**BIG DATA PROCESSING PROJECT**

**Stock Market Analysis and Prediction**

**Group-8**

**Group Members:**

**Dhruvi Kotecha – 202218009**

**Harshita Balani – 202218024**

**Keya Shah – 202218031**

**Bhavisha Chotrani – 202218051**

**Anushka Jain – 202218057**

**INTRODUCTION**

Real-time analysis of the stock market is crucial for successful investing as it requires tracking and interpreting market data, news, and trends to make informed investment decisions. This type of analysis involves accessing a variety of financial and economic data sources such as stock prices, market indices, and company financial statements. To process and analyse this data, analytics tools are employed to identify patterns, trends, and insights in real-time. Real-time analysis of the stock market can be utilized by both individual investors and professional traders to gain a competitive advantage and achieve better returns. By keeping track of market trends and making informed investment decisions, investors can mitigate risks and increase their chances of success in the stock market.

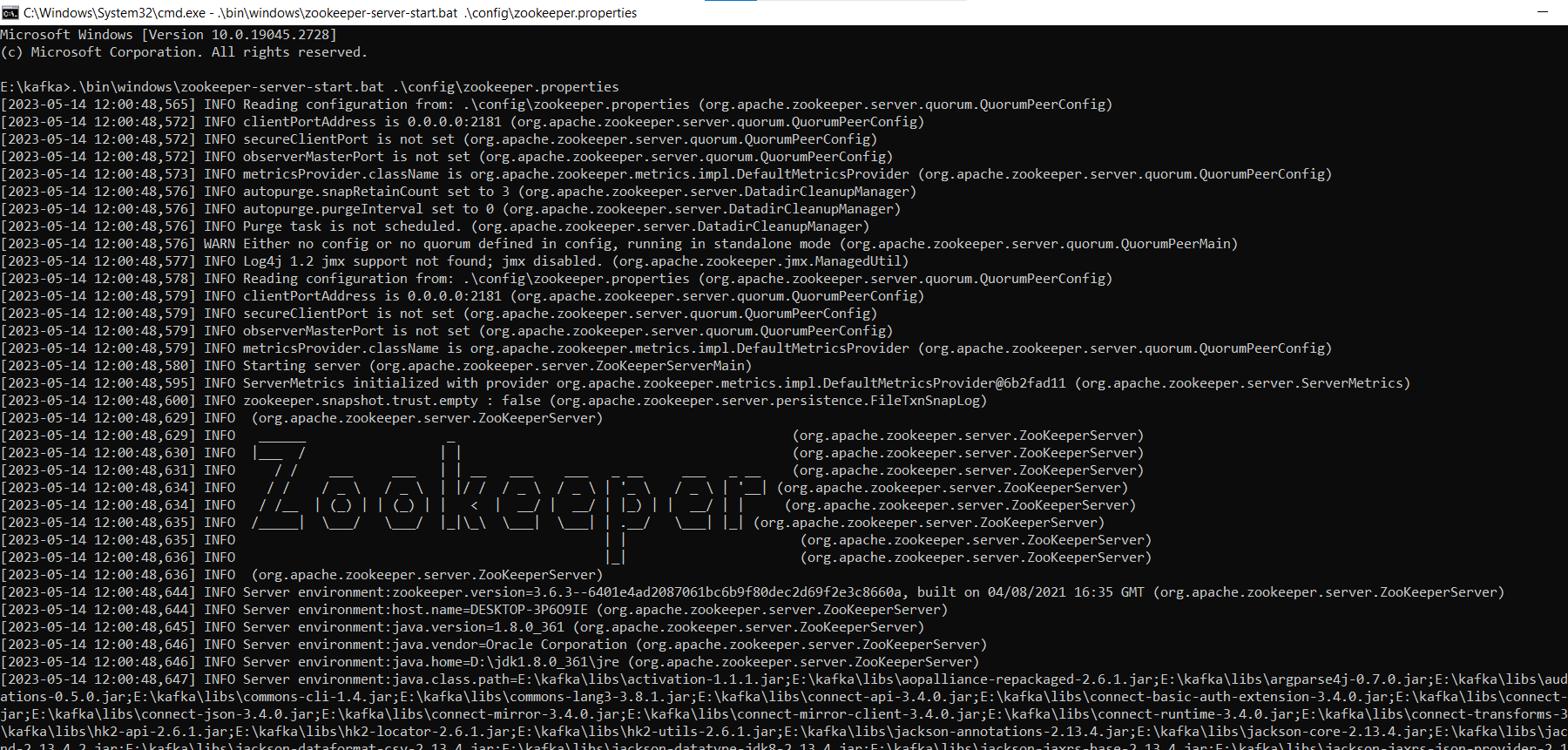
**DATA COLLECTION**

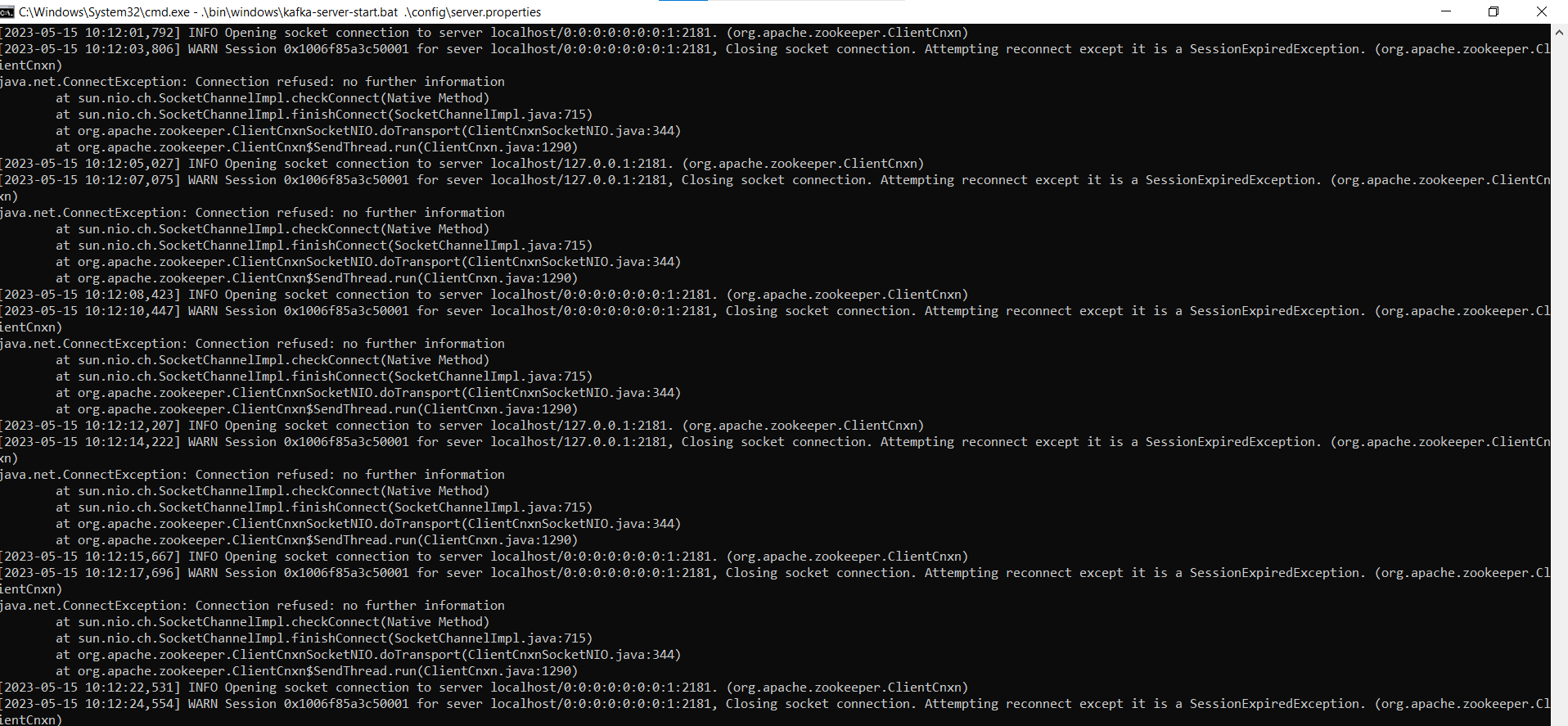
**Kafka:**

The open-source distributed event streaming platform Kafka is employed to manage both real-time and archived data sources. The fault-tolerant and highly scalable Kafka system can handle huge volume data streams in real time.

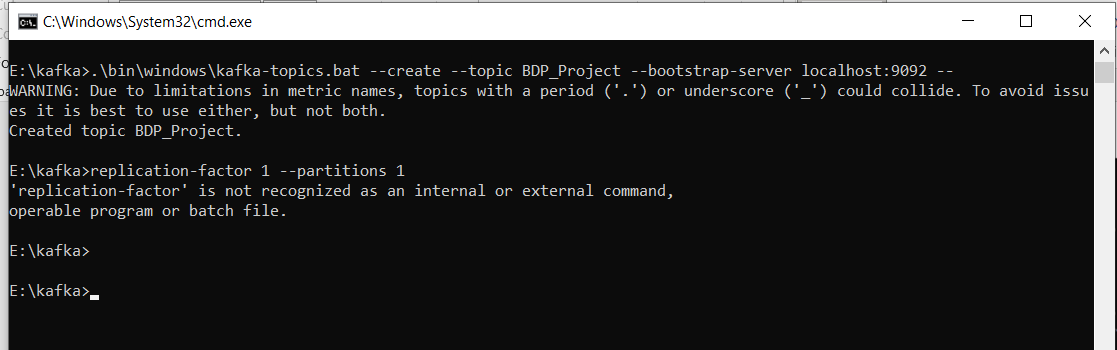
Applications can send and receive messages in real-time due to the publish-subscribe messaging system provided by Kafka. Consumers subscribe to Kafka topics in order to receive messages sent by producers to those topics.

**Steps to set-up kafka:**  
We first have run zookeeper-properties, then opened server-properties





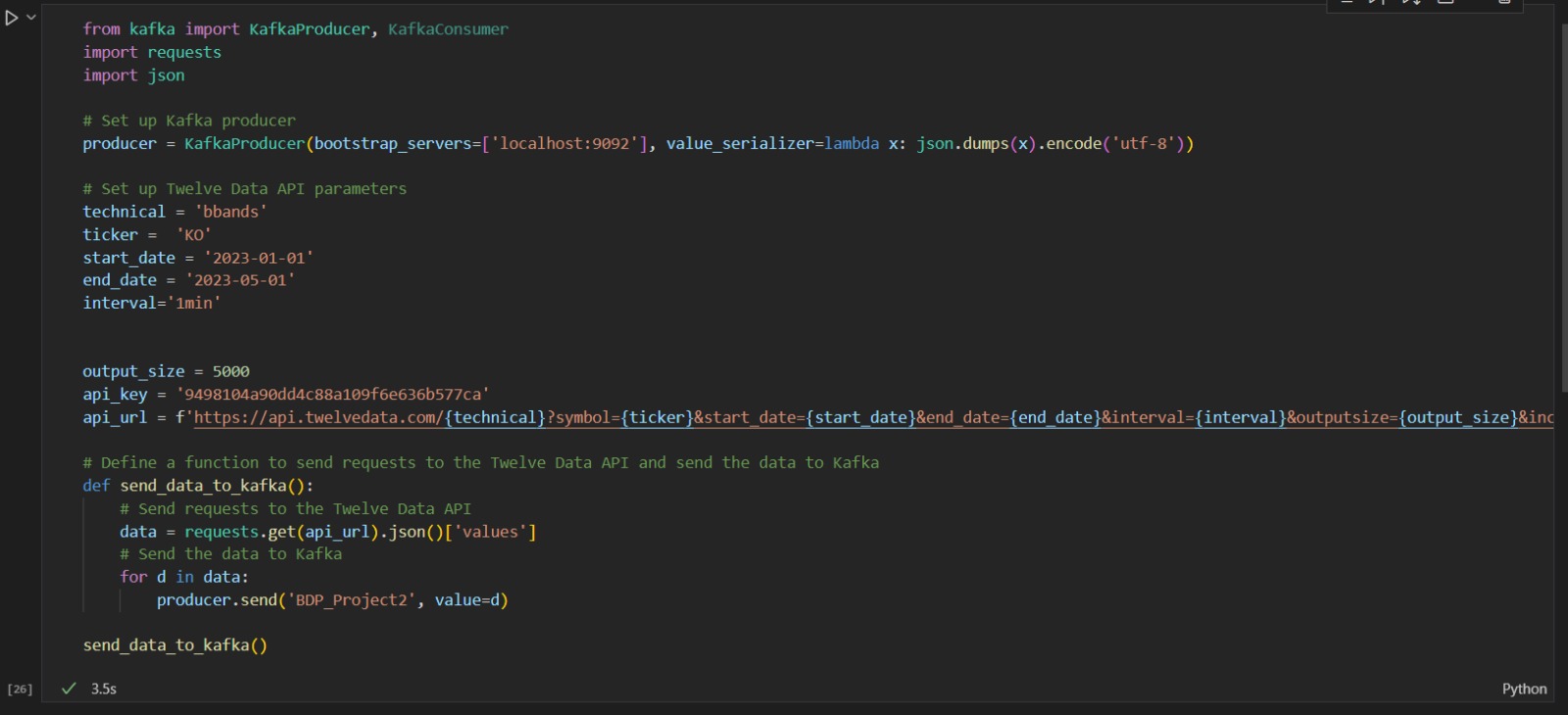
Then we have made a topic,named BDP\_Project



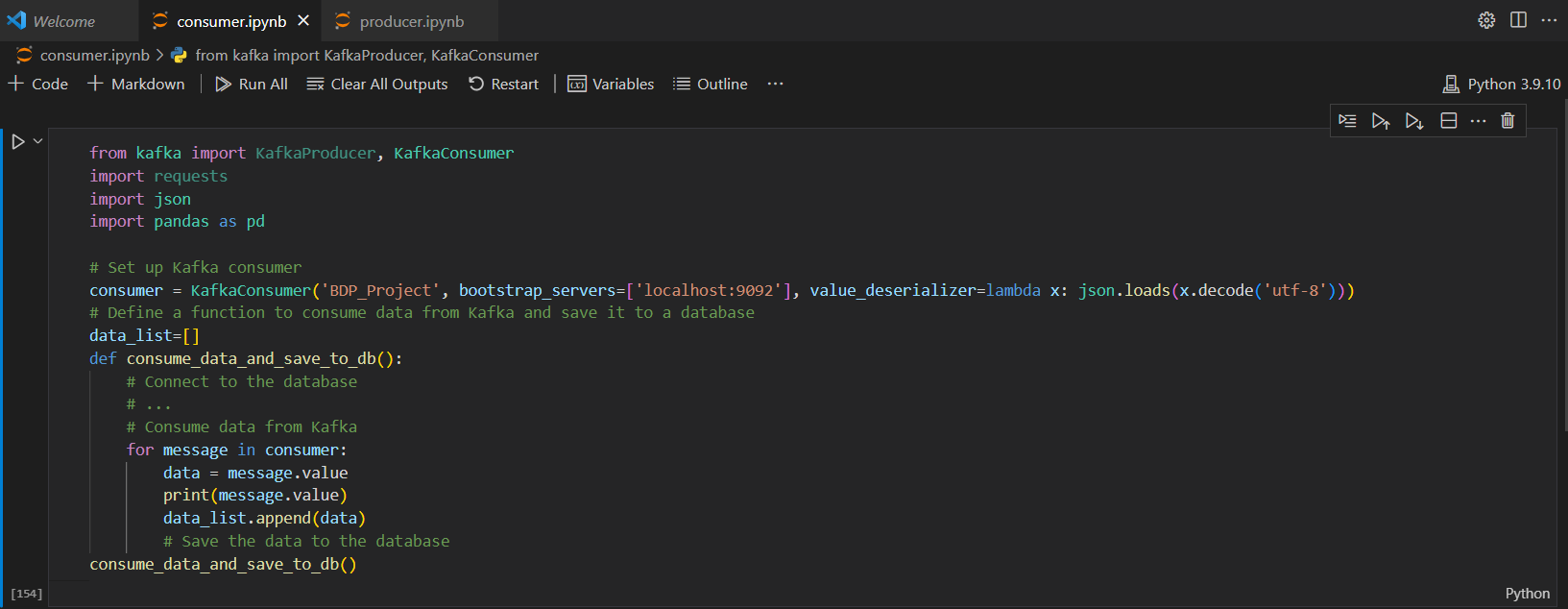
Then we have made two .ipynb files and used Twelve-Data API

Twelve-Data is an API service that provides financial market data, including real-time and historical stock prices, forex rates, cryptocurrency data, and more. It offers RESTful APIs that allows us to retrieve data programmatically and use it in their own applications. It offers a free tier that allows users to make up to 800 API requests per day, with limitations on data updates and historical data access.

**Producer.ipynb:**



**Consumer.ipynb:**

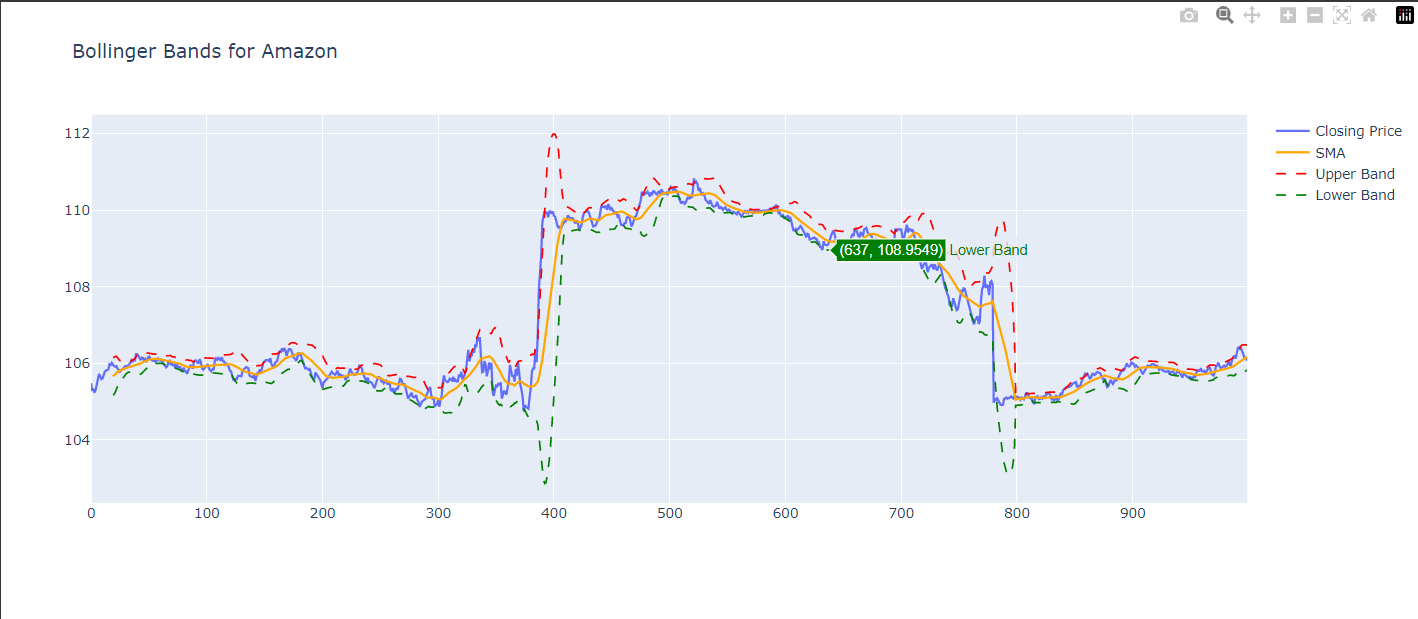


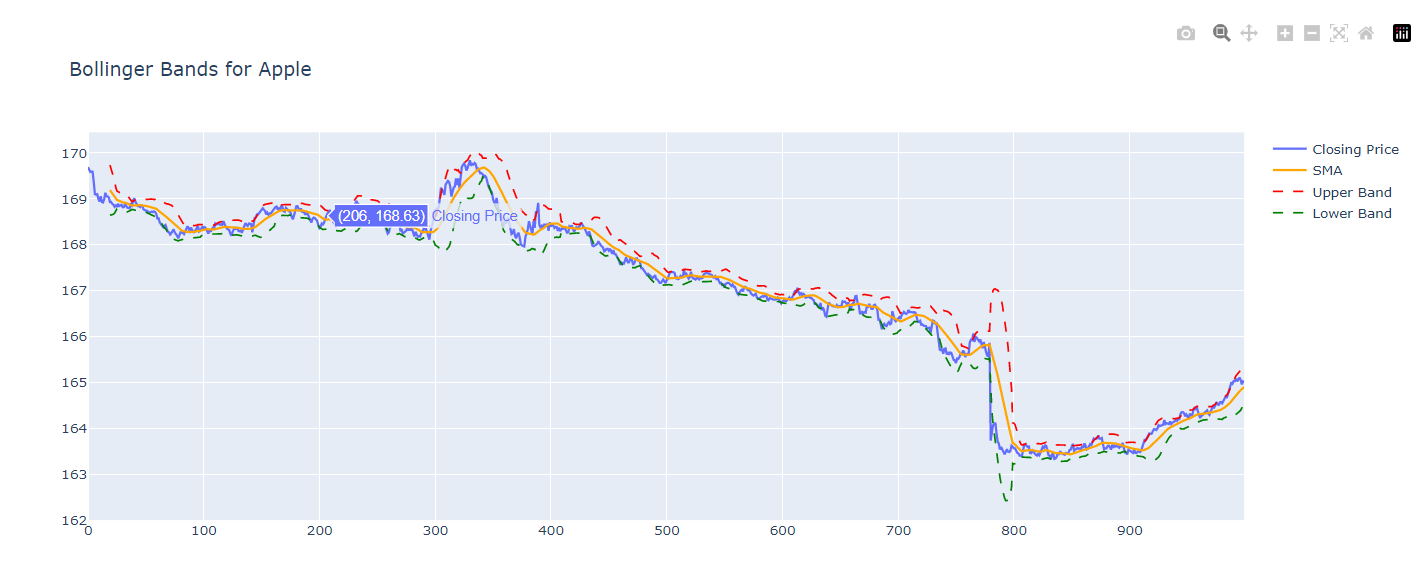
We have fetched about 20 csv files with columns: datatime,upper\_band,middle\_band,lower\_band,open,high,low,close,volume

In Producer file, we have taken limit, i.e., output\_size=5000, so each .csv file contains 5000 records.

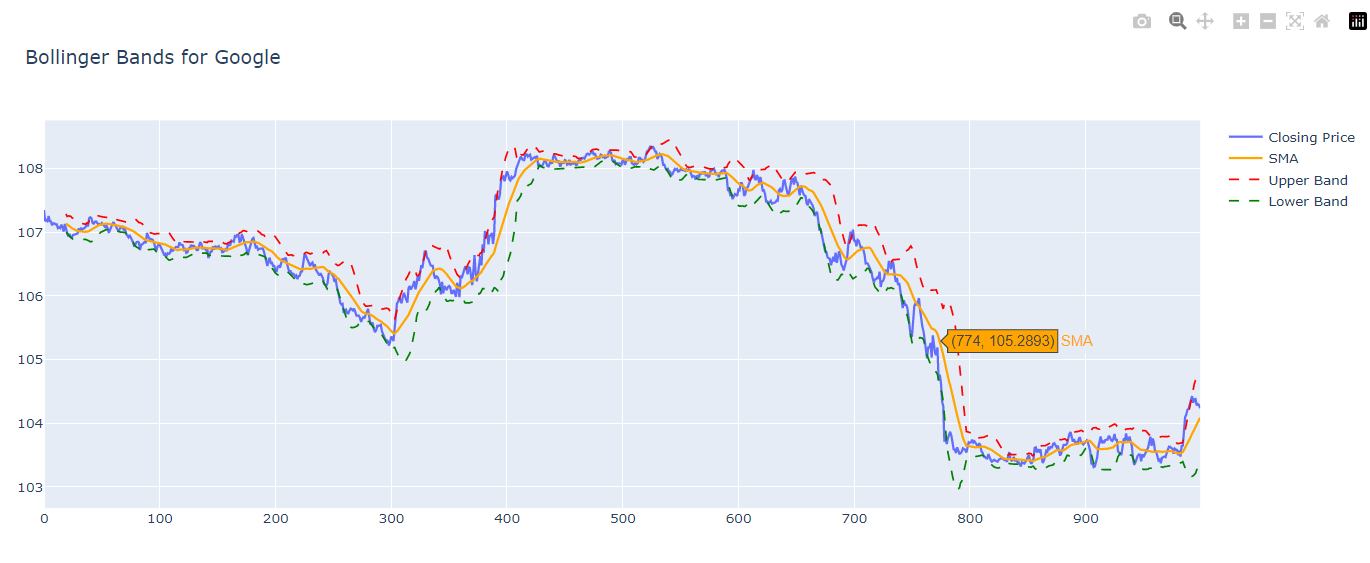
We have collected stock details of these companies:  
Meta, Amazon, Apple, Netflix,Google,JP Morgan,Nvidia,Samsung,Nokia, IBM,McDonald’s, Pepsi, Coca-Cola, Dominos,Starbucks,Honda, Toyota, Ford,Tesla,BMW

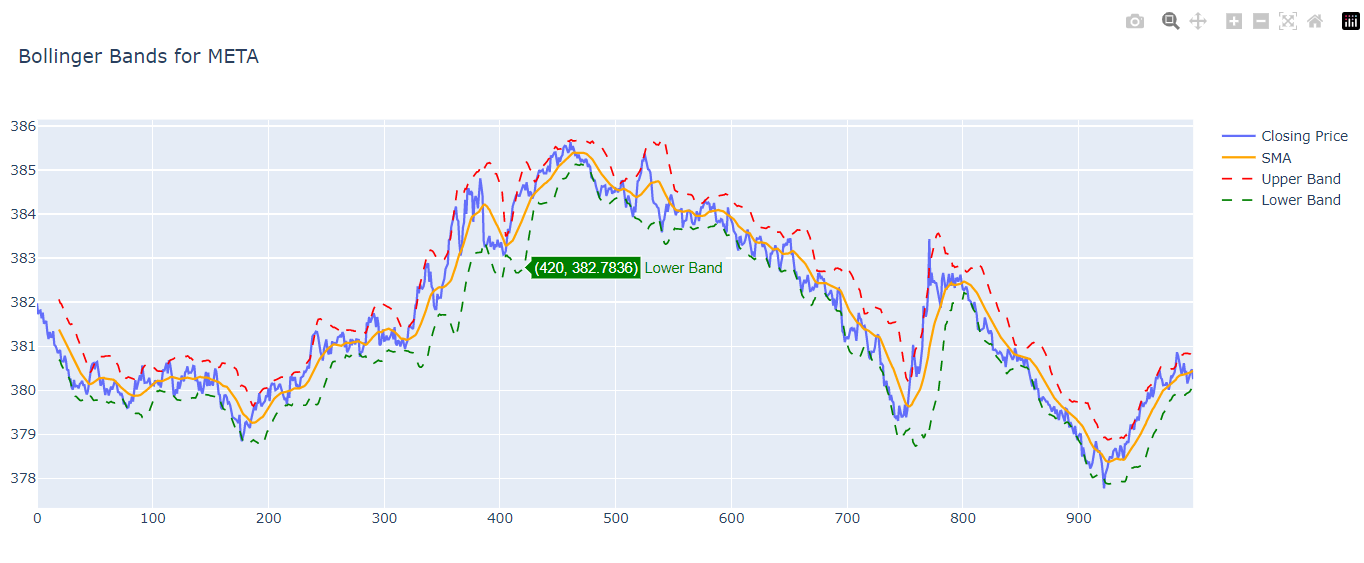
**EXPLORATORY DATA ANALYSIS PERFORMED**

**VOLATILITY ANALYSIS FOR MAANG COMPANIES**

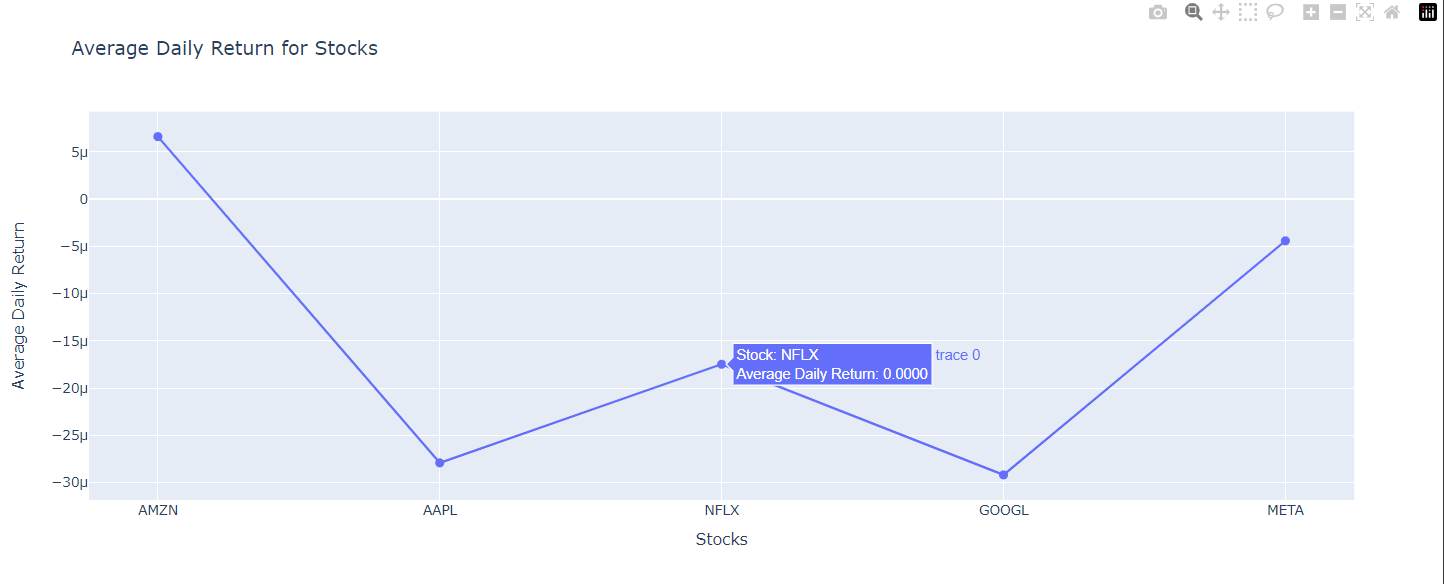




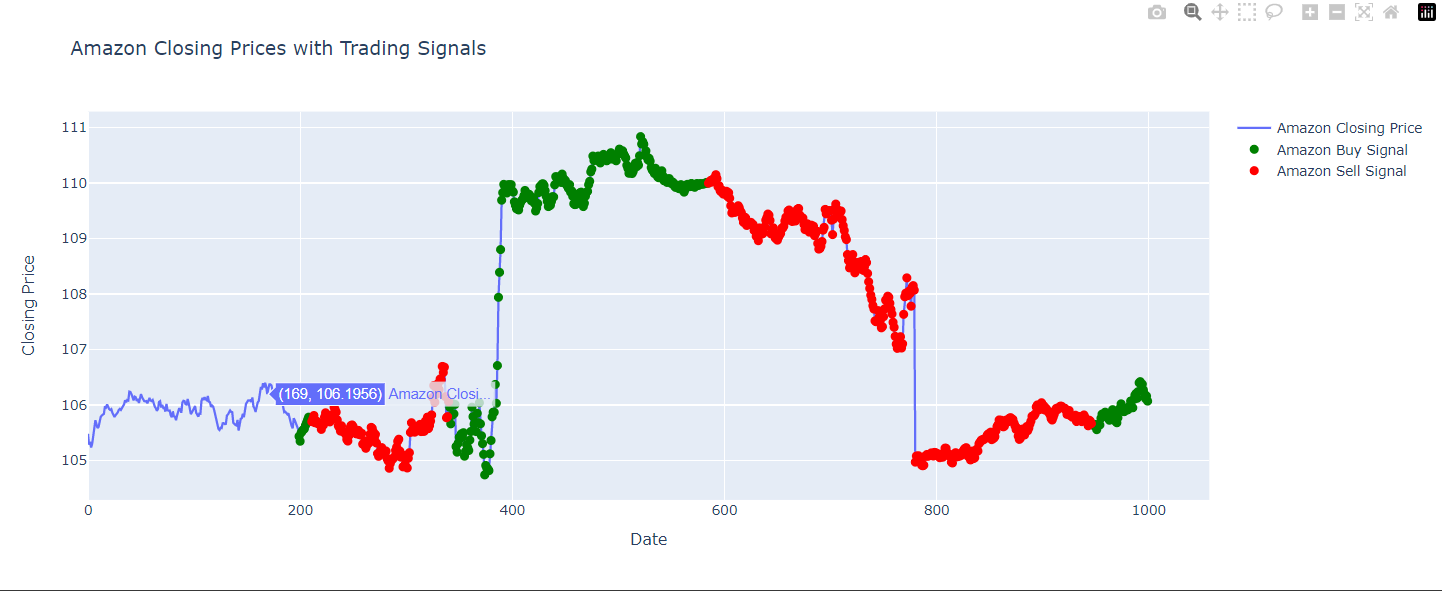


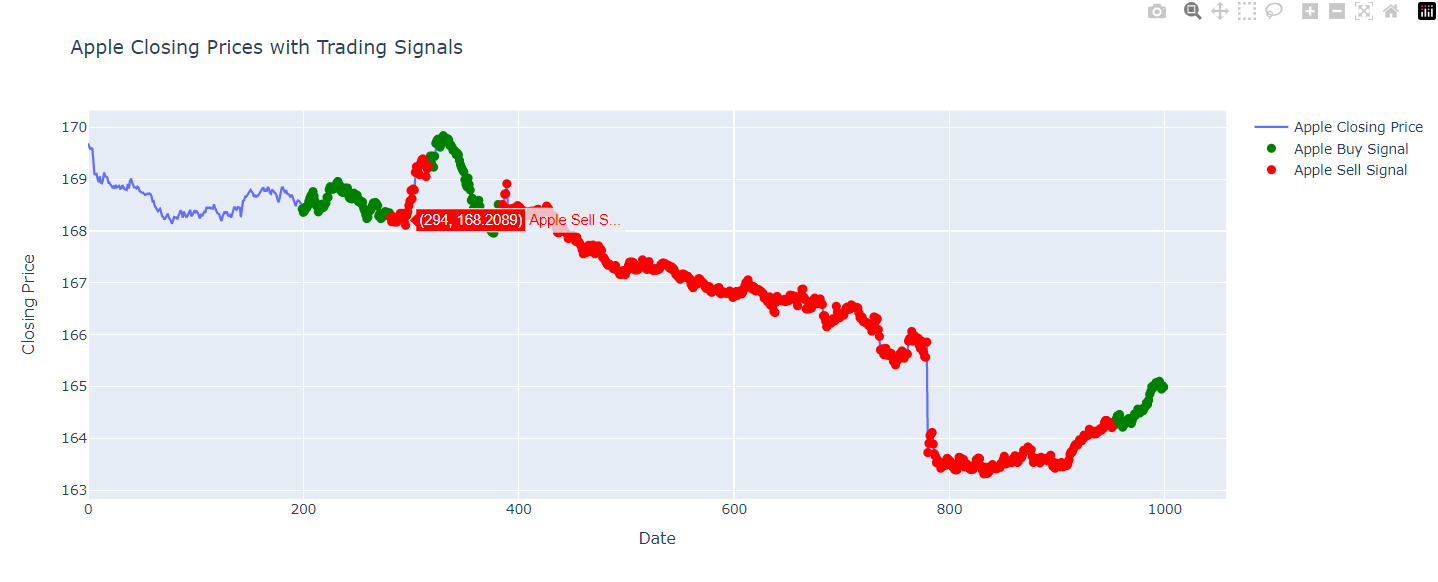


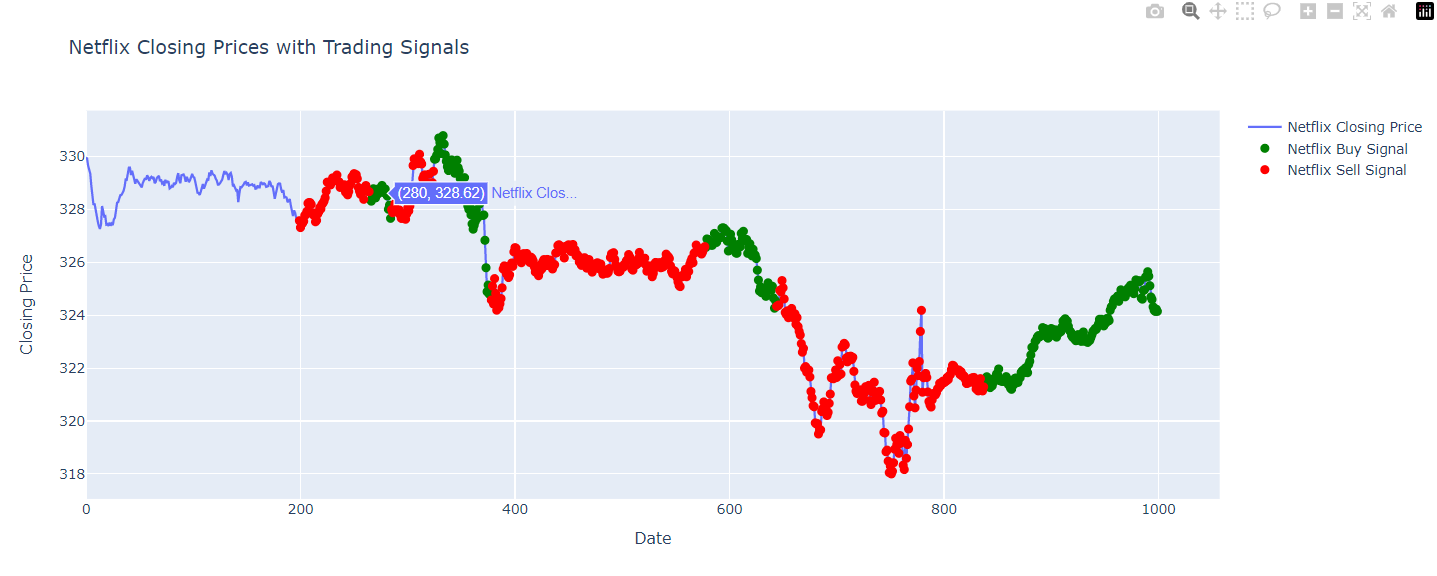
**DAILY RETURNS FOR MAANG COMPANIES**

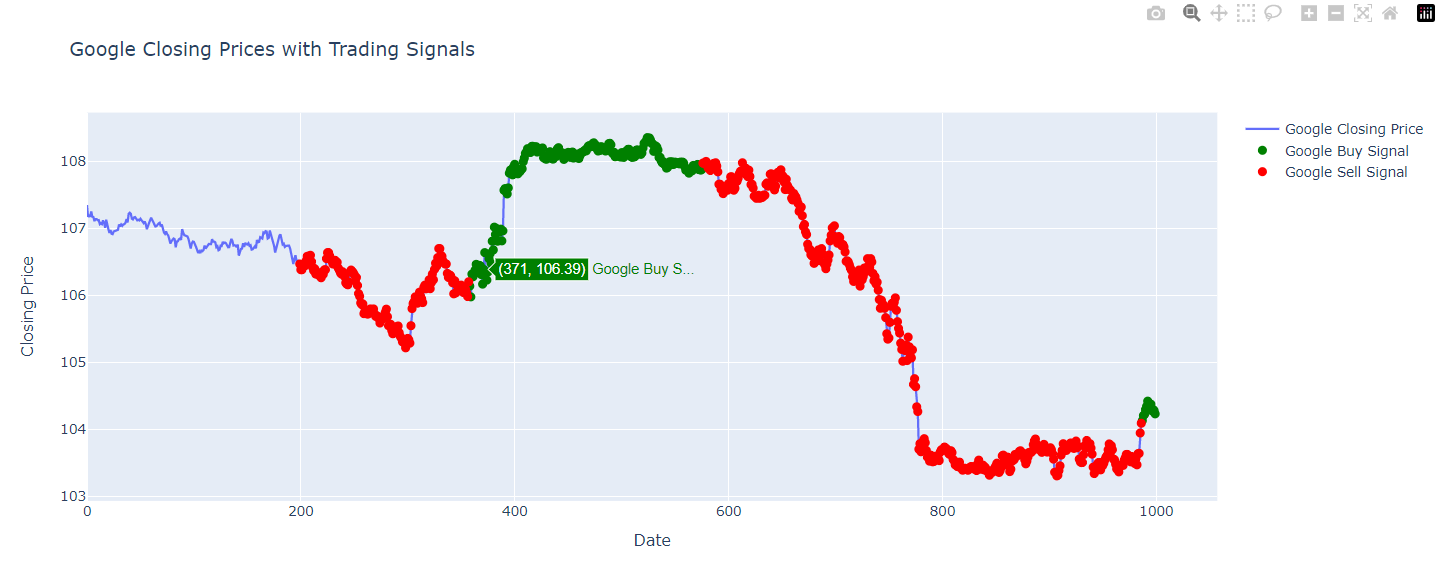
****

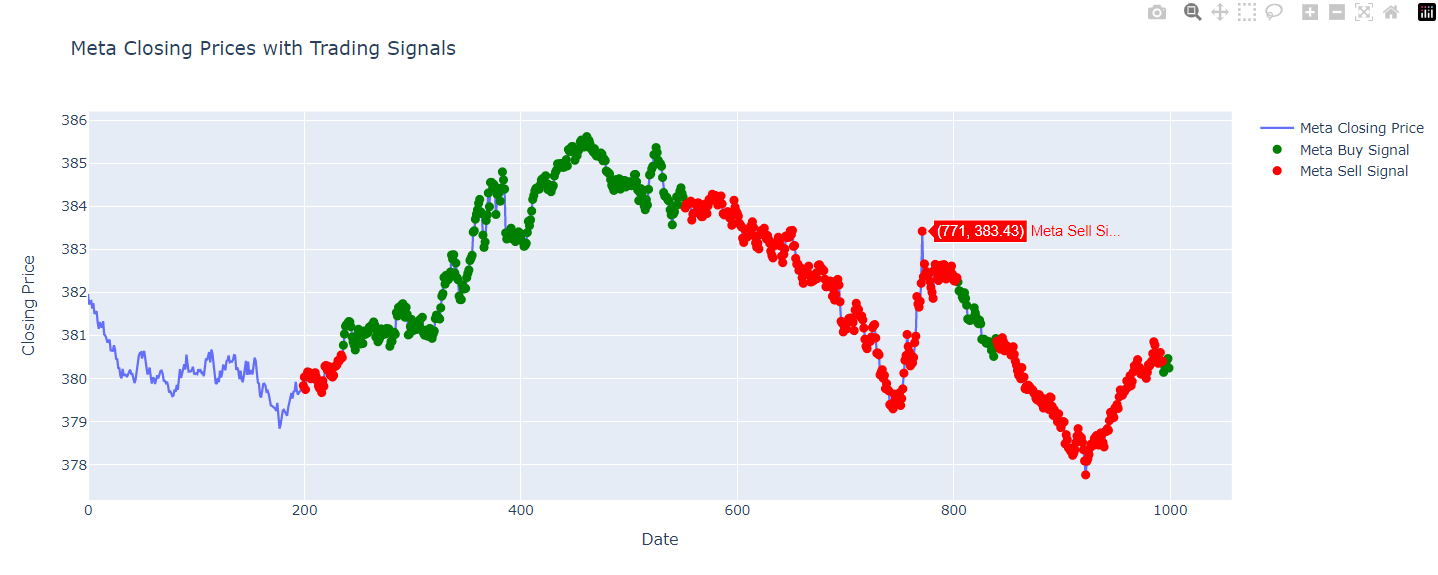
**TRADING SIGNALS FOR MAANG COMPANIES**

****

****

****

****

****

**MACHINE LEARNING**

**Linear Regression**:

We have evaluated linear regression models for all MAANG companies for comparative studies of stocks. Linear regression is a simple tool for extraction of useful insights and is useful for inferential studies.

We split our data using ‘randomsplit’. We have used 70% of data for training and 30 % code for testing data.

**Libraries used for Machine Learning:**

* pyspark.ml.feature.VectorAssembler: This module is used to assemble multiple columns of features into a single column of vectors. In the given code, it is used to assemble features into the "features" column.
* pyspark.ml.regression.LinearRegression: This module is used to perform linear regression analysis on the data. In the given code, it is used to train a linear regression model.
* pyspark.ml.evaluation.RegressionEvaluator: This module is used to evaluate the performance of the regression model using various metrics such as RMSE, R-squared, etc. In the given code, it is used to calculate RMSE and R-squared.
* pyspark.sql.SparkSession: This module is used to create a SparkSession, which is the entry point to programming Spark with the Dataset and DataFrame API. In the given codes, it is used to create a SparkSession.
* pyspark.sql.functions.abs: This module is used to apply the "abs" function to a column of data in a Spark DataFrame. In the given code, it is not used.
* matplotlib.pyplot: This module is used to create data visualizations in Python. This is used later for plots of linear regression.

Maximum number of iterations: 10

We have set our regularization parameter as 0.3 and alpha as elastic net parameter as 0.8.

Elastic net model is a special case we can use when our model is overfitting the data. It uses l1 and L2 norm.

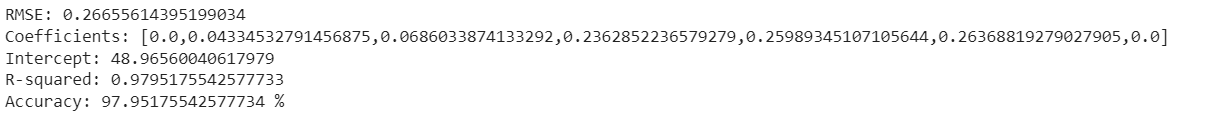
Evaluation methods:

To evaluate our model we have used RMSE, Model coefficients, intercept and R-Squared measure. These are the basic functionalities used for evaluating any regression models. We have calculated these using inbuilt libraries

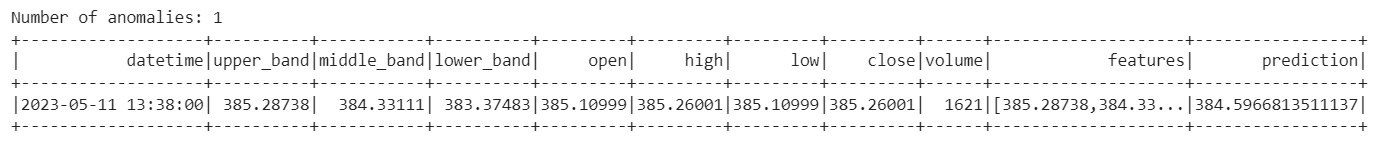
Outputs:

1. **META**

Evaluation:



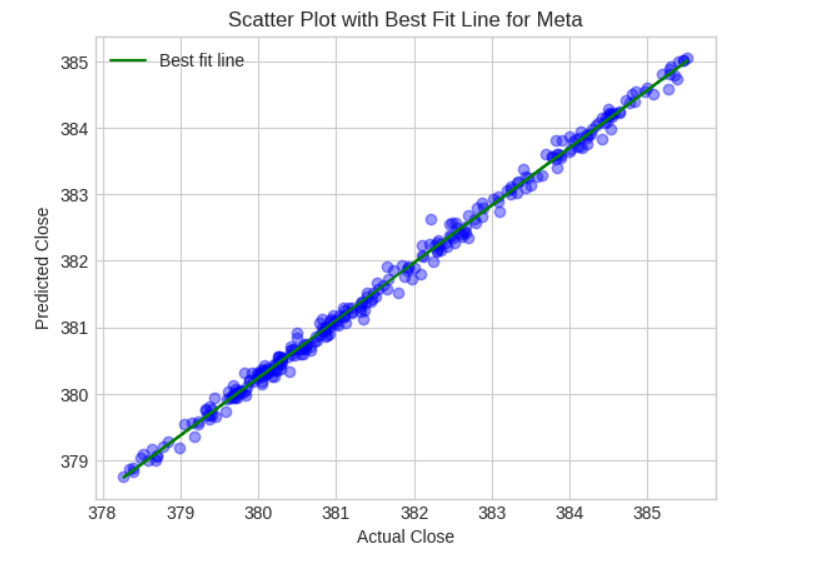
Anomalies detection:



The residuals—the absolute difference between the actual and anticipated values—are calculated by the code to find these anomalies, and it then determines a threshold for anomaly identification as 3 standard deviations from the mean. Any observation with a residual above the threshold is thought to be an anomaly.

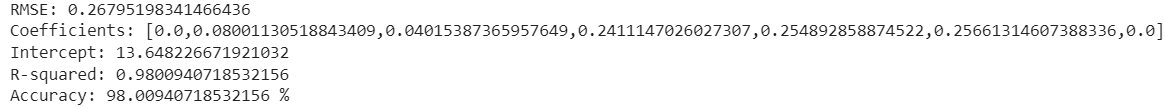
To find any outliers or unexpected observations in the projected values of the linear regression model is the goal of discovering anomalies in this situation. These anomalies could be a sign of model flaws or accuracy issues, or they could be actual abnormalities or outliers in the data. To ensure the precision and dependability of the model's predictions, it is critical to recognise and look into these abnormalities.

Plot:



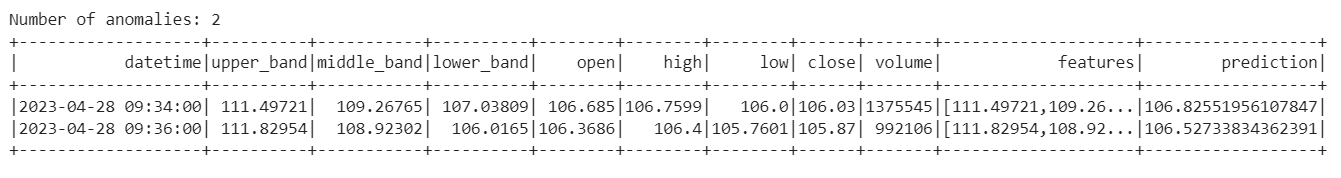
1. **AMAZON**

Evaluation:



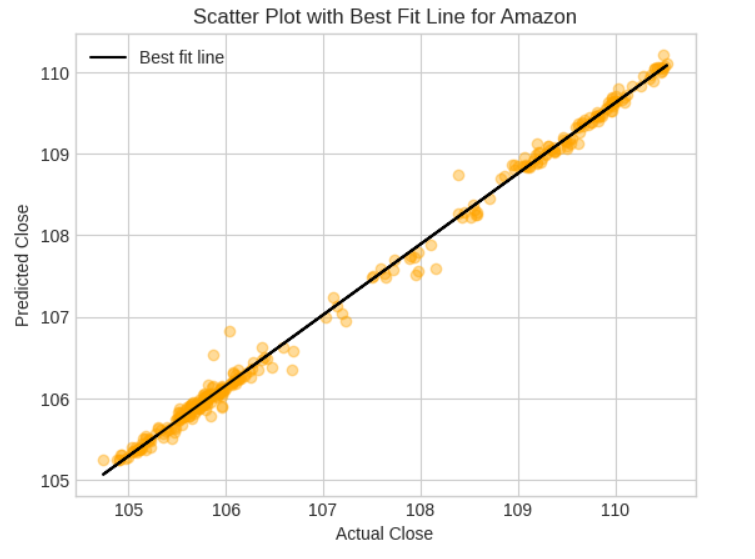
The R-squared measure of how much of the variance in the close prices is explained by the input features is used by the code to determine the model's accuracy. A high R-squared and accuracy value show that the model fits the data well and is capable of forecasting the close prices of Amazon stock.

Anomalies detection:



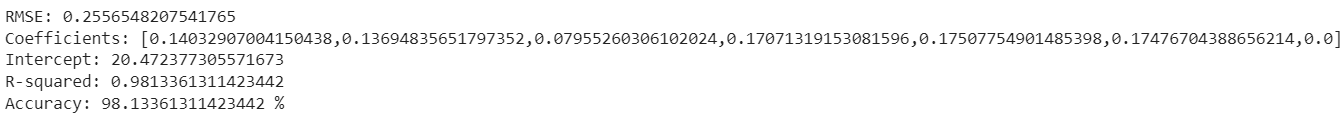
We have two anomalies present in this data.

Plot:

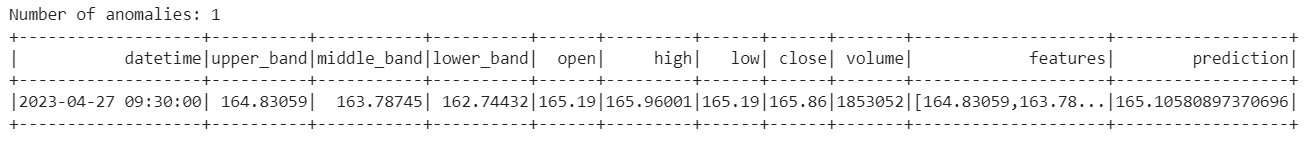


1. **APPLE**

Evaluation:

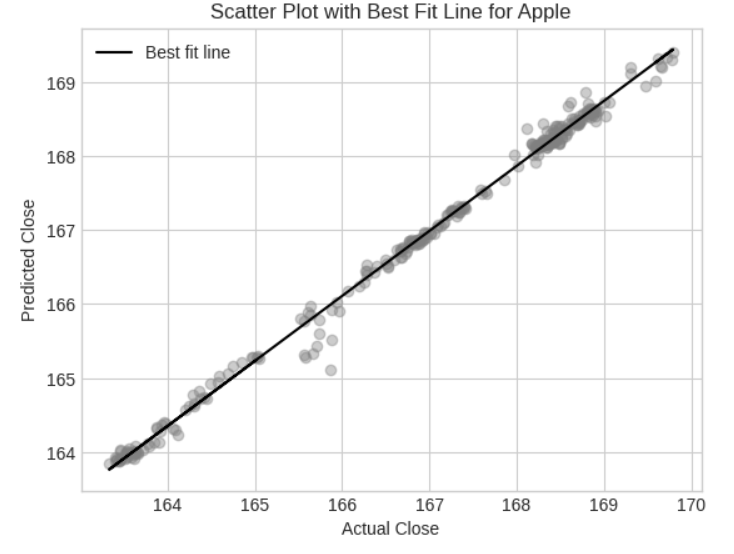


Anomalies detection:



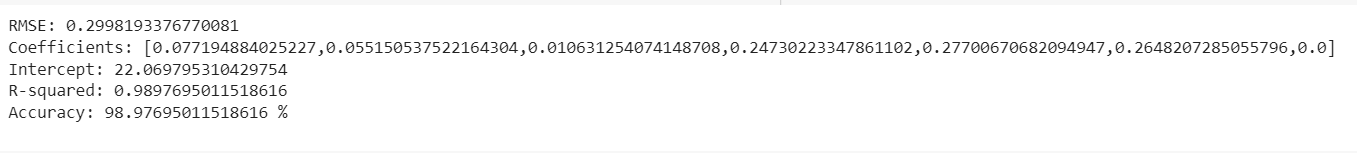
We have one anomaly present in the data.

Plot:



1. **NETFLIX**

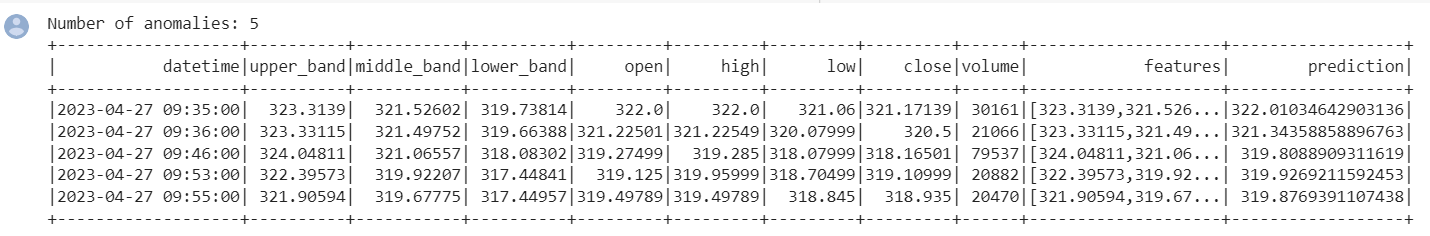
Evaluation:



The RMSE value of 0.2998 indicates the average difference between the actual close prices and the predicted close prices is relatively low

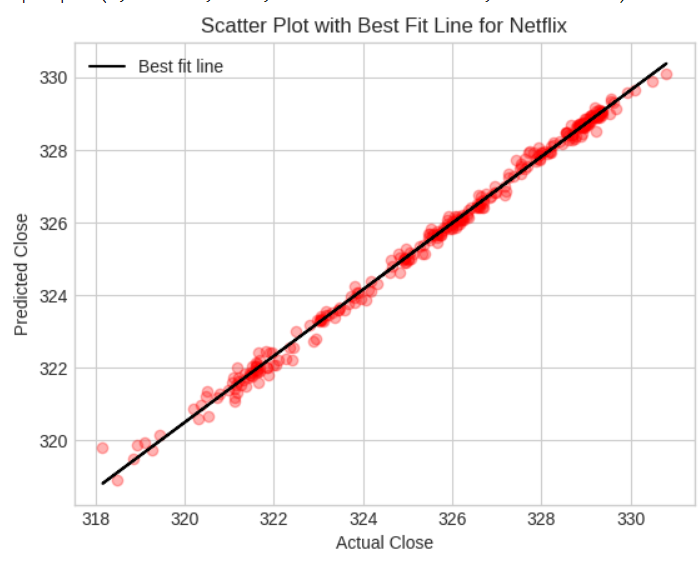
The interceptl is 22.0698. It represents the predicted close price when all the feature values are 0.

Anomalies Detection:

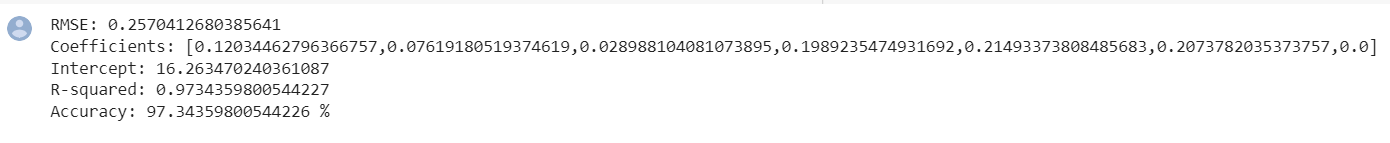


We got 5 anomalies in Netflix

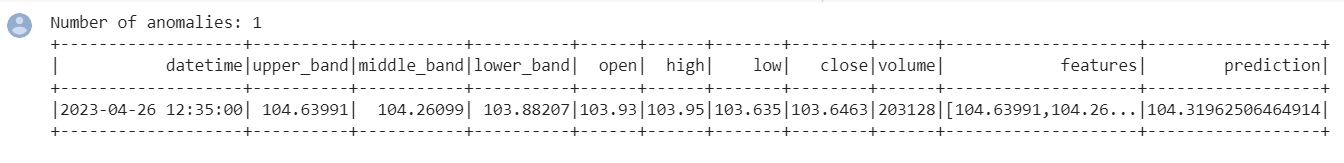
Plot Showing Actual Vs Predicted Closing Value:



1. **GOOGLE**

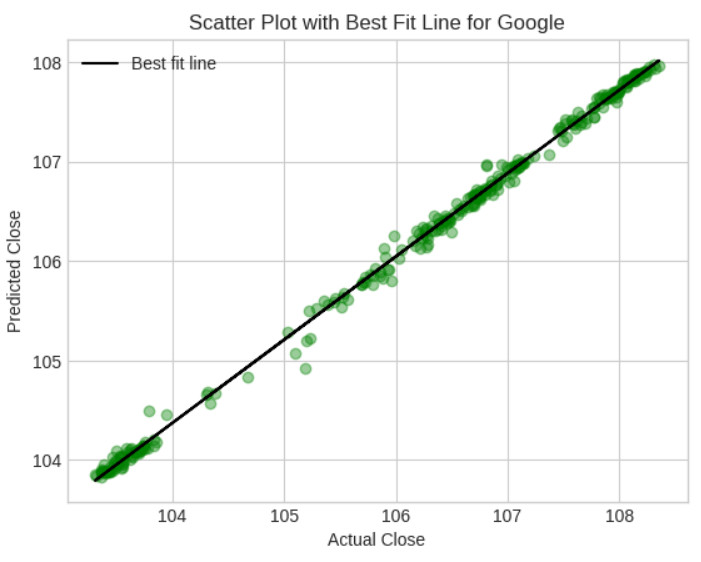
Evaluation:  


Anomalies Detection:



We have 1 anomaly in Google.

Plot Showing Actual Vs Predicted Closing Value:



**LONG-SHORT TERM MEMORY**

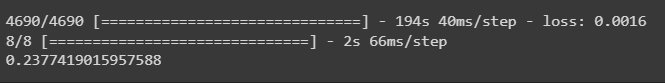
LSTM is a type of neural network architecture designed to remember and utilize important information from the past, making it particularly useful for tasks that involve sequences of data or where capturing long-term dependencies is crucial, such as language modeling, speech recognition, and time series forecasting.

The "long" and "short-term memory" parts refer to its ability to capture both short and long-term dependencies or relationships in the data.

Keeping the above points in mind, we performed LSTM and therefore it gives the best fit for some models as it is neither over-fitting nor under-fitting.

**LSTM for MAANG companies**

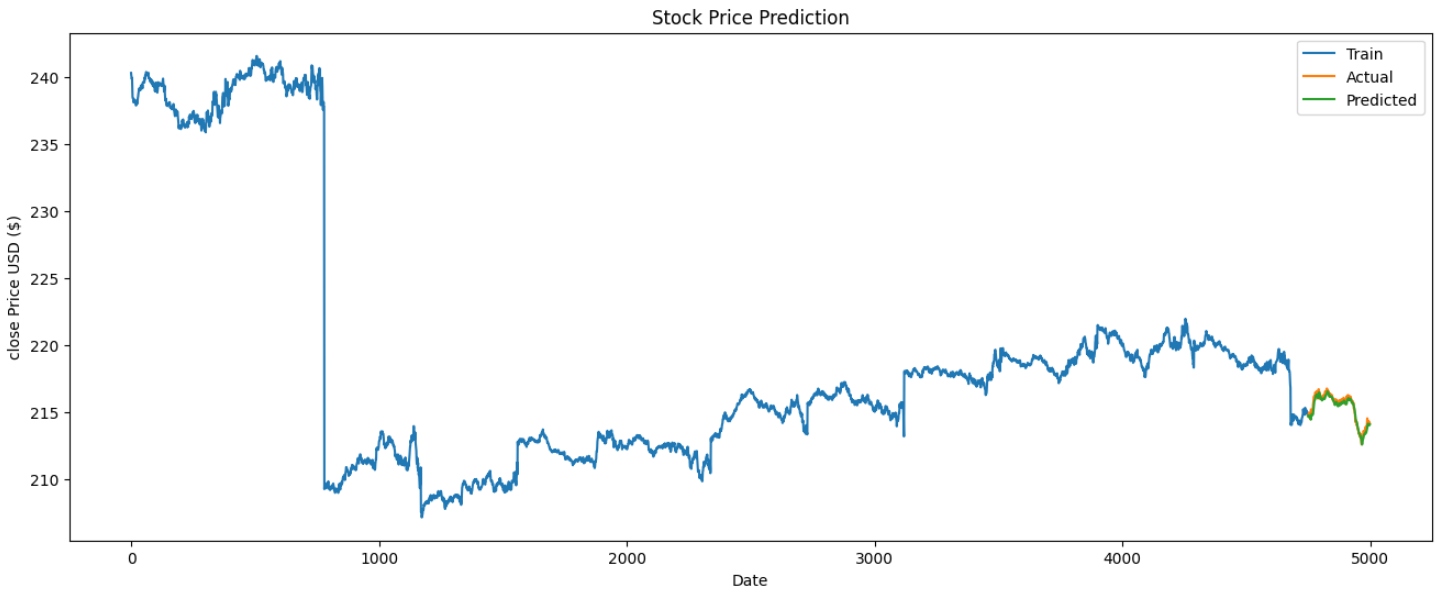
1. **META**

RMSE for META

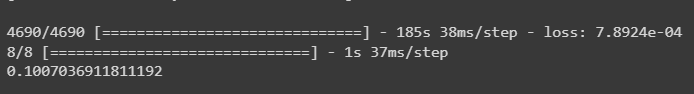
R Squared for META



Stock price prediction plot for META



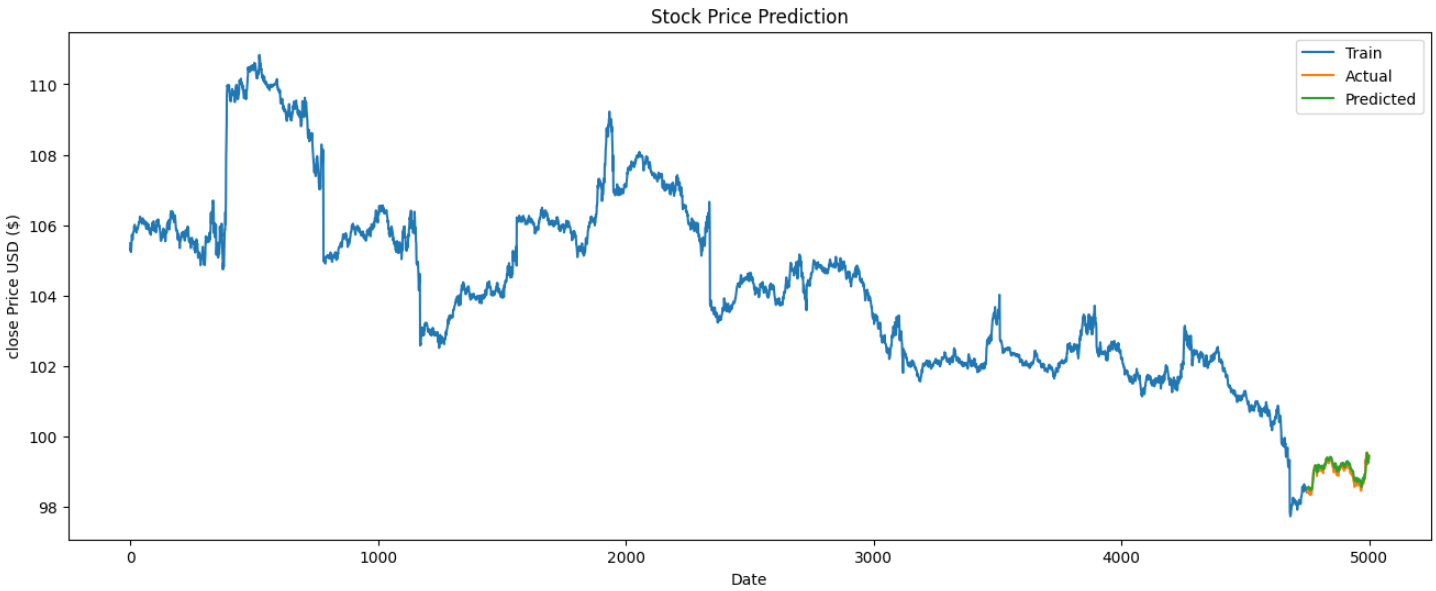
1. **AMAZON**

RMSE for Amazon

R squared for Amazon

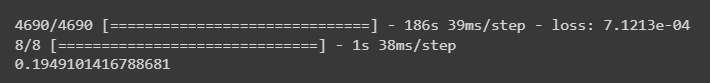


Stock price prediction plot for Amazon



1. **APPLE**

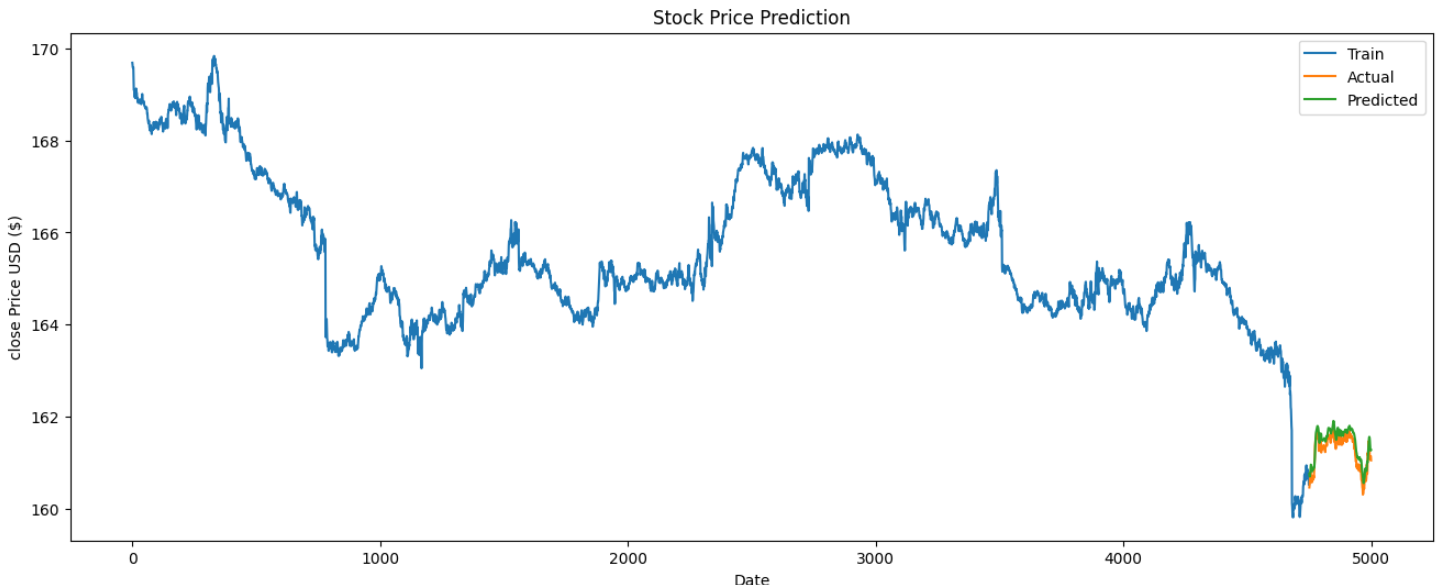
RMSE for Amazon

****

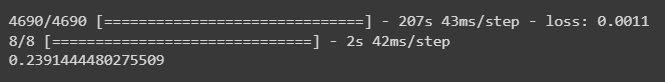
R squared for Amazon

****

Stock price prediction plot for Apple

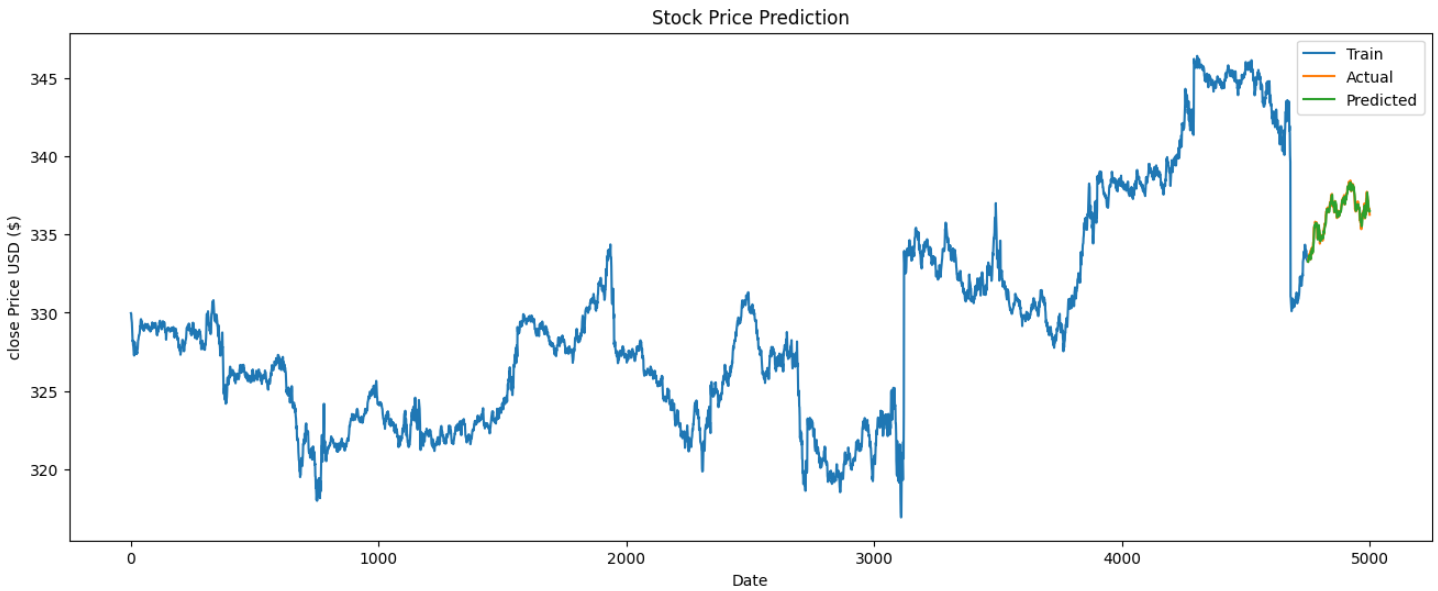
****

1. **NETFLIX**

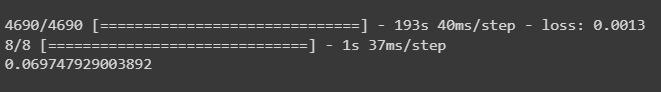
RMSE for Netflix****

R Squared for Netflix

****

Stock price prediction plot for Netflix****

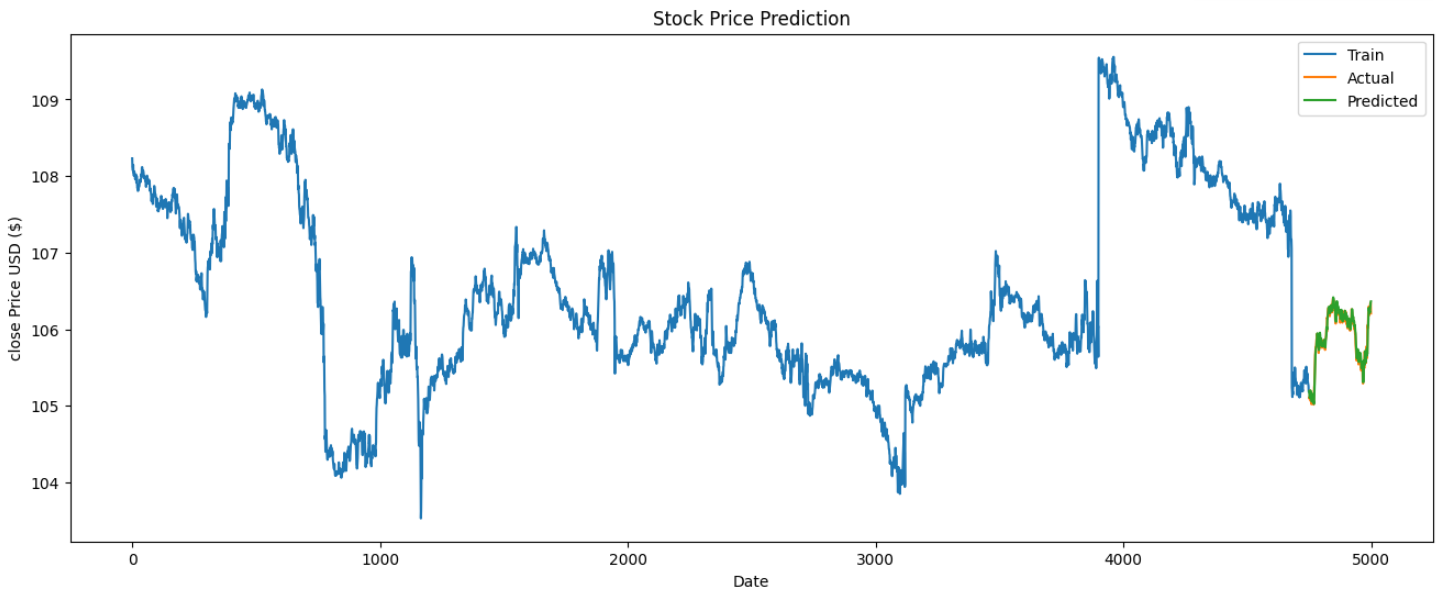
1. **GOOGLE**

RMSE for Google****

R squared for Google

****

Stock price prediction plot for Google

****

**RANDOM FOREST REGRESSION**

Random Forest is a machine learning algorithm that combines the power of decision trees and randomness to make accurate predictions. In Random Forest, a group of decision trees is created, each trained on a different subset of the data. Each decision tree independently makes its own prediction, and the final prediction is determined by aggregating the predictions of all the trees.

**RF for MAANG companies**

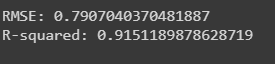
1. **META**

RMSE AND R SQUARED FOR META



1. **AMAZON**

RMSE and R Squared for Amazon

****

1. **APPLE**

RMSE AND R SQUARED FOR APPLE



1. **NETFLIX**

RMSE AND R SQUARED FOR APPLE



1. **GOOGLE**

RMSE AND R SQUARED FOR GOOGLE



**DECISION TREE**

Decision trees are a popular machine learning algorithm used for both classification and regression tasks. It is a graphical representation of a decision-making process that mimics how humans make decisions. It is a flowchart-like structure where each internal node represents a feature or attribute, each branch represents a decision or rule, and each leaf node represents an outcome or prediction.

**DT for MAANG companies**

1. **META**

RMSE and R Squared for META



1. **AMAZON**

RMSE and R Squared for Amazon

****

1. **APPLE**

RMSE and R Squared for Apple

****

1. **NETFLIX**

RMSE and R Squared for Netflix

****

1. **GOOGLE**

RMSE and R Squared for Netflix

****

**SIMPLE MOVING AVERAGE**

SMA stands for Simple Moving Average. It is a technical analysis indicator used to analyze and interpret price trends in financial markets, particularly in stocks, currencies, and commodities. SMA calculates the average price of an asset over a specific period by taking the sum of the closing prices for that period and dividing it by the number of data points. The resulting value represents the average price of the asset during that time frame.

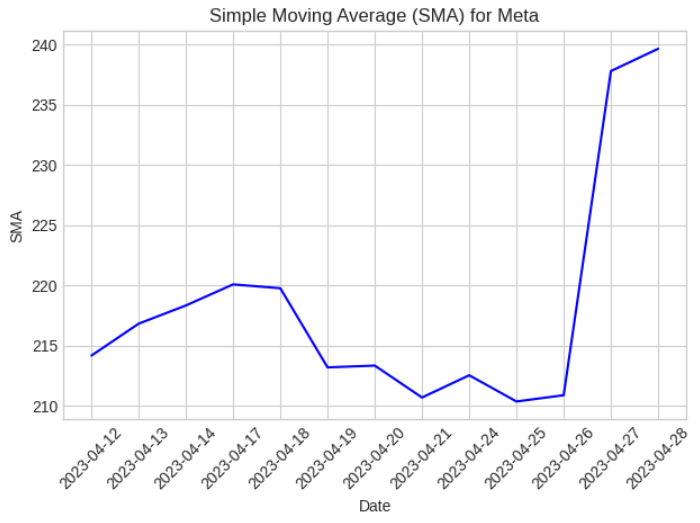
**SMA for MAANG companies**

1. **META**

RMSE and R Squared for META

****

Stock price prediction plot for META

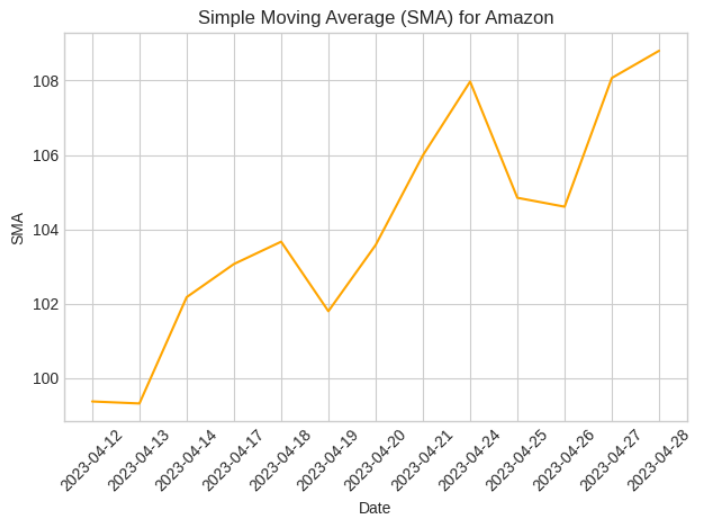
****

1. **AMAZON**

RMSE and R Squared for AMAZON

****

Stock price prediction plot for AMAZON

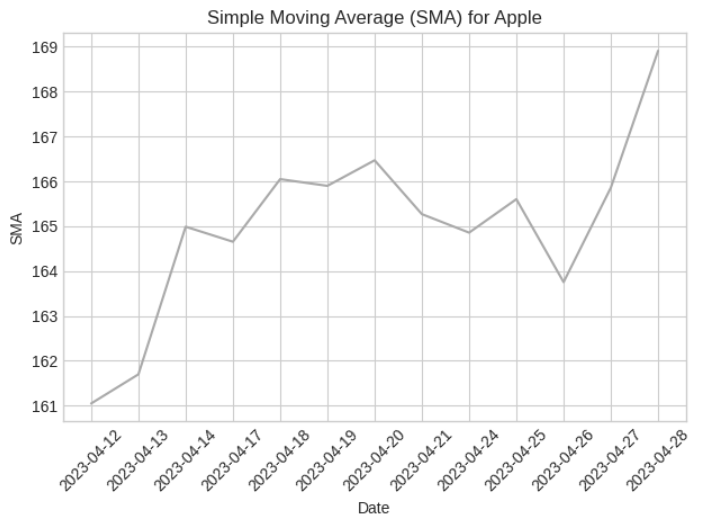
****

1. **APPLE**

RMSE and R Squared for Apple

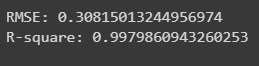
****

Stock price prediction plot for Apple

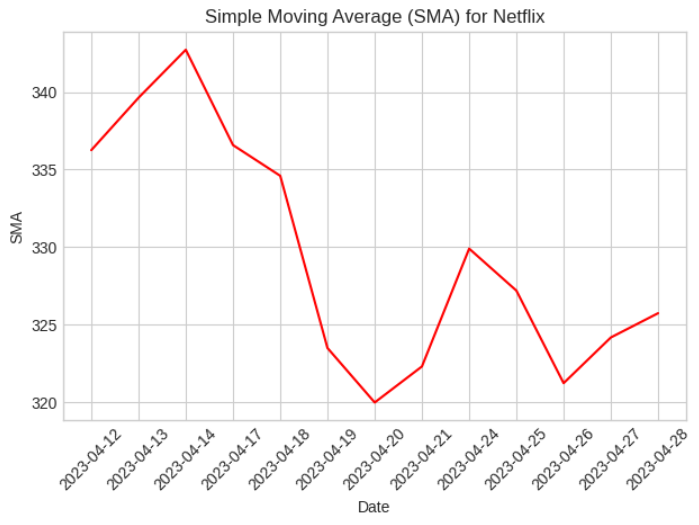
****

1. **NETFLIX**

RMSE and R Squared for Netflix

****

Stock price prediction plot for Netflix

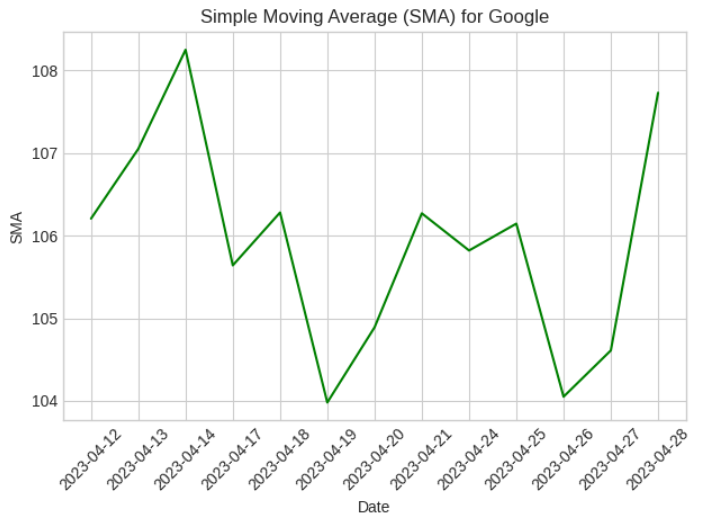
****

1. **GOOGLE**

RMSE and R Squared for Google

****

Stock price prediction plot for Google

****

**COMPARATIVE STUDY:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Linear Regression | Simple Moving Average | Random Forest | LSTM | Decision Trees |
| META | RMSE=0.2928  R^2=0.9768 | RMSE=0.1786  R^2=0.9996 | RMSE=1.5972  R^2=0.9703 | RMSE=0.2377  R^2=0.9494 | RMSE=0.615  R^2=0.9955 |
| AMAZON | RMSE=0.3104  R^2=0.9738 | RMSE=0.0992  R^2=0.99866 | RMSE= 1.5972  R^2=0.9703 | RMSE=0.1007  R^2=0.87183 | RMSE=0.176019  R^2=0.99582 |
| APPLE | RMSE=0.2646  R^2=0.9782 | RMSE=0.08298  R^2=0.9980 | RMSE=0.5498  R^2=0.9164 | RMSE=0.19491  R^2=0.7167 | RMSE=0.1255  R^2=0.9953 |
| NETFLIX | RMSE= 0.3125  R^2=0.9896 | RMSE=0.3081  R^2=0.9979 | RMSE=1.6900  R^2=0.93941 | RMSE=0.2391  R^2=0.9604 | RMSE=0.36604  R^2=0.997 |
| GOOGLE | RMSE=0.2825  R^2=0.9708 | RMSE=0.08018  R^2=0.9959 | RMSE=0.3268  R^2=0.9317 | RMSE=0.06974  R^2=0.9606 | RMSE=0.0839  R^2=0.9954 |

**INSIGHTS:**

* LSTM is neural network model, so LSTM turns out to be the best regression model for the MAANG companies.
* META: Other than LSTM, Linear Regression performs better than other regression models.
* AMAZON: In Amazon, SMA seems to perform better than all the other as it has low RMSE, and comaparatively average accuracy.
* APPLE: We noticed, that the R^2 value of random forest is lesser than other models, so the plot wont overfit.
* NETFLIX: In this, both LSTM and Random Forest works just fine, and better than the other three.
* GOOGLE: RMSE is the least in LSTM, but accuracy is the best in Random Forest, so if we want to compute errors, we will go LSTM model, and in the case of getting the best accuracy we work with Random Forest.

**K-means clustering**

K-means clustering is a centroid-based algorithm, where each cluster is associated with a centroid. This algorithm's primary goal is to reduce the total distances between each data point and its corresponding clusters.

The algorithm starts with an unlabeled dataset as its input, separates it into k clusters, and then continues the procedure until it runs out of clusters to use. In this algorithm, the value of k should be predetermined.

The two major functions of the k-means clustering algorithm are:

* uses an iterative technique to choose the best value for K centre points or centroids.
* each data point is matched with the nearest k-center. A cluster is formed by the data points that are close to a specific k-center.

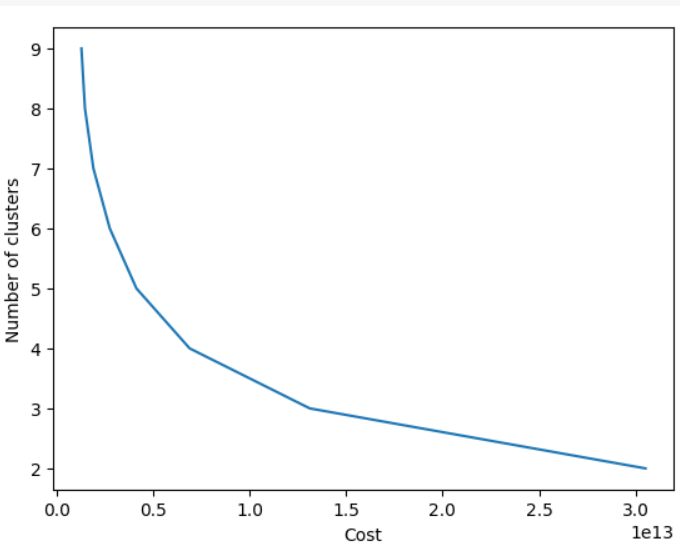
Hence each cluster has datapoints with some commonalities, and it is away from other clusters.

**K-means for MAANG companies**

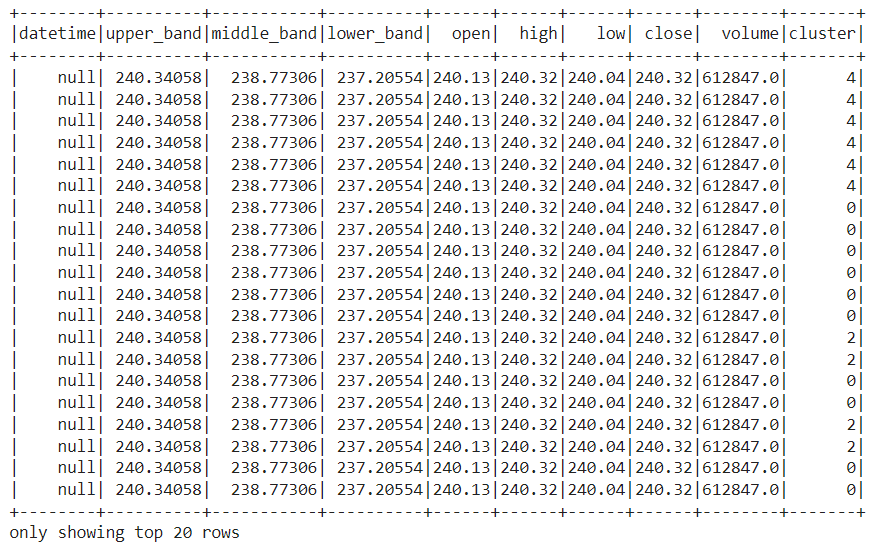
1. **META**

Plotted optimal number of clusters needed for META using ELBOW METHOD

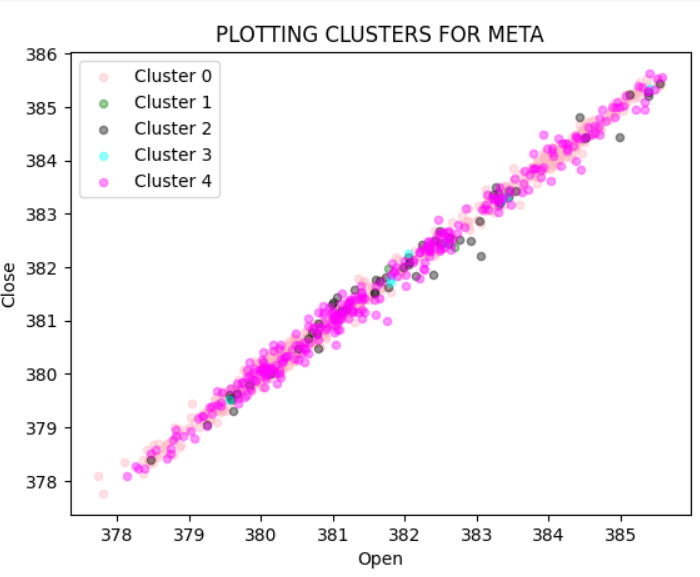
Sharp cuts shows the number of clusters



Clusters:

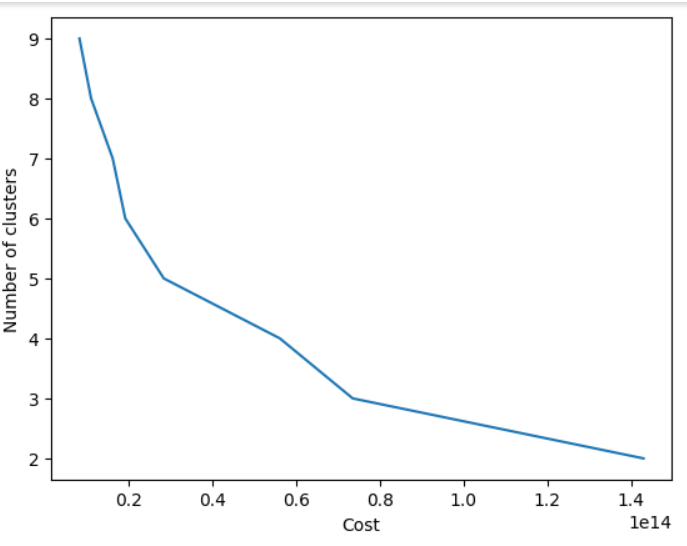


Plotting Scatter Plot for Open and Close

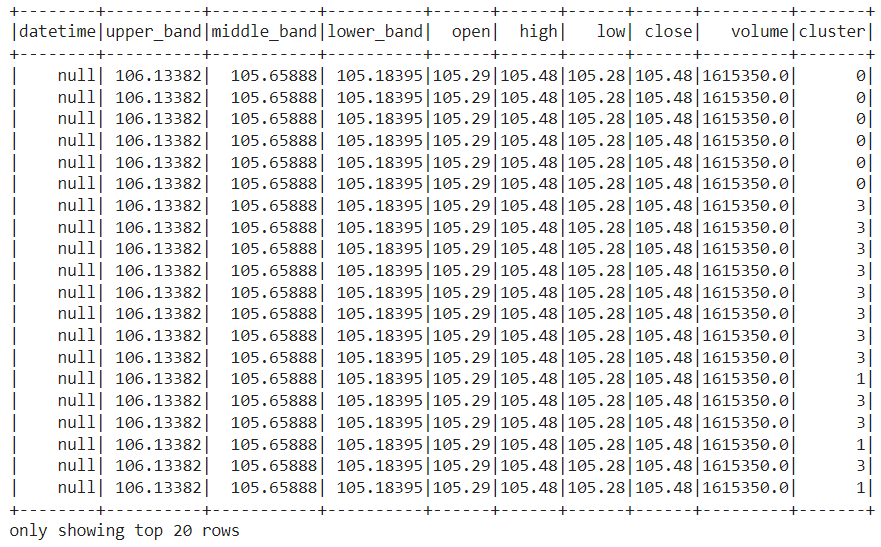


1. **AMAZON**

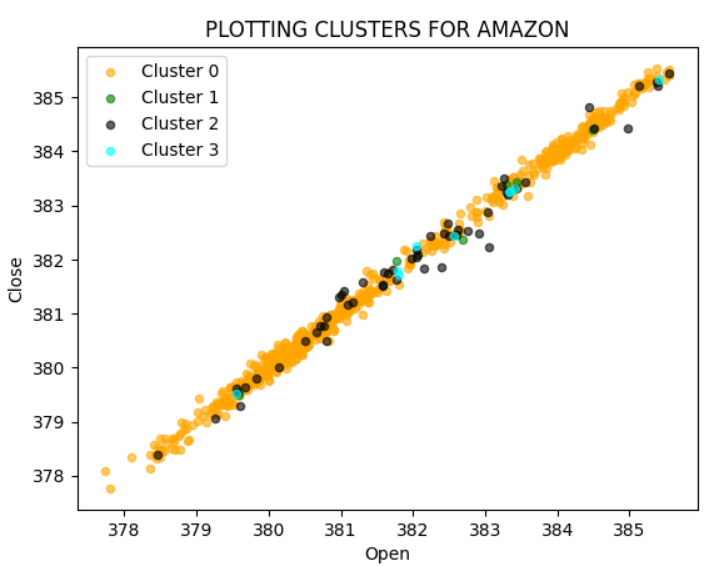
Plotted optimal number of clusters needed for META using ELBOW METHOD



Clusters:

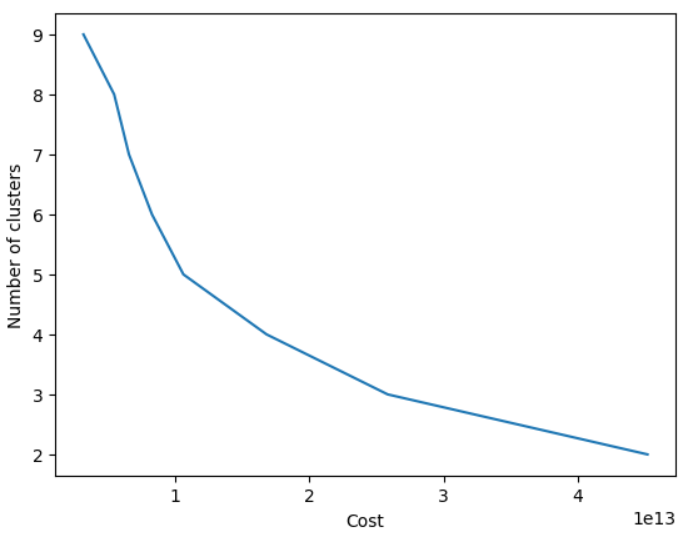


Plotting Scatter Plot for Open and Close

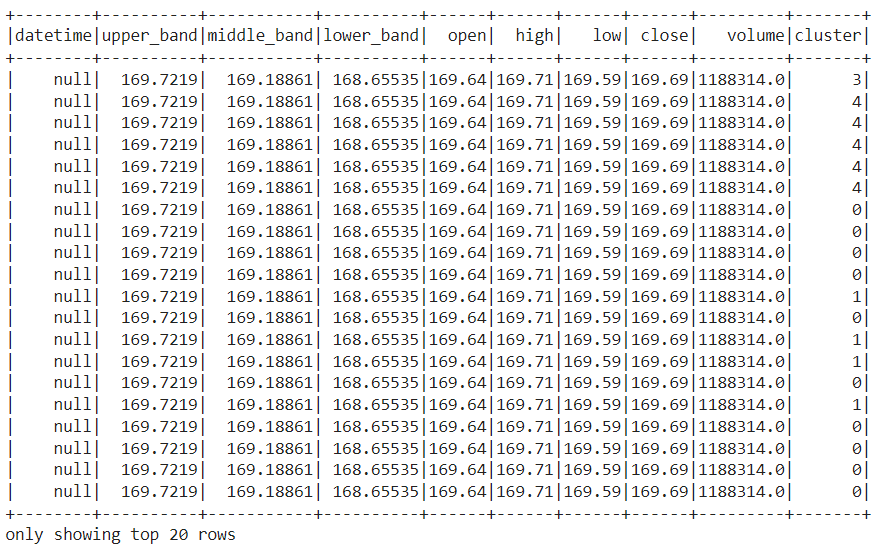


1. **APPLE**

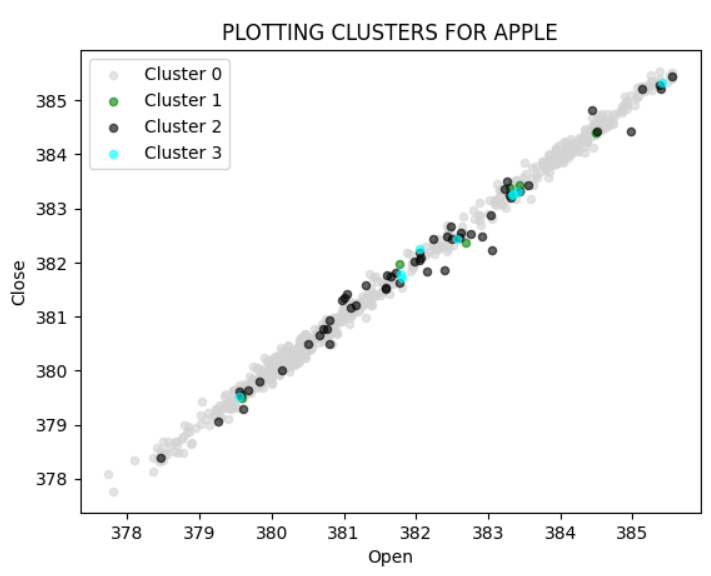
Plotted optimal number of clusters needed for META using ELBOW METHOD



Clusters:

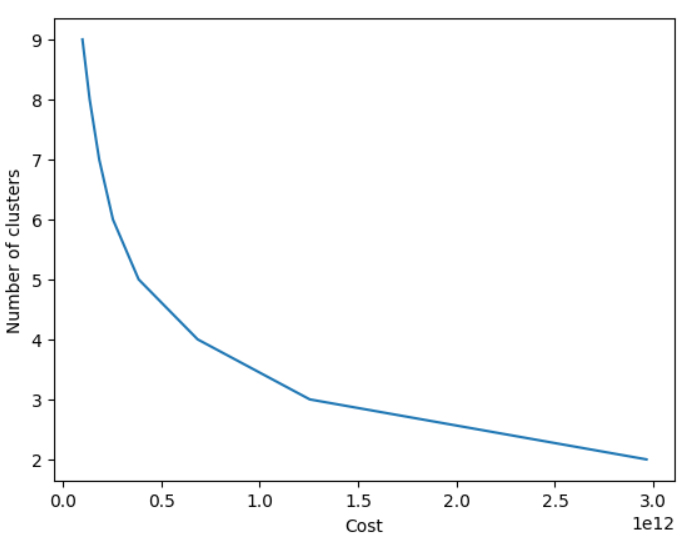


Plotting Scatter Plot for Open and Close

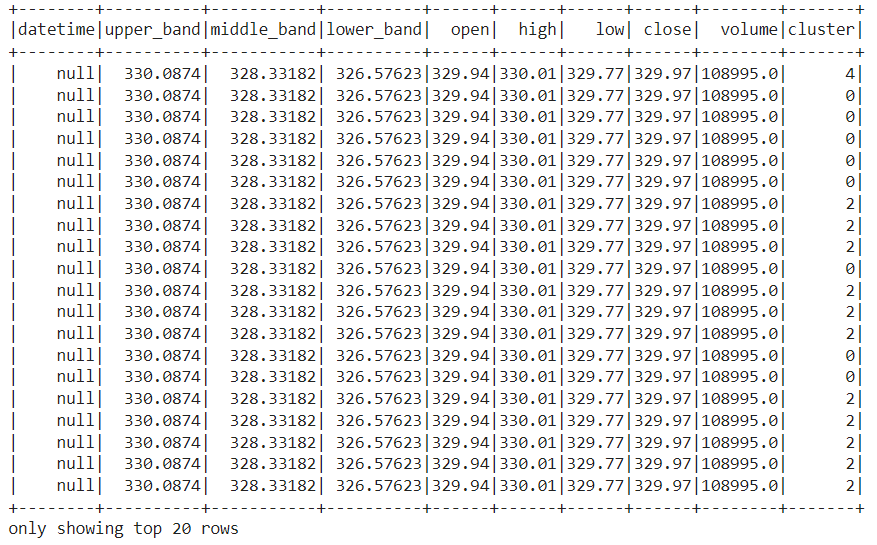


1. **NETFLIX**

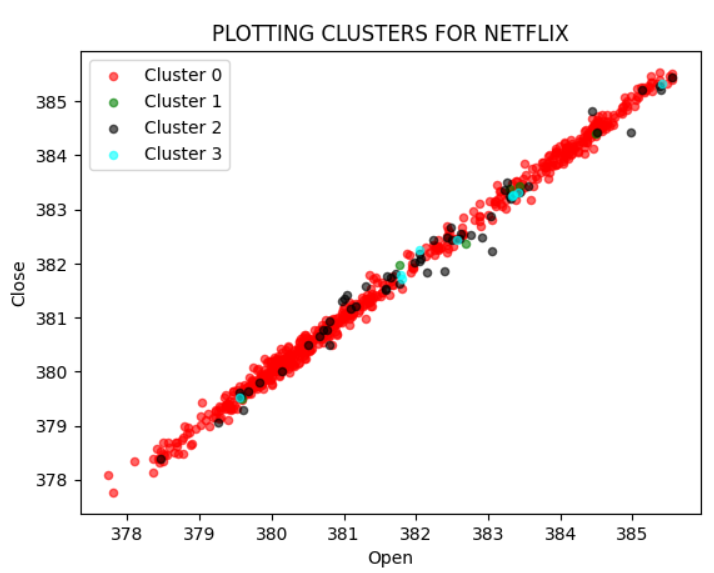
Plotted optimal number of clusters needed for META using ELBOW METHOD



Clusters:

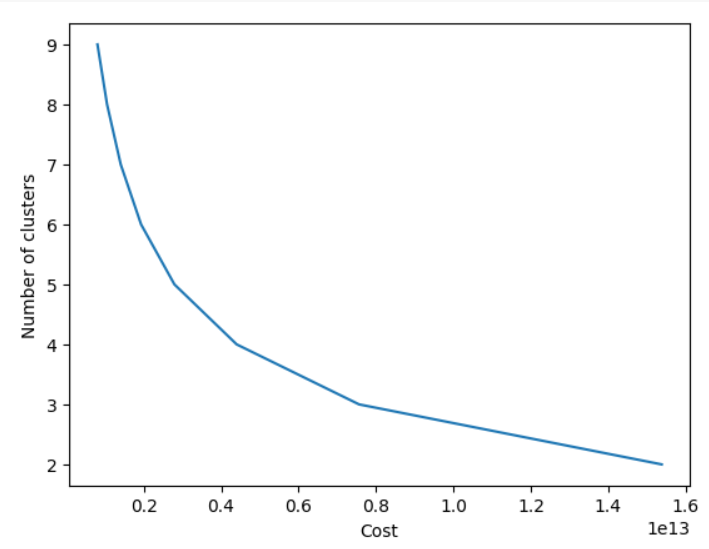


Plotting Scatter Plot for Open and Close

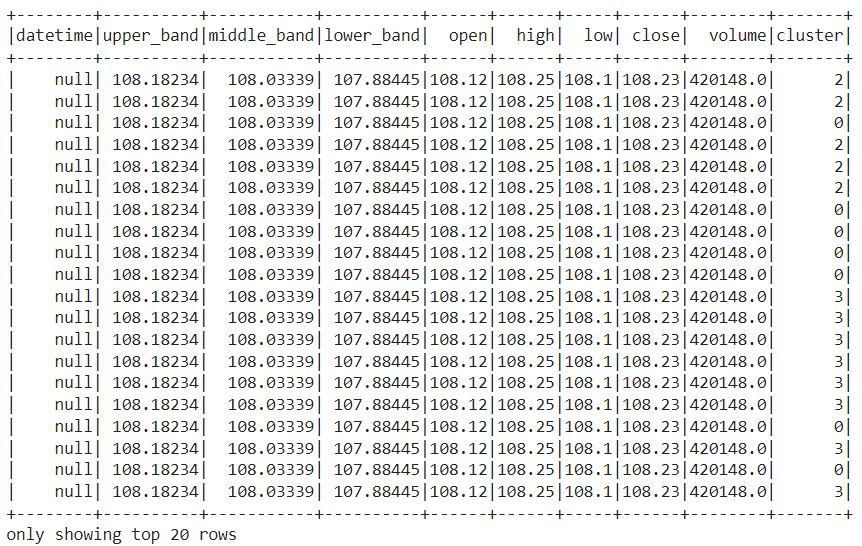


1. **GOOGLE**

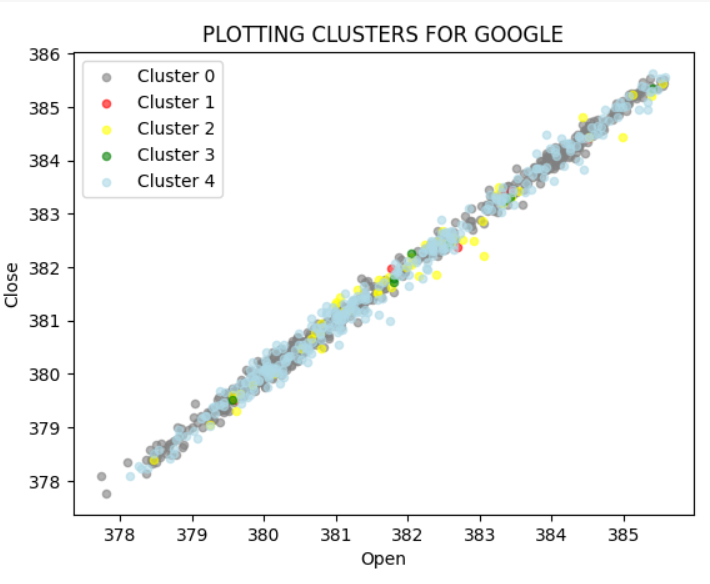
Plotted optimal number of clusters needed for META using ELBOW METHOD



Clusters:



Plotting Scatter Plot for Open and Close:



**USE CASES OF FINANCIAL ANALYSIS:**

Stock market financial analysis can help investors make better decisions about buying and selling stocks. Use cases of this analysis:

**Valuation**

Financial analysis can be used to figure out how much a company's stock is worth. Analysts look at the company's financial information and use different methods to estimate its true value.

**Industry analysis**

Financial analysis can be used to compare how well different companies in the same industry are doing. Investors look at financial information to see if there are trends in the industry that could affect their investments.

**Risk assessment**

Financial analysis can help investors understand how risky an investment is. Analysts look at things like how much debt a company has, how much money it has in the bank, and other financial information to see if the investment is safe.

**Portfolio optimization**

Financial analysis can help investors build a collection of different stocks that will give them good returns without being too risky. Investors use different tools to decide how much money to put into different stocks.

**Trading strategies**

Financial analysis can help investors decide when to buy or sell stocks. Investors look at historical stock prices and use different methods to find patterns that can help them make good decisions.

**CONCLUSION**

Real-time stock market data will be gathered, processed, and analysed by the system, which will also offer insights into market trends, volatility, and prospective investment opportunities. Kafka-based real-time data analysis of the stock market can enhance decision-making, efficiency, risk management, and customer satisfaction. This approach can be used by financial institutions to give their clients real-time updates on their investments, improving the client experience overall.