

# Assignment 2

---

**Sillah Babar 19i-2029**

**Aliza Adnan 19i-1654**

**Section: AI-J**

**MLOps**

---

**GITHUB URL:** <https://github.com/Aliza-adnan1/MLOps-Assignment-2.git>

---

## Finding a datastream and creating a model:

In this assignment we chose to use the stock market shares for apple to run our simple linear model on. The model takes into account the latest data as uploaded on the api and trains the model to predict the closed value of the share once an open value has been provided. With this investors can get a sense of how their shares' worth will change over the upcoming months by looking at the previous trends.

### Call to get data from api

```
import pandas_datareader as pdr
import pandas as pd
import os
import matplotlib.pyplot as plt
import numpy as np
import pickle
key="0d7d08b1ef617c881687c625f95bd9b9af0a12e8"
df= pdr.get_data_tingo("AAPL",start='2021-01-25', end='2021-07-29', api_key=key)

print(df.head(5))

X = df.iloc[:, 3:4].values
Y = df.iloc[:, 0].values
print(X)
print(Y)
```

### Creating the model

```
17
18
19 # Splitting the dataset into the Training set and Test set
20 from sklearn.model_selection import train_test_split
21 X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size = 0.2, random_state = 1)
22
23 from sklearn.preprocessing import StandardScaler
24 sc = StandardScaler()
25 X_train = sc.fit_transform(X_train[:, 0].reshape(-1, 1))
26 X_test = sc.transform(X_test[:, 0].reshape(-1, 1))
27
28 from sklearn.linear_model import LinearRegression
29 regressor = LinearRegression()
30 regressor.fit(X_train, y_train)
31
32 y_pred = regressor.predict(X_test)
33
34 with open("model.pkl", "wb") as f:
35     pickle.dump(regressor, f)
36
37 print(len(X_train))
38 print(len(y_train))
39 print(y_pred)
40 # Visualising the Training set results
41 plt.scatter(X_train, y_train, color = 'red')
42 plt.plot(X_train, regressor.predict(X_train), color = 'blue')
43 plt.title('Salary vs Experience (Training set)')
44 plt.xlabel('Years of Experience')
45 plt.ylabel('Salary')
46 plt.show()
47
```

## Creating a flask file and other necessary files:

Our flask app takes an open value and tells the predicted value alongside the metrics for linear regression that are the y intercept and the coefficient.

```
del > inter.py > ...
1 from flask import Flask, render_template, request
2 import numpy
3 import pickle
4 from sklearn import preprocessing, svm
5
6 app = Flask(__name__) #creating the Flask class object
7
8 def getResults(num):
9     # save the model to disk
10    num1=int(num)
11    arr=numpy.array(num,dtype=float)
12
13    num2=arr.reshape(1,-1)
14    print(num2)
15
16    with open("C:\\Users\\sillah\\Documents\\Tweet\\model\\model.pkl", "rb") as f:
17        loaded_model = pickle.load(f)
18        value= loaded_model.predict(num2)
19        print(value[0])
20        return str(value[0])
21
22 @app.route('/', methods =["GET", "POST"])
23 def home():
24     if request.method == "POST":
25         # getting input with name = fname in HTML form
26         years = request.form.get("open")
27         predictedSal=getResults(years)
28         return "Predicted closed value for apple share is: "+predictedSal
29     return render_template("home.html")
30
31 if __name__ == '__main__':
32     app.run(debug = True)
```

## Building a container on local machine through docker:

```
C:\Users\aliza\Downloads\model>docker build -t sampleww .
[+] Building 1141.4s (11/11) FINISHED
=> [internal] load build definition from Dockerfile 1.0s
=> => transferring dockerfile: 32B 0.1s
=> [internal] load .dockerignore 1.4s
=> => transferring context: 2B 0.0s
=> [internal] load metadata for docker.io/library/python:3.10 3.8s
=> [1/6] FROM docker.io/library/python:3.10@sha256:96f52897a41344ac2c11a4a058403d50478027c7a20867ef9b9be7aafb3f 0.0s
=> [internal] load build context 0.7s
=> => transferring context: 220B 0.1s
=> CACHED [2/6] RUN pip install --upgrade pip 0.0s
=> CACHED [3/6] WORKDIR /model 0.0s
=> CACHED [4/6] COPY . . 0.0s
=> [5/6] RUN pip install -r requirements.txt 1105.8s
=> [6/6] RUN python stock.py 9.4s
=> exporting to image 18.0s
=> => exporting layers 17.2s
=> => writing image sha256:18ddd379d29941699ebab14b88f06f77c25d0006a11191c6987ca598673cf994 0.1s
=> => naming to docker.io/library/sampleww 0.1s
```

## Running the image and testing the app:

```
C:\Users\aliza\Downloads\model>docker run --name mlops_app -p 8000:5000 sampleww
* Serving Flask app 'infer'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://172.17.0.2:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 108-043-654
172.17.0.1 - - [22/Mar/2023 17:31:38] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [22/Mar/2023 17:31:38] "GET /favicon.ico HTTP/1.1" 404 -
172.17.0.1 - - [22/Mar/2023 17:31:41] "POST / HTTP/1.1" 200 -
```







### Containers

[Give feedback](#)

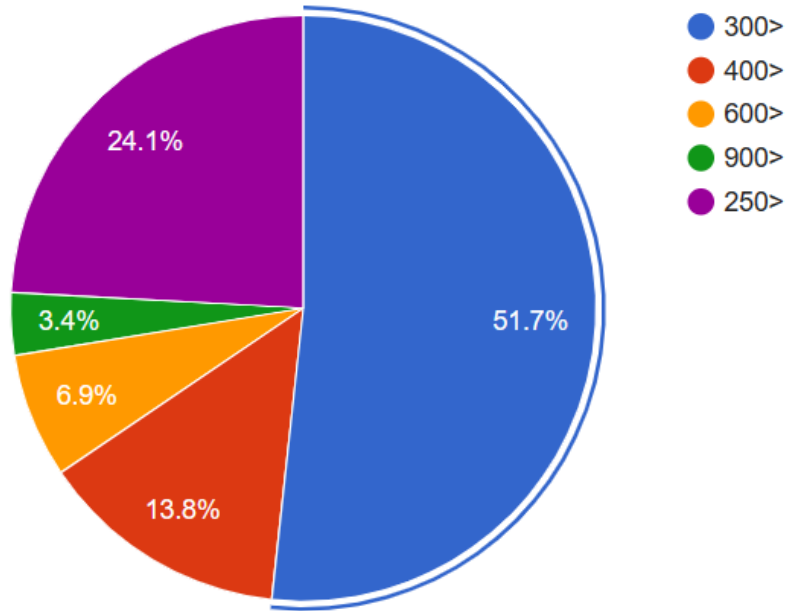
A container packages up code and its dependencies so the application runs quickly and reliably from one computing environment to another. [Learn more](#)

☐ Only show running containers

Search

<input type="checkbox"/>	Name	Image	Status	Port(s)	Last started	Actions
<input type="checkbox"/>	 <b>mlops_app</b> c955b8403f78 	<a href="#">sampleww</a>	Running	<a href="#">8000:5000</a> 	1 second ago	  

### CLOSED PRICES OF APPLE SHARE OVER THE PAST YEAR \$



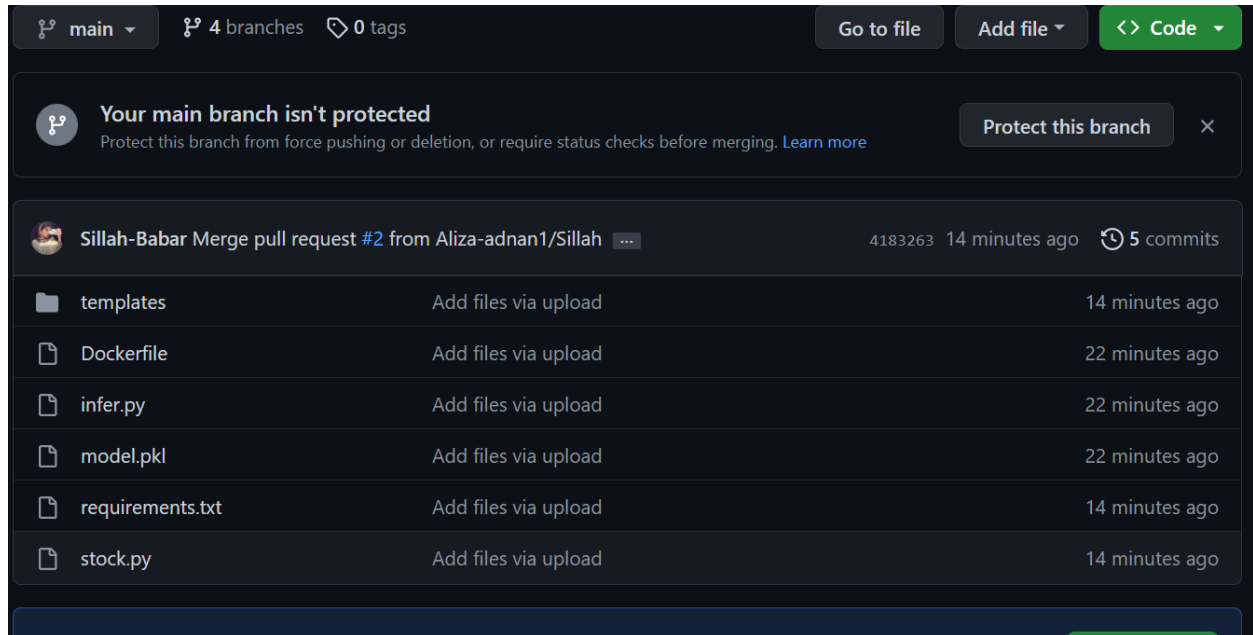
Enter an Open Value:

## Predicted Values

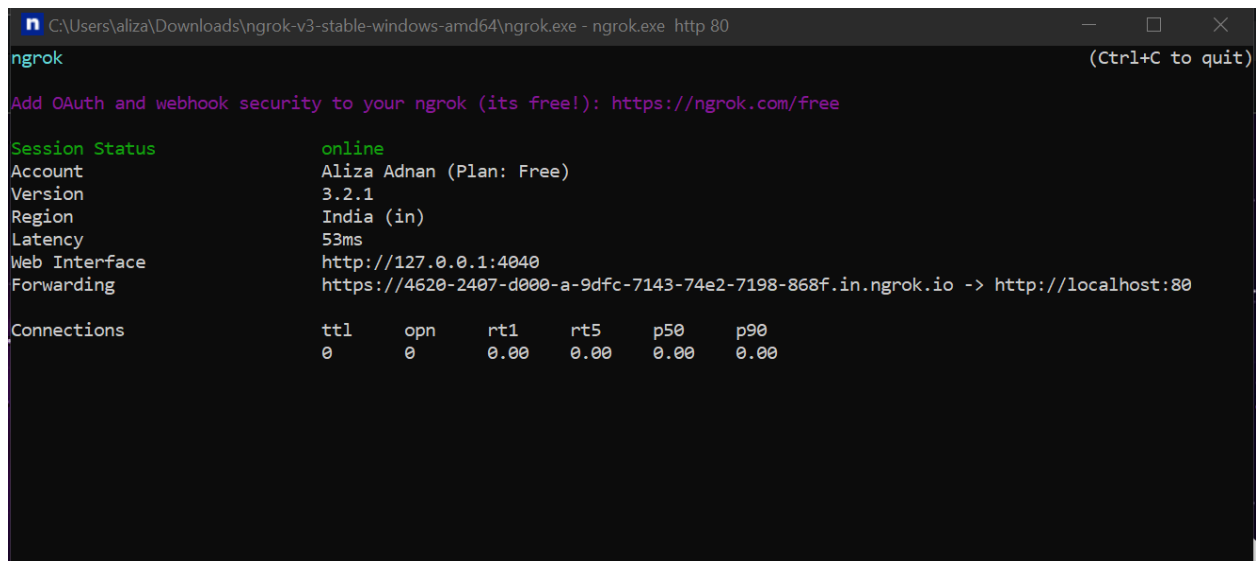
← → ↺ ⓘ 127.0.0.1:8000

Predicted closed value for apple share is: 209.37904190609697 and the intercept of model is: 131.28923076923076 and the accuracy is : -17168.66260358888

## Uploading the files on github:



## Connecting ngrok:



## Connecting jenkins through webhook:

Creating webhook on github:

## Webhooks

Add webhook

Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our [Webhooks Guide](#).

- <https://4620-2407-d000-a-9dfc-714...> (push)

Edit

Delete

## Creating and building the item on jenkins:

Inserting docker commands in the configuration settings:

Build Steps

≡

Execute Windows batch command ?

×

Command

See [the list of available environment variables](#)

```
docker rm --force flask_container
docker rmi flask_image
docker build -t flask_image .
docker create --name flask_container -p 8000:5000 flask_image
```

Advanced...

Add build step ▾

Building the item:

```

Started by user Aliza Adnan
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\.jenkins\workspace\MLOps Assignment2
[WS-CLEANUP] Deleting project workspace...
[WS-CLEANUP] Deferred wipeout is used...
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/Aliza-adnan1/MLOps-Assignment-2.git
  > C:\Program Files\Git\cmd\git.exe init C:\ProgramData\Jenkins\.jenkins\workspace\MLOps Assignment2 # timeout=10
Fetching upstream changes from https://github.com/Aliza-adnan1/MLOps-Assignment-2.git
  > C:\Program Files\Git\cmd\git.exe --version # timeout=10
  > git --version # 'git version 2.38.0.windows.1'
  > C:\Program Files\Git\cmd\git.exe fetch --tags --force --progress -- https://github.com/Aliza-adnan1/MLOps-Assignment-2.git +refs/heads/*:refs/remotes/origin/* # timeout=10
  > C:\Program Files\Git\cmd\git.exe config remote.origin.url https://github.com/Aliza-adnan1/MLOps-Assignment-2.git # timeout=10
  > C:\Program Files\Git\cmd\git.exe config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
  > C:\Program Files\Git\cmd\git.exe rev-parse "refs/remotes/origin/main^{commit}" # timeout=10
Checking out Revision 813a790c0d18c307a6132c0ae6aee819c296f120 (refs/remotes/origin/main)
  > C:\Program Files\Git\cmd\git.exe config core.sparsecheckout # timeout=10
  > C:\Program Files\Git\cmd\git.exe checkout -f 813a790c0d18c307a6132c0ae6aee819c296f120 # timeout=10
Commit message: "Merge pull request #3 from Aliza-adnan1/analytics"

```

Dashboard > MLOps Assignment2 > #1

```

#10 15.25 145.40012268 129.18999477 133.46835676 135.77132656 131.75105601
#10 15.25 134.88785969 121.69541634 128.94182992 120.4545921 126.0234113
#10 15.25 143.91113359]
#10 DONE 16.1s

#11 exporting to image
#11 sha256:e8c613e07b0b7ff33893b694f7759a10d42e180f2b4dc349fb57dc6b71dcab00
#11 exporting layers
#11 exporting layers 30.6s done
#11 writing image sha256:8f2de3325275874ced262efc5ca0ffdf91cb9dd057da647f5bacc270dd1dbd1
#11 writing image sha256:8f2de3325275874ced262efc5ca0ffdf91cb9dd057da647f5bacc270dd1dbd1 0.1s done
#11 naming to docker.io/library/flask_image 0.1s done
#11 DONE 31.2s

C:\ProgramData\Jenkins\.jenkins\workspace\MLOps Assignment2>docker create --name flask_container -p 8000:5000 flask_image
573b6be763cd87ceb2bfa7ec5d1116c73d380ea6173eae6f613e2c6c7d4884035

C:\ProgramData\Jenkins\.jenkins\workspace\MLOps Assignment2>exit 0
[WS-CLEANUP] Deleting project workspace...
[WS-CLEANUP] Deferred wipeout is used...
[WS-CLEANUP] done
Finished: SUCCESS

```