

# STOCK MARKET ANALYSIS PROJECT

## RELIANCE INDUSTRIES STOCK FORECAST

**GROUP 5**

**TEAM MEMBERS:-**

NANDYALA NARMADA

KAVYA SHRUTHI

DEEPTI GUPTA

NEERAJA AR

PRIYAL VINOD MOTGHARE

OMKAR



# PROBLEM STATEMENT

1. Predict the Reliance Industries Stock Price for the next 30 days.  
There are Open, High, Low, and Close prices that you need to obtain from the web for day starting from 2000 to 2022 for Reliance Industries stock.
2. Split the last year into a test set- to build a model to predict stock price.
3. Find short term, & long term trends.
4. Understand how it is impacted from external factors or any big external events.
5. Forecast for next 1 year.



# TABLE OF CONTENTS

1	IMPORTING REQUIRED LIBRARIES AND DATASETS	5	MODEL EVALUATION
2	EDA and VISUALIZATION	6	DEPLOYMENT
3	ADDITIONAL ACTIONS PERFORMED		
4	DATA PREPROCESSING		



# DATA SUMMARY

Date	Open	High	Low	Close	Adj Close	Volume
03-01-2000	36.747459	38.944569	36.747459	38.944569	29.684193	28802010
04-01-2000	39.981236	42.062302	38.882679	42.062302	32.060585	61320457
05-01-2000	39.710464	44.545658	39.710464	43.710136	33.316597	173426953
06-01-2000	44.715855	46.52615	44.715855	45.54364	34.714115	101355113
07-01-2000	45.644211	49.187443	45.334759	48.669109	37.096405	128426756
10-01-2000	49.110077	49.311222	47.237892	47.733013	36.382893	86714866
11-01-2000	47.647915	48.112095	43.919018	44.638493	34.0242	81074352
12-01-2000	44.715855	47.191475	43.655979	46.680878	35.580936	78264129
13-01-2000	47.346199	48.955353	47.052219	48.251347	36.777981	110363002
14-01-2000	47.887741	49.767666	47.887741	48.93988	37.302795	86996234
17-01-2000	49.241596	49.961071	47.508663	47.771698	36.412376	69343031



- Date : Date of trade
- Open: Opening Price of Stock
- High : Highest price of stock on that day
- Low : Lowest price of stock on that day
- Close: Close price adjusted for splits.
- Adj Close: Adjusted close price adjusted for splits and dividend and/or capital gain distributions.
- Volume: Volume of stock on that day



# IMPORTING THE REQUIRED LIBRARIES AND DATASETS

- We import the necessary libraries required for the Stock Market Analysis (Reliance Industries stock forecast) and we import the dataset

```
#import Libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

import warnings
warnings.filterwarnings('ignore')
```

```
data = pd.read_csv("reliance_stock.csv")
data.head()
```

	Date	Open	High	Low	Close	Adj Close	Volume
0	2000-01-03	36.747459	38.944569	36.747459	38.944569	29.684193	28802010.0



# EDA and VISUALIZATION

- **EDA (exploratory data analysis)**

- Here we are trying to remove the unwanted null values
- And there is no duplicated values

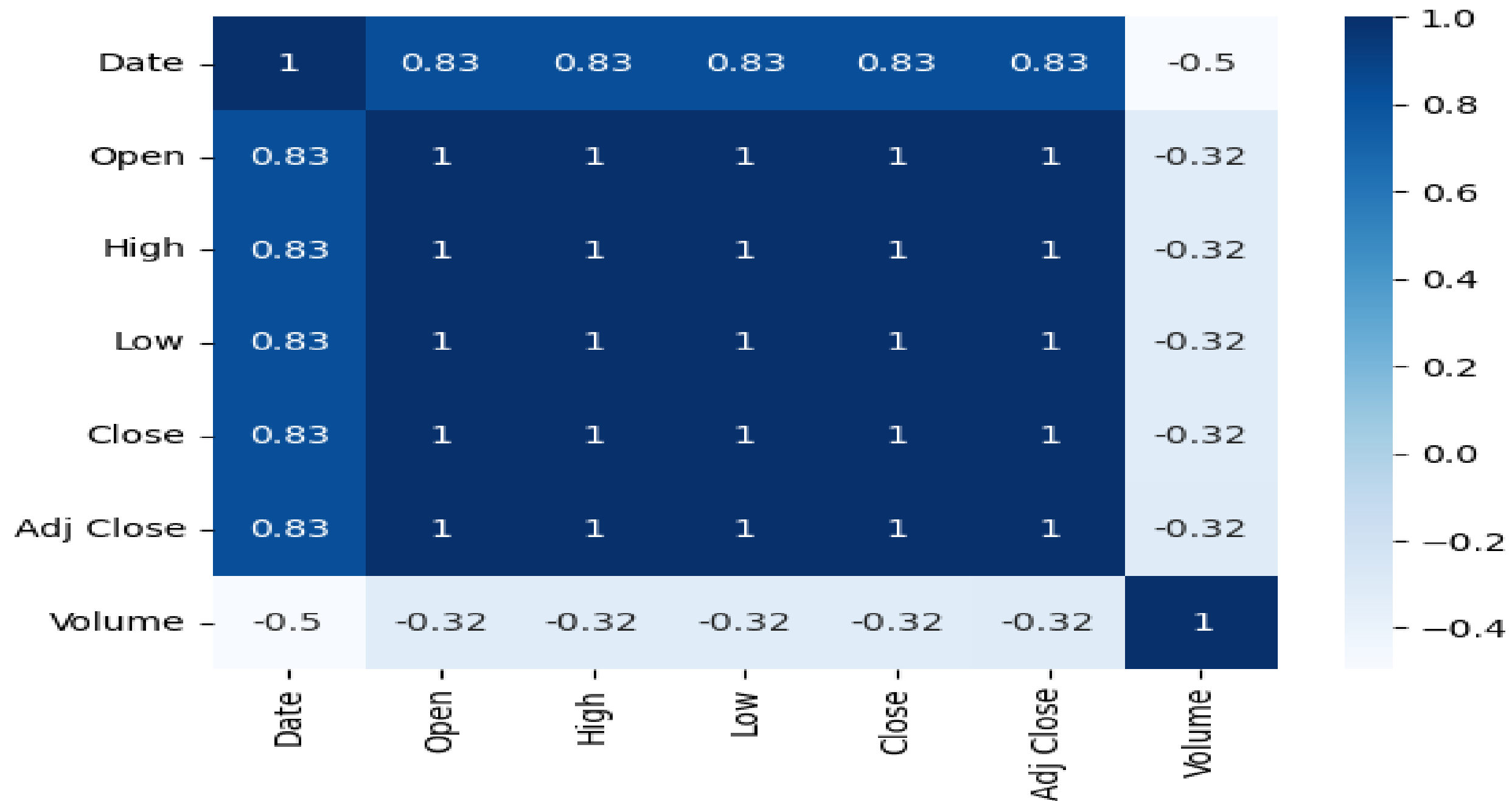
- **VISUALIZATION**

- In the visualization part we have done the histogram
- used boxplot to find outliers
- Used log transformation to reduce the outliers



# EDA and VISUALIZATION

- *Correlatin using heatmap*



There is negative 32% correlation between independent variables.



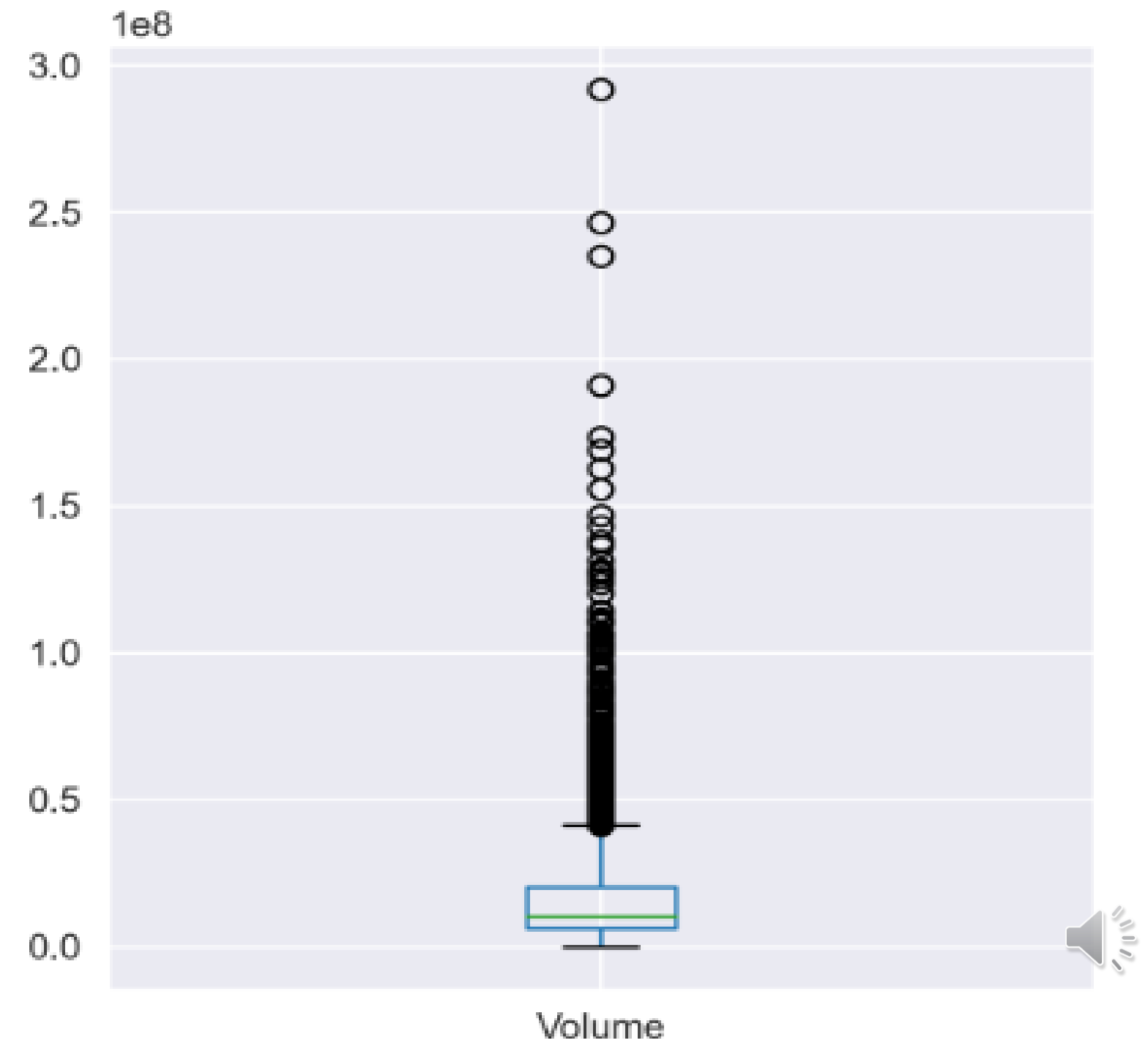
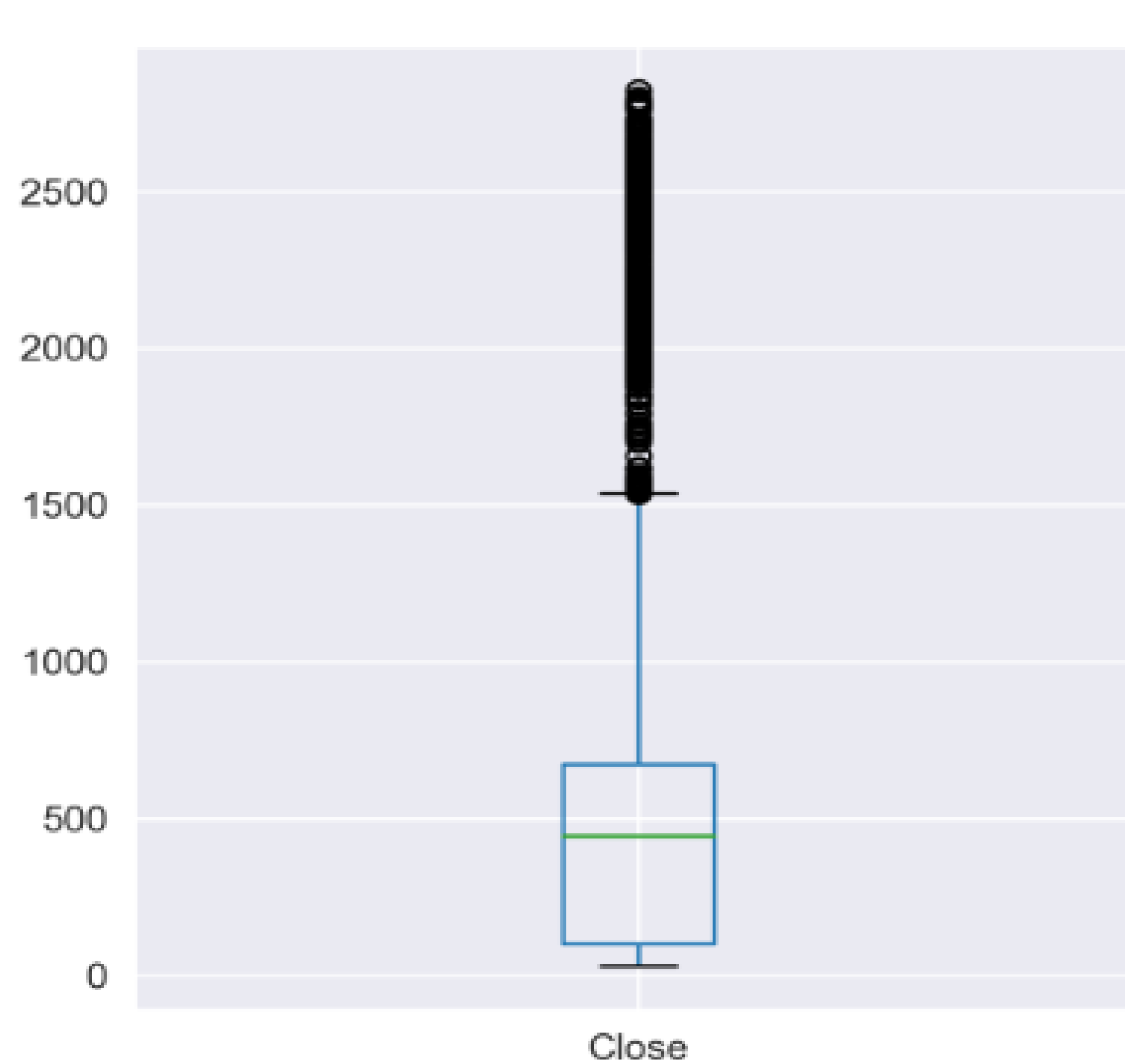


# EDA and VISUALIZATION

*Outliers using boxplot for 'Close' & 'Volume'*

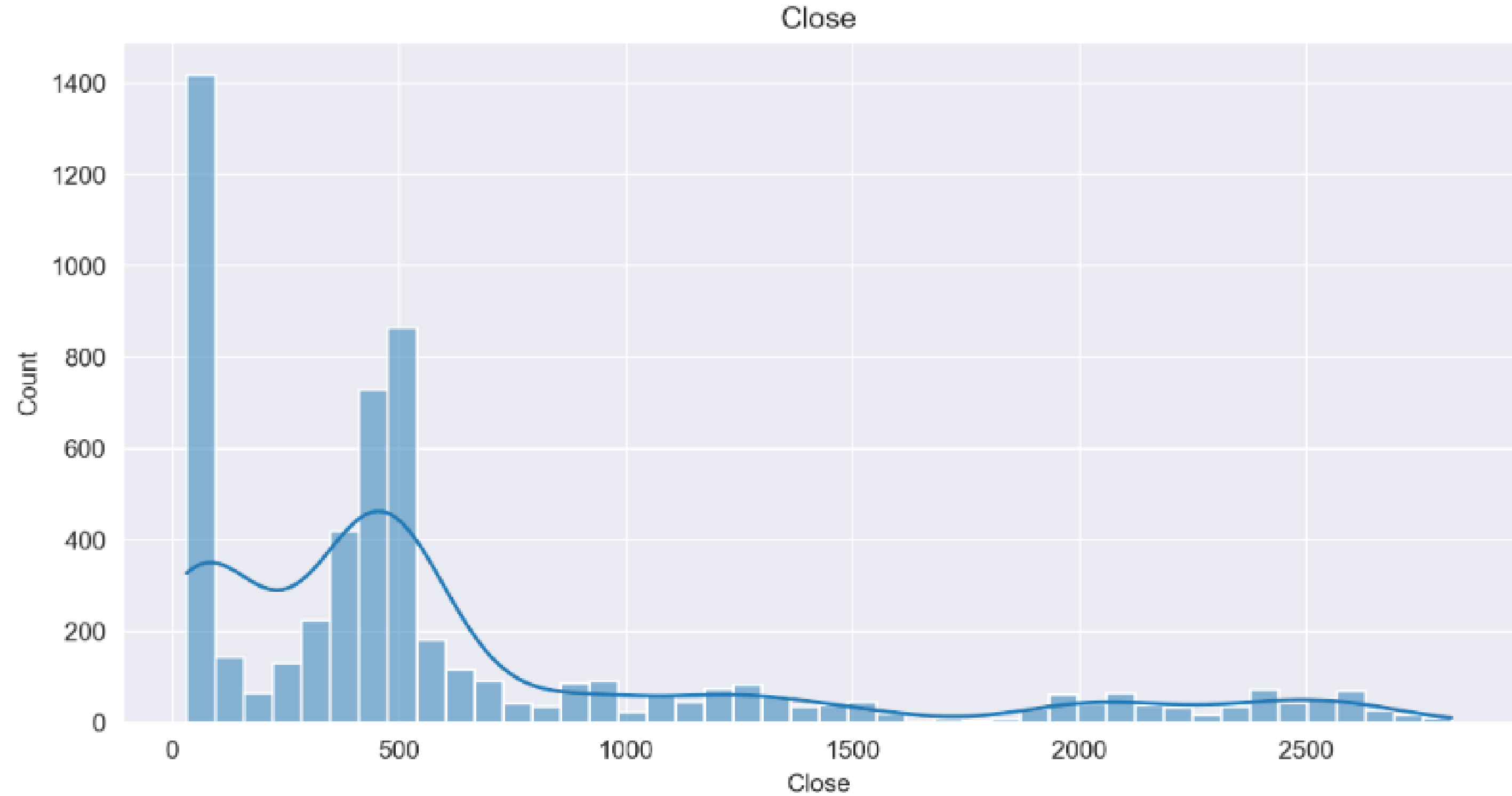
```
data[['Close']].boxplot()  
plt.subplot(1,2,1)  
data[['Volume']].boxplot()
```

[1]: Close: >



# EDA and VISUALIZATION

*data distribution using histogram*

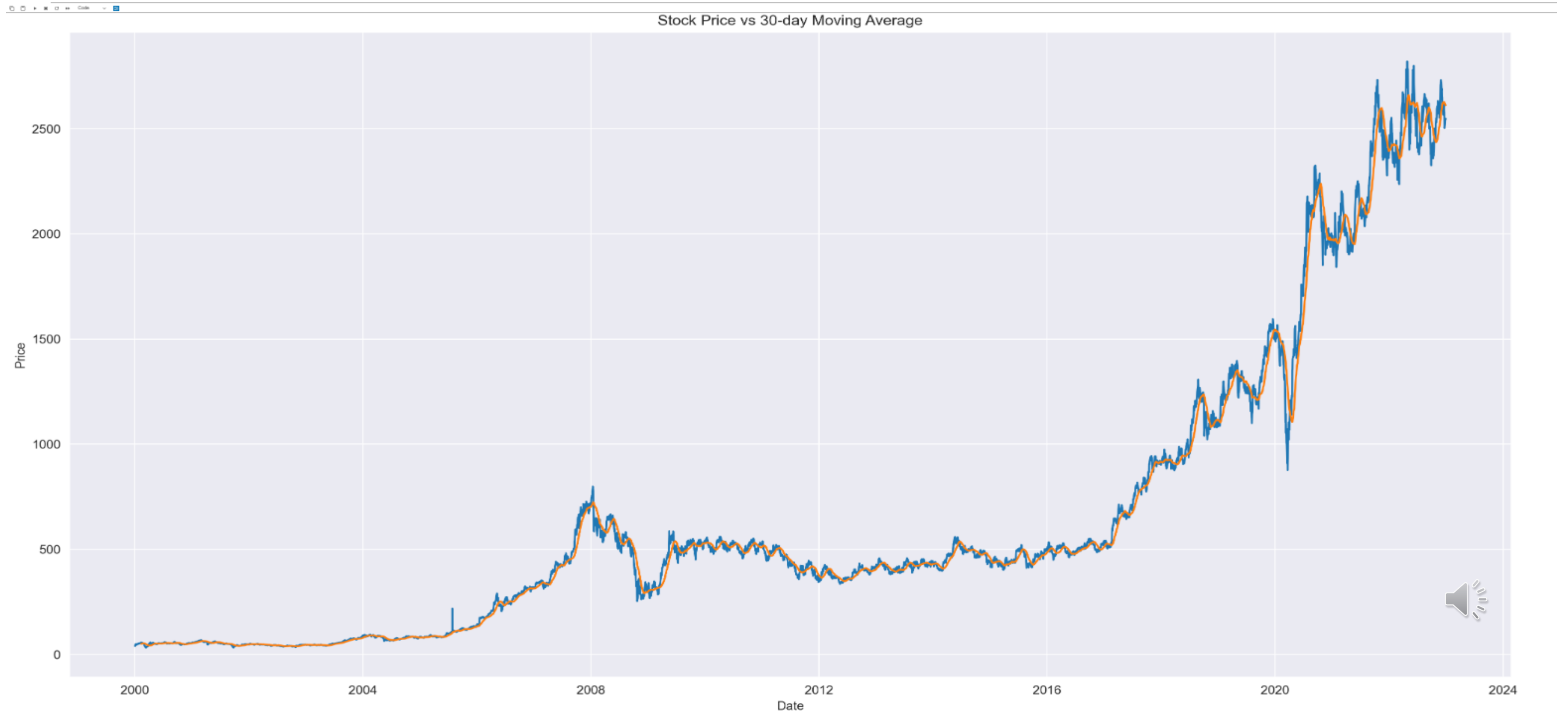


‘Close’ data is positively skewed



# EDA and VISUALIZATION

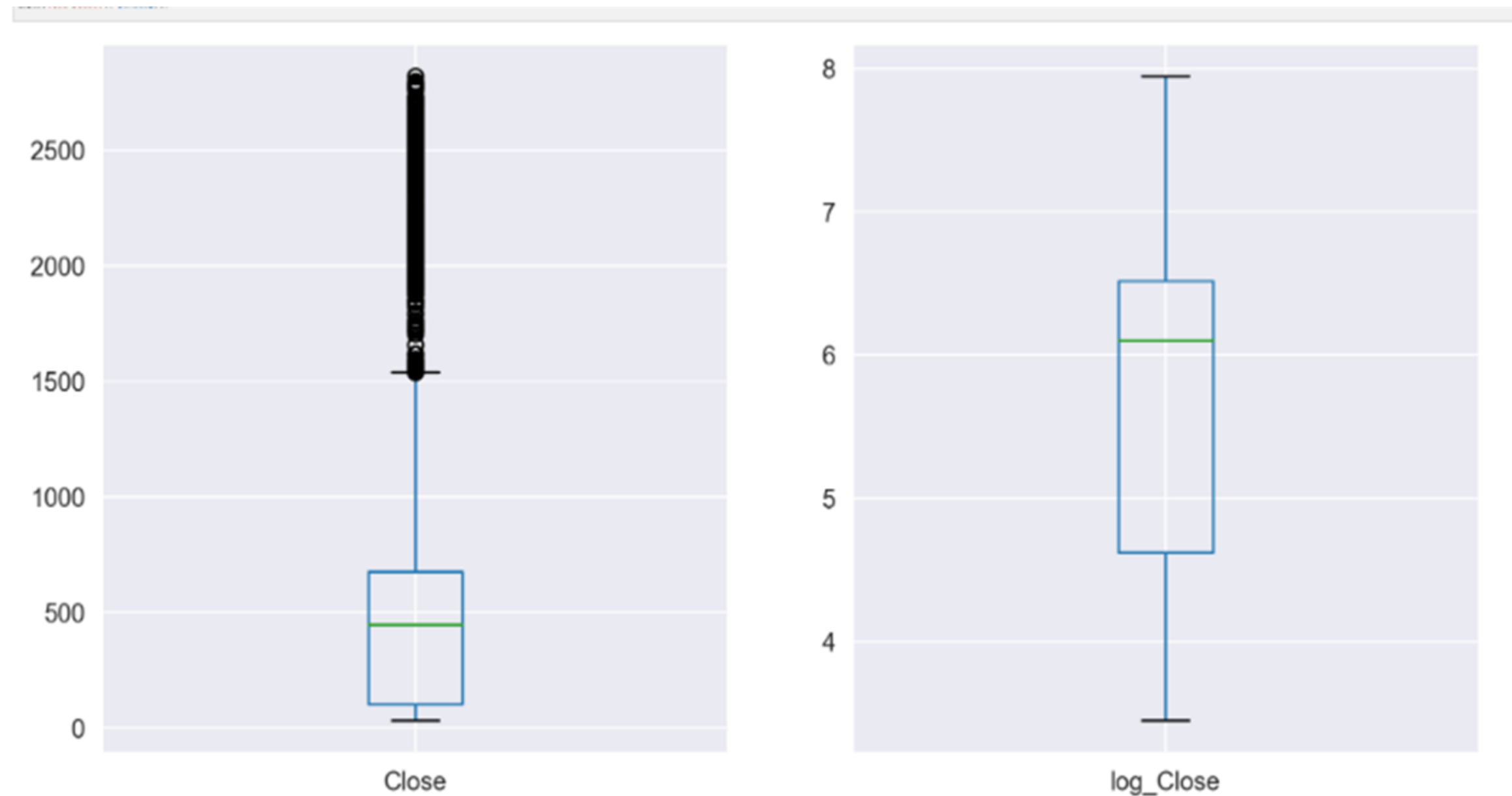
## Stock price vs 30 days moving average



# ADDITIONAL ACTIONS PERFORMED

- *LOG TRANSFORMATION.*

*Applied log transformation to remove outliers and to get normal distribution.*



- After log transformation no outliers present in 'Close'.



# DATA PREPROCESSING

- For this dataset there were 10 null values which are removed null values.
- There were zero duplicate records in the dataset
- 'Date' column was not in datetime datatype, it was in object so we changed it into datetime
- Renamed the columns names
- Set 'Date' column as index column for visualization purpose
- There was no need of 'Adj Close' column so we removed it



# MODEL - EVALUATION

- STEPS TO FOLLOW :-
  - Before applying converted numerical data to the model we should first split the data into train test
  - Import the model
  - Make the model learn the training data
  - Predict the output
  - Check the accuracy
  - Choose the model with the better accuracy
- THE MODELS WE HAVE TRIED:-
  - LSTM
  - GATED RECURRENT UNIT(GRU)



# PREDICTION

```
data={"Model": [ "LSTM", "GRU"],  
      "Train R2 Score": [train_r2_lstm, train_r2_gru],  
      "Test R2 Score": [test_r2_lstm, test_r2_gru]}  
df=pd.DataFrame(data)  
df
```

	Model	Train R2 Score	Test R2 Score
0	LSTM	0.998995	0.952913
1	GRU	0.998285	0.923218

By Looking into this table we can say that our LSTM model have best R2 score.

so we are going to use LSTM model for our deployment part.



# DEPLOYMENT

- For deployment we have used LSTM model because we felt that other models were facing the issue of overfitting (memorizing the data)
- We have used pickle file to dump our LSTM MODEL .
- And for backend code we have loaded back our model from pickle library We have used streamlit library
- And we have demonstrated our backend code in pycharm (IDLE) as it supports many languages
- WORKING:-
  - The input (text) is provided in the web app where we are doing deployment
  - Where the input goes to the backend python script and operations are performed and the output is returned back to the web app.





Show next 30 days predictions

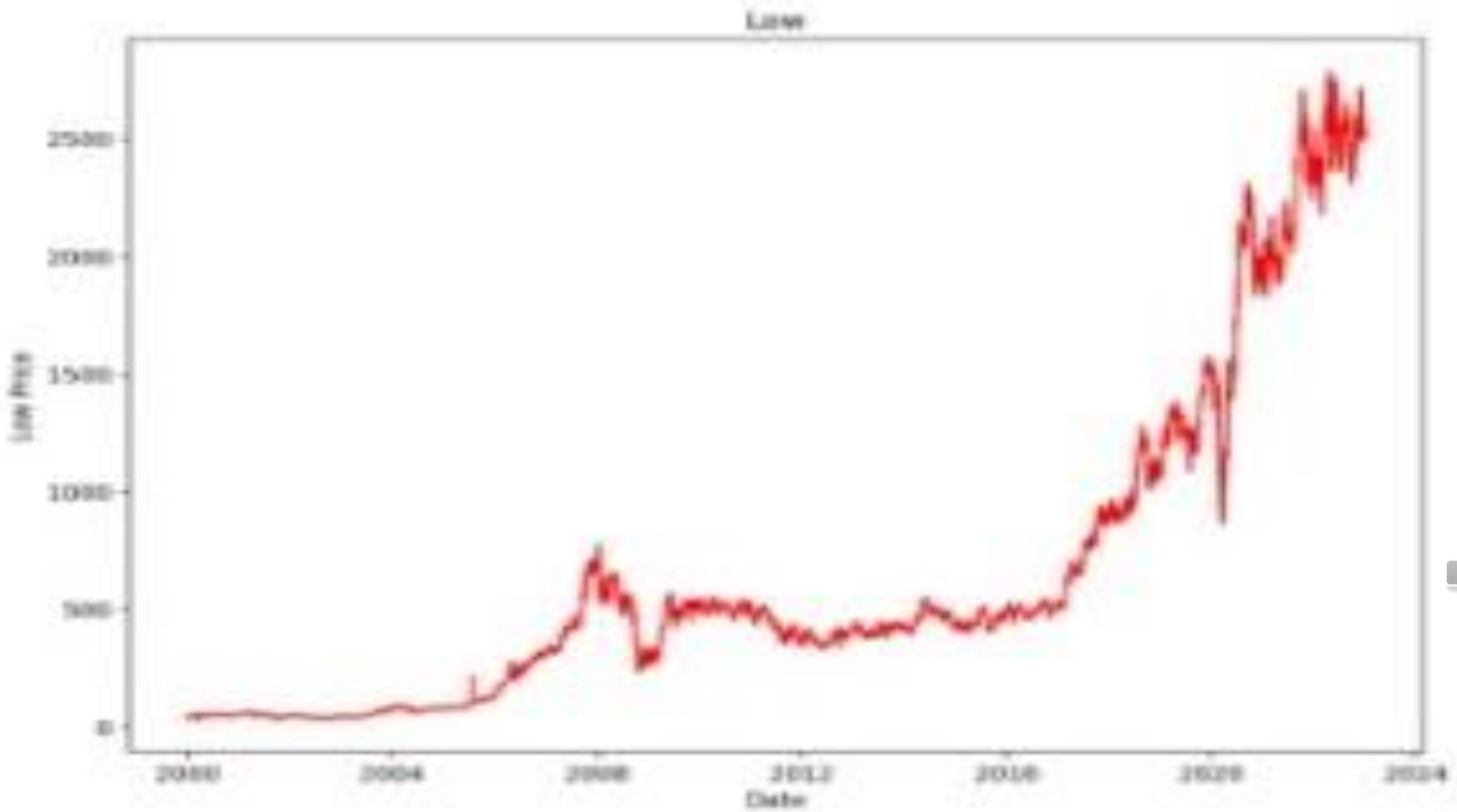
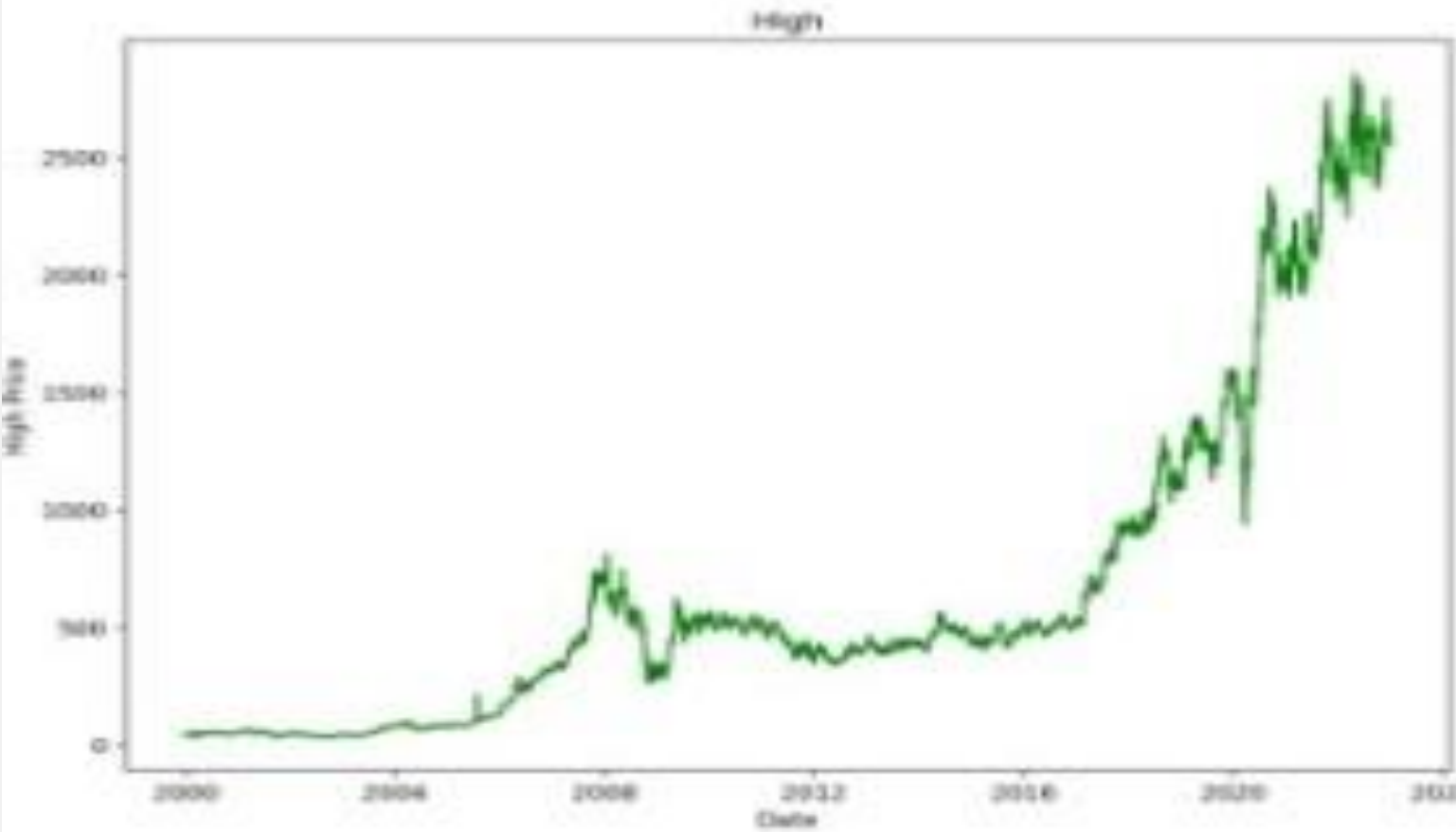
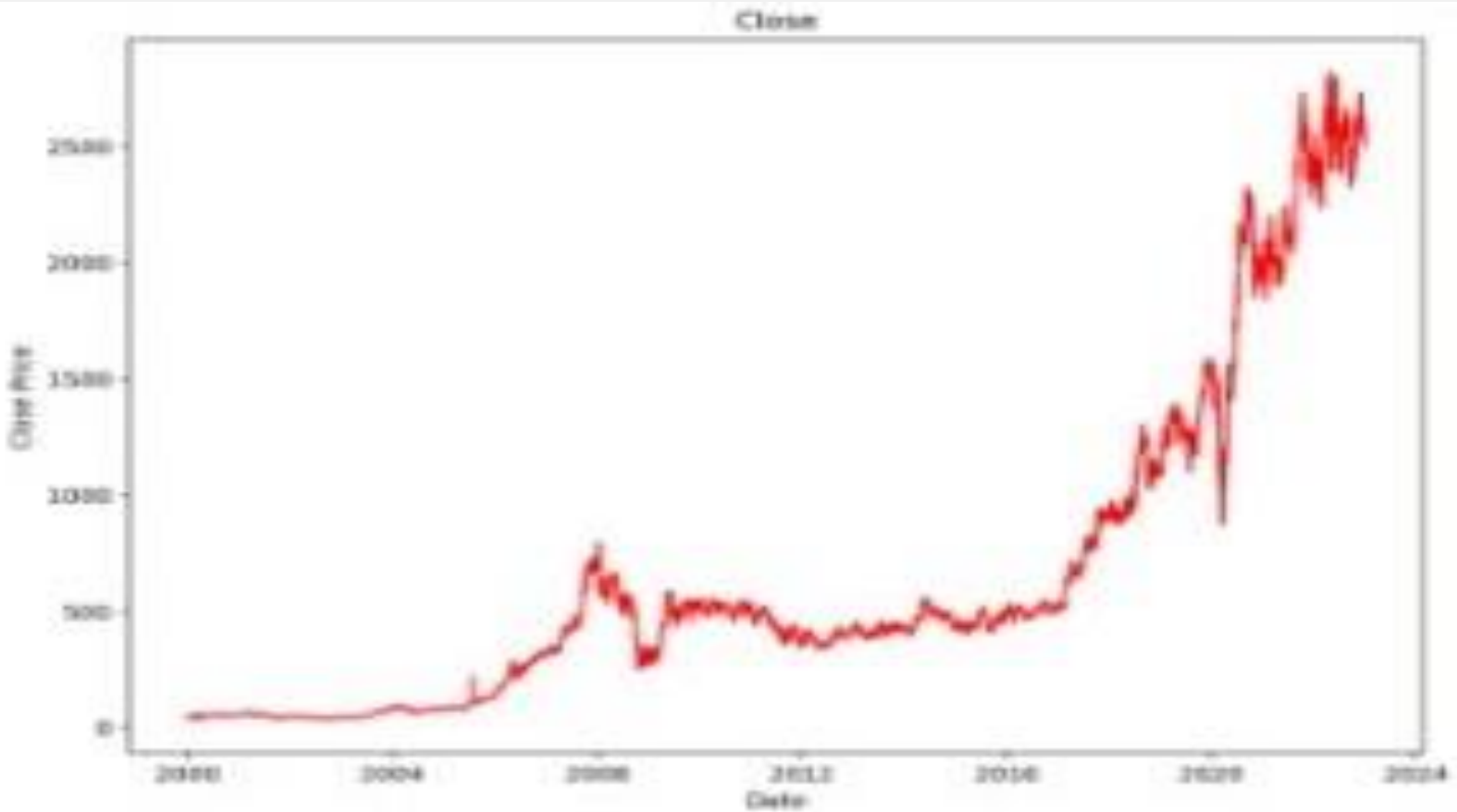
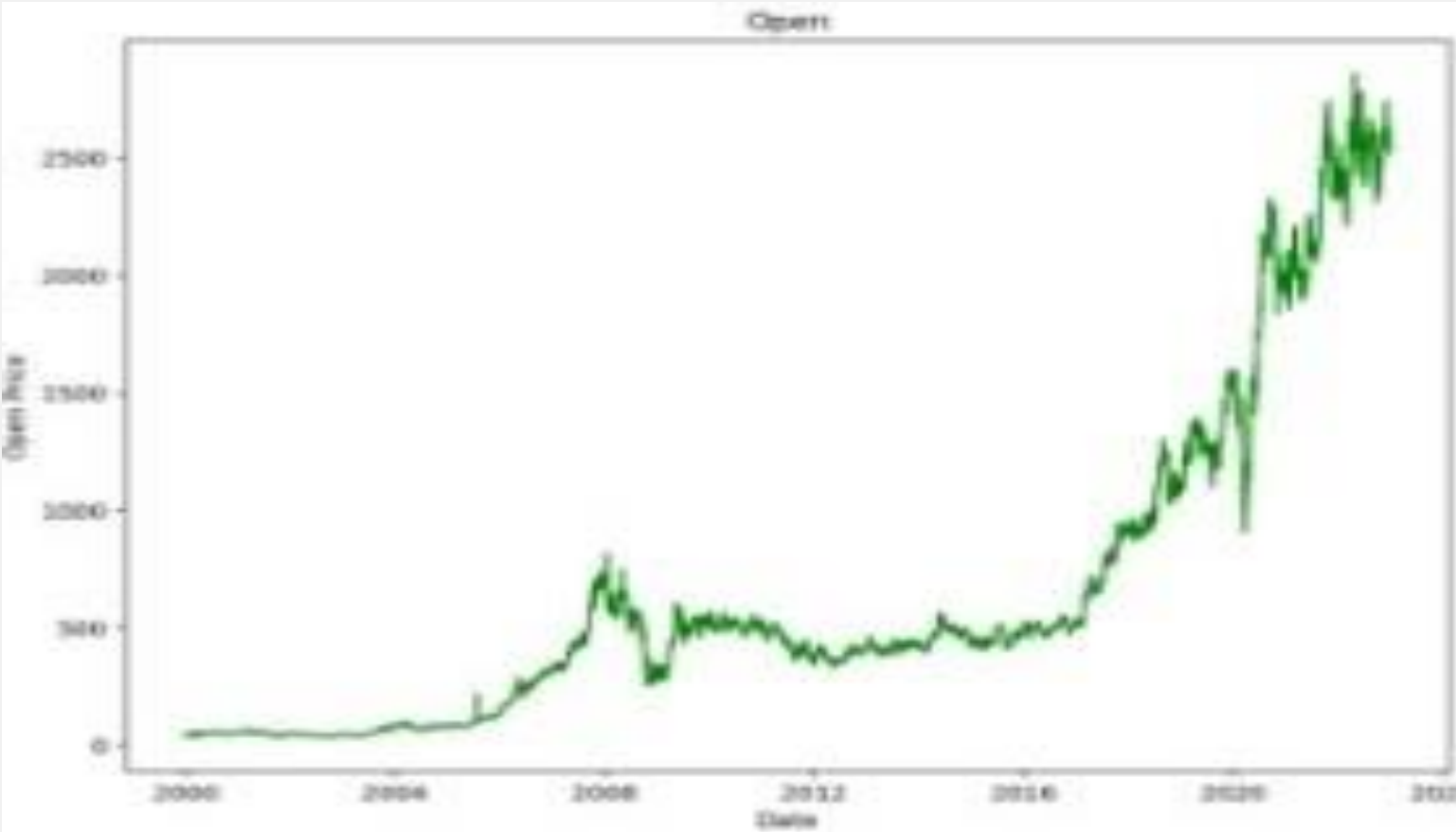
	Price	date
5,715	2,547.1649	2023-01-01 00:00:00
5,716	2,546.8984	2023-01-02 00:00:00
5,717	2,547.1369	2023-01-03 00:00:00
5,718	2,546.9235	2023-01-04 00:00:00
5,719	2,547.1144	2023-01-05 00:00:00
5,720	2,546.9436	2023-01-06 00:00:00
5,721	2,547.0964	2023-01-07 00:00:00
5,722	2,546.9597	2023-01-08 00:00:00
5,723	2,547.082	2023-01-09 00:00:00
5,724	2,546.9726	2023-01-10 00:00:00

## Deployment

As It shows the Prediction of the stock price of next 30 days of the reliance industries.



# Graphs of 30 days predicton stock price for High' , 'Close', 'Open', 'Low'



# Thank You

