ABSTRACT

In this project we attempt to implement machine learning approach to predict stock prices. Machine learning is effectively implemented in forecasting stock prices. The objective is to predict the stock prices in order to make more informed and accurate investment decisions. We propose a stock price prediction system that integrates mathematical functions, machine learning, and other external factors for the purpose of achieving better stock prediction accuracy and issuing profitable trades . Predicting the future in all the areas using machine learning techniques was the recent research in the current scenario. Stock market is one among them which needs the prediction future market to invest in the new enterprise or to sell their existing shares to get profit. This need the efficient prediction technique which studies the previous exchanges of stock market and gives the future prediction based on that. The proposed prediction system of stock market price based on the exchange takes place in previous scenario. The system studies the diversing effect of market price of product in a particular time gap and analyze its future trend whether it's loss or gain. During the system of thinking about diverse strategies and variables that should be taken into account, we observed out that strategies with linear regression.

TABLE OF CONTENT

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 OBJECTIVE

The financial market is a dynamic and composite system where people can buy and sell currencies, stocks, equities and derivatives over virtual platforms supported by brokers. The stock market allows investors to own shares of public companies through trading either by exchange or over the counter markets. This market has given investors the chance of gaining money and having a prosperous life through investing small initial amounts of money, low risk compared to the risk of opening new business or the need of high salary career. Stock markets are affected by many factors causing the uncertainty and high volatility in the market.

Stock market prediction and analysis are some of the most difficult jobs to complete. There are numerous causes for this, including market volatility and a variety of other dependent and independent variables that influence the value of a certain stock in the market. These variables make it extremely difficult for any stock market expert to anticipate the rise and fall of the market with great precision.However , Machine Learning and its strong algorithms, the most recent market research and Stock Market Prediction advancements have begun to include such approaches in analyzing stock market data.

1.2 SCOPE OF THE PROJECT

Analysis of stocks using data mining will be useful for new investors to invest in stock market based on the various factors considered by the software. Stock market includes daily activities like sensex calculation, exchange of shares. The exchange provides an efficient and transparent market for trading in equity, debt instruments and derivatives.Our software will be analyzing sensex based on company’s stock value. The stock values of company depend on many factors.

DemandandSupply:

Demand and Supply of shares of a company is a major reason price change in stocks. When Demand Increase and Supply is less, price rises and vice versa.

Popularity:

Main Strength in hands of share buyer. Popularity of a company can effect on buyers. Like if any good news of a company, may result in rise of stock price. And a bad news may break dreams.The stock value depends on other factors as well, but we are taking into consideration only these main factors.

CHAPTER 2

ANALYSING THE SYSTEM

2.1 PROBLEM STATEMENT

This project proposes the question of whether it is possible to predict the stock trends thorough machine learning models. Specially the aim of the project is to determine the ideal model that is efficient in predicting stock prices. “Stock Market” It is widely used source for people to invest money in companies with high growth potential.

2.1.1CHALLENGES

Challenge 1: Quantity of required data

The first issue was to have an extreme amount of data as many factors can influence the stock prices. This is the first challenge of getting hold of large amounts of data from all walks of life. This includes accessing data that might not be publicly available and unstructured data.

Challenge 2: Feature Selection

The second issue is that once a large amount of data is gathered, it is difficult to decide which feature to keep and drop. This is the area Modern-day ***deep learning algorithms outshines vs. machine learning***, and solves this problem as their weights can be tweaked; still, this is the main issue that restricts a data scientist from creating the successful stock price predicting model.

Challenge 3: Integration of Human's Behavioral effect on the market

The third problem was integrating human input, which includes human bias, misjudgments, anxiety, nervousness, etc. Thus typical behavior of traders such as Loss aversion, anchoring, endowment effect is to quantified and fed to the model.

Challenge 4: Identification of distant relationships

The fourth issue is the problem of identifying relationships between different stocks. For example, if the price of oil increases, then the share of aviation-based companies decreases. By assessing this relationship, predictions can take place. Still, the challenge here is to identify the exact set of variables involved as it is a multivariate problem, and several features are interested in finding such relationships.

Challenge 5: Requirement of Expertise in multiple fields

The sixth challenge is the level of advanced mathematics required for creating an extremely successful model. Renaissance Technologies employed people with PhDs who were masters in Hidden Makarov Processes, Algebra, Kernel methods, Game theory, Computer Science, and Finance. Today, we have a pre-built linear regression model, but a high level of knowledge in all these fields is required to master the model.

2.2EXISTING SYSTEM

In the finance world stock trading is one of the most important activities. Stock market prediction is an act of trying to determine the future value of a stock other financial instrument traded on a financial exchange. The technical and fundamental or the time series analysis is used by the most of the stockbrokers while making the stock predictions. The programming language is used to predict the stock market using machine learning is Python. In this paper we propose a Machine Learning (ML) approach that will be trained from the available stocks data and gain intelligence and then uses the acquired knowledge for an accurate prediction. We use a machine learning technique called linear regression to predict stock prices for the large and small capitalizations.

DISADVANTAGES

The existing system fails when there are rare outcomes or predictors, as the algorithm is based on bootstrap sampling. The previous results indicate that the stock price is unpredictable when the traditional classifier is used. The existence system reported highly predictive values, by selecting an appropriate time period for their experiment to obtain highly predictive scores. The existing system does not perform well when there is a change in the operating environment. It doesn‟t focus on external events in the environment, like news events or social media. It exploits only one data source, thus highly biased. The existing system needs some form of input interpretation, thus need of scaling. It doesn‟t exploit data pre-processing techniques to remove inconsistency and incompleteness of the data

2.3 PROPOSED SYSTEM

In this proposed system, we focus on predicting the stock Values using machine learning algorithms like linear regression. We proposed the system “Stock market price prediction” we have predicted the stock market price using the linear regression. In this proposed system, we were able to train the machine from the various data points from the past to make a future prediction. We took data from the previous year stocks to train the model. We majorly used two machine-learning libraries to solve the problem. The first one was numpy, which was used to clean and manipulate the data, and getting it into a form ready for analysis. The other was scikit, which was used for real analysis and prediction. The data set we used was from the previous year’s stock markets collected from the public database available online, 80 % of data was used to train the machine and the rest 20 % to test the data. The basic approach of the supervised learning model is to learn the patterns and relationships in the data from the training set and then reproduce them for the test data. We used the python pandas library for data processing which combined different datasets into a data frame. The tuned up dataframe allowed us to prepare the data for feature extraction. The dataframe features were date and the closing price for a particular day. We used all these features to train the machine on linear regression model and predicted the object variable, which is the price for a given day. We also quantified the accuracy by using the predictions for the test set and the actual values. The proposed system touches different areas of research including data pre-processing, linear regression, and so on.

2.4 SYSTEM REQUIREMENTS:

HARDWARE REQUIREMENTS:

HARDWARE	SPECIFICATION
PROCESSOR	64-BIT, Intel(R) Core(TM),1.70GHz 1
RAM	MINIMUM-4GB RECOMMENDED8.00GB
MEMORY SPACE	4.00GB
OPERATING SYSTEM	WINDOWS/LINUX/MAC OS

SOFTWARE REQUIREMENTS:

Following software's are used to implement an Stock Market Prediction Using Python and Machine learning:

1.Python 3.9

2.Google Collaboratory

3.Anaconda Power shell

To Import a Python Library's

1.scikit-learn==0.22.1

2.pandas

3.numpy

4.Flask

FRONT END:

1.HTML(Hyper Text Mark-up Language)

2. CSS (cascading Style sheets)

BACK END:

1.PYTHON 3.9

CHAPTER 3

LITERATURE SURVEY

INTRODUCTION

The various research works on the existing stock market price prediction using machine learning have been discussed and analysed .

Stock market predication using a linear regression

AUTHOR NAME : Dinesh ; Girish Kaushal ; Ashish Sharma ; Upendra Singh

PUBLISHER:IEEE

PUBLISHED IN: 2017 International conference of Electronics , Communication and Aerospace Technology(ICECA) 2,510-513,2017

ABSTRACT: It is a serious challenge for investors and corporate stockholders to forecast the daily behavior of stock market which helps them to invest with more confidence by taking risks and fluctuations into consideration. In this paper, by applying linear regression for forecasting behavior of TCS data set, we prove that our proposed method is best to compare the other regression technique method and the stock holders can invest confidentially based on that.

Survey of stock market prediction using machine learning approach

AUTHOR NAME :Ashish Sharma , Dinesh Bhuriya , Upendra Singh

PUBLISHER : IEEE

PUBLISHED IN: 2017 International conference of Electronics , Communication and Aerospace Technology(ICECA) 2,510-509,2017

ABSTRACT: Stock market is basically nonlinear in nature and the research on stock market is one of the most important issues in recent years. People invest in stock market based on some prediction. For predict, the stock market prices people search such methods and tools which will increase their profits, while minimize their risks. Prediction plays a very important role in stock market business which is very complicated and challenging process. Employing traditional methods like fundamental and technical analysis may not ensure the reliability of the prediction. To make predictions regression analysis is used mostly. In this paper we survey of well-known efficient regression approach to predict the stock market price from stock market data based. In future the results of multiple regression approach could be improved using more number of variables.

Regression techniques for the prediction of stock price trend

AUTHOR NAME **:** Malaysian Institute of Information Technology, University Kuala Lumpur, Kuala Lumpur, Malaysia

PUBLISHER:IEEE

PUBLISHED IN : 2012 International Conference on Statistics in Science, Business and Engineering (ICSSBE)

ABSTRACT : This paper examines the theory and practice of regression techniques for prediction of stock price trend by using a transformed data set in ordinal data format. The original pretransformed data source contains data of heterogeneous data types used for handling of currency values and financial ratios. The data formats in currency values and financial ratios provide a process for computation of stock prices. The transformed data set contains only a standardized ordinal data type which provides a process to measure rankings of stock price trends. The outcomes of both processes are examined and appraised. The primary design is based on regression analysis from WEKA machine learning software. The stock price movement in Bursa Malaysia is used as our research setting. The data sources are corporate annual reports which included balance sheet, income statement and cash flow statement. The variables included in the data set were formed based on stock market trading fundamental analysis approach. Classifiers in WEKA were used as algorithms to produce the outcomes. This study showed that the outcomes of regression techniques can be improved for the prediction of

stock price trend by using a dataset in standardized ordinal data format.

Stock Market Prediction using Linear Regression

AUTHOR NAME **:** Vaishnavi Gururaj , Shriya V R and Dr. Ashwini K

PUBLISHER:IEEE

PUBLISHED IN : International Journal of Applied Engineering Research ISSN 0973-4562 Volume 14, Number 8 (2019) pp. 1931-1934

ABSTRACT : Machine learning (ML) is a technology that gives the systems the ability to learn on its own through real-world interactions and generalizing from examples without being explicitly programmed as in the case of rule-based programming. Machine Learning can play a key role in a wide range of critical applications. In machine learning, Linear Regression (LR) is a basic technique by which a linear trend can be obtained. But Support Vector Machines (SVMs) have advanced features such as high accuracy and predictability. In this paper we survey the pros and cons of using both these techniques to predict values and compare both algorithms.

Stock Closing Price Prediction using Machine Learning Techniques

AUTHOR NAME : Mehar vijh , Deeksha Chandola , Viny Anand Tikkiwal , Arun Kumar

PUBLISHED IN : Procedia Computer Science 167,599-606,2020

ABSTRACT : Accurate prediction of stock market returns is a very challenging task due to volatile and non-linear nature of the financial stock markets. With the introduction of artificial intelligence and increased computational capabilities, programmed methods of prediction have proved to be more efficient in predicting stock prices. In this work, Artificial Neural Network and Random Forest techniques have been utilized for predicting the next day closing price for five companies belonging to different sectors of operation. The financial data: Open, High, Low and Close prices of stock are used for creating new variables which are used as inputs to the model. The models are evaluated using standard strategic indicators: RMSE and MAPE. The low values of these two indicators show that the models are efficient in predicting stock closing price.

CHAPTER-4
LANGUAGE SPECIFICATION

4.1 ABOUT PYTHON

Python is a high-level, interpreted, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly procedural), objectoriented and functional programming. It is often described as a "batteries included" language due to its comprehensive standard library. Guido van Rossum began working on Python in the late 1980s as a successor to the ABC programming language and first released it in 1991 as Python 0.9.0. Python is meant to be an easily readable language. Its formatting is visually uncluttered and often uses English keywords where other languages use punctuation. Unlike many other languages, it does not use curly brackets to delimit blocks, and semicolons after statements are allowed but rarely used. It has fewer syntactic exceptions and special cases than C or Pascal. Python 3.9.13 is the latest 3.9 version, and from now on 3.9 (and older; 3.8 and 3.7) will only get security updates. Design philosophy and features Python is a multi-paradigm programming language.

4.2 ABOUT HTML AND CSS

HyperText Mark-up Language or HTML is the standard mark up for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages.

HTML describes the structure of a web page semantically and originally included cues for the appearance of the document. HTML can embed programs written in a scripting language such as JavaScript, which affects the behaviour and content of web pages. Inclusion of CSS defines the look and layout of content. The World Wide Web Consortium (W3C), former maintainer of the HTML and current maintainer of the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997. A form of HTML, known as HTML5, is used to display video and audio, primarily using the element, in collaboration with JavaScript. Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript. CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility; provide more flexibility and control in the specification of presentation characteristics; enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, which reduces complexity and repetition in the structural content; and enable the .css file to be cached to improvethe page load speed between the pages that share the file and its formatting.

CHAPTER-5

STRUCTURED DESIGNED METHODOLOGY:

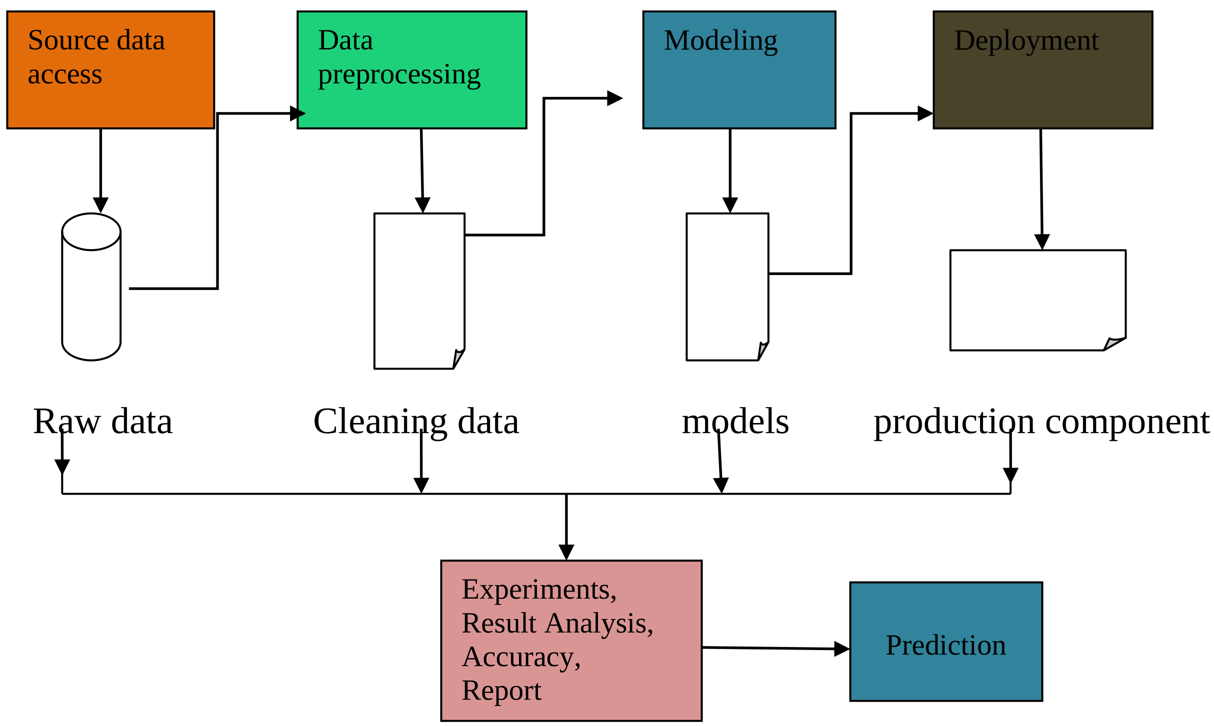


Figure 5 : Structured designed methodology

CHAPTER-6
METHODOLOGIES

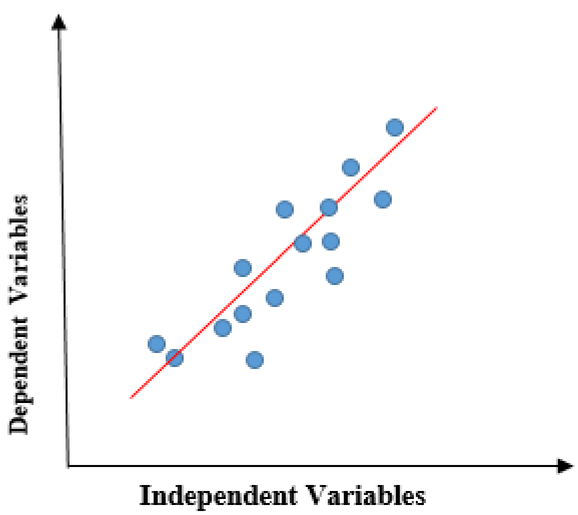
6.1 MACHINE LEARNING MODEL

Machine Learning is the field of study that gives computers the capability to learn without being explicitly programmed. ML is one of the most exciting technologies that one would have ever come across. As it is evident from the name, it gives the computer that makes it more similar to humans: The ability to learn. Machine learning is actively being used today, perhaps in many more places than one would expect.

Machine Learning is a field of inquiry devoted to understanding and building methods that 'learn', that is, methods that leverage data to improve performance on some set of tasks. Machine learning algorithms build a model based on sample data, known as training data, in order to make predictions or decisions without being explicitly programmed to do so. Machine learning algorithms are used in a wide variety of applications, such as in medicine, email filtering, speech recognition, and computer vision, where it is difficult or unfeasible to develop conventional algorithms to perform the needed tasks.A subset of machine learning is closely related to computational statistics, which focuses on making predictions using computers; but not all machine learning is statistical learning. The study of mathematical optimization delivers methods, theory and application domains to the field of machine learning. Data mining is a related field of study, focusing on exploratory data analysis through unsupervised learning. Some implementations of machine learning use data and neural networks in a way that mimics the working of a biological brain. In its application across business problems, machine learning is also referred to as predictive analytics.

6.1.1 LINEAR REGRESSION

Linear regression is a quiet and simple statistical regression method used for predictive analysis and shows the relationship between the continuous variables. Linear regression shows the linear relationship between the independent variable (X-axis) and the dependent variable (Y-axis), consequently called linear regression. If there is a single input variable (x), such linear regression is called simple linear regression. And if there is more than one input variable, such linear regression is called multiple linear regression. The linear regression model gives a sloped straight line describing the relationship within the variables.



The above graph presents the linear relationship between the dependent variable and independent variables. When the value of x (**independent variable**) increases, the value of y (**dependent variable**) is likewise increasing. The red line is referred to as the best fit straight line. Based on the given data points, we try to plot a line that models the points the best.

To calculate best-fit line linear regression uses a traditional slope-intercept form.

$$y = mx+b \Longrightarrow y = a_0+a_1x$$

y= Dependent Variable.

x= Independent Variable.

a0= intercept of the line.

a1 = Linear regression coefficient.

Linear Regression Mathematical Calculation:

Linear regression model tries to produce the best possible straight line for the dataset. For determining the best fit we attempt to minimize the distance between all points and their distance to our line. In this regression technique we predict the value of one variable of Y from the other values of X, where Y is the Criterion Variable and X is called the Predictor Variable that we are basing our Prediction.

The model calculates the prediction with the following formulae

yy = *aa* + *bbbb*

Here 'a' represents the index or the intercept and 'b' represents the slope or the coefficient of the variable 'x'. *aa* = *yy*� − *bbxx̅*

The value of 'b' is calculated by the formulae:

bb = *nn* ∑ *xxxx* − (∑ *xx* ∑ *yy*) *nn* ∑ *xx*2 − (∑ *xx*)2

The values of the above terms are calculated as follows:

yy� = ∑ *yy nn* Where, ∑y = y1+ y2 + y3 +…+ yn *xx̅*= ∑ *xx nn* Where, ∑x =

x1 + x2 + x3 +…+ xn

After calculating the values of (4) and (5) substitute those in equation (2).

Nowcalculate equation (3) and finally substitute (2) and (3) in equation (1)

AlgorithmLinear Regression

Input: x_train: Open values of a company

y_train: Close values of a company

x_test: Open value for which close value is predicted

n:- Size of training dataset

Output: predict_value:- Predicted value.

1: deflinear_regression (x_train, y_train, x_test, n)

2: begin

3: fori = 0 to n 4: begin

5: x_sum = 0,y_sum=0

6: x_sum + = x [i]

7: y_sum + = y [i]

8: end

9: x_mean = x_sum / n

10: y_mean = y_sum / n

11: for i = 0 to n 12: begin

13: xy_sum = 0

14: xy_sum + = x[i] * y[i]

15: x2_sum + = x[i] * x[i]

16: end

17: x_sum_sq = x_sum * x_sum

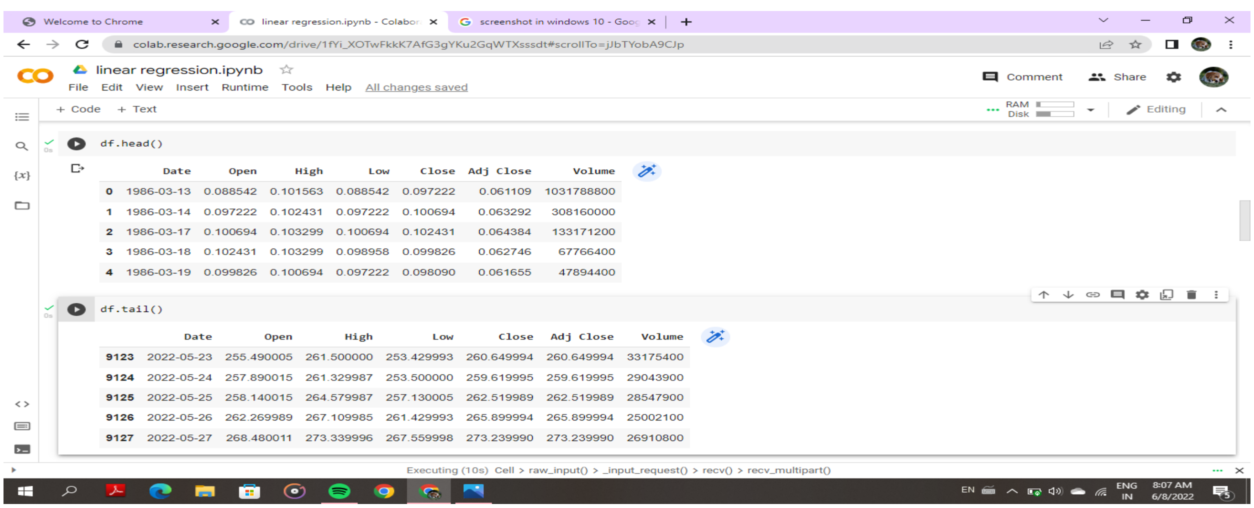
```
l= n * xy_sum - (x_sum * y_sum)
m = n * x2_sum - x_sum_sq
b = l / m a = y_mean - ( b * x_mean)
predict_value = a + b * x_test
return predict_value
end.
```

6.2 DATA COLLECTION AND FEATURE ANALYSIS

Data collection is a very basic module and the initial step towards the project. It generally deals with the collection of the right dataset. The dataset that is to be used in the market prediction has to be used to be filtered based on various aspects. Data collection also complements to enhance the dataset by adding more data that are external. Our data mainly consists of the previous year stock prices. Initially, we will be analyzing the yahoo finance and according to the accuracy, we will be using the model with the data to analyze the predictions accurately.

FEATURE ANALYSIS

From the compiled samples data obtained above, we formed our experimental dataset based on 0 features which are ’OPEN’, ’CLOSE’, ’HIGH’ ,’LOW’,’ADJ CLOSE’,’VOLUME’



```
df.head()
```

	Date	Open	High	Low	Close	Adj Close	Volume
0	1986-03-13	0.088542	0.101563	0.088542	0.097222	0.061109	1031788800
1	1986-03-14	0.097222	0.102431	0.097222	0.100694	0.063292	308160000
2	1986-03-17	0.100694	0.103299	0.100694	0.102431	0.064384	133171200
3	1986-03-18	0.102431	0.103299	0.098958	0.099826	0.062746	67766400
4	1986-03-19	0.099826	0.100694	0.097222	0.098090	0.061655	47894400

```
df.tail()
```

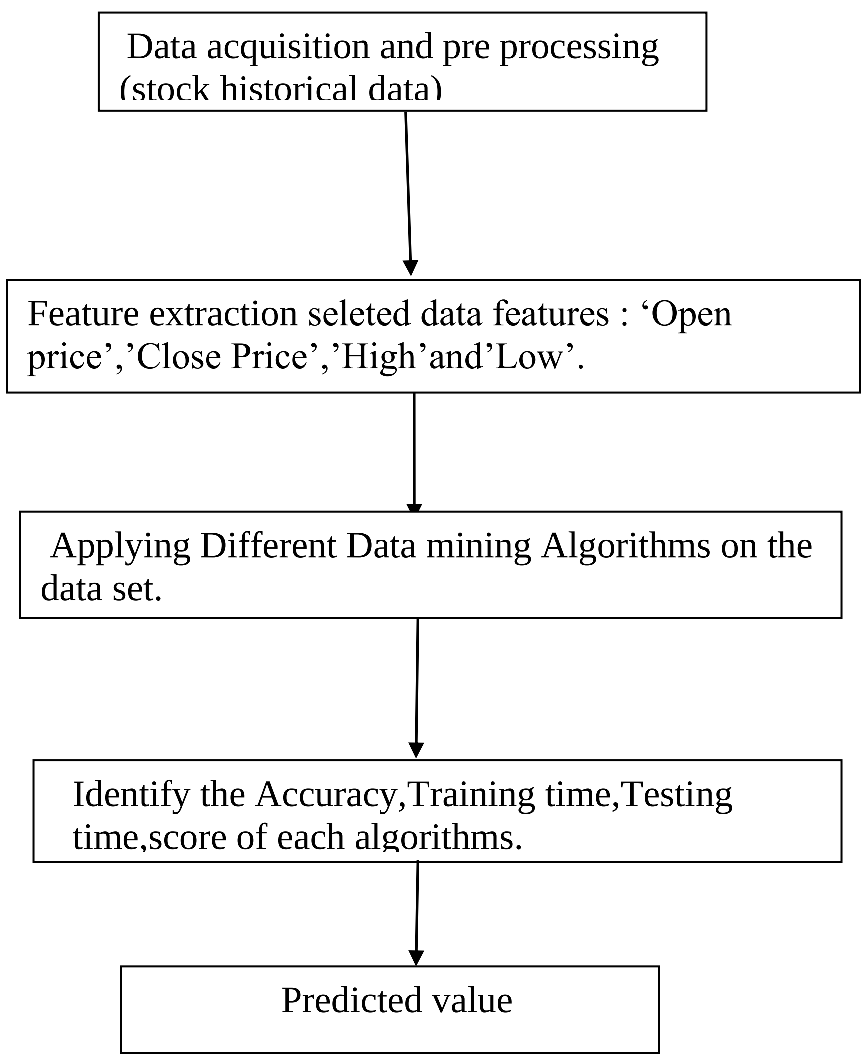
	Date	Open	High	Low	Close	Adj Close	Volume
9123	2022-05-23	255.490005	261.500000	253.429993	260.649994	260.649994	33175400
9124	2022-05-24	257.890015	261.329987	253.500000	259.619995	259.619995	29043900
9125	2022-05-25	258.140015	264.579987	257.130005	262.519989	262.519989	28547900
9126	2022-05-26	262.269989	267.109985	261.429993	265.899994	265.899994	25002100
9127	2022-05-27	268.480011	273.339996	267.559998	273.239990	273.239990	26910800

CHAPTER-7

PROJECT IMPLEMENTATION

7.1 PROJECT WORKFLOW

Stock Market Price Prediction - is based on a Linear Regression Classification Algorithm. It was developed in Python using Jupiter Lab IDE. In thisreport, we will be outlining the algorithms and pseudo-codes used through well elaboratedflowcharts and explanations of the above mentioned Methodologies



Project Detailed Workflow: Steps

• Collecting Sample data related to Stock market Analysis.

• Building of a dataset from a set of known and well-done datasets.

• Loading and Analyzing dataset.

• Splitting the dataset into training and testing sets.

• Preprocessing of the dataset.

• Choose a Learning Model, Methodology, or Schema for training the dataset.

• Fitting the Model with proper parameters and Predicting a feasible outcome (likelihood).

• Determining the Model Accuracy Score.

• Report and Visualization of the predicted outcomes.

• If the results are not that convincing, then Tuning and Optimizing Model with necessary algorithms, is needed.

• Testing the Optimized Model and Reporting its whereabouts and results.

7.2 FEATURE ENGINEERING:

Feature engineering: Feature engineering- is the process of using domain knowledge to extract features from raw data via data mining techniques. These features can be used to improve the performance of machine learning algorithms. Feature engineering can be considered as applied machine learning itself.

Feature Extraction- It uses data reduction which allows the elimination of less important features. There are many features available about stock prices,to yield a better results we have seleted a certain features to develop our model.The seleted data features are ‘Open price’,’Close Price’,’High’ and ’Low’.

Feature Selection: We select the feature of open,close,high , low as input data and feature ‘closing price’ of stock we have taken as target data or result/output from the input data.

7.2.1 TRAINING AND TEST SET GENERATION:

The train-test split procedure is used to estimate the performance of machine learning algorithms when they are used to make predictions on data not used to train the model. It is a fast and easy procedure to perform, the results of which allow you to compare the performance of machine learning algorithms for your predictive modeling problem.

In this research project, we used a split percentage of Train: 80%, Test: 20% which is also called an 8:2 split ratio on the compiled dataset. We split the dataset with the selected features. After splitting the dataset, we got:

• X-Train: which represents 80% of the dataset.

• X-Test: }which represents 20% of the dataset.

• Y-Train and Y-Test: which are the target data for X-Train and X-Test respectively with the same split ratio.

7.2.2 ACCURACY:

ML algorithm , comes with a score method .The score method calculates a widely accepted scoring metric based on type of algorithm . For example, classification algorithms calculate the classification accuracy score when the score method is called.

This score represents the precent of correct classification predictions, and ranges between 0 and 1with a higher score being more accurate.

As you can see the code to score a model is very simple . Where we passed the train versions of X and Y to the fit method, we pass the test versions of X and y to the score method. In one line ,this will generate predictions using the features contained in X_test and compare the predictions to the actual values contained in y_test . Our baseline model correctly classified 0.99% of the reviews in the test set.

7.2.3 RESULTS ANALYSIS

We were able to purchase a share at the exact price open price recorded .

We were able to sell that share just before closing at the exact price recorded.

We’re applying this model on data very close to the training data;

CHAPTER-8

SYSTEM DESIGN

System Design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm and area of application. Design is the first step in the development phase for any engineered product or system. The designer's goal is to produce a model or representation of an entity. The importance can be stated with a single word "Quality". Design is the place where Quality is fostered in software development. Design provides us with representation of software that can assess for Quality. Design is the only way that we can accurately translate a customer's view into a finished software product or system.

8.1 ARCHIETECTURE DIAGRAM

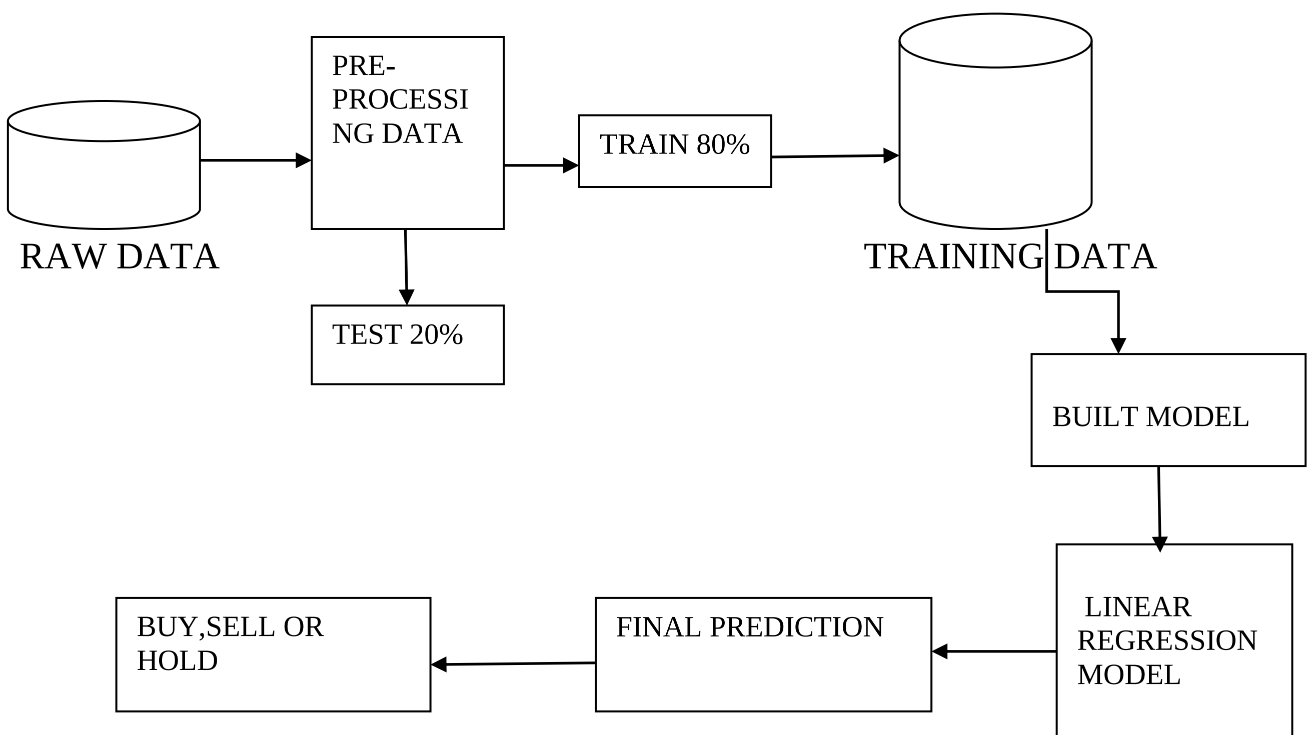


Figure8.1 : Architecture diagram of stock market prediction

8.1.1 USE CASE DIAGRAM

In the Unified Modelling Language (UML), a use case diagram can summarize the details of your system's users (also known as actors) and their interactions with the system. To build one, you'll use a set of specialized symbols and connectors. An effective use case diagram can help your team discuss and represent:

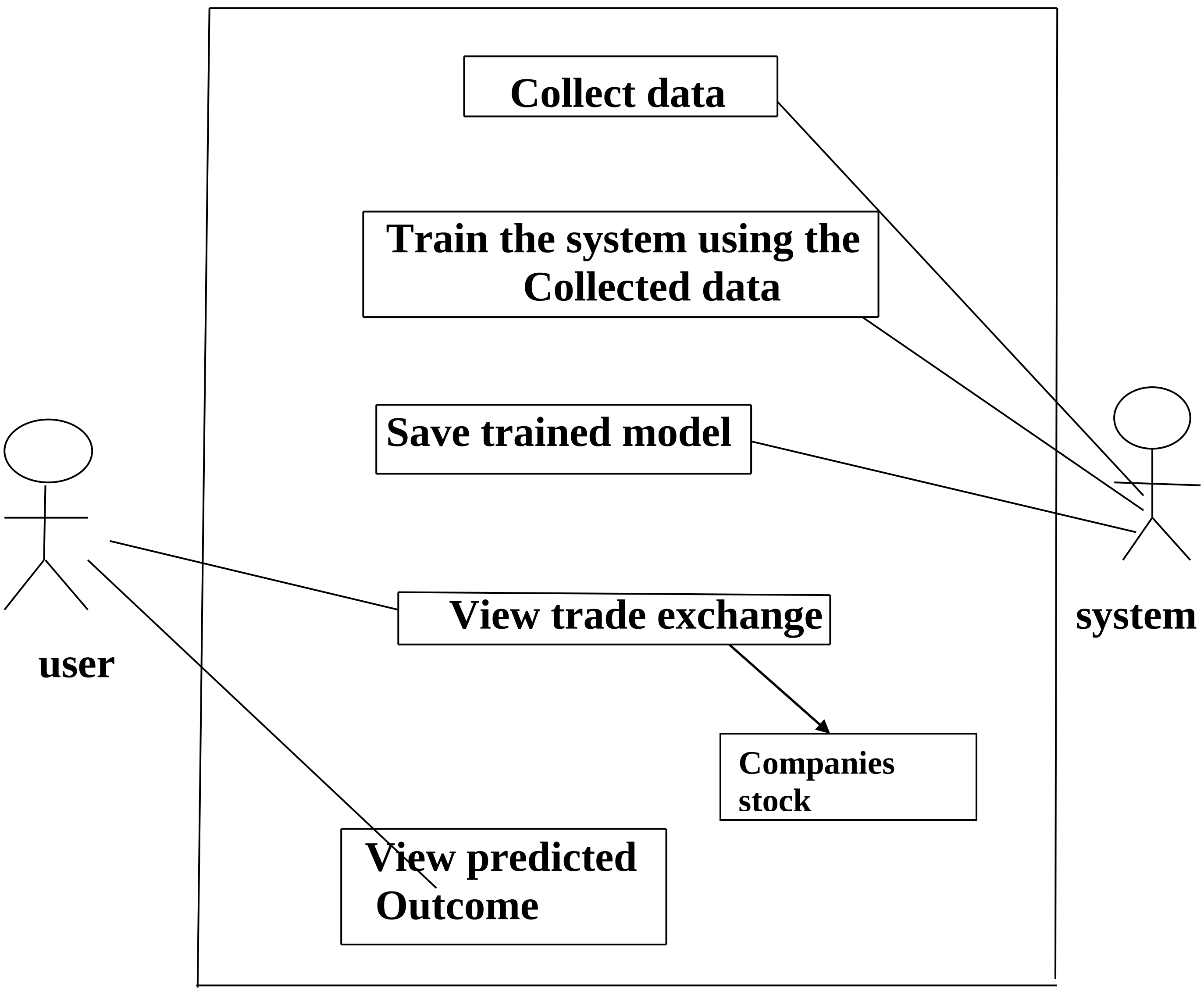


Figure 8.1.1: Use case diagram of stock market prediction

8.1.2 Sequence Diagram

A sequence diagram is a type of interaction diagram because it describes how and in what order a group of objects works together. These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process. Sequence diagrams are sometimes known as event diagrams or event scenarios.

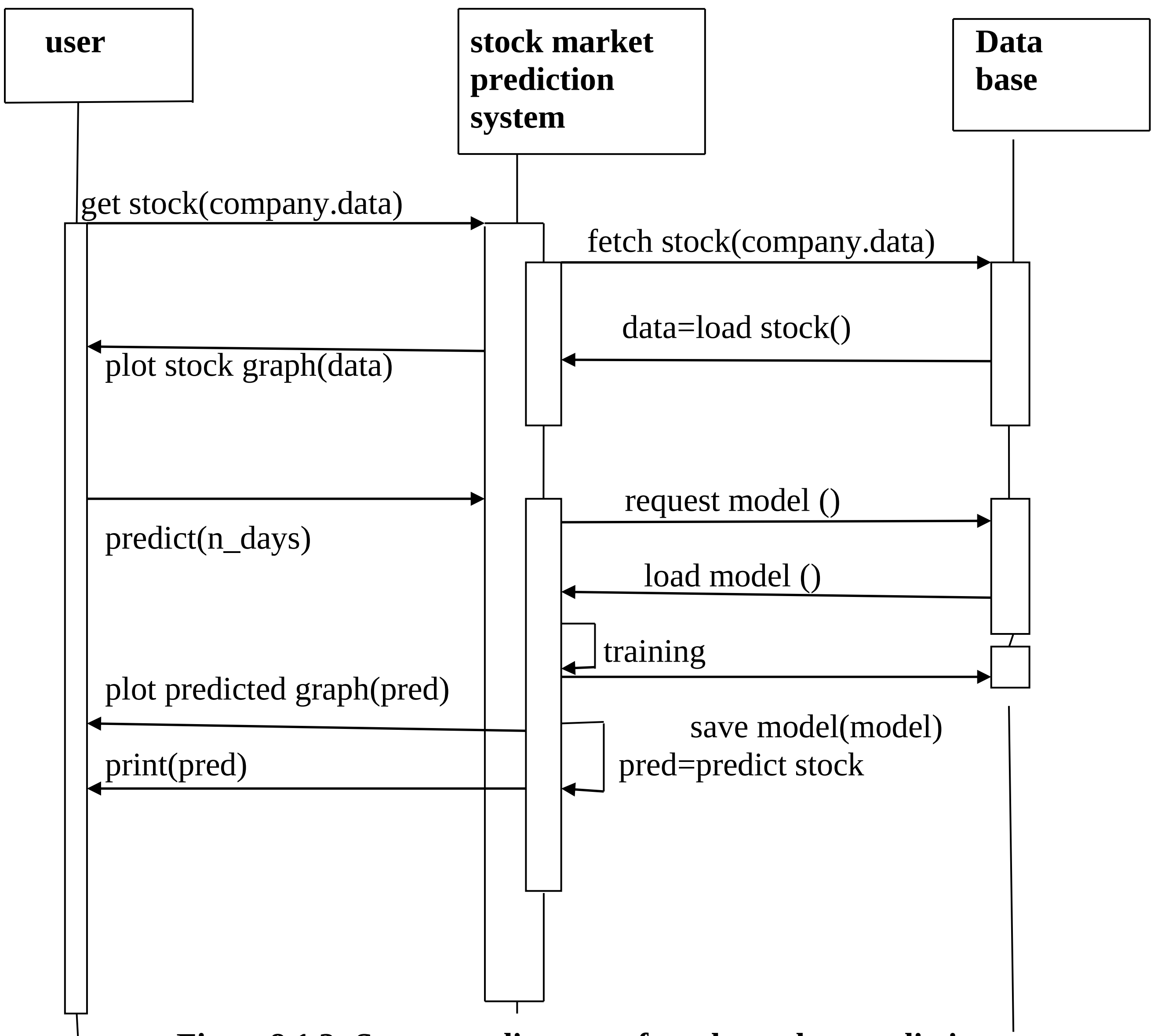


Figure8.1.2: Sequence diagram of stock market prediction

8.1.3 Activity Diagram

An activity diagram is a behavioral diagram i.e. it depicts the behavior of a system. An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed.

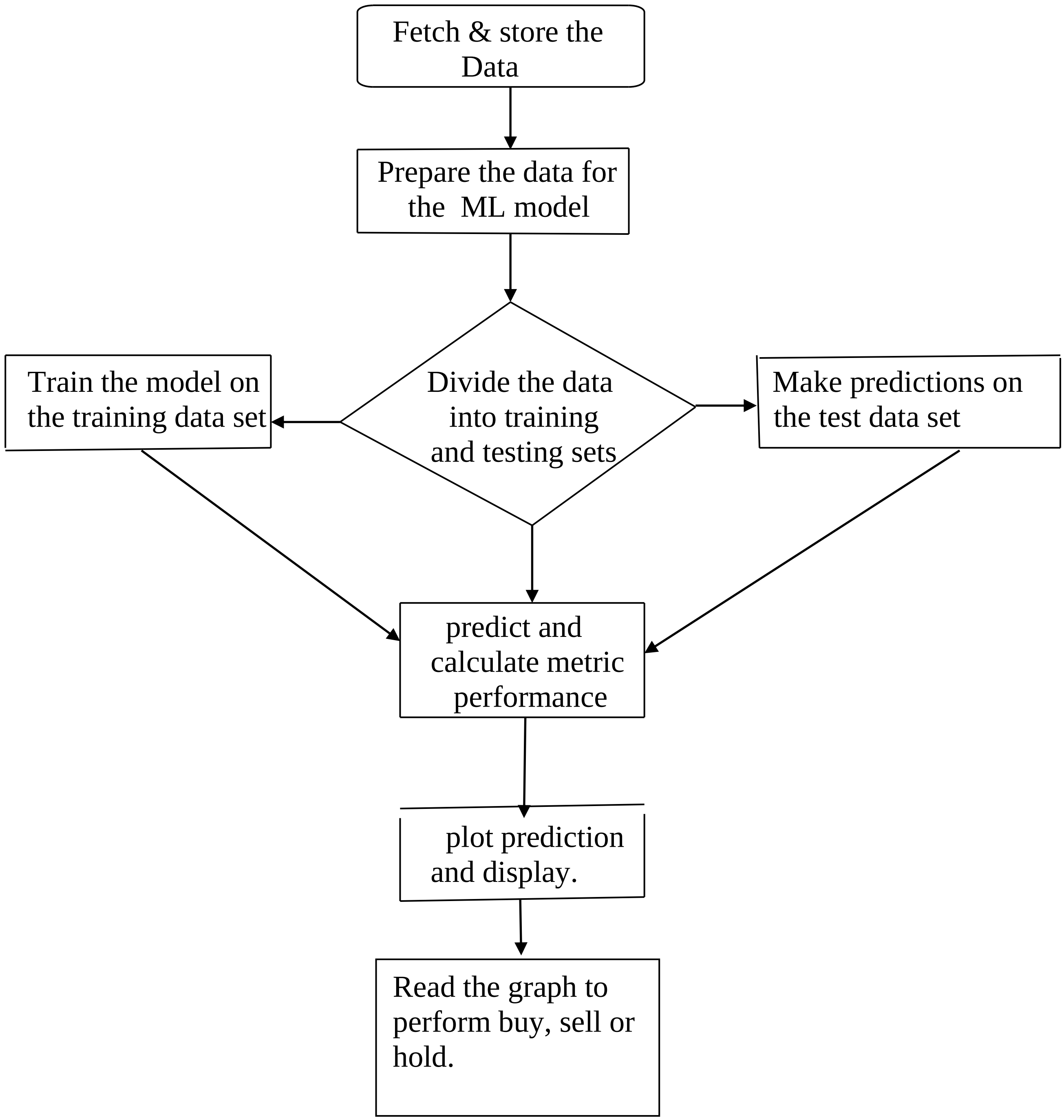


Figure 8.1.3: Activity Diagram

CHAPTER 9
CONCLUSION AND FUTURE ENHANCEMENT

9.1 CONCLUSION

There are specific problems in the world that push the capabilities of data science and the technologies available in this field to their edge. Among them is the stock market prediction. It is challenging for a person to create such a model, but there are ways through which this art can be learned. By measuring the accuracy of the different algorithms, we found that the most suitable algorithm for predicting the market price of a stock based on various data points from the historical data is the Linear Regression algorithm. The algorithm will be a great asset for brokers and investors for investing money in the stock market since it is trained on a huge collection of historical data and has been chosen after being tested on a sample data . The project demonstrates the machine learning model to predict the stock value with more accuracy as compared to previously implemented machine learning models.

9.2 FUTURE ENHANCEMENT

Future scope of this study involves considering more multiple companies from any stock exchange of different countries and also performing comparison of any different techniques of prediction so that one can understand which technique has less MSE, MAE and R2 Score.

CHAPTER-10

APPENDIX-1

SAMPLE CODE

```
!pip install yfinance
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

from sklearn.linear_model import LinearRegression

import warnings
warnings.filterwarnings("ignore")
import pickle

import yfinance as yf
yf.pdr_override()

symbol = 'MSFT'
start = '1986-03-13'
end = '2022-05-28'
df = yf.download(symbol, start, end)
df = df.reset_index()

df.head()

df.tail()

df.describe()

df.plot.line(y="Close", use_index=True)
plt.scatter(x = df.Date, y = df.Close)
```

```
plt.xlabel("Trading days")
plt.ylabel("Closing days")
plt.title("Scatter Plot of Trading days vs Closing days")
plt.show()

X = df['Open']
Y = df['Adj Close']
X_train = df[df.columns[1:5]] # data_aal[['open', 'high', 'low', 'close']]
Y_train = df['Adj Close']

X_train = X_train.values[:-1]
Y_train = Y_train.values[1:]

lr = LinearRegression()
lr.fit(X_train, Y_train)
X_test = df[df.columns[1:5]].values[:-1]
Y_test = df['Adj Close'].values[1:]

lr.score(X_test, Y_test)
lr.score(X_train, Y_train)

opening_price = float(input('Open: '))
high = float(input('High: '))
low = float(input('Low: '))
close = float(input('Close: '))
print('My Prediction the opening price will be:', lr.predict([[opening_price, high, lo
w, close]])[0])
```

We will build a dashboard to analyze stocks. Dash is a python framework that provides an abstraction over flask and react.js to build analytical web applications. Before moving ahead, you need to install dash. Run the below command in the terminal.

```
pip3 install dash
pip3 install dash-html-components
pip3 install dash-core-components
```

Now make a new python file stock_app.py

```
import dash
import dash_core_components as dcc
import dash_html_components as html
import pandas as pd
import plotly.graph_objs as go
from dash.dependencies import Input, Output
from keras.models import load_model
from sklearn.preprocessing import minmax_scaler
import numpy as np
app = dash.Dash()
server = app.server
scaler=MinMaxScaler(feature_range=(0,1))
df_nse = pd.read_csv("./NSE-MSFT.csv")
df_nse["Date"]=pd.to_datetime(df_nse.Date,format="%Y-%m-%d")
df_nse.index=df_nse['Date']
data=df_nse.sort_index(ascending=True,axis=0)
new_data=pd.DataFrame(index=range(0,len(df_nse)),columns=['Date','Close'])
for i in range(0,len(data)):
new_data["Date"][i]=data['Date'][i]
new_data["Close"][i]=data["Close"][i]
new_data.index=new_data.Date
new_data.drop("Date",axis=1,inplace=True)
dataset=new_data.values
train=dataset[0:987,:]
valid=dataset[987:,:]
scaler=MinMaxScaler(feature_range=(0,1))
scaled_data=scaler.fit_transform(dataset)
x_train,y_train=[],[]
for i in range(60,len(train)):
x_train.append(scaled_data[i-60:i,0])
y_train.append(scaled_data[i,0])
x_train,y_train=np.array(x_train),np.array(y_train)
x_train=np.reshape(x_train,(x_train.shape[0],x_train.shape[1],1))
model=load_model("saved_model.h5")
inputs=new_data[len(new_data)-len(valid)-60:].values
inputs=inputs.reshape(-1,1)
inputs=scaler.transform(inputs)
X_test=[]
for i in range(60,inputs.shape[0]):
X_test.append(inputs[i-60:i,0])
```

```
X_test=np.array(X_test)
X_test=np.reshape(X_test,(X_test.shape[0],X_test.shape[1],1))
closing_price=model.predict(X_test)
closing_price=scaler.inverse_transform(closing_price)
train=new_data[:987]
valid=new_data[987:]
valid['Predictions']=closing_price
df= pd.read_csv("./stock_data.csv")
app.layout = html.Div([
html.H1("Stock Price Analysis Dashboard", style={"textAlign": "center"}),
dcc.Tabs(id="tabs", children=[
dcc.Tab(label=MSFT  Stock Data',children=[
html.Div([
html.H2("Actual closing price",style={"textAlign": "center"}),
dcc.Graph(
id="Actual Data",
figure={
"data":[
go.Scatter(
x=train.index,
y=valid["Close"],
mode='markers'
)
],
"layout":go.Layout(
title='scatter plot',
xaxis={'title':'Date'},
yaxis={'title':'Closing Rate'}
)
}
),
html.H2("LR Predicted closing price",style={"textAlign": "center"}),
dcc.Graph(
id="Predicted Data",
figure={
"data":[
go.Scatter(
x=valid.index,
y=valid["Predictions"],
mode='markers'
```

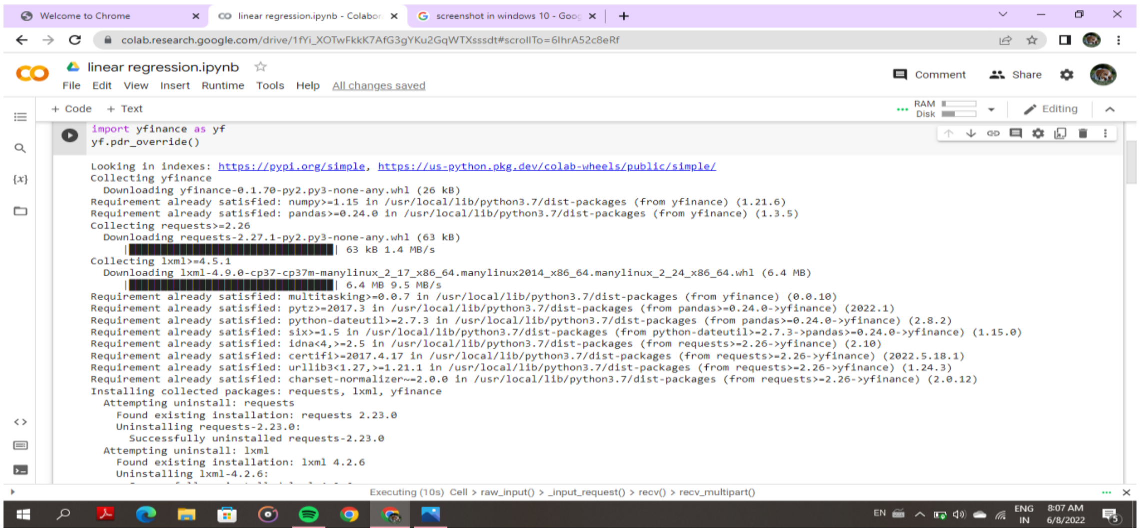
```
)
],
"layout":go.Layout(
title='scatter plot',
xaxis={'title':'Date'},
yaxis={'title':'Closing Rate'}
)
}
)
])
]),
options={'label': 'Microsoft','value': 'MSFT'}],
multi=True,value=['FB'],
style={"display": "block", "margin-left": "auto",
"margin-right": "auto", "width": "60%"}),
dcc.Graph(id='highlow'),
html.H1("Facebook Market Volume", style={'textAlign': 'center'}),
dcc.Dropdown(id='my-dropdown2',
options=[{'label': 'Tesla', 'value': 'TSLA'},
{'label': 'Apple','value': 'AAPL'},
{'label': 'Facebook', 'value': 'FB'},
{'label': 'Microsoft','value': 'MSFT'}],
multi=True,value=['FB'],
style={"display": "block", "margin-left": "auto",
"margin-right": "auto", "width": "60%"}),
dcc.Graph(id='volume')
], className="container"),
])
])
])
@app.callback(Output('highlow', 'figure'),
[Input('my-dropdown', 'value')])
def update_graph(selected_dropdown):
dropdown = {"TSLA": "Tesla","AAPL": "Apple","FB": "Facebook","MSFT":
"Microsoft",}
trace1 = []
trace2 = []
for stock in selected_dropdown:
trace1.append(
go.Scatter(x=df[df["Stock"] == stock]["Date"],
```

```
y=df[df["Stock"] == stock]["High"],
mode='lines', opacity=0.7,
name=f'High {dropdown[stock]}',textposition='bottom center'))
trace2.append(
go.Scatter(x=df[df["Stock"] == stock]["Date"],
y=df[df["Stock"] == stock]["Low"],
mode='lines', opacity=0.6,
name=f'Low {dropdown[stock]}',textposition='bottom center'))
traces = [trace1, trace2]
data = [val for sublist in traces for val in sublist]
figure = {'data': data,
'layout': go.Layout(colorway=["#5E0DAC", '#FF4F00', '#375CB1',
'#FF7400', '#FFF400', '#FF0056'],
height=600,
title=f"High and Low Prices for {', '.join(str(dropdown[i]) for i in
selected_dropdown)} Over Time",
xaxis={"title":"Date",
'rangeselector': {'buttons': list([{'count': 1, 'label': '1M',
'step': 'month',
'stepmode': 'backward'},
{'count': 6, 'label': '6M',
'step': 'month',
'stepmode': 'backward'},
{'step': 'all'}])},
'rangeslider': {'visible': True}, 'type': 'date'},
yaxis={"title":"Price (USD)"})}
return figure
@app.callback(Output('volume', 'figure'),
[Input('my-dropdown2', 'value')])
def update_graph(selected_dropdown_value):
dropdown = {"TSLA": "Tesla","AAPL": "Apple","FB": "Facebook","MSFT":
"Microsoft",}
trace1 = []
for stock in selected_dropdown_value:
trace1.append(
go.Scatter(x=df[df["Stock"] == stock]["Date"],
y=df[df["Stock"] == stock]["Volume"],
mode='lines', opacity=0.7,
name=f'Volume {dropdown[stock]}', textposition='bottom center'))
traces = [trace1]
```

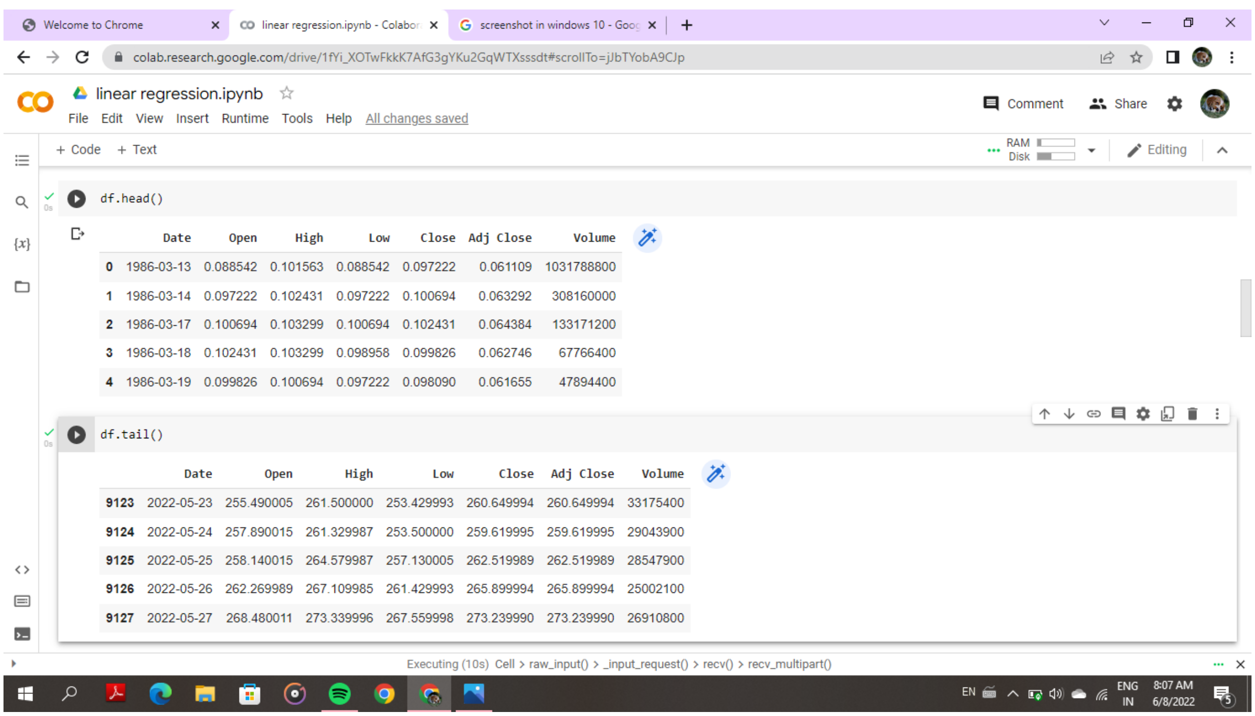
```
data = [val for sublist in traces for val in sublist]
figure = {'data': data,
'layout': go.Layout(colorway=["#5E0DAC", '#FF4F00', '#375CB1',
'#FF7400', '#FFF400', '#FF0056'],
height=600,
title=f"Market Volume for {', '.join(str(dropdown[i]) for i in
selected_dropdown_value)} Over Time",
xaxis={"title":"Date",
'rangeselector': {'buttons': list([{'count': 1, 'label': '1M',
'step': 'month',
'stepmode': 'backward'},
{'count': 6, 'label': '6M',
'step': 'month',
'stepmode': 'backward'},
{'step': 'all'}])},
'rangeslider': {'visible': True}, 'type': 'date'},
yaxis={"title":"Transactions Volume"})}
return figure
if __name__=='__main__':

app.run_server(debug=True)
```

APPENDIX 2(SCREENSHOTS)

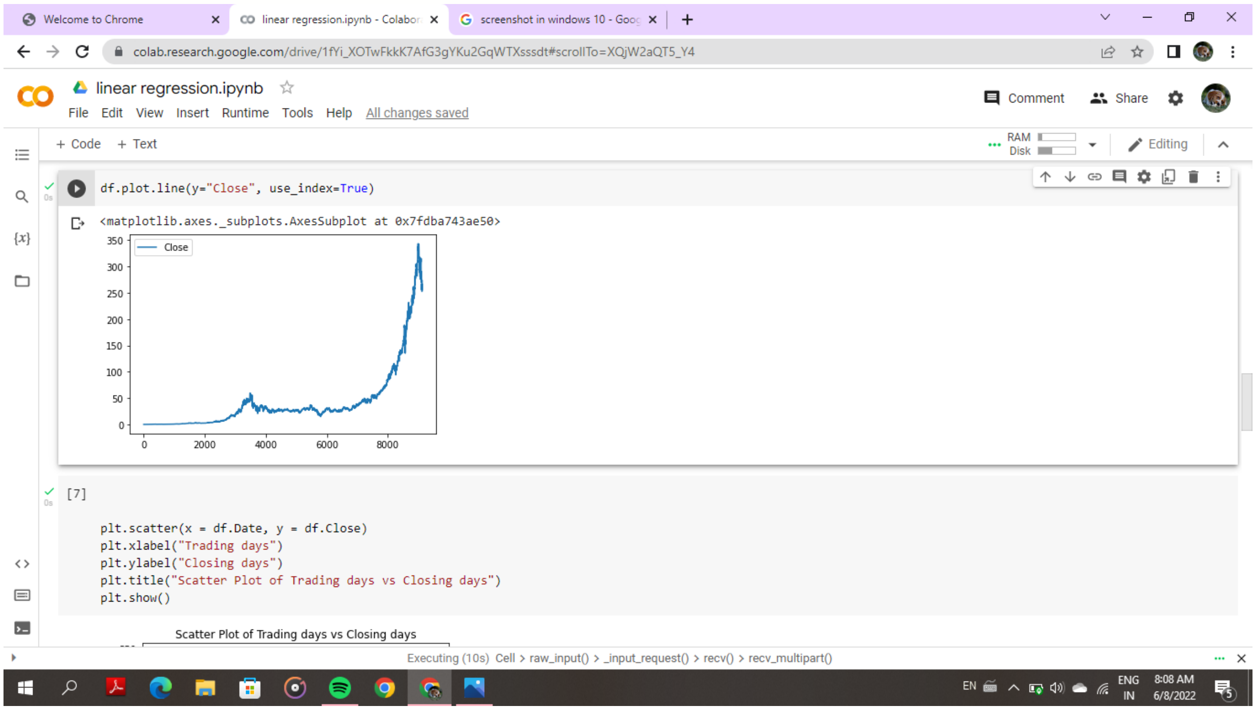
STEP 1: IMPORTING MODULES



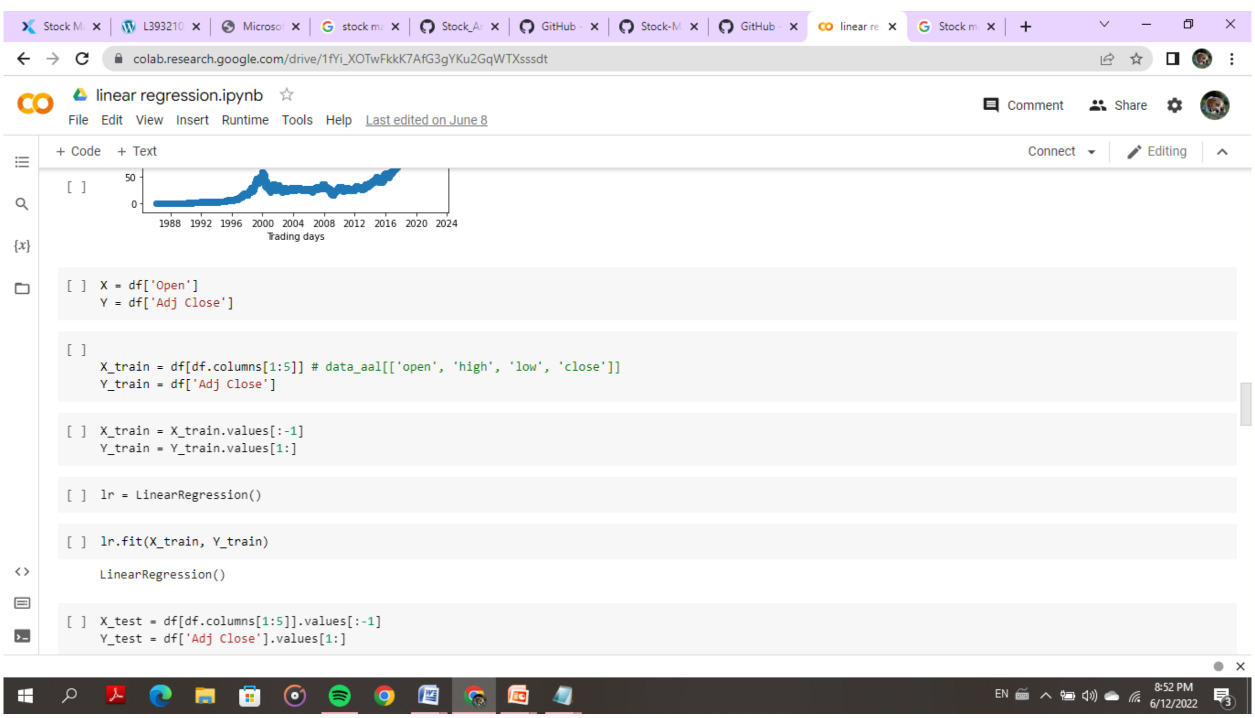
STEP2. LOADING AND PREPARATION



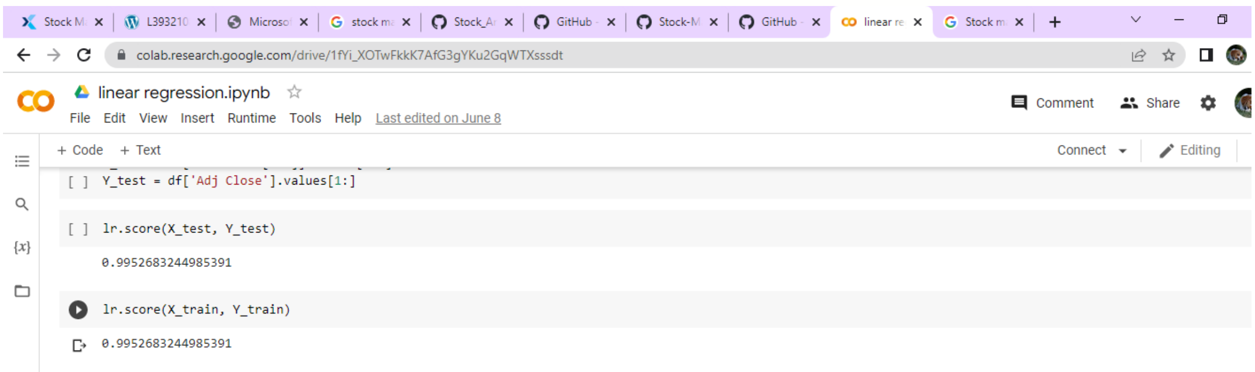
STEP 3: VISUALIZING THE DATA



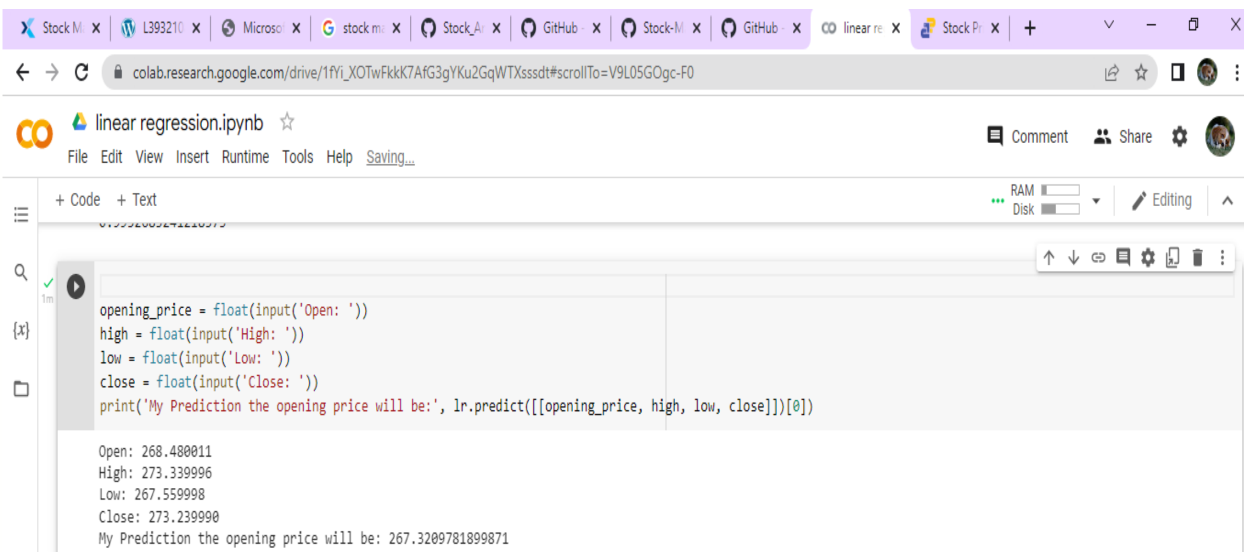
STEP 4 :CREATING A LINEAR REGRESSION MODEL



STEP 5 :CHECKING THE ACCURACY SCORE:



STEP 6: MAKE PREDICTION



REFERENCES

1.Ishita Parmar, Navanshu Agarwal, Sheirsh Saxena, Ridam Arora, Shikhin Gupta, Himanshu Dhiman, Lokesh Chouhan Department of Computer Science and Engineering National Institute of Technology, Hamirpur – 177005, INDIA - Stock Market Prediction Using Machine Learning.

2.Pranav Bhat Electronics and Telecommunication Department, Maharashtra Institute of Technology, Pune. Savitribai Phule Pune University - A Machine Learning Model for Stock Market Prediction.

3.Anurag Sinha Department of computer science, Student, Amity University Jharkhand Ranchi, Jharkhand (India), 834001 - Stock Market Prediction Using Machine Learning.

4.V Kranthi Sai Reddy Student, ECM, Sreenidhi Institute of Science and Technology, Hyderabad, India - Stock Market Prediction Using Machine Learning.

5.Asset Durmagambetov currently works at the mathematics, CNTFI. Asset does research in Theory of Computation and Computing in Mathematics, Natural Science, Engineering and Medicine. Their current project is 'The Riemann Hypothesis-Millennium Prize Problems' - stock market predictions.

6. Tiwari S, Bharadwaj A, Gupta S. Stock price prediction using data analytics. In 2017 InternationalConference on Advances in Computing, Communication and Control (ICAC3) 2017 Dec 1 (pp. 1-5).IEEE

7.Usmani M, Adil SH, Raza K ,Ali SS. Stock market prediction using machine learning techniques.In 2016 3rd international conference on computer and information sciences(ICCOINS) 2016 Aug 15 (pp.322-327).IEEE.

8. Shunrong Shen, Haomiao Jiang, and T. Zhang, "Stock Market Forecasting Using Machine Learning Algorithms," IEEE Transactions on Neural Networks, vol. 84, no. 4, pp. 21, 2015

9. Yan, Robert, and Charles Ling. 2007. "Machine Learning for Stock Selection." Industrial andGovernment Track Short Paper Collection 1038-1042.

10. Hurwitz, E, and T Marwala. 2009. "Common Mistakes when Applying Computational Intelligence and Machine Learning to Stock Market modelling." University of Johannesburg Press.

11. Sharanya Banerjee, Neha Dabeeru, R. Lavanya SRM Institute of Science and Technology, Chennai in Computer Science and Engineering - Stock Market Prediction.