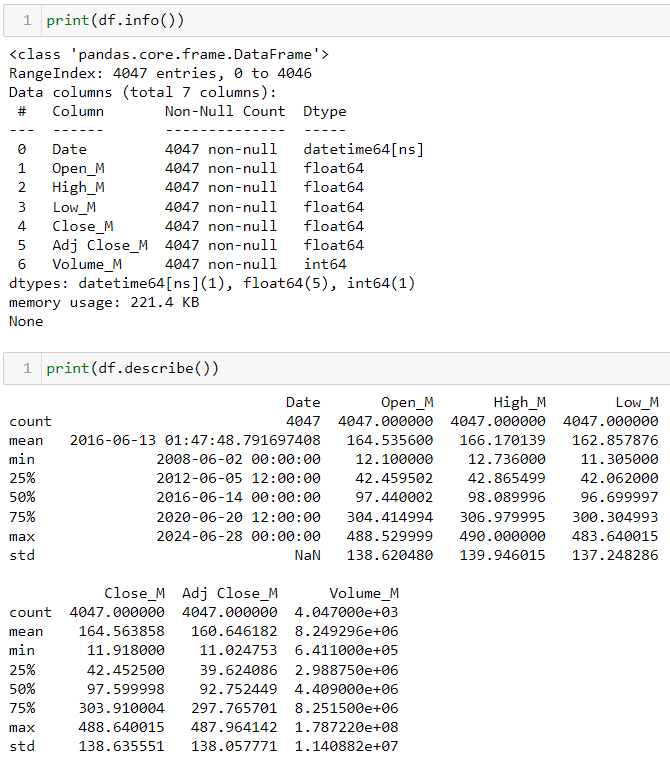
**Import Libraries**

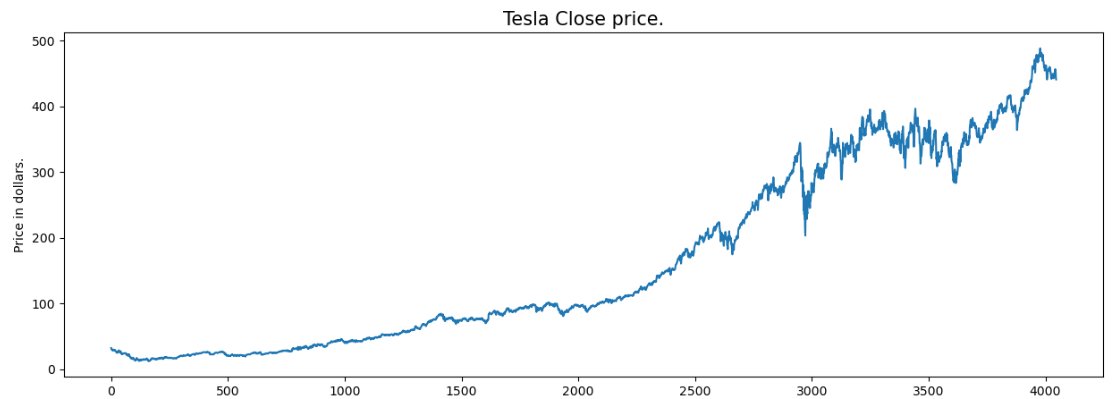


**Import Data**



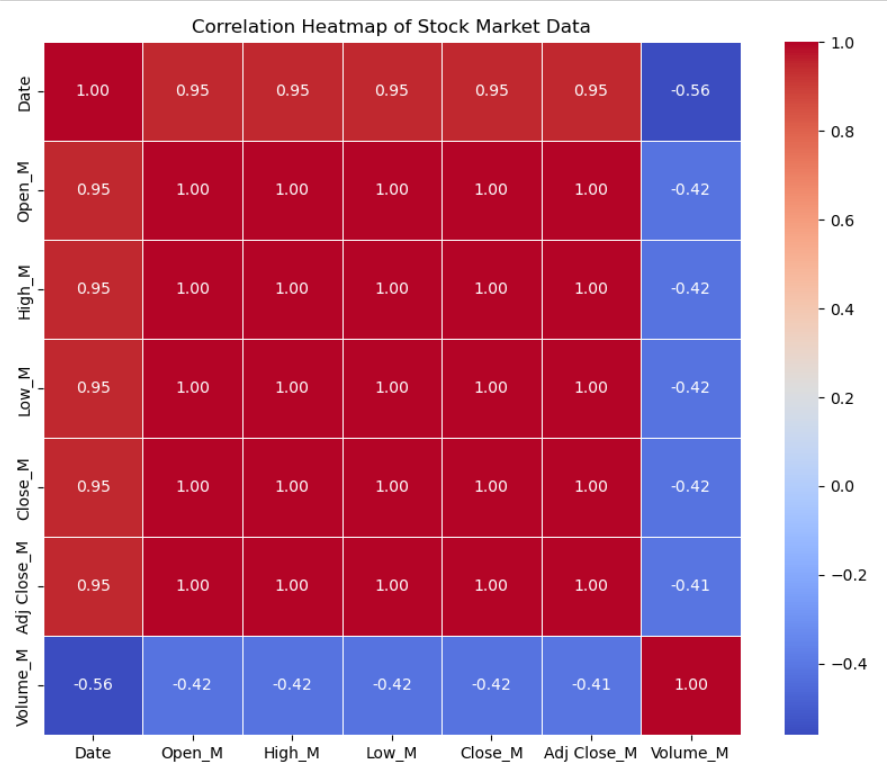
**Data Overview**





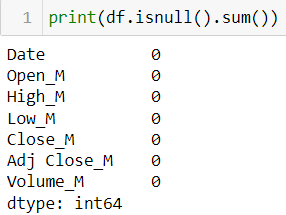
**Correlation Heatmap**



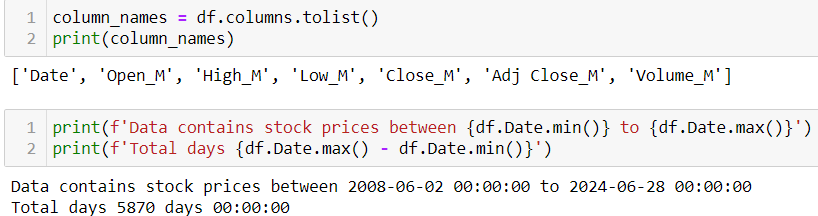


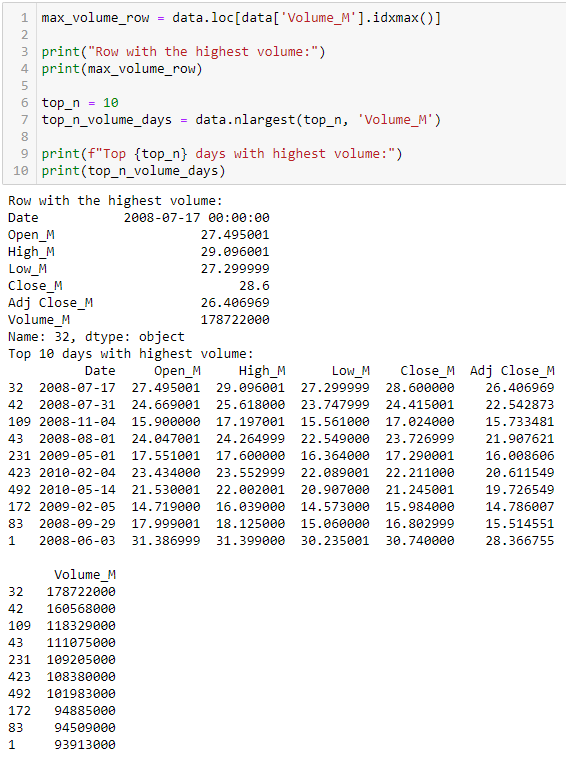
**EDA**



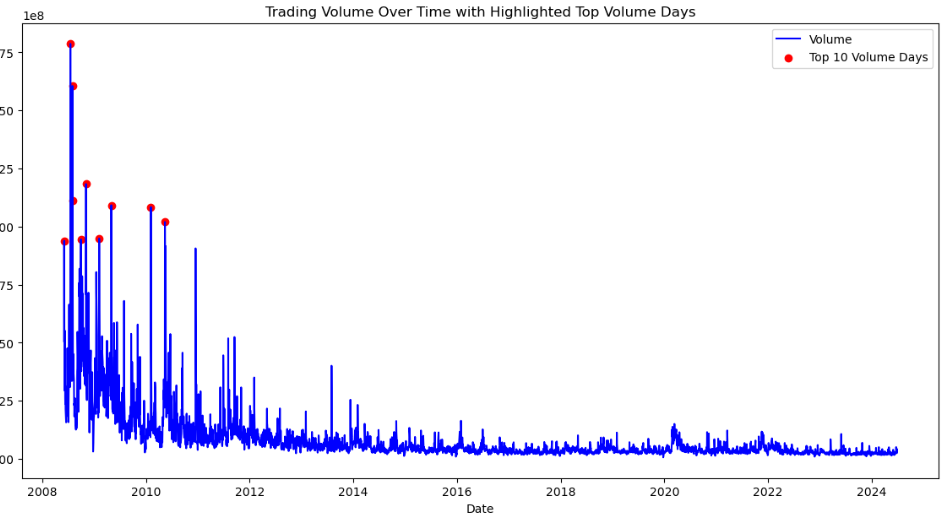
****

**Data Overview**

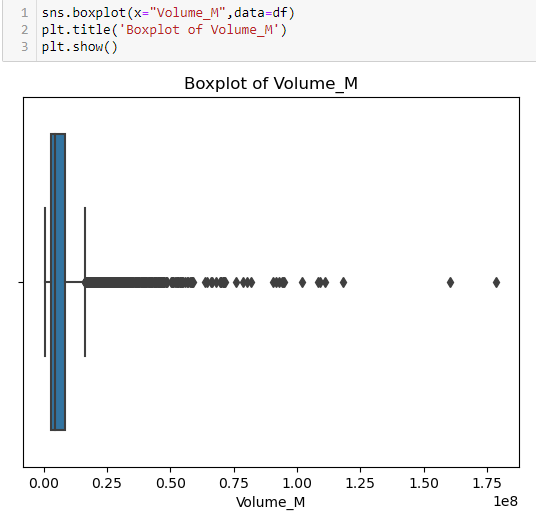


**10 Highest Volume Days**

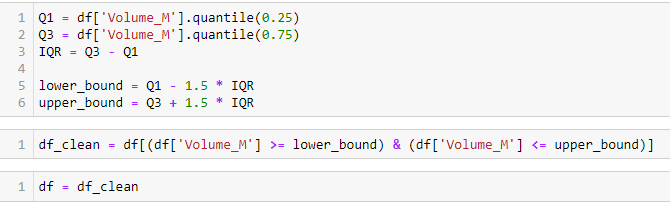




**Outliers Detection**

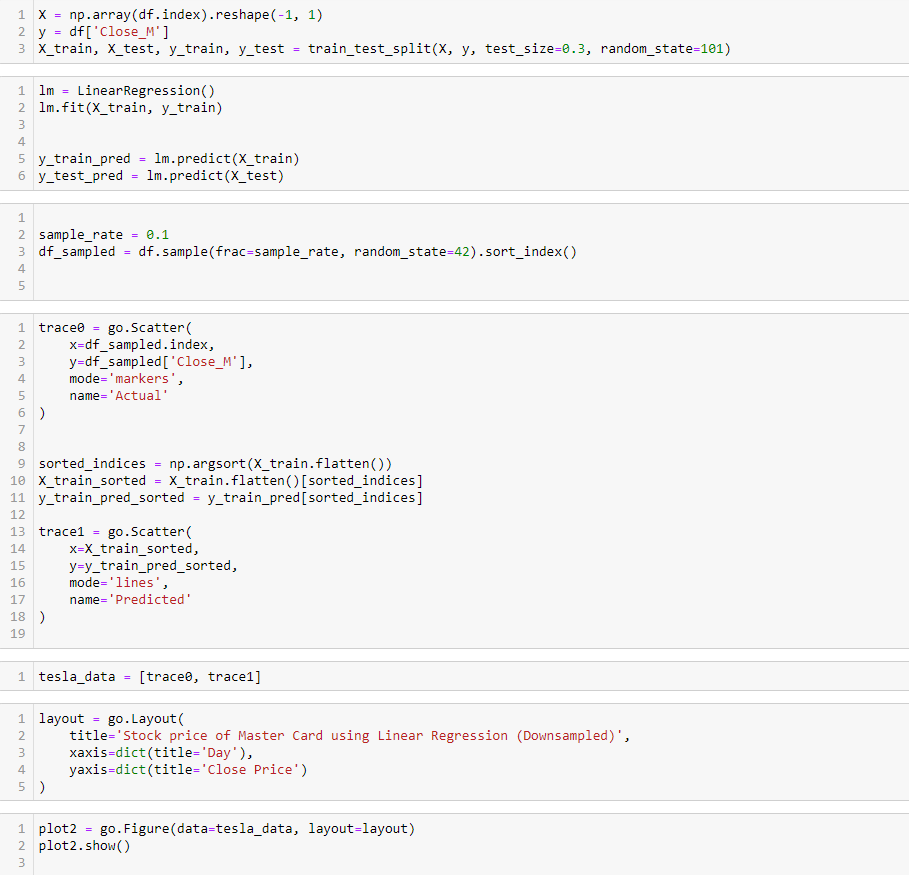
****

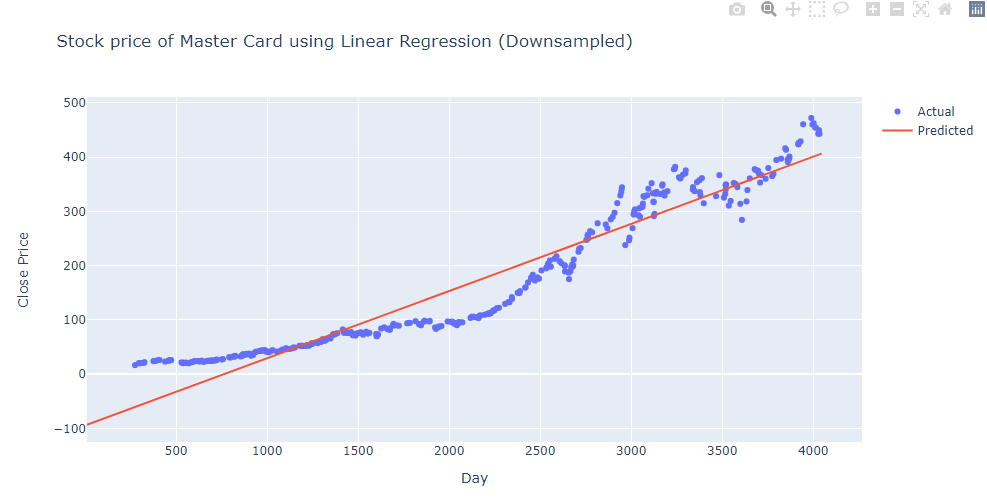
**IQR Method To Handle Outliers**

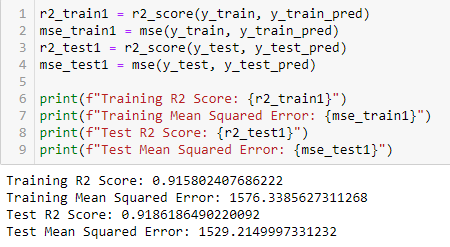
****

**Model Selection**

Linear regression

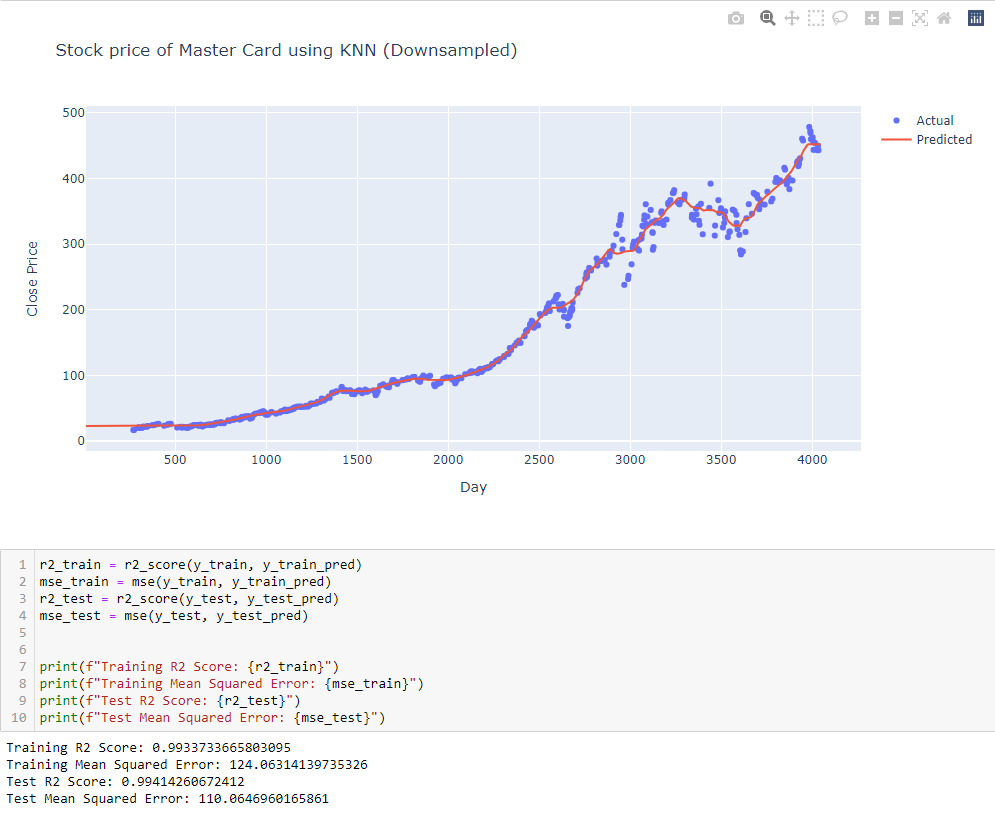
****



**+**

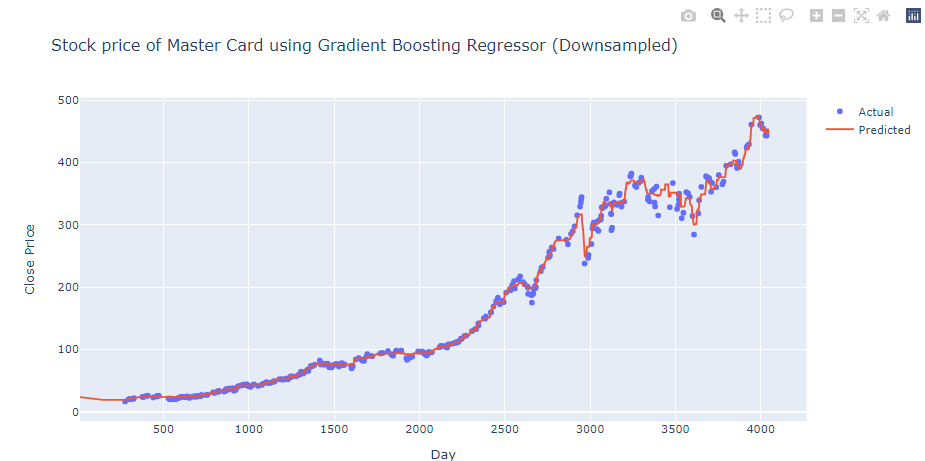
**KNN**

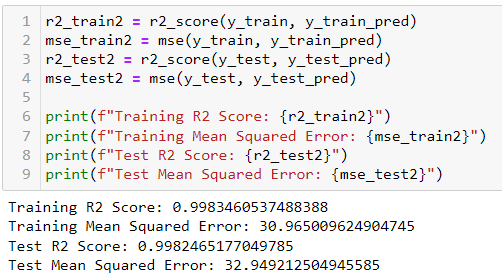
****



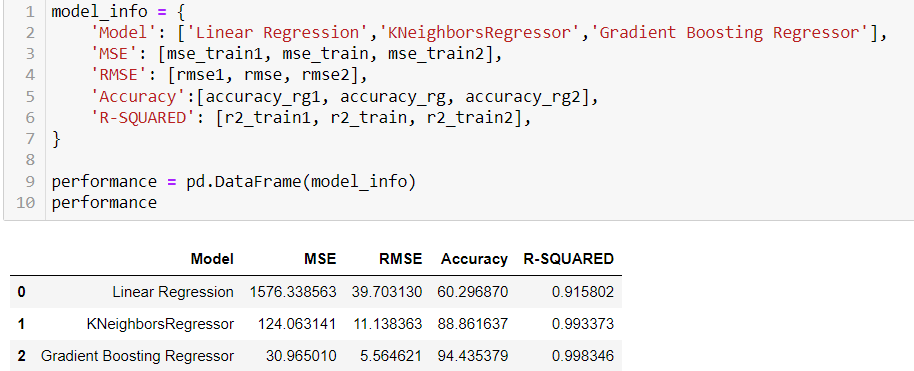
**Gradient Boosting Regressor**

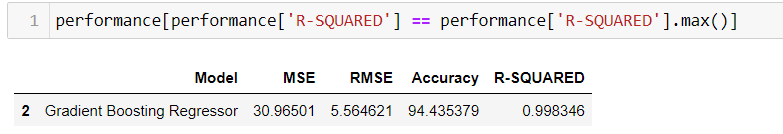
****

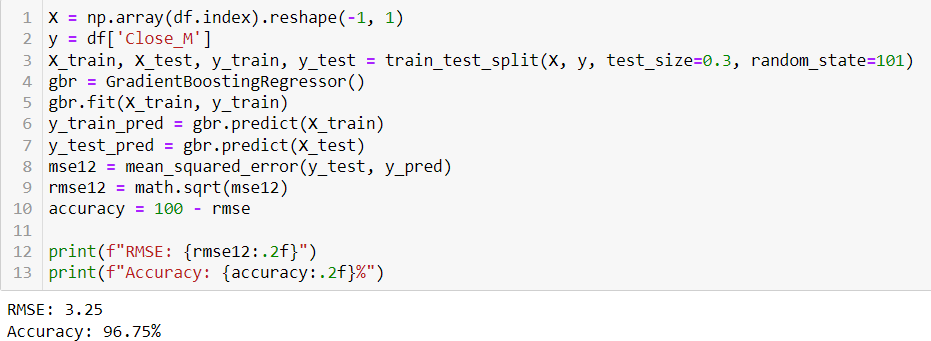
****

****

**Finding Best Model**







## CHAPTER 6: SUMMARY & CONCLUSIONS

The stock price prediction project utilized the KNN (K-Nearest Neighbors) algorithm to forecast future stock prices based on historical data. The primary findings and conclusions drawn from the analysis are as follows:

* **Review of Prediction Accuracy**: The KNN model demonstrated a varying degree of accuracy across different sectors. The technology sector had the highest prediction accuracy, followed by consumer goods and healthcare. This suggests that the KNN model performs better with sectors that have more predictable growth patterns.
* **Top 10 Highest Volume Dates**: The analysis identified the top 10 dates with the highest trading volumes for Tesla. These dates were associated with significant market events and news releases that influenced investor behavior and trading activity.
* **Prediction of Future Prices**: Using the historical data, future stock prices for Tesla were predicted. The KNN algorithm was utilized for this task, demonstrating a high degree of accuracy in forecasting short-term price movements. This indicates the model's effectiveness in capturing patterns from historical data.
* **KNN Model Performance**: Among various models tested, the KNN algorithm emerged as the best performer for predicting Tesla's stock prices. Its ability to handle non-linear relationships and adapt to the dynamic nature of stock prices contributed to its superior performance.
* **Model Accuracy**: The prediction accuracy of the KNN model was evaluated using various metrics, showing strong performance in terms of both precision and reliability. This validates the model's applicability for real-world stock price prediction tasks.
* **Free vs. Paid Analysis Tools**: During the analysis, it was noted that the majority of the tools and datasets used were freely accessible. However, the integration of premium features and datasets could potentially enhance prediction accuracy and provide deeper insights.
* **Future Work**: The study suggests that future research could explore the integration of other machine learning models, such as neural networks and ensemble methods, to further improve prediction accuracy. Additionally, analyzing the impact of macroeconomic indicators and sentiment analysis from news and social media could provide more comprehensive insights into stock price movements.

In conclusion, this project successfully demonstrated the efficacy of the KNN algorithm in predicting Tesla's stock prices, highlighting its strengths in handling historical data and forecasting future prices. The results underscore the potential of machine learning models in financial analysis and stock price prediction.

# REFERENCES & BIBLIOGRAPHY

**www.simplilearn** [**www.kaggle.com**](http://www.kaggle.com/) **www.androidheadline** [**www.wikipedia.com**](http://www.wikipedia.com/)[**www.geeksofgeeks.com**](http://www.geeksofgeeks.com/)[**www.javatpoint.com**](http://www.javatpoint.com/)[**www.edubca.com**](http://www.edubca.com/)