META STOCK PRICE PREDICTION ANALYSIS

A project work made under the guidance of **Vigor Council**

Submitted by:

Kirti

Data Analyst Internship @ Vigor Council

June, 2024

ACKNOWLEDGEMENT

I would like to express our heartfelt gratitude to our supervisor, **Dr. B.P. Sharma**, for his invaluable guidance and support throughout this project. This project helped us to explore and undertake research work and gaining practical knowledge and expertise to perform tasks. We also extend our thanks to the **Vigor Council** for providing the necessary resources and facilities.

Thanking You

Kirti

DECLARATION

I, **Kirti**, hereby declare that the project entitled "Meta Stock Price Prediction Analysis" is a result of our original research work carried out under the guidance and supervision of **Dr. B.P. Sharma**, **President** at **Vigor Council**.

This project work is undertaken as part of our **internship as Data Analysts** and is submitted to **Vigor Council**. We affirm that the research and findings presented in this project are genuine. All sources of information and data have been acknowledged appropriately. We also declare that any help received in carrying out this project and preparing the report has been duly acknowledged.

ABSTRACT

The prediction of stock prices has always been a challenging task due to its dynamic nature and the influence of various factors such as market sentiment, economic indicators, and company performance. This study focuses on predicting the stock prices of Meta platforms, leveraging historical stock price data and advanced machine learning algorithms.

Our approach begins with data collection, where I gather historical stock prices and relevant financial indicators of Meta. This data is then processed to handle missing values.

The core of my analysis involves the implementation of various machine learning models, including Linear Regression, Random Forest, XGBoost, Support Vector Regressor, and KNN. These models are evaluated based on their predictive performance, using metrics such as Mean Squared Error (MSE), Mean Absolute Error (MEA), and R-Squared.

This study highlights the potential of machine learning techniques in stock prices prediction and provides a comprehensive analysis of their effectiveness. The results can aid investors and financial analysts in making informed decisions and developing robust trading strategies.

INTRODUCTION

The stock market is a complex, dynamic system influenced by numerous factors, making the prediction of stock prices a formidable challenge. Accurate stock price prediction is highly valuable to investors, traders, and financial analysts as it aids in making informed investment decisions, optimizing portfolios, and managing risks.

This study focuses on predicting the stock prices of Meta platforms, Inc., a leading technology known for its social media platforms and technological innovations.

Meta, as a publicly traded company, experiences stock price fluctuations driven by variety of factors, including company performance, market trends, economic indicators, and investor sentiment. Given its significant market capitalization and influence, predicting Meta's stock prices is not only relevant for individual investors but also for institutional stakeholders.

Traditional financial models, while useful, often fall short in capturing the non-linear and complex patterns inherent in stock market data. The advent of machine learning and advanced computational techniques offers new avenues for improving the accuracy of stock price predictions. Machine learning models can process large volumes of data, identify hidden patterns, and adapt to new information, making them well-suited for stock market analysis.

RESEARCH OBJECTIVE

The primary goal of this project is to develop a robust and accurate predictive model for forecasting the stock prices of Meta platforms. To achieve this, the study will address the following specific research objectives:

1. Data Collection and Preprocessing:

- Gather historical stock price data and relevant financial indicators for Meta.
- Clean and preprocess the data to handle missing values, outliers, and normalize features for model readiness.

2. Model Development and Implementation:

- Implement various machine learning algorithms including Linear Regression, Random Forest, XGBoost, SVR, KNN.

3. Model Evaluation:

- Evaluate the performance of the predictive models using metrics such Mean Absolute Error, Mean Squared Error, and R-squared.
- Compare the models to identify which one provides the best accuracy and robustness in predicting stock prices.

4. Model Interpretation and Insights:

- Interpret the results of the predictive models to understand the key drivers and patterns influencing Meta's stock prices.

5. Visualization and Reporting:

- Visualize the predicted vs. actual stock prices to illustrate model performance.

The project aims to contribute valuable insights into the use of machine learning techniques for stock price prediction and provide a practical tool for forecasting the stock prices of Meta.

METHODOLOGY

DATA COLLECTION

I took this dataset from "Kaggel.com" in csv format. It contains 633 rows and 7 columns. It's a historical dataset as shown in the below:

Date	Open	High	Low	Close	Adj Close	Volume
28-10-2021	312.98999	325.519989	308.11	316.92	316.5841	50806800
29-10-2021	320.190002	326	319.6	323.57	323.2271	37059400
01-11-2021	326.040009	333.450012	326	329.98	329.6303	31518900
02-11-2021	331.380005	334.790009	323.8	328.08	327.7323	28353000
03-11-2021	327.48999	332.149994	323.2	331.62	331.2685	20786500
04-11-2021	334.01001	337.269989	332.65	335.85	335.494	22495300
05-11-2021	340.309998	346.790009	339.64	341.13	340.7684	26872800
08-11-2021	344.420013	344.790009	338.34	338.62	338.2611	18342500
09-11-2021	340	341.309998	334.47	335.37	335.0145	17556700
10-11-2021	332.48999	333.190002	325.51	327.64	327.2928	21872600
11-11-2021	329.820007	332.459991	327	327.74	327.3926	12376600
12-11-2021	330.179993	341.859985	329.78	340.89	340.5287	25414300
15-11-2021	344.339996	353.649994	343.2	347.56	347.1916	25076600
16-11-2021	343.829987	346.649994	340.87	342.96	342.5965	18181100
17-11-2021	344.23999	347.299988	340.1	340.77	340.4088	13602800
18-11-2021	339.720001	342.459991	335.3	338.69	338.331	17487200
19-11-2021	342.200012	352.100006	339.9	345.3	344.934	26488500

DATA PREPARATION

Later the data was uploaded in jupyter notebook, I clean the data and there is no null values in the dataset. I set Date column as index of the dataset.

DETAILED INFORMATION OF COLUMNS OF DATASET

- **Date**: The date of the stock data.
- **Open**: The opening price of the stock.
- **High**: The highest price of the stock during the day.
- **Low**: The lowest price of the stock during the day.
- **Close**: The closing price of the stock.
- **Adj Close**: The adjusted closing price of the stock.
- **Volume**: The trading volume of the stock.

<u>EXPLORATORY DATA ANALYSIS (EDA)</u>

Exploratory Data Analysis (EDA) is a critical step in the data analysis process, involving the thorough examination of a dataset to uncover underlying patterns, spot anomalies, test hypotheses, and check assumptions through visual and quantitative methods. In this project, EDA is employed to analyse the survey data collected on users' preferences and experiences with online and offline coaching.

This includes data cleaning and pre-processing to handle missing values and outliers, followed by the use of descriptive statistics and visualizations such as histograms, bar charts, and box plots to summarize the demographic distribution and general perceptions of coaching among respondents. Through EDA, we aim to identify key trends and insights, such as the demographic factors that influence coaching preferences and the common reasons cited for favouring one mode over the other. This analysis not only helps in understanding the data better but also guides the feature selection and engineering process for the subsequent classification models. EDA provides a foundation for making informed decisions throughout the project, ensuring that the final models are built on a solid understanding of the underlying data patterns.

Time Series Graph

A time series graph depicts data points in a sequential order, typically over equal intervals of time. It is used to visualize trends, patterns, and fluctuations over time, allowing for the analysis of how variables change. Here's what a time series graph typically includes:

1. **Time on the X-Axis**: The horizontal axis represents time intervals (e.g., days, months, years).
2. **Data Points on the Y-Axis**: The vertical axis represents the values of the variable being measured (e.g., stock prices figures).
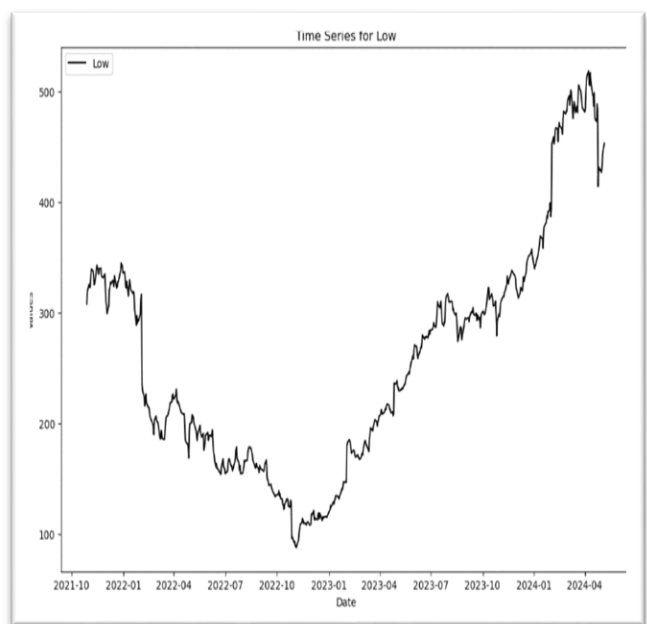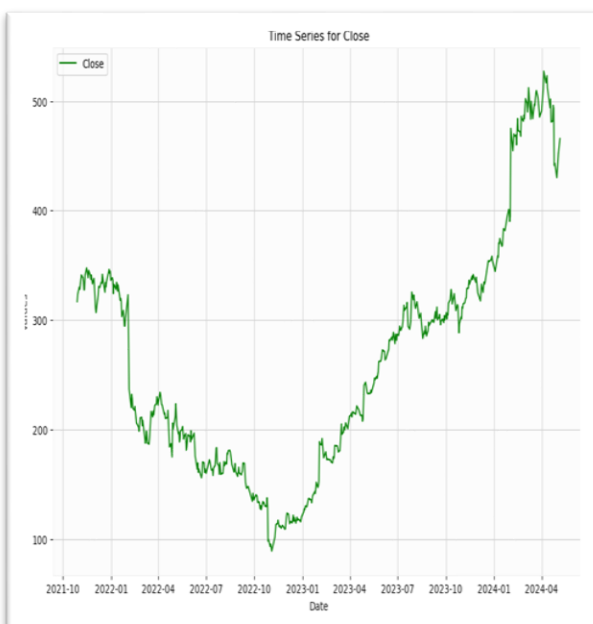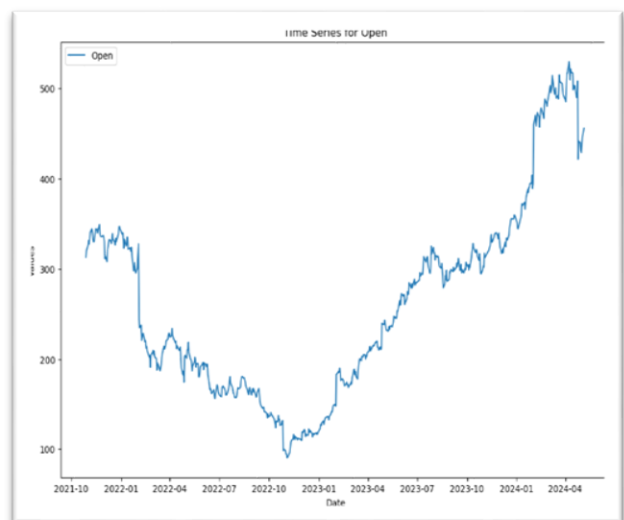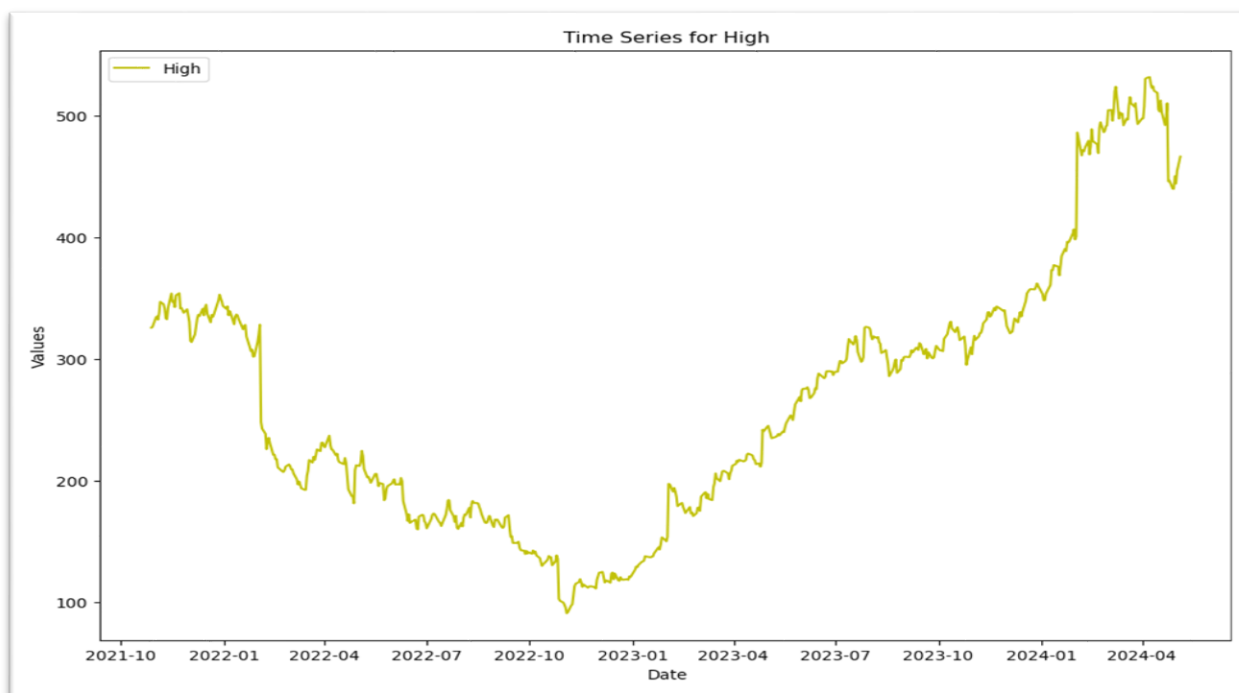
Candle Stick Plot

The 'mpf.plot' function from the 'mplfinance' library in Python is used to create candlestick, line, and other types of financial plots, specifically designed for visualizing financial time series data.

Candlestick Plots: Visualize open, high, low, and close prices for a given time period.

Distribution Plot

It combines a histogram with a kernel density estimate (KDE) or a fitted probability density function (PDF) to provide a comprehensive view of the distribution.

 Histogram: A bar plot representing the frequency of data points in each bin.
 KDE Plot: A smoothed line representing the estimated probability density function of the variable.
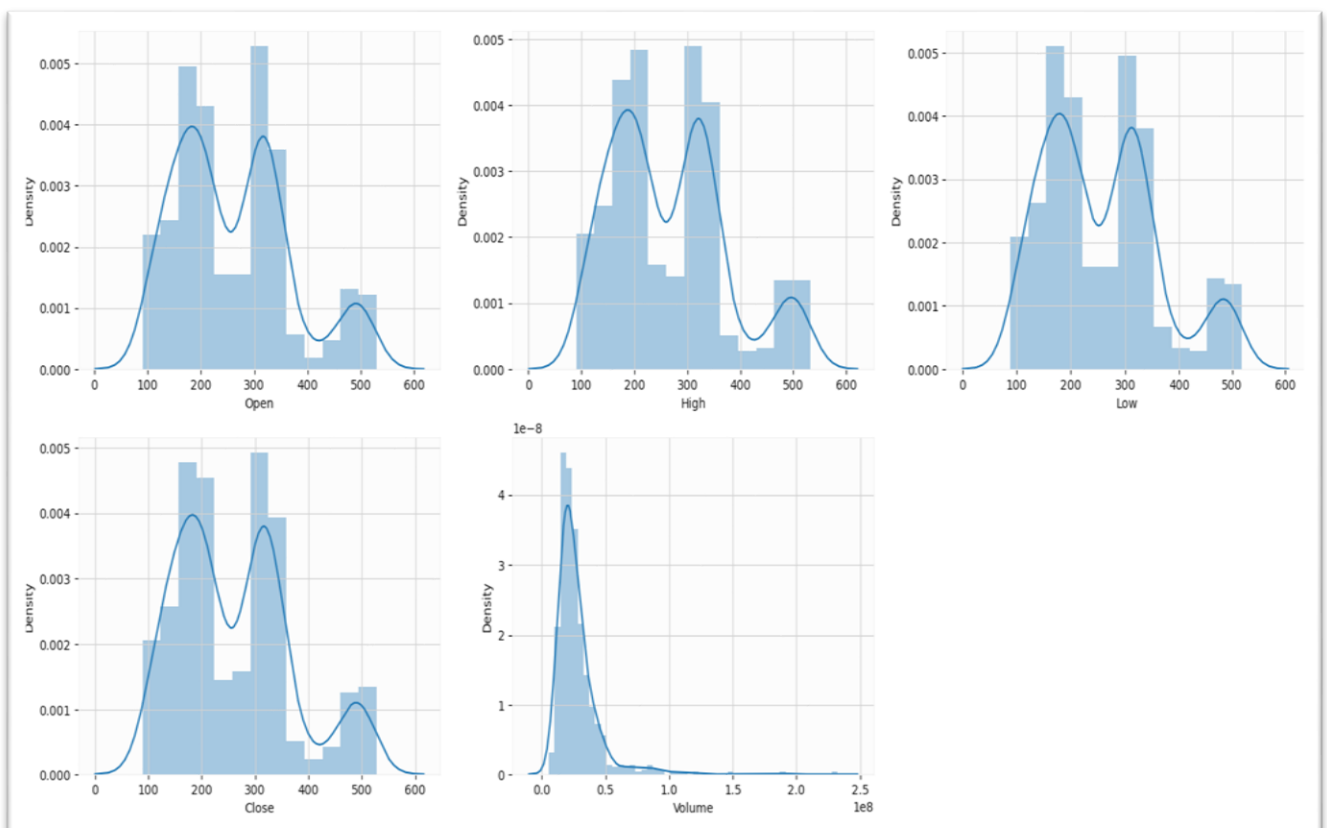
Correlation Matrix

A correlation matrix is a table showing the correlation coefficients between many variables. Each cell in the table shows the correlation between two variables. The value is in the range of -1 to 1. If the variables have a high positive correlation, the value is closer to 1. If they have a high negative correlation, the value is closer to -1. A value of 0 indicates no correlation.
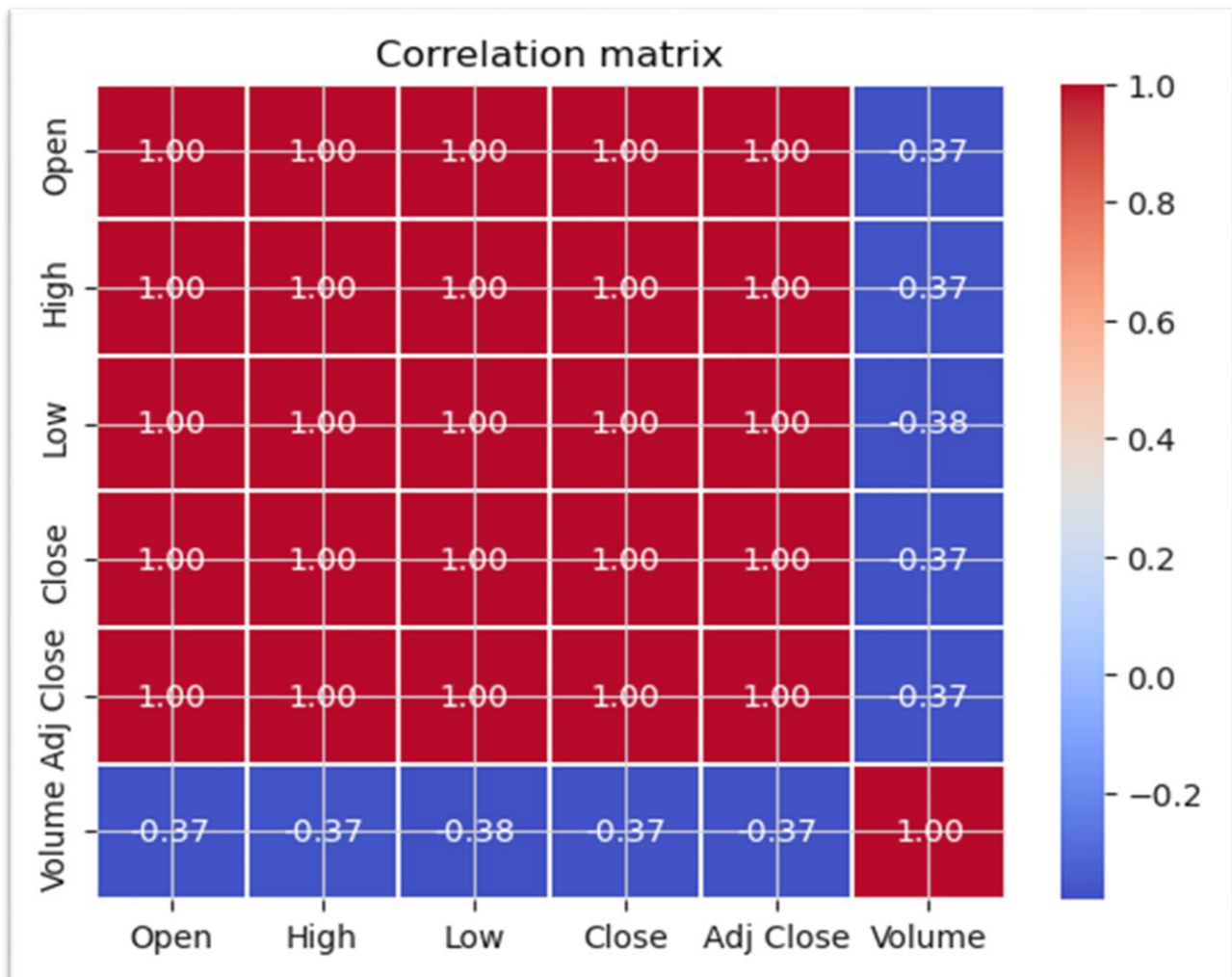
In this correlation matrix "Volume" column have the highly negative correlation with other variables.

Moving Average Plot

A moving average plot is a visualization that helps smooth out short-term fluctuations in data and highlight longer-term trends. It is especially useful in time series analysis, such as stock prices, to better understand the underlying patterns without the noise of daily volatility.
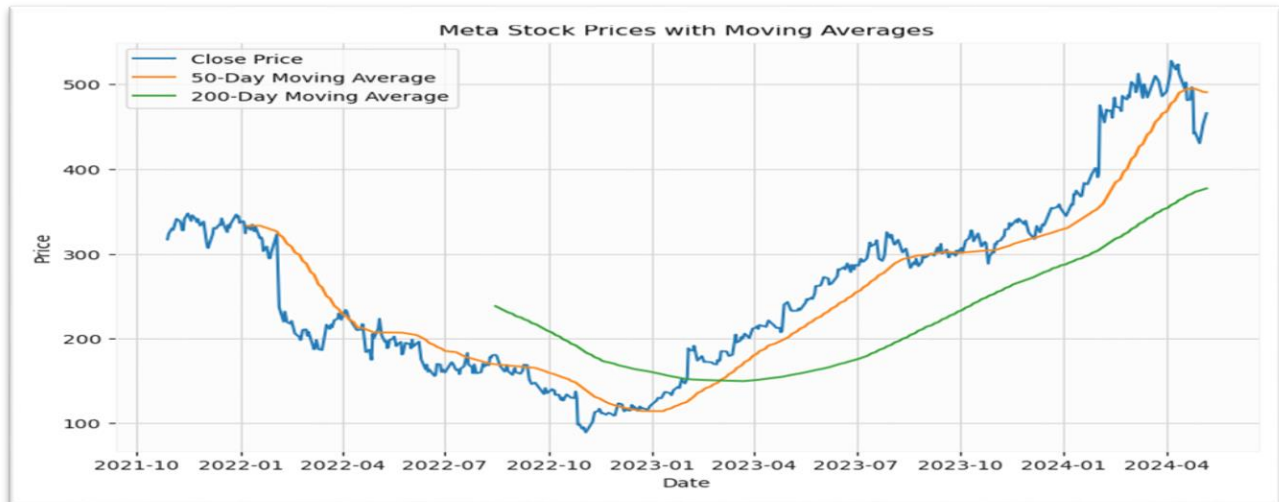
Downtrend (Early 2022):

- The closing price is declining, and the 50-day moving average is below the 200-day moving average.
- This period shows a downward trend, possibly indicating that the stock was underperforming during this time.

Consolidation (Mid 2022 to Early 2023):

- The stock price fluctuates, and both moving averages appear to stabilize.
- This phase might indicate a consolidation period where the stock price is neither strongly rising nor falling.

Uptrend (From Early 2023):

- The closing price starts to rise sharply, and the 50-day moving average crosses above the 200-day moving average (potential golden cross).
- This indicates a strong upward trend, suggesting that the stock was performing well during this period.

Recent Correction (Early 2024):

- There's a noticeable dip in the closing price.
- The 50-day moving average shows a downward turn while the 200-day moving average remains stable but upward-sloping, indicating a potential short-term correction in a longer-term uptrend.

<u>REGRESSION ALGORITHMS</u>

Regression algorithms are a fundamental part of machine learning, used for predicting continuous outcomes. In my project Meta stock price prediction, I employ various regression algorithms to analyse and predict outcomes based on the historical data.

The use of multiple regression algorithms allows us to compare their performance and ensure the robustness of our findings.

LINEAR REGRESSION

Linear regression is a fundamental statistical and machine learning technique used to model the relationship between a dependent variable (also called the target or response variable) and one or more independent variables (also called features or predictors). The goal of linear regression is to find the best-fitting straight line through the data points that minimizes the sum of the squared differences between the observed values and the values predicted by the model.

XGBOOST REGRESSION

XGBoost is a machine learning algorithm that belongs to ensemble learning category, specifically the gradient boosting framework. It utilizes decision tress as base learners and employs regularization techniques to enhance model generalization. XGBoost is famous for its computational efficiency, offering efficient processing, insightful feature importance analysis and seamless handle of missing values.

XGBoost is a powerful approach for building supervised regression models. The validity of this statement can be inferred by knowing about its objective function and base learners. The objective function contains loss function and regularization term. It tells about the difference between actual and predicted values, i.e. how far the model results are from the real values.

SUPPORT VECTOR REGRESSION

Support Vector Regression (SVR) is a type of machine learning algorithm used for regression analysis. The goal of SVR is to find a function that approximates the relationship between the input variables and a continuous target variable, while minimizing the prediction error.

Unlike Support Vector Machines (SVMs) used for classification tasks, SVR seeks to find a hyperplane that best fits the data points in a continuous space. This is achieved by mapping the input variables to a high-dimensional feature space and finding the hyperplane that maximizes the margin (distance) between the hyperplane and the closest data points, while also minimizing the prediction error.

RANDOM FOREST REGRESSION

Random forest is a supervised learning algorithm, meaning that the data on which it operates contains labels or outcomes. It works by creating many decision trees, each built on randomly chosen subsets of the data. The model then aggregates the outputs of all of these decision trees to make an overall prediction for unseen data points. In this way, it can process larger datasets and capture more complex associations than individual decision trees.

K-NEAREST NEIGHBORS REGRESSION

K-Nearest Neighbours (KNN) is a non-parametric machine learning algorithm that can be used for both classification and regression tasks. In the context of regression, KNN is often referred to as "K-Nearest Neighbours Regression" or "KNN Regression." It's a simple and intuitive algorithm that makes predictions by finding the K nearest data points to a given input and averaging their target values (for numerical regression) or selecting the majority class (for classification).
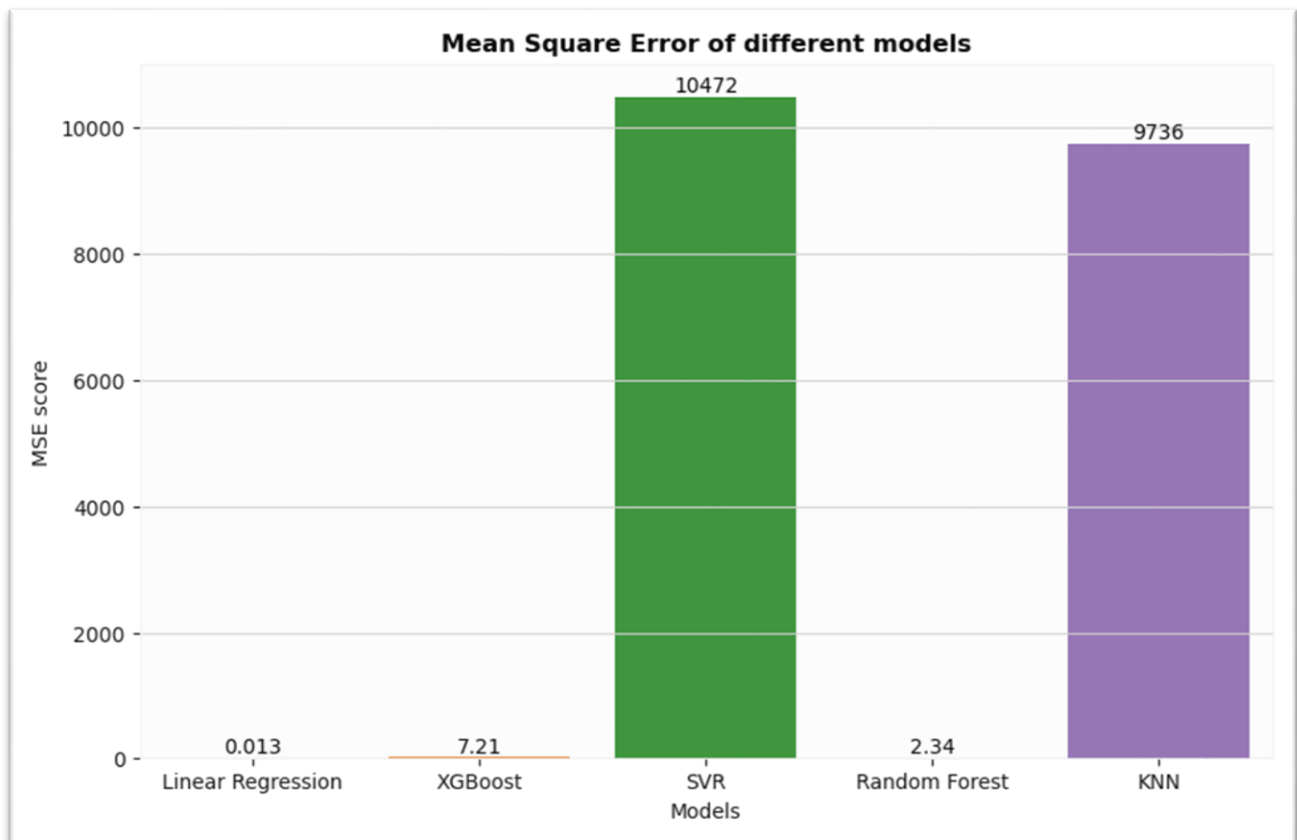
MODEL DEVELOPMENT

- I have considered 'Adj Close 'price as a dependent variable.
- I then split the data using Sklearn library into Training data (80% of dataset) and Test data (20% of dataset).

REGRESSION REPORT

Adj Close Prediction with XGBoost

R² of different models

Mean Square Error of different models
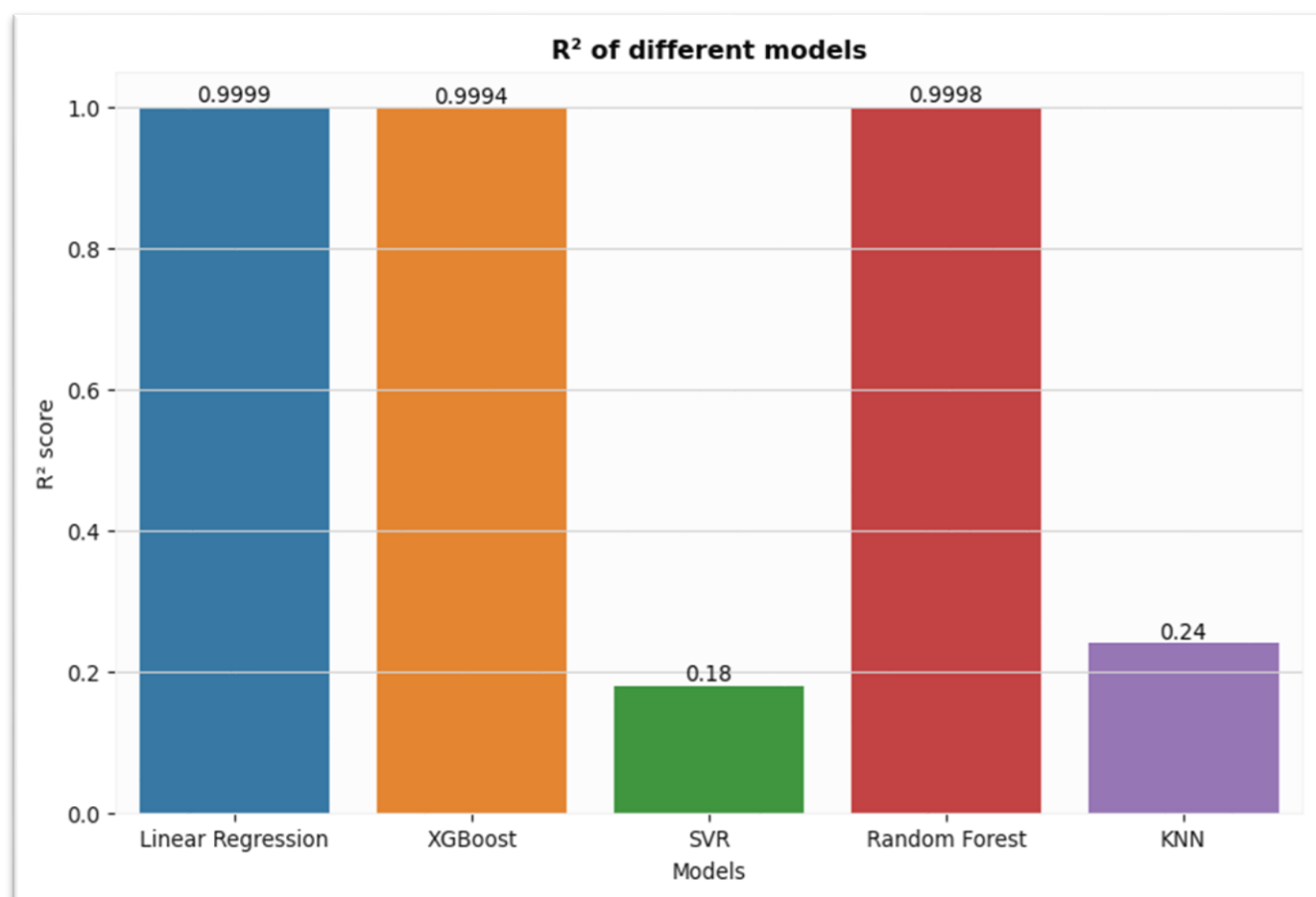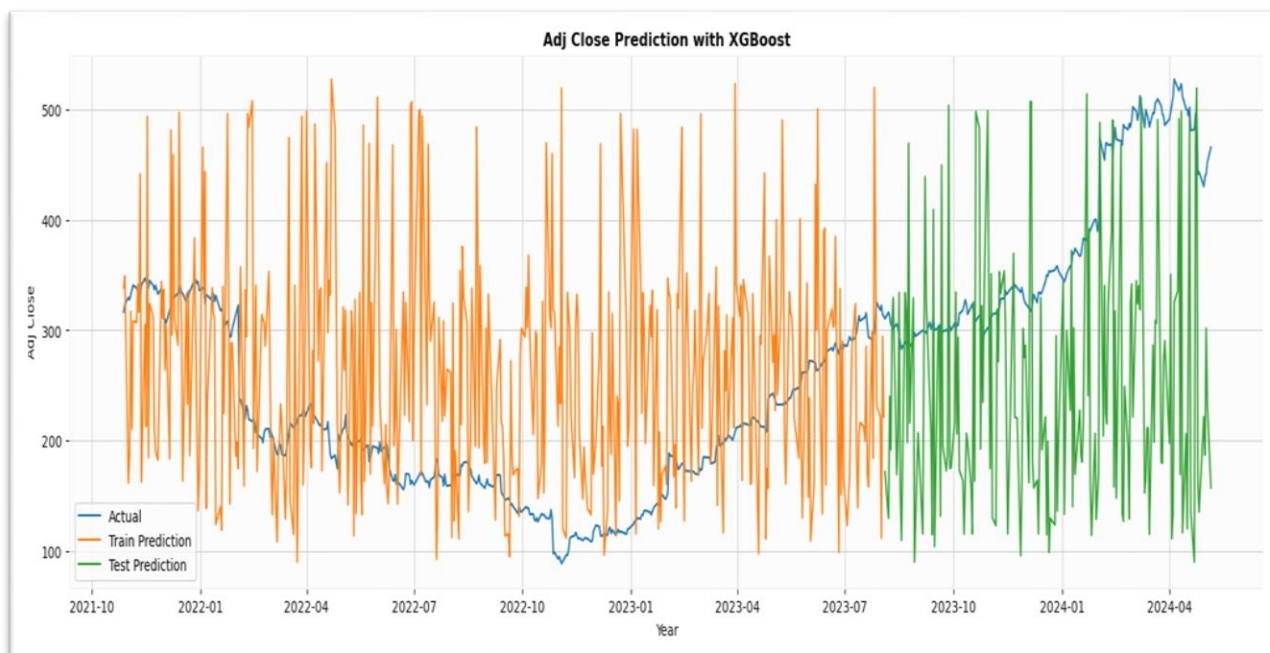
<u>INTERPRETATIONS</u>

After analysing the performance of five regression algorithms—Linear Regression, Support Vector Regression (SVR), Random Forest, XGBoost, and KNN we can draw the following conclusions:

- R-squared is a statistical measure that represents the proportion of the variance for the dependent variable that's explained by the independent variables in the model
- MSE is the average of the squared differences between the actual and predicted values. A lower MSE indicates better model performance

Linear Regression:

R-squared = 0.999, means that 99.9% of the variance in the dependent variable is explained by the model. The model fits the data extremely well.

Mean-square-error = 0.013, is very low, indicating that the model's predictions are very close to the actual values.

XGBoost Regression:

R-squared = 0.9994, means that 99.4% of the variance in the dependent variable is explained by the model. The model fits well.

Mean-squared-error = 7.21, is quite high, indicating that the model prediction are less accurate.

Support Vector Regression:

R-squared = 0.18, means that 18% of the variance in dependent variable is explained by the model. This relatively low, suggesting that the model is not capturing much of the underlying relationship between the independent variables and the dependent variable.

Mean-squared-error = 10472, is extremely very high, suggests that the model's predictions are far off from the actual values.

Random Forest:

R-squared = 0.9998, means that 99.98% of the variance in dependent variable is explained by the model. The model fits the data extremely well.

Mean-squared-error = 2.34. This means that the predictions made by the Random Forest model are quite close to the actual values.

K-Nearest Neighbor:

R-squared = 0.24, means that 24% of the variance in dependent variable is explained by the model. This relatively low, suggesting that the model is not capturing much of the underlying relationship between the independent variables and the dependent variable.

Mean-squared-error = 9736, is extremely very high, suggests that the model's predictions are far off from the actual values.

CONCLUSION

In this project, we aimed to predict Meta's (formerly Facebook) stock prices using various machine learning models. We explored different regression techniques, including Linear Regression, XGBoost, SVR, and Random Forest, to determine which model provides the most accurate predictions.

☐ The Linear Regression model showed the best performance in terms of both R-squared and MSE, suggesting that the relationship between the features and the target variable (Meta's stock prices) was largely linear.

☐ The Random Forest model, while not outperforming Linear Regression, still provided reasonably accurate predictions with an MSE of 2.34.

BIBLIOGRAPHY

https://www.google.com/search?sca_esv=4d1091d9e59bc338&rlz=1C1VDKB_enIN1089IN1089&sxsrf=ADLYWIKejCIXg_0f03stCZ6RbIrYKOnCPQ:1717952123587&q=vigor+council&uds=ADvngMj53AHIJt2KifqzVbYVXLyJYJFmr_CSCIdb8JLCdoJXd57e7CUzevkpTrMtPqwDiQ_FLQDXteKYUYmctl-_4YZuwu5BRzMjAUcXiE4K52bE4HoB1hA_fbdiUB2j4d-ICeWyb2Is7muE6D2NIeLsbpMWJD3RgvNEOqlfbRbMsXYY3afenD85DJLkh31d0TINdcxI-C3QOVvkzYtJ9orvIcdtsY-EMN83ik8rqQfhKHigpWYCu7bBNrfHeZRvbhPlWiBcYfsDTcTK7TxMDV_AJwAPs00YX73u4YSPtQOa2RQDncYmp_JtmCVIVKJe2ejxlWKbjkhs&udm=2&prmd=insvmbtz&sa=X&ved=2ahUKEwis3sSw_s6GAxVfTGwGHXmWAkIQtKgLegQIDBAB&biw=1536&bih=695&dpr=1.25#vhid=qNjXSQ53ju_-CM&vssid=mosaic

https://www.google.com/search?sca_esv=23275d10bb72eb94&rlz=1C1VDKB_enIN1089IN1089&sxsrf=ADLYWIKwPISsV3k-vWhix1mkS7oVqJjnYw:1717957414942&q=meta+stock+symbol&uds=ADvngMgi54_t_sBvBWQ1IdPXuRRdtbMz83tR9Ao6LgjamWVx7euUJuo9CEnfV0ShW5f13jdQHyJ7i0zBOyuNxVbaBQEZx71yR9PrbrRZdsvOQhRes0TL_QOD0RosdpWBtDkyNuM2f5csAsOotVGuhoApz7BZbhc7ADZPwVCxTa56a_N3MrZQyu6PeZmEoJBISwS4uxDWbH53mGp2wmlV44NUNykVgGTFh7oVIE-2Vdy-0spfGv1SjOuH0-q9KaEwtE4rof1HowyqqHqpd78HRpV9sLHN7SOoEANDAr2h1_0FyWQfDXH04akqqkd65GABjX7cia_6ofhe0ml6a6th_o4mUFzJQ2wt4Q&udm=2&prmd=zinvsbmt&sa=X&ved=2ahUKEwjWv9OLks-GAxXeS2wGHUrKBSIQtKgLegQICxAB&biw=1536&bih=695&dpr=1.25

https://www.geeksforgeeks.org/xgboost-for-regression/

https://www.analyticsvidhya.com/blog/2020/03/support-vector-regression-tutorial-for-machine-learning/#:~:text=Support%20Vector%20Regression%20(SVR)%20is,while%20minimizing%20the%20prediction%20error.

https://www.analytixlabs.co.in/blog/random-forest-regression/

https://medium.com/@nandiniverma78988/understanding-k-nearest-neighbors-knn-regression-in-machine-learning-c751a7cf516c