STOCK PRICE PREDICTION

Project submitted in partial fulfillment of the requirements for the award of the degree

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

BY

PANDIRI VJAYALAXMI **(18C91A0568)**

RAHUL SINGH **(18C91A0578)**

PEDDI ADITHYA RAM **(18C91A0573)**

Under the Esteemed guidance of

DR. BIRRU DEVENDER M.Tech(Ph D)

Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

HOLY MARY INSTITUTE OF TECHNOLOGY & SCIENCE

(COLLEGE OF ENGINEERING)

(Approved by AICTE New Delhi, Permanently Affiliated to JNTU Hyderabad, Accredited by NAAC with 'A' Grade)

Bogaram (V), Keesara (M), Medchal District -501 301.

2021 – 2022

HOLY MARY INSTITUTE OF TECHNOLOGY & SCIENCE

(COLLEGE OF ENGINEERING)

(Approved by AICTE New Delhi, Permanently Affiliated to JNTU Hyderabad, Accredited by NAAC with 'A' Grade)

Bogaram (V), Keesara (M), Medchal Dist-501301.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the mini project entitled "STOCK PRICE PREDICTION" is being submitted by VIJAYALAXMI (18C91A0568), RAHUL SINGH (18C91A0578), P ADITHYA RAM (18C91A0573), in Partial fulfillment of the academic requirements for the award of the degree of Bachelor of Technology in"COMPUTER SCIENCE AND ENGINEERING" HOLY MARY INSTITUTE OF TECHNOLOGY & SCIENCE, JNTU Hyderabad during the year 2021-2022.

INTERNALGUIDE

DR BIRRU DEVENDER

Professor

Dept. of Computer Science&Engineering.

HEAD OF THEDEPARTMENT

DR .B.NARSIMHA M.Tech,Ph.D.

Professor&HoD

Dept. of Computer Science &Engineering

EXTERNAL EXAMINER

II

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completionof any task would be incomplete without the mention of the people who made it possible, who's constant guidance and encouragement crowns all effort with success.

I take this opportunity to express my profound gratitude and deep regards to My Guide **Dr.B Devender Dept.** of Computer Science & Engineering, Holy Mary Institute of Technology & Science for his / her exemplary guidance, monitoring and constant encouragement throughout the project work.

My special thanks to **Dr.B.Narsimha, Head of the Department**, Dept.of Computer Science & Engineering, Holy Mary Institute of Technology & Science who has given an immense support throughout the course of the project.

I also thank to **Dr. P. Bhaskara Reddy,** the **honorable Director** of my college Holy Mary Institute of Technology & Science for providing me the opportunity to carry out this work.

At the outset, I express my deep sense of gratitude to the beloved **Chairman A. Siddartha Reddy** of **Holy Mary Institute of Technology & Science**, for giving me the opportunity to complete my course of work

I am obliged to **staff members** of Holy Mary Institute of Technology & Science for the valuable information provided by them in their respective fields. I am grateful for their cooperation during the period of my assignment.

Last but not the least I thank **ALMIGHTY** and My **Parents**, and **Friends** for their constant encouragement without which this assignment would not be possible.

P VIJAYALAXMI **(18C91A0568)**

RAHUL SINGH **(18C91A0578)**

P ADITHYA RAM **(18C91A0573)**

DECLARATION

This is to certify that the work reported in the present project titled **"STOCK PRICE PREDICTION"** is a record of work done by me in the Department of Computer Science & Engineering, Holy Mary Institute of Technology and Science.

No part of the thesis is copied from books/journals/internet and wherever the portion is taken, the same has been duly referred in the text the reported are based on the project work done entirely by me not copied from any other source.

P VIJAYALAXMI	**(18C91A0568)**
RAHUL SINGH	**(18C91A0578)**
P ADITHYA RAM	**(18C91A0573)**

INDEX

ABSTRACT

Content Page No.

1. INTRODUCTION

1.1 Problem Statement ……2

1.2 Objectives ……3

1.3 Motivation……..3

1.4 Existing System ……3

1.5 Proposed System ……...4

2. LITERATURE SURVEY

2.1 Existing system……..…….6

2.2 Proposed system ……..…… 7

2.3 Summery ……...…… 8

3. SOFTWARE REQUIREMENTS SPECIFICATIONS

3.1 Software Requirements……..……. 9

3.2 Hardware Requirements……..……... 10

4.SYSTEM DESIGN

System Architecture ……..…11

5 .IMPLEMENTATION

5.1 Environmental Setup …….15

5.2 Module Description ……...25

5.3 Software Description …….26

5.4 Sample Code ……..27

6.SYSTEM TESTING

5.5 Tests …….32

7.RESULTSCREEN SHOTS …….35

8 .CONCLUSION ……....42

9. BIBLIOGRAPHY……...……...43

9. LIST OF FIGURES :

4.1System Architecture …….……11

4.2 Use case Daigram…….……13

4.3 Data Flow Daigram…….……14

10 .LIST OF SCREENSHOTS

Capturing the dataset…… 35

Getting data Information…… 37

Prediction of values…… 38

Linear Regression graph……..….39

Lstm graph……..……..40

ABSTRACT

In this we present the stock price prediction. The aim is to predict the future value of financial stocks of company. Prediction of stock market is a challenging task in predicting future prices due to fluctuating nature of stock, soo stock market is too difficult to predict.

The recent trend in stock market prediction technologies, the use of machine learning which makes prediction based on their previous values.

Open,high,low,close prices of stock are used for creating new variables which are inputs to the model.

1.INTRODUCTION

INFORMATION ON STOCK

we all have heard the word stock one way or the other. Particularly stock is related with the associates and companies which are commercialized and are to settling in the world of marketization. The other word used for stock is share which is prominently used in day to day life. People even term is as an investment plan and its something people see as a long term investment that secures and provides an abundant funds during the retirement age. Buying a company stock is purchasing a small share of it. People invest on the same to get a long term benefit which they think is less value for now but has to potential to grow with the time. Its an investment that provides the long time run and deals with long time goals with the fair objectives. The value of share you invest today has to give you an yield of best tomorrow but its not the same.

Market is unpredictable so are the resources and the factors that are taken to drive it off or on on the set. Its never been on the same level and the pattern of the same is still unpredictable till the time. Some closeness and prediction method had been derived and approximates values and the rough figures are generated hoping for the best but all of the resource can't be trusted and are still unpredictable in nature. Knowing the market situation and researching on the same is the best way to find the reliability for which there are many agents who have taken the same as a profession and are making a fortune out of it. They predict and advise but the advisory cost and the charge is higher and the stock evaluation is never less the same.

Market is changing in an instantaneous rate even in a day there are many highs and lows in the market and having said the resources and the timing the external and internal agent. Stock is a fascinating resource to start with.Stock in other term is defined as the fair share or the ownership representation explaining the security measures and the agreement between two parties which are an individual and the company. Stock is there from the start and due to its tendency of uncertainty it has been a word of fancy. People researching on the same and implementing on the daily basis had made a fortune out of it. There are various agents available in market for making you understand and invest on the same and the charges of the same are hectic and insanely expensive.

1.1 Problem statement:

STOCK PRICE PREDICTION USING LSTM

Stock is an unpredictable curve that had been in picture ever since. Its essence had been ever long living and indulging. It had grown its popularity with respect to time. People are more fascinating and interested on the same then before times. Same for the case for the organization. Organization had created it as a better source of revenue generation rather than investing and taking a loan approval from the bank It's way efficient and less hectic from the firm point of view.

Stock is unpredictable and its been the same from the start. Its way of escalating and deescalating had been phenomenon and experiencing the same is the best integral part of it. It has its upper hand and flexibility with the changes that has the chances of uprising as well as crashing the whole market. Its easily defined in few words but making an essence and understanding the same is way more hectic and time consuming.Simpler it sound complex are its phenomenon and integrating the same. Its has its whole different sets of dependencies and integration from different agents which fluctuate the same in the market. Finding an accurate and getting the exact values out of the same is still unaligned and no particular model of the same is seen in the market value. Finding the closest and getting an accurate proximate value out of such an unpredictability is a problem in itself. Merging of the data getting the best prediction to increase the efficiency alongside considering the different expects of the moderator is tough and we took the same in consideration and implemented with every aspect to generate the best out of the same and get a result that can be better interrupted and the efficiency remains the same with the value of different aspects of creating an impact of reducing the risk and influencing the same over the time period to gain the most out of it.This is totally based on Machine Learning Algorithm to proceed and provide an effective result.

1.2.Objectives:

The project aims at building a machine learning model that will be able the predict the future prices of stocks. we will develop a stock prediction, which is prediction of the stock prices by using past data of a company.
In our project we basically on past data of a company .The low, high, close, open price of stock are used for creating new variables which are inputs to our model.

1.3.Motivation:

The most fundamental motivation for trying to predict the stock market prices is financial gain.

Stock price would make the owner of the model very wealthy. Thus researchers, investors and investment professionals are always attempting to find a stock market model that would yield them higher returns than their counterparts.

1.4.Existing system:

There is model support vector machine, which is used for stock market prediction

SVM: Support vector machine, it is also one type of machine learning technique designed to solve regression.

1.5.Proposed system:

We focus on predicting the stock values using machine learning algorithms like long term short term prediction and linear regression.

Lstm is a type of recurrent nueral network.
Linear regression is supervised learning algorithm to predict the prediction continuous variables.
Lstm is most powerful and are known for retaining long term memory.
we can predict the stock price by trying to guess stock's future PE(Price earning ratio),EPS(earning per ratio).

2.LITERATURE SURVEY

One of the integral part to maintain the consistency is the literature survey. It's the crucial steps to be followed in the development process. The Software Development needs authenticity of the resources and the availability of the same. This part helps in discovering the content that been worked on and find the utilization and implementation of the same in today's time. The key factor to the development is the economy and the strength of the product. Once the innovation of the same undergoes through the building phase the support and the resource flow is to be monitored and computed. This is also known as the Research phase where all the research is embedded and done to carry the flow.

MACHINE LEARNING

One of the finest word heard in today time is Machine Learning. Either it be at work or different places the machine learning has been an integral part of todays technology. Though its evolutionalizing and developing in a rapid rate and development and deployment of the same is still in progress. The machine learning itself had brought a random changes in today worlds because of which automation is in frame which was a make existence in the past. It's an aspiring term in todays time.

One of the move that all the firm are interested into. It's a leading pillar for tomorrow leading the world to a better future of evolution where the customization and labor work can be reduce to half and the safety of the survival can be with held to stand tall for the better utilization of human mind. Since Machine is considered most efficient and the level of mistakes are kept at the minimum the level of work flow can be a work of hazard and further improvement on the same may create a thousands sitting idle in home creating a larger impact on unemployment and livelihood. Which in other way is a threat to the society too.

2.1 EXISTING SYSTEM:

As many have invested their time and effort in this world trade for getting it closer and more reliable to the people for carrying out the resources and make their lifestyle more deliberate than the previous. In the past few years various strategies and the plans had been derived and deployed ever since it's continuation and the topic is still a point of research where people are coming up with ideas to solve. Intelligence fascinates mankind and having one in machine and integrating on the same is the hot key of research. There are various people contributing on the same research.

All the learning system from the past are limited and are simplest in nature where learning of the simple algorithm for a computational mean is not enough which can even be done by human brain itself. The main motto of learning was limitized and learning model was not efficient. The existing models can't cope up with the vulnerabilities and remove the rarest information that they can't process causing it a major data loss which creates a problem in forecasting. Observation is the integral part in the resource and prediction management. If the outcome can't be observed it's point predictions are apparently biased because it consider a only source point for data source. Before the prediction of the data set a simple data retrieval should be generated and tested on the training data set which are more flexible and versatile in nature.

Loss of sights is a major problem in the existing system as the stock varies each days and the loss margin can be higher with respect to time. An initial instance is taken for prediction.

There are some similar algorithms, which are mainly used for stock price prediction.

Support vector machine:

Support Vector Machine is a **machine learning technique** used in recent studies to forecast stock prices. ... These are used as parameters to the SVM model. The model attempts to predict whether a stock price sometime in the future will be higher or lower than it is on a given day.

Prediction of continuous values based on observations from the data. SVM is commonly used for classification (assigning a discrete class) and sometimes used for clustering (separate data points to some homogeneous classes). It does't work well when we have large dataset.

2.2.PROPOSED SYSTEM:

Stock is unpredicted and liberal in nature. The follow of the same is impressive and reluctant in nature. Finding the predictability and getting the nearest is the best hit goal for the same. The exact and accurate estimation of the same is never-less possible. There are various constrains that influctuate the pricing and the rate of stock.

Those constrains had to be taken in consideration before jumping to the conclusion and report derivation. Here as described in the figure above, the proposed system will have an input from the dataset which will be extracted featured wise and Classified underneath. The classification technique used is supervised and the various techniques of machine level algorithms are implemented on the same.

We use long term short term algorithm for stock price prediction.
Which is a type of recurrent nueral network. It is most power full and are known for retaining long term memory.

Training Dataset are created for training the machine and the test cases are derived and implemented to carry out the visualization and the plotting's. The result generated are passed and visualized in the graphical form.

2.3.SUMMARY:

In summary, Machine Learning Algorithms are widely utilized by many organizations in Stock market prediction. This article will walk through a simple implementation of analyzing and forecasting the stock prices in Python using various Machine Learning Algoritom.

The process of stock price prediction is shown in this module.

The main objective of this project is to predict the future values of a company.

3.SOFTWARE REQUIREMENTS SPECSIFICATIONS

3.1.Software Reqirements:

Oprating system : Windows & Linux

Coading language : Python

IDE : Jupyter Notebook

Data Set : .csv file

Visualization : mat plot lib, pandas.

TensorFlow

3.2.Hardware Requirements:

Processor : Intel i5 or above

RAM : 4 GB

Input Device : Keyboard

Output Device : Screens of Monitor or a Laptop

4.SYSTEM DESIGN

4.1.System Architecture:

The dataset we use for the proposed project is been taken from Kaggle. But, this data set is in raw format. The data set is a collection of valuation of stock market information about some companies. The initial step is to convert raw data into processed data. Which is done by feature extraction, since the raw data collected have multiple attributes but only some of those attributes are needed for the prediction.

Feature extraction is a reduction process.

The structure, behavior and views of a system is given by structural mode

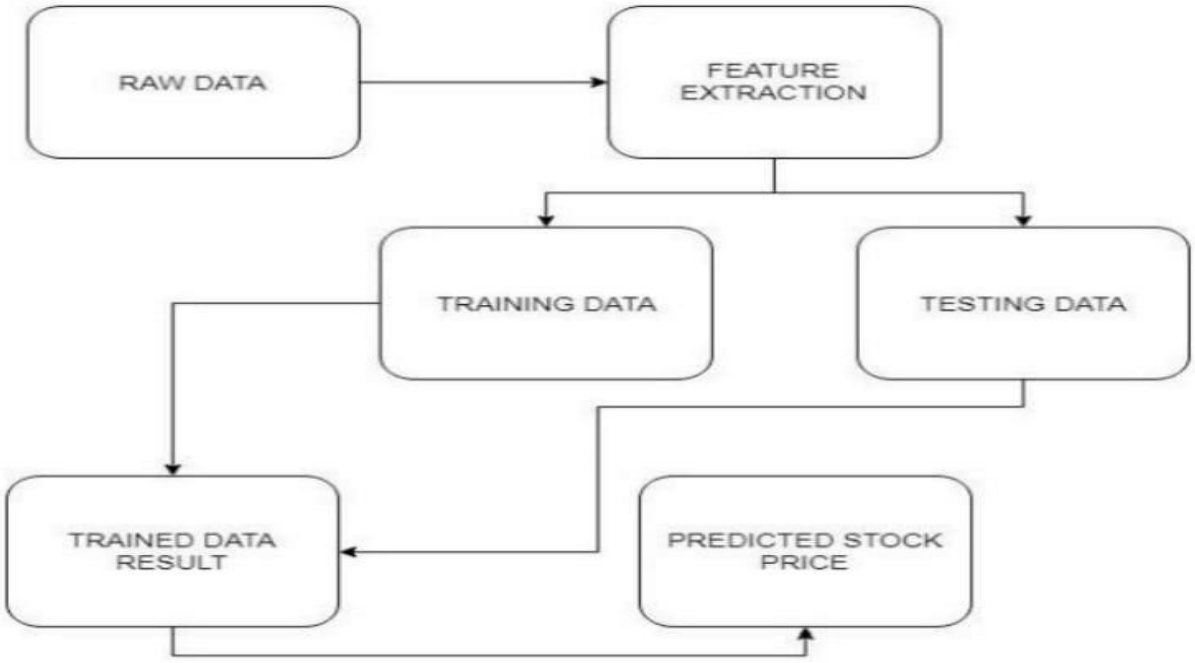


Fig 4.1:Architectural Diagram

Design goals

To make the project runs smoothly it's required that we make plan and design some accepts like flowcharts and system architecture which are defined below.

Data Collection

Data collection is one of the important and basic thing in our project. The right dataset must be provided to get robust results. Our data mainly consists of previous year or To weeks stock prices. We will be taking and analyzing data from Kaggle. After that seeing the accuracy we will use the data in our model.

Data Preprocessing

Human can understand any type of data but machine can't our model will also learn from scratch so it's better to make the data more machine readable. Raw data is usually inconsistent or incomplete .Data preprocessing involves checking missing values, splitting the dataset and training the machine etc.

Training Model

Similar to feeding somethings, machine/model should also learn by feeding and learning on data. The data set extracted from Kaggle will be used to train the model. The training model uses a raw set of data as the undefined dataset which is collected from the previous fiscal year and from the same dataset a refine view is presented which is seen as the desired output. For the refining of the dataset various algorithms are implemented to show the desired output.

USECASE DIAGRAM:

A dynamic and behavioral diagram in UML is use case diagram. Use cases are basically set of actions, services which are used by system. To visualize the functionality requirement of the system this use case diagram are used. The internal and external events or party that may influence the system are also picturized. Use case diagram specify how the system acts on any action without worrying to know about the details how that functionality is achieved.

For the project we have created the below mentioned use case diagram.

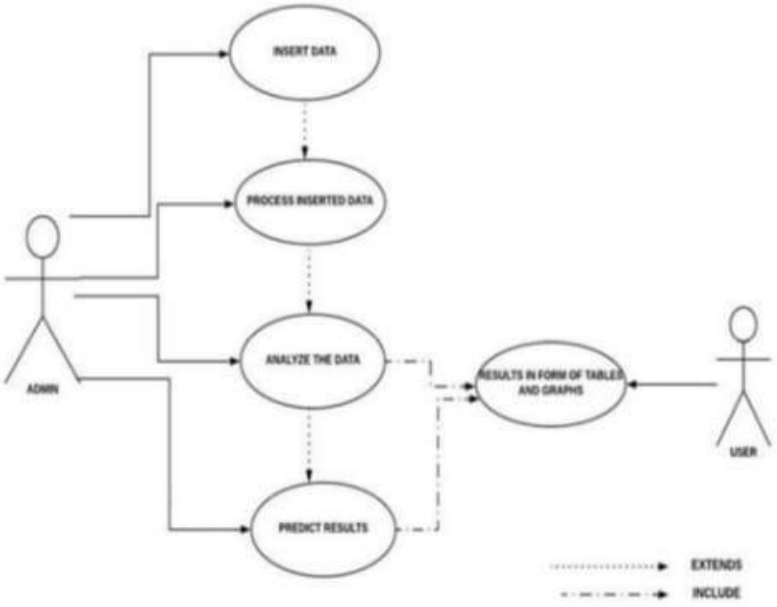


Fig 4.2: Use Case Diagram

The above **figure 4.2** shows the use-case diagram of the entitled project and it's flow. From the diagram it's seen that the user gives the raw dataset as input and with the flow of the input in the system.

The system evaluates and process the dataset train itself with the provided dataset and extract the meaningful dataset to process and refine the cluster data and from the given cluster of the data, the plotting of the data values are shown and with the given range the system plots the data gives a figurative output display on screen.

DATA FLOW DAIGRAM:

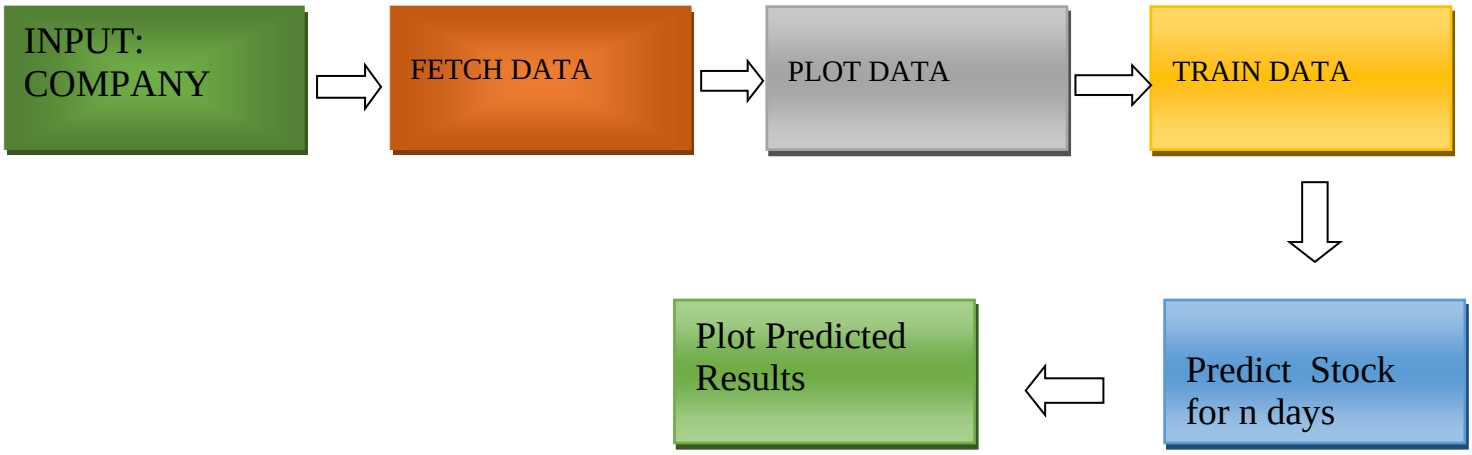


Fig 4.3 Data flow Diagram

In the above fig 4.3 we are taking a company fetching the data of the company from the panda's data-reader library then we are plotting the data, then we train the data to predict the stock for certain number of days. In this way data is flowing in our system

5.IMPLEMENNTATION

5.1.Environmental setup:

Tools:

The following tools which are used to develop the ML application

1.Jupyter:

upyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations, and narrative text. Uses include data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.

2.Umbrello:

Umbrello is a Unified Modelling Language(UML) modeling tool and code generator.It can create industry-standard UML format,and can also generate code from UML diagrams in a variety of programming languages

Features:

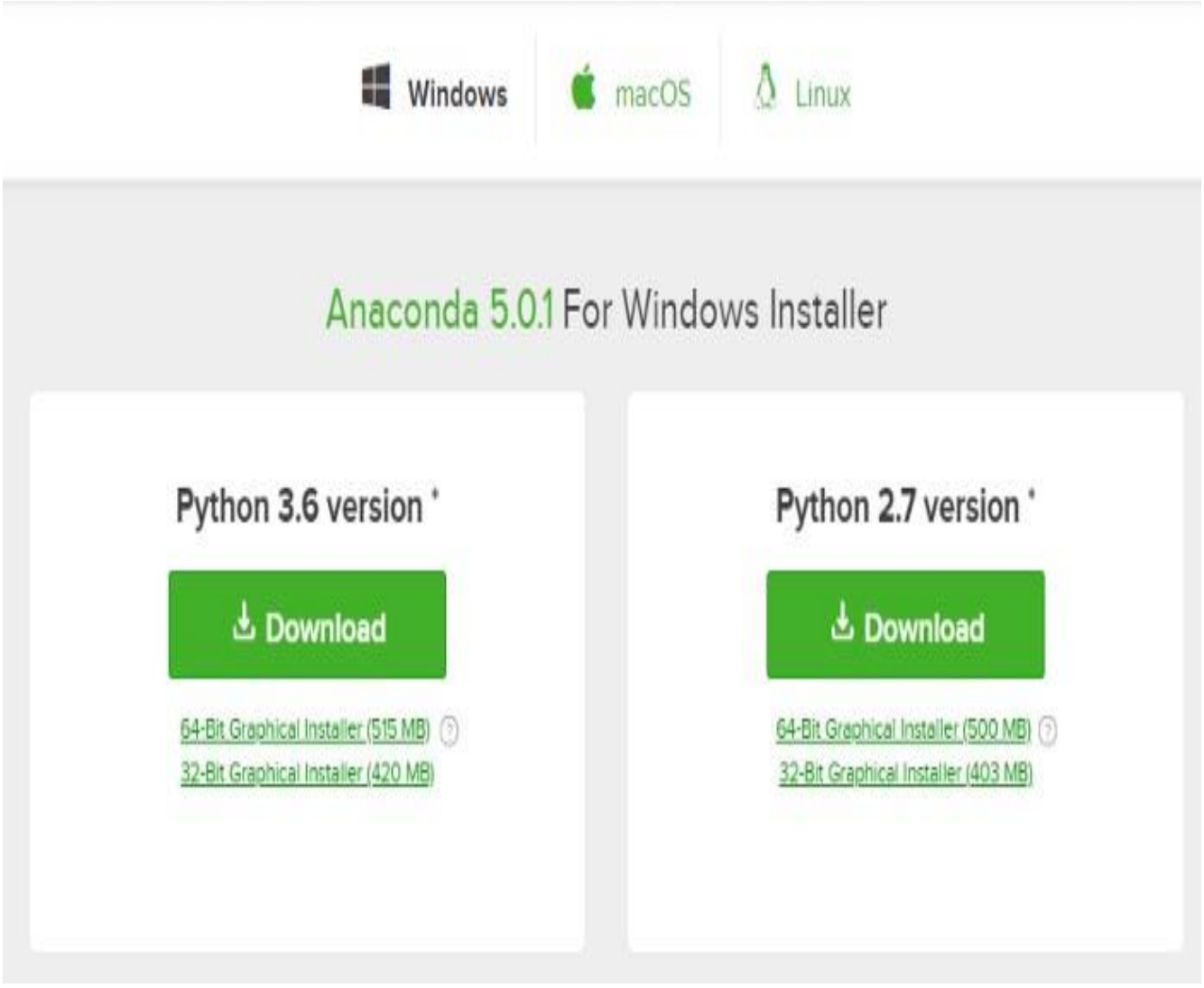
- Supported formats: XMI
- Several type of diagrams supported: use case, class, sequence, communication, state, activity , component, deployment, entity relationship.

Sofware installation procedure

Step 1) Installing Anaconda

1. Install **Anaconda** from https://repo.anaconda.com/archive/Anaconda32021.05-Windows-x86_64.exe.

2.After opening link u can see this download

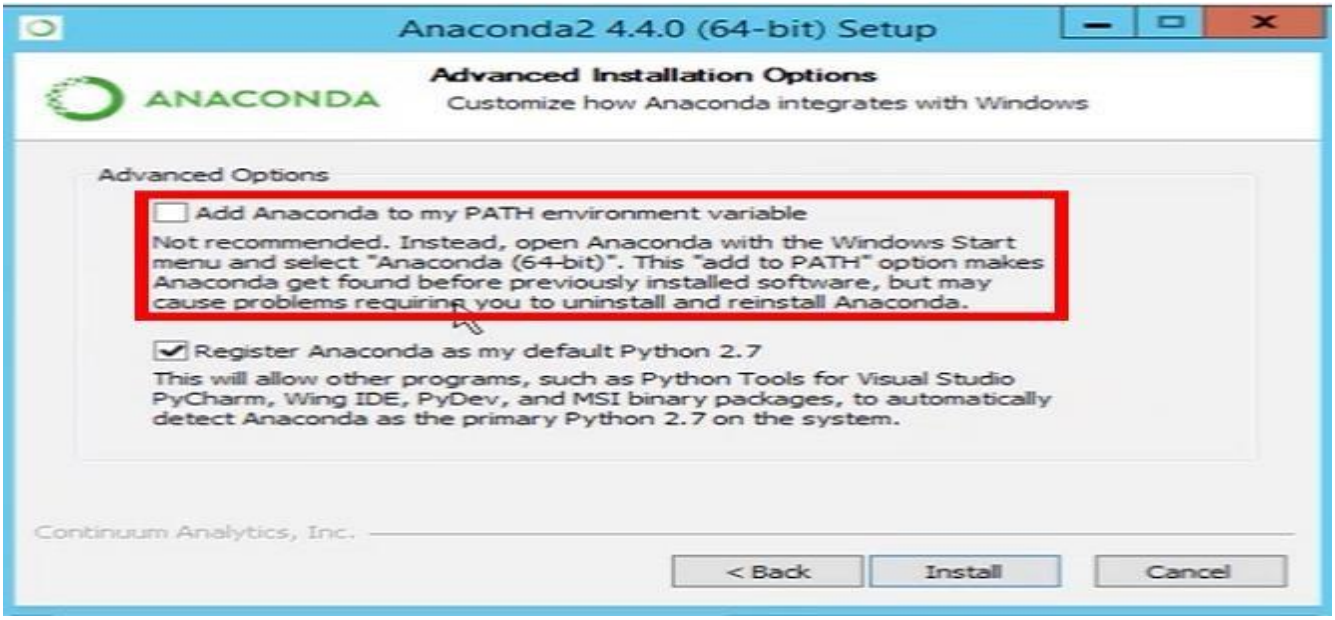


3.click on the download option in above image.

4. after downloading start installation.

4. The default options when prompted during the installation of Anaconda as shown above.

5. Ensure that the path to the folder where Anaconda is installed is added to your computer/system.



6.Open “Anaconda Prompt” by finding it in the Windows (Start) Menu.

7.Type the command in red to verified Anaconda was installed.

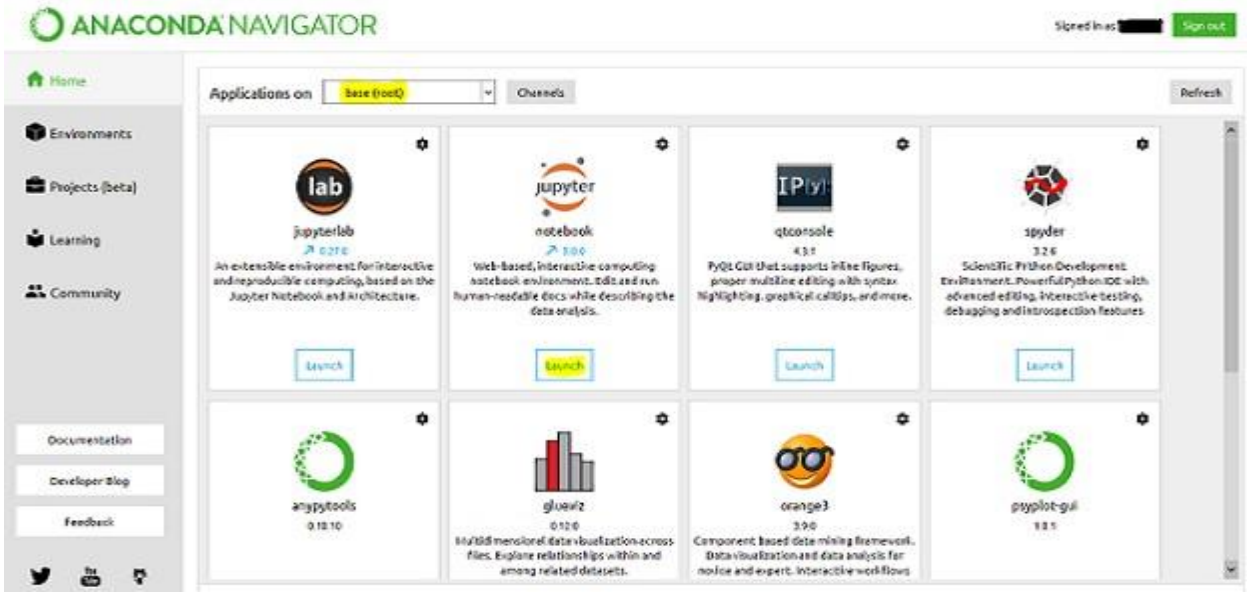
```
python --version
P> ython 3.7.3
```

8.Type the command in red to update Anaconda.

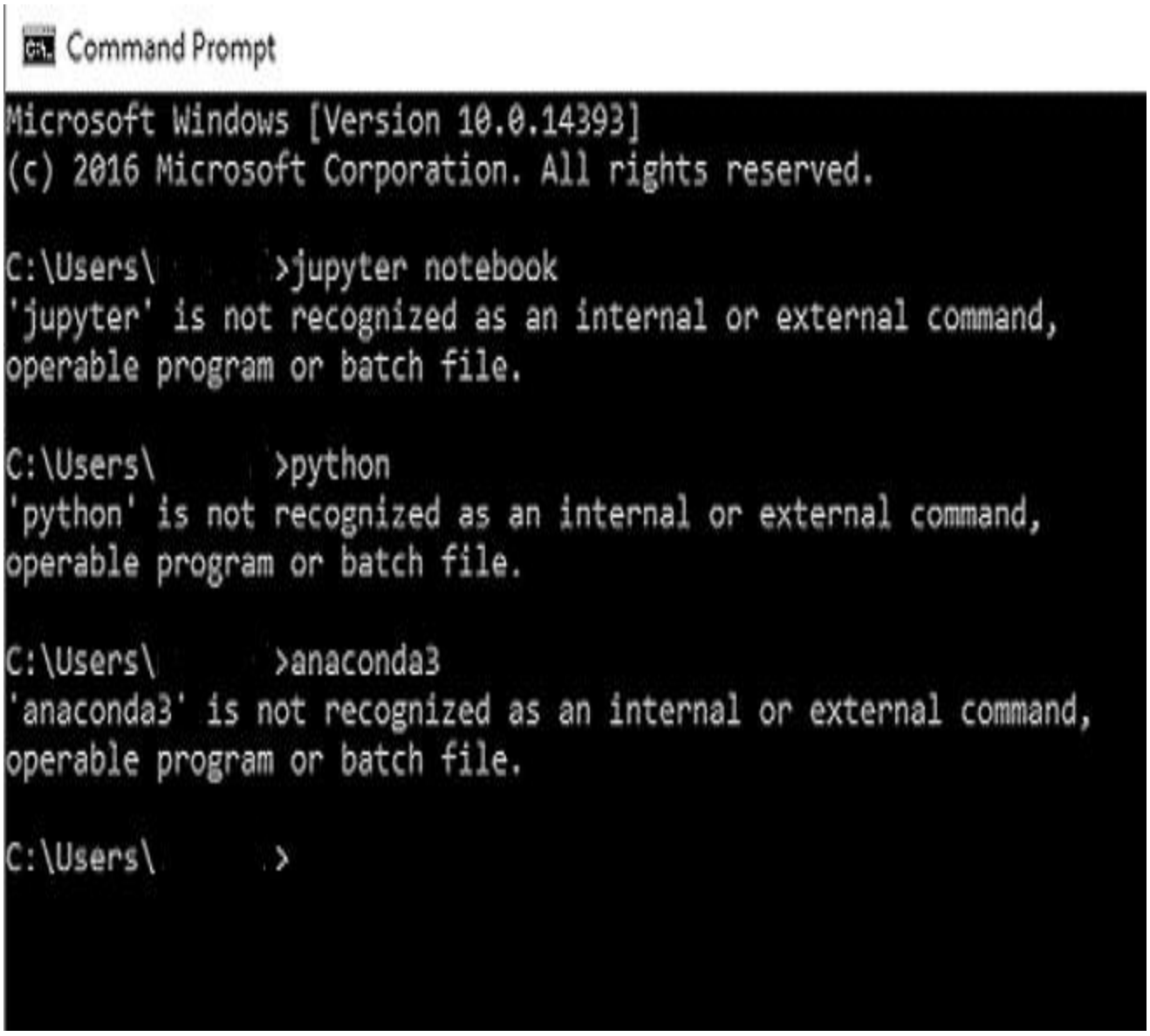
```
> conda update --all --yes
```

Start Jupyter Notebook

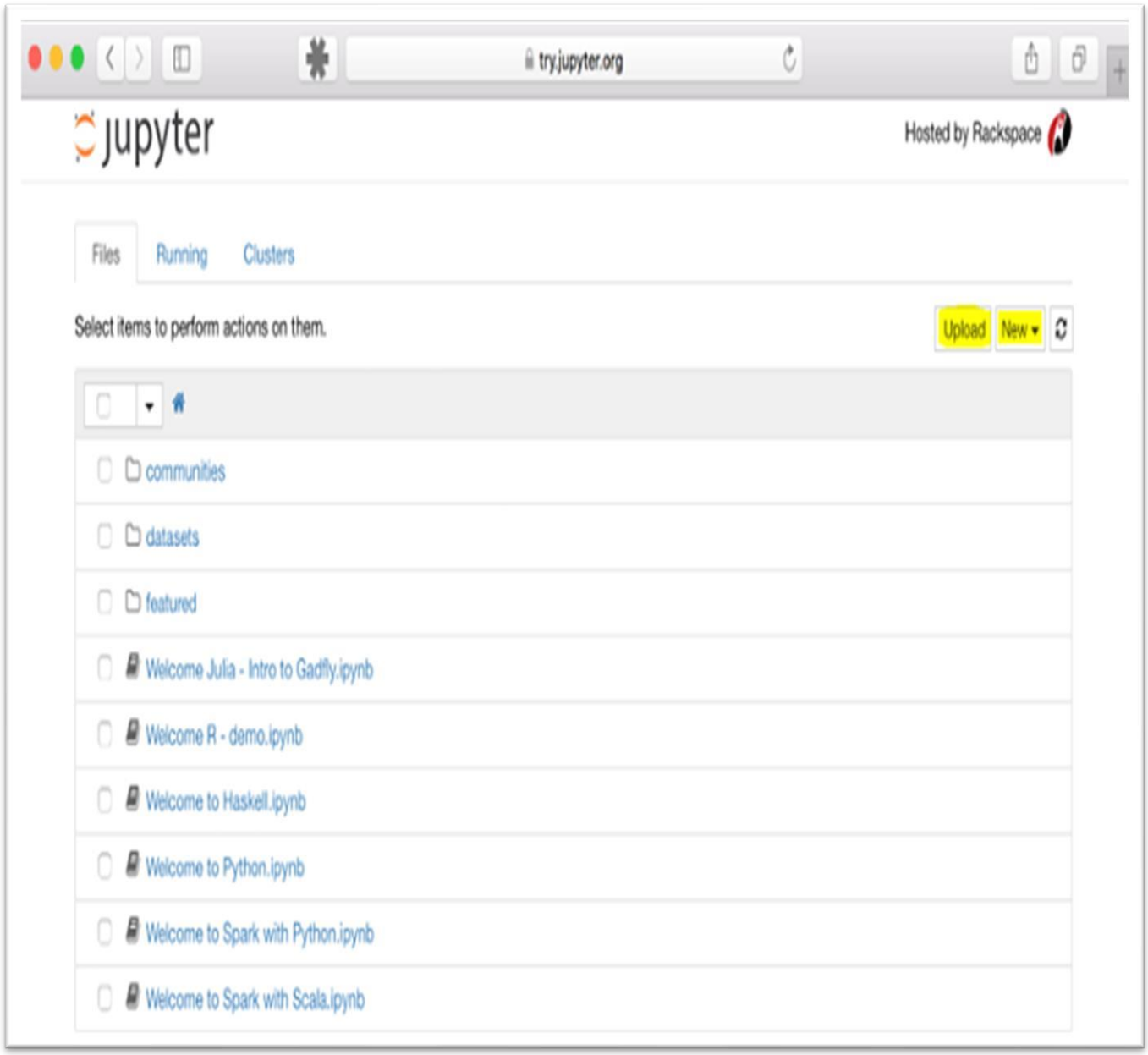
1.open anaconda navigator and the screen which is similar to below appears.

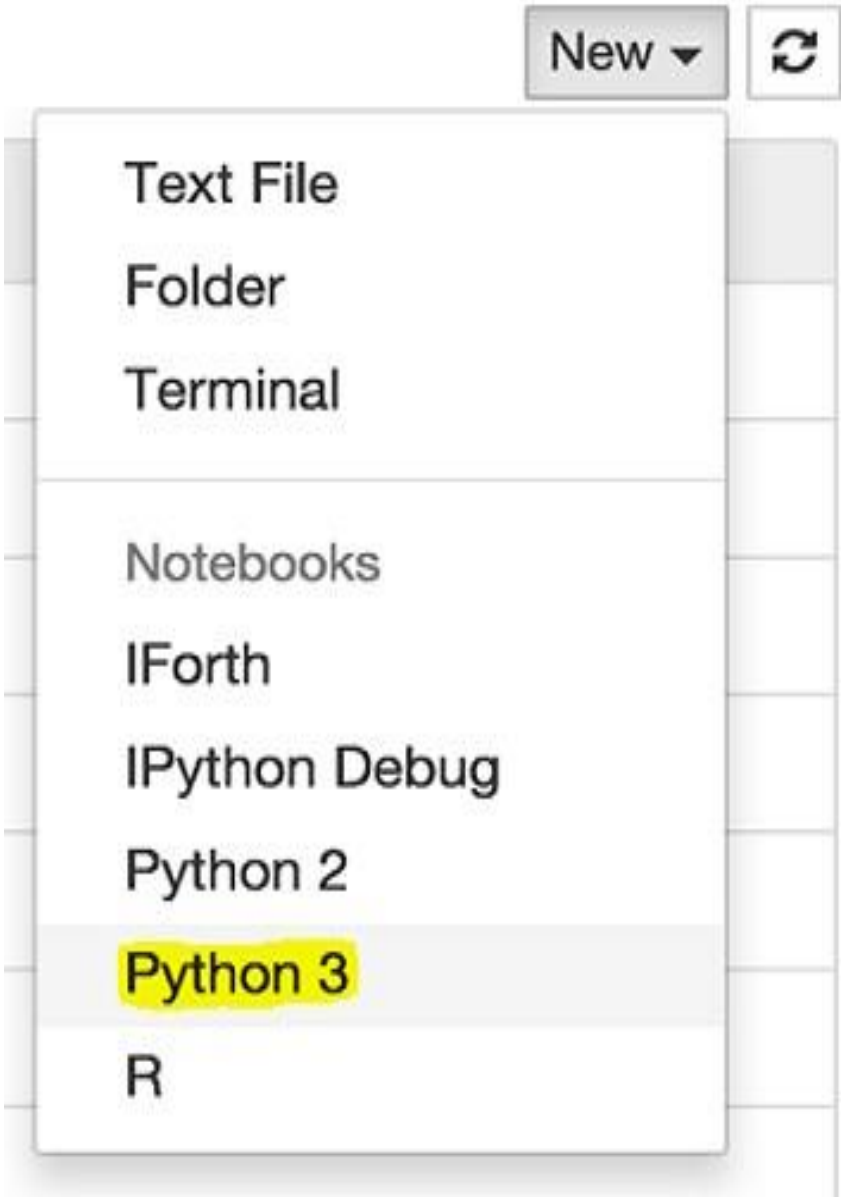


open anaconda prompt to open jupyter note book


Command Prompt
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.
C:\Users\ >jupyter notebook
'jupyter' is not recognized as an internal or external command,
operable program or batch file.
C:\Users\ >python
'python' is not recognized as an internal or external command,
operable program or batch file.
C:\Users\ >anaconda3
'anaconda3' is not recognized as an internal or external command,
operable program or batch file.
C:\Users\ >


localhost:8888/tree
jupyter
Logout
Files
Running
Clusters
Select items to perform actions on them.
Upload
New
Name
Last Modified
FreshStart
3 months ago
GitHub
2 months ago
My Tableau Repository
2 months ago
Python Scripts
3 months ago


try.jupyter.org
jupyter
Hosted by Rackspace
Files
Running
Clusters
Select items to perform actions on them.
Upload
New
communities
datasets
featured
Welcome Julia - Intro to Gadfly.ipynb
Welcome R - demo.ipynb
Welcome to Haskell.ipynb
Welcome to Python.ipynb
Welcome to Spark with Python.ipynb
Welcome to Spark with Scala.ipynb



Installing required packages

MODULE DESCRIPTION:

We present a system to predict stock prices.. It is based on long short term memory technique. We train the algorithm using the past stock prices (open,close,low,high) data of a certain company and we predit future stock prices of any company.

The programming language used here is Python. Machine learning problems are all solved using various algorithms which use python or java as the basic programming. One such algorithm we are using is the Lstm algorithm.

LSTM is one of the most simple and commonly used Machine Learning algorithms(Recurrent nueral network) .which is mostly used for prediction.It is powerfull and are known for retaining lomg term memory.

SOFTWARE DESCRIPTION:

- **Lstm implemented using python.**
- **Pakages used for pandas,numpy,matplotlib.**

Now we turn to the implementation of lstm model. There are several general steps you'll take when you're preparing your classification models:

classification models:

- Import packages, functions.
- Get the data information.
- Predict the values
- Plot the values

Sample code

```
import numpy as np import pandas as pd
import matplotlib.pyplot as plt import seaborn
as sns
data= pd.read_csv("prices-split-adjusted.csv") df = pd.DataFrame(data) df.head() df.describe() for i in
df.columns:
print(i,"\t-\t", df[i].isna().mean()*100)
def get_corelated_col(cor_dat, threshold):
# Cor_data to be column along which corelation to be measured
#Threshold be the value above which of corelation to considered   feature=[]   value=[]
for i ,index in enumerate(cor_dat.index):                if
abs(cor_dat[index]) > threshold:
feature.append(index)       value.append(cor_dat[index])
df = pd.DataFrame(data = value, index = feature, columns=['corr val ue'])   return df
top_corelated_values = get_corelated_col(cormap['close'], 0.60) top_corelated_values
df = df[top_corelated_values.index] df.head()
df.shape sns.pairplot(df) plt.tight_layout()
X = df.drop(['close'], axis=1) y = df['close']
 from sklearn.preprocessing import MinMaxScaler
 scaler = MinMaxScaler()
X = pd.DataFrame(scaler.fit_transform(X), columns=X.columns)
X.head()
#now lets split data in test train pairs
from sklearn.model_selection import train_test_split
 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size =
 0.2, shuffle=False)
 Acc = []
```

```
from sklearn.linear_model import LinearRegression
 # model training
 model_1 = LinearRegression() model_1.fit(X_train, y_train)
# prediction
y_pred_1 = model_1.predict(X_test) pred_df = pd.DataFrame({'Actual': y_test, 'Predicted':
y_pred_1}) pred_df.head()
from sklearn.metrics import r2_score
  print("Accuracy score of the predictions: {0}".format(r2_score(y_test
, y_pred_1)))
Acc.append(r2_score(y_test, y_pred_1))


plt.figure(figsize=(8,8)) plt.ylabel('Close Price', fontsize=16)
plt.plot(pred_df) plt.legend(['Actual Value', 'Predictions'])
plt.show()
 X_train = np.array(X_train).reshape(X_train.shape[0], X_train.shape[1 ], 1)
X_test = np.array(X_test).reshape(X_test.shape[0], X_test.shape[1], 1 )
from tensorflow.keras import Sequential,utils from tensorflow.keras.layers import Flatten, Dense,
Conv1D, MaxPool1D
, Dropout  def
reg():
    model = Sequential()
  model.add(Conv1D(32, kernel_size=(3,), padding='same', activation
='relu', input_shape = (X_train.shape[1],1)))
    model.add(Conv1D(64, kernel_size=(3,), padding='same', activation
='relu'))
    model.add(Conv1D(128, kernel_size=(5,), padding='same', activatio n='relu'))
   model.add(Flatten())
model.add(Dense(50,  activation='relu'))          model.add(Dense(20,  activation='relu'))
model.add(Dense(units = 1))
```

```
    model.compile(loss='mean_squared_error', optimizer='adam')
       return model
 # Model Training
 model_3 = reg() model_3.fit(X_train, y_train, epochs=100, validation_split=0.2)
# Prediction
 y_pred_3 = model_3.predict(X_test) pred_df = pd.DataFrame({'Actual': y_test, 'Predicted': y_pred_3.flatt
en()}) pred_df.head()
 # Measure the Accuracy Score
 from sklearn.metrics import r2_score
  print("Accuracy score of the predictions: {0}".format(r2_score(y_test
, y_pred_3)))
Acc.append(r2_score(y_test, y_pred_3)) plt.figure(figsize=(8,8))
plt.ylabel('Close Price', fontsize=16) plt.plot(pred_df)
plt.legend(['Actual Value', 'Predictions']) plt.show()
```

Now converting data in a time series data and applying some more models.

```
 close = df.reset_index()['close'] close.head() plt.plot(close) plt.show() time_step = 30
X, y = [], []
 for i in range(len(close)-time_step-1):
X.append(close[i:(i+time_step)])
    y.append(close[(i+time_step)])
 X = np.array(X) y = np.array(y) X[:5] y[:5]
 from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler() X = scaler.fit_transform(X) pd.DataFrame(X).head()
```

Prediction

#now lets split data in test train pairs

from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size =
0.2, shuffle=False)
Acc = []

LSTM:

X_train_ = X_train.reshape(X_train.shape[0],X_train.shape[1],1) X_test_ =
X_test.reshape(X_test.shape[0],X_test.shape[1],1) from tensorflow.keras.layers
import LSTM
def Reg():
  model = Sequential()
    model.add(LSTM(70, return_sequences=True, input_shape=(30,1)))
model.add(LSTM(70, return_sequences=True)) model.add(LSTM(70))
model.add(Dense(1))
    model.compile(loss='mean_squared_error', optimizer='adam')
    return model

# Model Training
model_1 = reg() model_1.fit(X_train_, y_train, epochs=100, validation_split=0.2)
# Prediction
y_pred_1 = model_1.predict(X_test_) pred_df = pd.DataFrame({'Actual': y_test, 'Predicted':
y_pred_1.flatten()}) pred_df.head()

# Measure the Accuracy Score
from sklearn.metrics import r2_score

```
print("Accuracy score of the predictions: {0}".format(r2_score(y_test, y_pred_1 )))
Acc.append(r2_score(y_test, y_pred_1))
plt.figure(figsize=(8,8)) plt.ylabel('Close Price',
fontsize=16) plt.plot(pred_df) plt.legend(['Actual Value',
'Predictions']) plt.show()
```

CNN:

```
# Model Training

model_3 = reg()
model_3.fit(X_train_, y_train, epochs=100, validation_split=0.2) # Prediction

y_pred_3 = model_3.predict(X_test_) pred_df = pd.DataFrame({'Actual':
 y_test, 'Predicted': y_pred_3.flatt en()}) pred_df.head()

# Measure the Accuracy Score

from sklearn.metrics import r2_score

print("Accuracy score of the predictions: {0}".format(r2_score(y_test
, y_pred_3)))
Acc.append(r2_score(y_test, y_pred_3))
plt.figure(figsize=(8,8)) plt.ylabel('Close Price', fontsize=16)
plt.plot(pred_df) plt.legend(['Actual Value', 'Predictions'])
plt.show()
```

6.SYSTEM TETING

The purpose of testing is to get errors. Testing is that the process of trying to get every conceivable fault or weakness during a work product. It provides how to see the functionality of components, sub assemblies, assemblies and/or a finished product it's the method of exercising software with the intent of ensuring that the software meets its requirements and user expectations and doesn't fail in an unacceptable manner.

There are various sorts of test. Each test type addresses a selected testing requirement. The various types of testing that follows are listed as below.

UNIT TESTING

Unit testing involves the planning of test cases that validate that the interior program logic is functioning properly, which program inputs produce valid outputs. All decision branches and internal code flow should be validated. it's the testing of individual software units of the appliance.

It is done after the completion of a private unit before integration. this is often a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a selected business process, application, and/or system configuration.

Unit tests make sure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

INTEGRATION TESTING

Integration tests are designed to check integrated software components to work out if they really run together program. Testing is event driven and is more concerned with the essential outcome of screens or fields.

Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the mixture of components is correct and consistent. Integration testing is specifically aimed toward exposing the issues that arise from the mixture of components.

VALIDATION TESTING

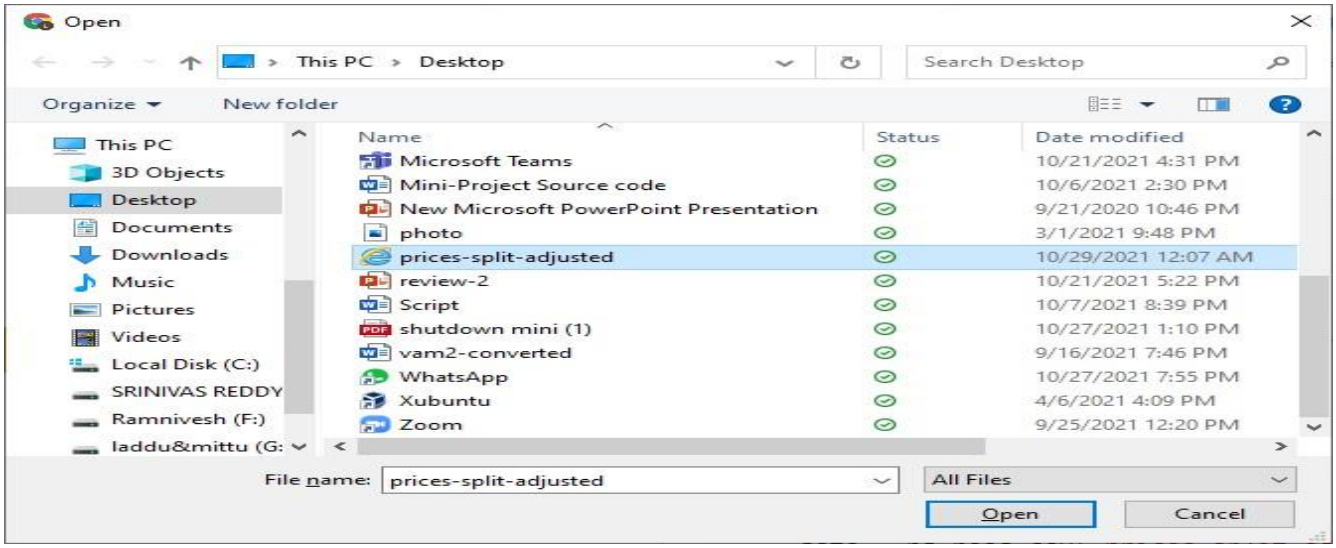
Validation testing is that the process of ensuring if the tested and developed software satisfies the client /user needs. The business requirement logic or scenarios need to be tested intimately . All the critical functionalities of an application must be tested here.

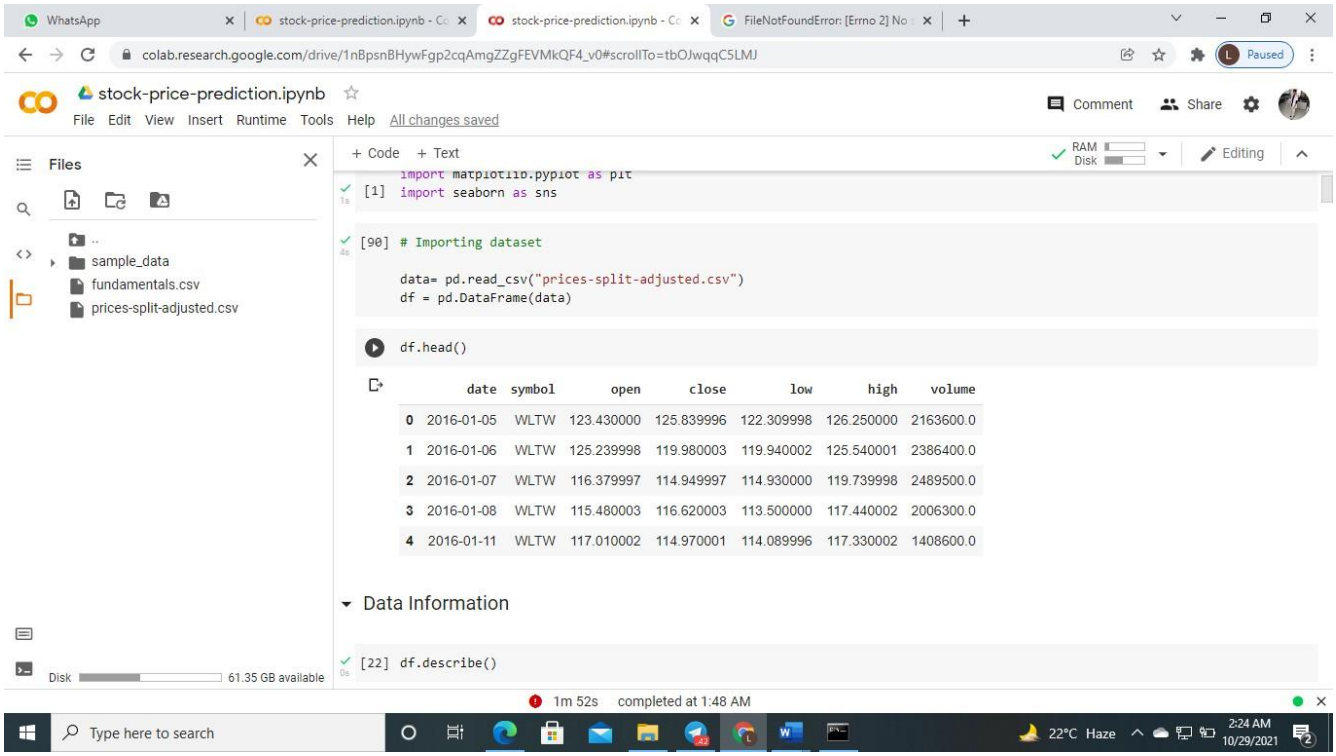
As a tester, it's always important to understand the way to verify the business logic or scenarios that are given to you. One such method that helps intimately evaluation of the functionalities is that the Validation Process

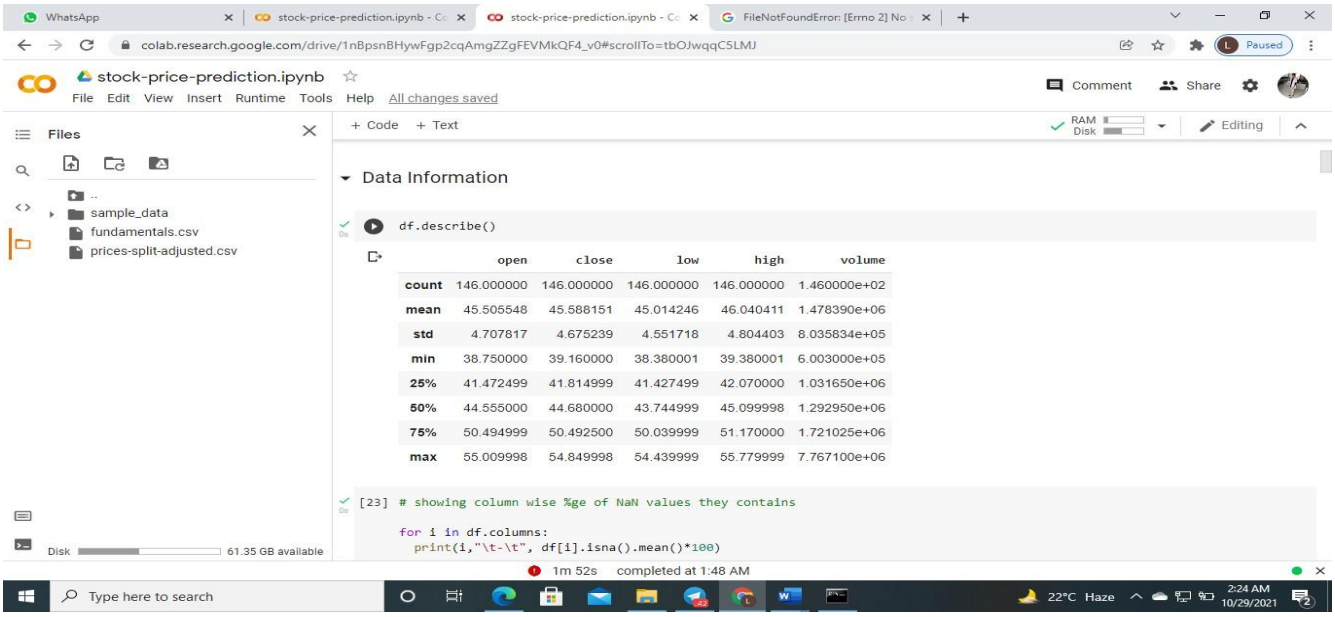
System testing of software or hardware is testing conducted on an entire , integrated system to guage the system's compliance with its specified requirements.
System testing falls within the scope of recorder testing, and intrinsically , should require no knowledge of the inner design of the code or logic.

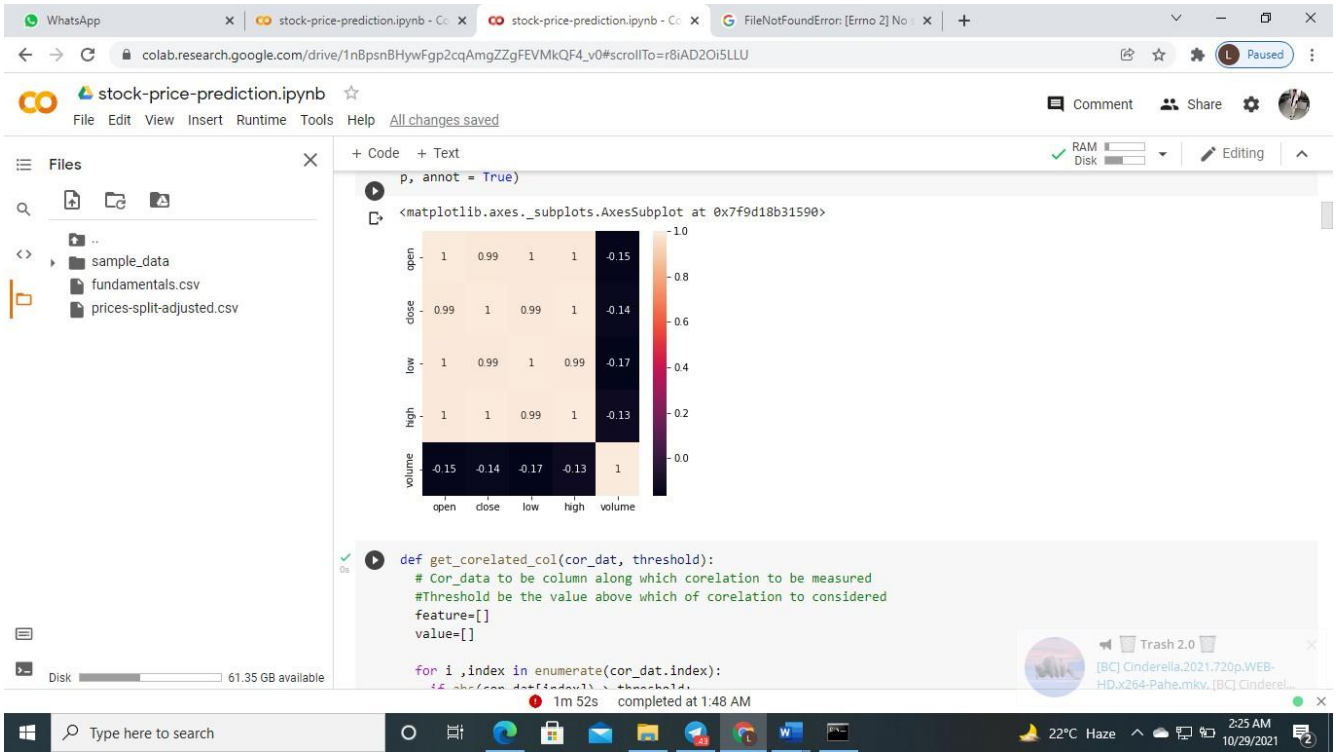
As a rule, system testing takes, as its input, all of the "integrated" software components that have successfully passed integration testing and also the software itself integrated with any applicable hardware system(s).
System testing may be a more limited sort of testing; it seeks to detect defects both within the "inter-assemblages" and also within the system as an entire . System testing is performed on the whole system within the context of a Functional Requirement Specification(s) (FRS) and/or a System Requirement Specification (SRS).
System testing tests not only the planning , but also the behavior and even the believed expectations of the customer. it's also intended to check up to and beyond the bounds defined within the software/hardware requirements specification(s)

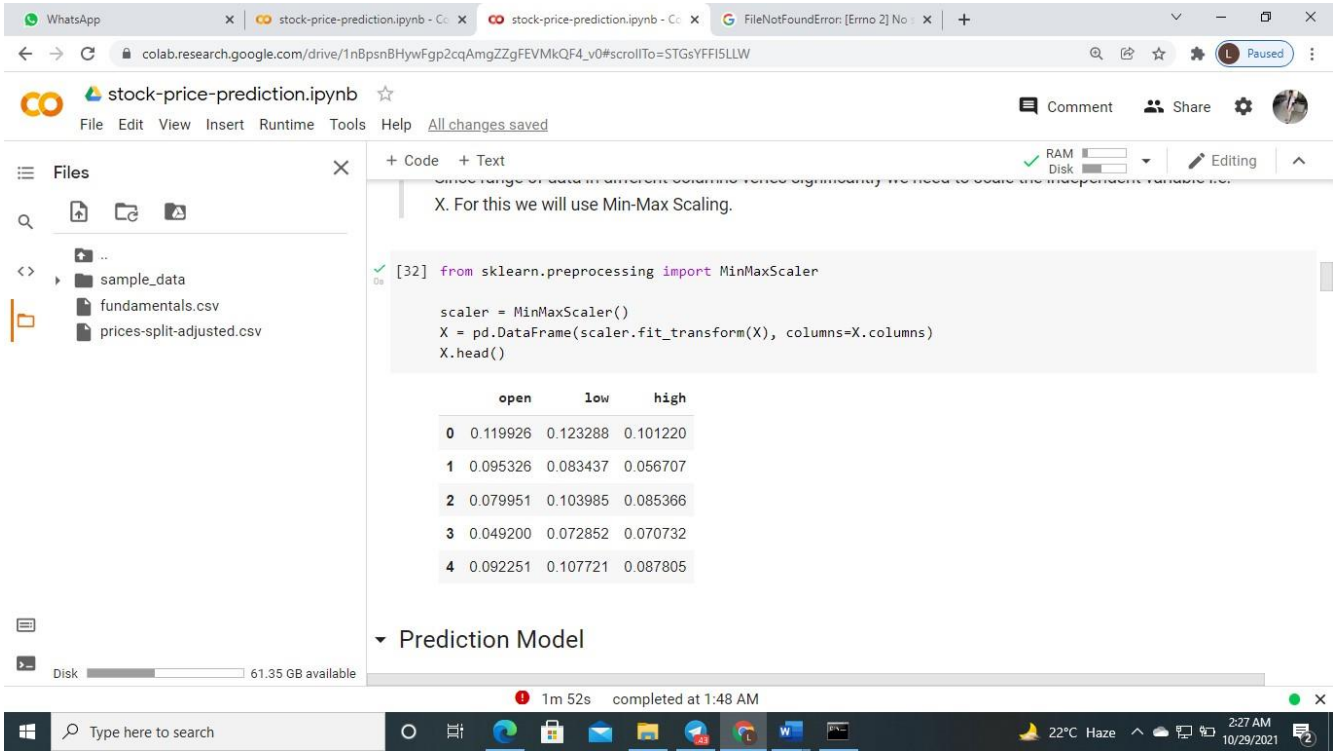
7.RESULTS AND SCREENSHOTS

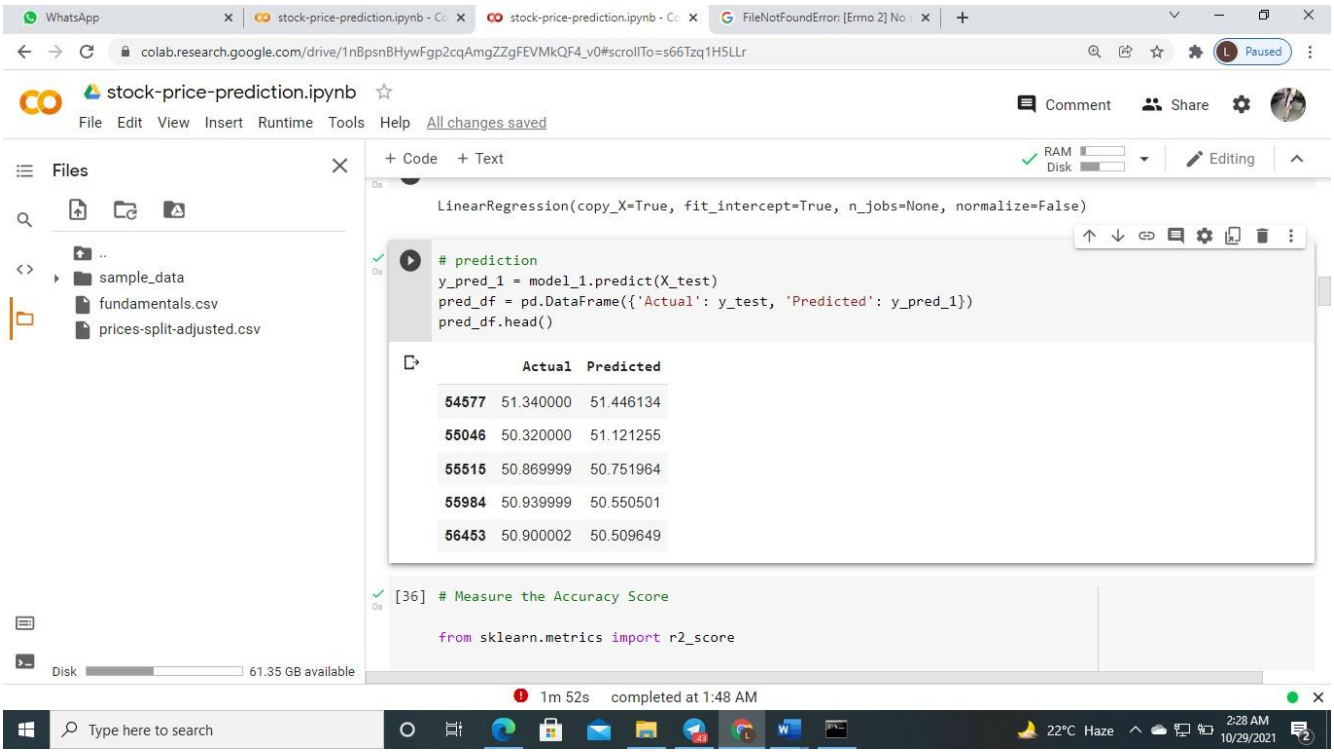



stock-price-prediction.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment Share
Files
sample_data
fundamentals.csv
prices-split-adjusted.csv
+ Code + Text
RAM Disk
Editing
[1] import seaborn as sns
[90] # Importing dataset
data= pd.read_csv("prices-split-adjusted.csv")
df = pd.DataFrame(data)
df.head()
date symbol open close low high volume
0 2016-01-05 WLTW 123.430000 125.839996 122.309998 126.250000 2163600.0
1 2016-01-06 WLTW 125.239998 119.980003 119.940002 125.540001 2386400.0
2 2016-01-07 WLTW 116.379997 114.949997 114.930000 119.739998 2489500.0
3 2016-01-08 WLTW 115.480003 116.620003 113.500000 117.440002 2006300.0
4 2016-01-11 WLTW 117.010002 114.970001 114.089996 117.330002 1408600.0
Data Information
[22] df.describe()
Disk 61.35 GB available
1m 52s completed at 1:48 AM
Type here to search
22°C Haze
2:24 AM
10/29/2021


stock-price-prediction.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment Share
Files
sample_data
fundamentals.csv
prices-split-adjusted.csv
+ Code + Text
RAM Disk
Editing
Data Information
df.describe()
open close low high volume
count 146.000000 146.000000 146.000000 146.000000 1.460000e+02
mean 45.505548 45.588151 45.014246 46.040411 1.478390e+06
std 4.707817 4.675239 4.551718 4.804403 8.035834e+05
min 38.750000 39.160000 38.380001 39.380001 6.003000e+05
25% 41.472499 41.814999 41.427499 42.070000 1.031650e+06
50% 44.555000 44.680000 43.744999 45.099998 1.292950e+06
75% 50.494999 50.492500 50.039999 51.170000 1.721025e+06
max 55.009998 54.849998 54.439999 55.779999 7.767100e+06
[23] # showing column wise %ge of NaN values they contains
for i in df.columns:
print(i,"\t-\t", df[i].isna().mean()*100)
Disk 61.35 GB available
1m 52s completed at 1:48 AM
Type here to search
22°C Haze
2:24 AM
10/29/2021

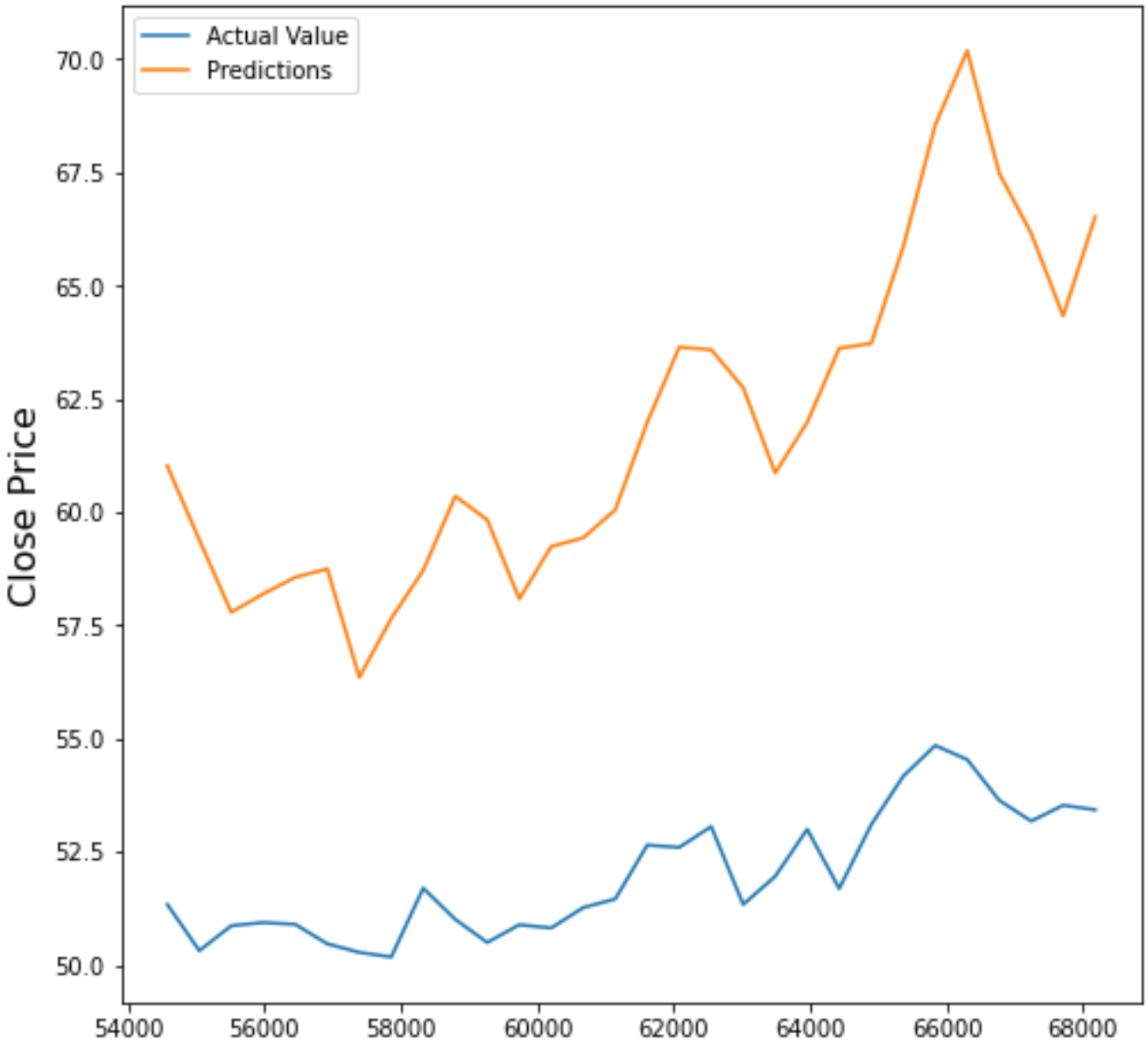

stock-price-prediction.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment Share
+ Code + Text
Editing
Files
sample_data
fundamentals.csv
prices-split-adjusted.csv
p, annot = True)
<matplotlib.axes._subplots.AxesSubplot at 0x7f9d18b31590>
open close low high volume
1 0.99 1 1 -0.15
0.99 1 0.99 1 -0.14
1 0.99 1 0.99 -0.17
1 1 0.99 1 -0.13
-0.15 -0.14 -0.17 -0.13 1
1.0 0.8 0.6 0.4 0.2 0.0
def get_corelated_col(cor_dat, threshold):
# Cor_data to be column along which corelation to be measured
#Threshold be the value above which of corelation to considered
feature=[]
value=[]
for i ,index in enumerate(cor_dat.index):
Disk 61.35 GB available
1m 52s completed at 1:48 AM
Type here to search
22°C Haze
2:25 AM
10/29/2021

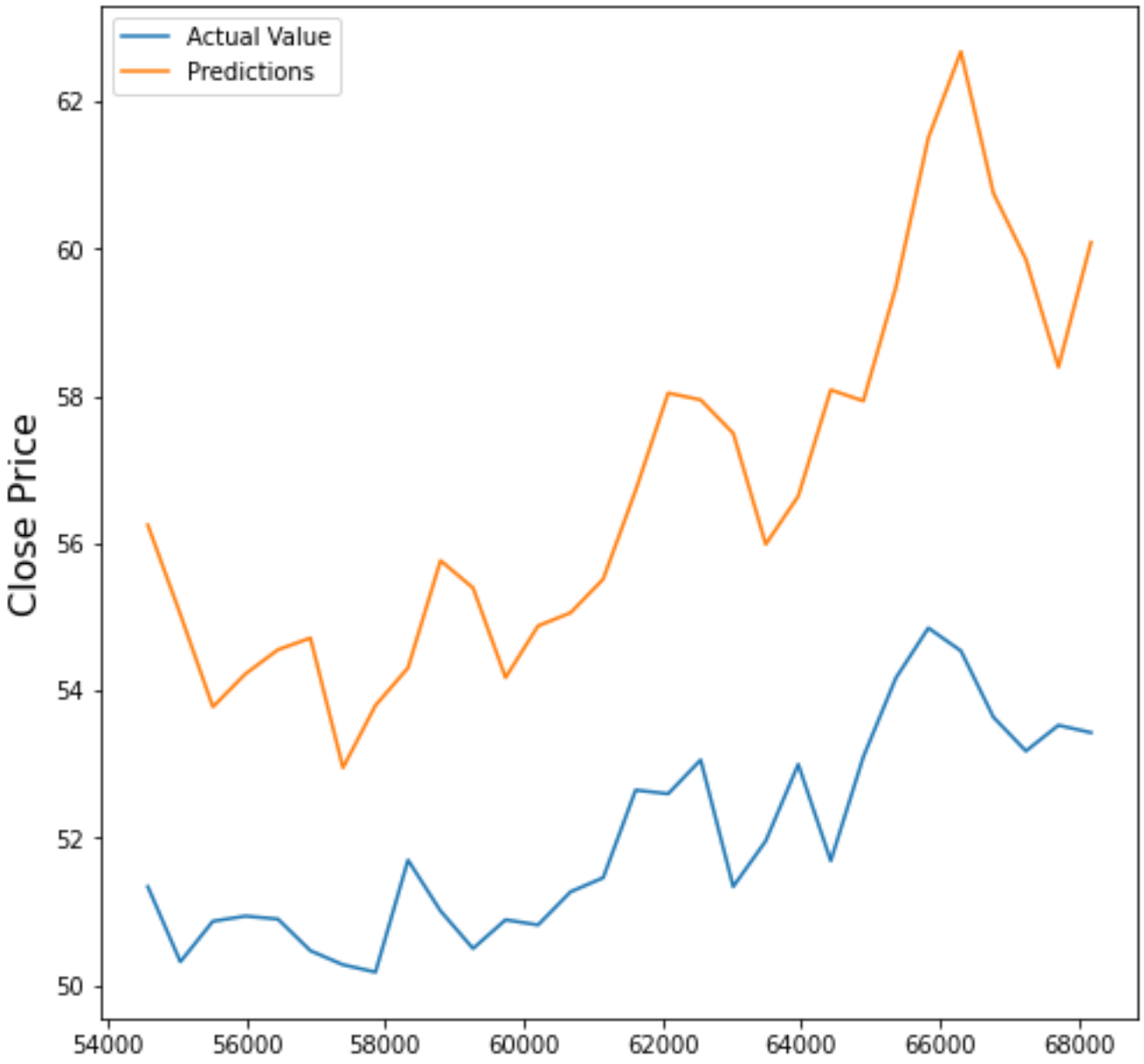

stock-price-prediction.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment Share
+ Code + Text
Editing
Files
sample_data
fundamentals.csv
prices-split-adjusted.csv
X. For this we will use Min-Max Scaling.
[32] from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()
X = pd.DataFrame(scaler.fit_transform(X), columns=X.columns)
X.head()
open low high
0 0.119926 0.123288 0.101220
1 0.095326 0.083437 0.056707
2 0.079951 0.103985 0.085366
3 0.049200 0.072852 0.070732
4 0.092251 0.107721 0.087805
Prediction Model
Disk 61.35 GB available
1m 52s completed at 1:48 AM
Type here to search
22°C Haze
2:27 AM
10/29/2021


stock-price-prediction.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
Files
+ Code + Text
RAM
Disk
Editing
sample_data
fundamentals.csv
prices-split-adjusted.csv
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=False)
# prediction
y_pred_1 = model_1.predict(X_test)
pred_df = pd.DataFrame({'Actual': y_test, 'Predicted': y_pred_1})
pred_df.head()
Actual Predicted
54577 51.340000 51.446134
55046 50.320000 51.121255
55515 50.869999 50.751964
55984 50.939999 50.550501
56453 50.900002 50.509649
[36] # Measure the Accuracy Score
from sklearn.metrics import r2_score
Disk 61.35 GB available
1m 52s completed at 1:48 AM
Type here to search
22°C Haze
2:28 AM
10/29/2021

LINEAR REGRESSION:

LSTM:


Actual Value
Predictions
Close Price
70.0
67.5
65.0
62.5
60.0
57.5
55.0
52.5
50.0
54000
56000
58000
60000
62000
64000
66000
68000


Actual Value
Predictions
Close Price
62
60
58
56
54
52
50
54000
56000
58000
60000
62000
64000
66000
68000

CONCLUSION

To conclude stock is an unpredictable mechanism which follows the segments of chain and the dependencies of the same are unpredictable. It is defined to be an curve which keeps on changing and turning the price from low to high and vice-versa. As the integration of the same is higher with other dependencies so leaving one dependencies compromises the level of accuracy. Accuracy is not the term used over in stock as the actual prediction is not possible for any fiscal days it keeps on changing and turning the tables day and night. Having higher component assets and the dependencies makes it more feasible and flexible in nature causing it even harder to predict. The approx value are taken into consideration and the hit or profit or the gain rate is calculated for the same.

In the project various high level machine learning algorithms are implemented and integrated and the output is generated from the same making a user visible with the outputs in the form of graph which makes it easier for them to see and interpret what's the scenario and they can decide on the same to invest and get the benefit out of it, The proposed software takes the raw set of data from the dataset or the .csv file and process it. The cleaning and cleansing of data is done and then further processed to gain the effective outcomes. After the computational mean the output is displayed in the screen in the form of graph

BIBLIOGRAPHY

- Reminiscences of a Stock Operator
  1994 Edition - Edwin Lefevre

- Kimoto T, Asakawa K, Yoda M and Takeoka M (1990), Stock market prediction system with modular neural networks, Proc. International
  Joint Conference on Neural Networks,

- Abraham A, Nath B and Mahanti P K (2001), Hybrid Intelligent Systems for Stock Market Analysis,Proceedings of the International Conference on Computational Science.

- Hannon, "5 Rules For Prediciting Stock Market Trends StockTrader.com",

GITHUB LINK :

https://github.com/vijayapandiri/Mini-Project/upload/main