Predicting Stock Prices: A Comparative Analysis of Time Series and Regression-Based Approaches

By Team-08

Srichand Mendagani (EF69050)
Shailesh Kumar Shetty (GT32214)
Jo Young (GM45575)
Aakash Shoraan (TE92013)
Sree Rashmika Talagampala (IK13932)
Anand Yenigalla (WK99013)

ABSTRACT

Predicting Stock Prices: A Comparative Analysis of Time Series and Regression-Based Approaches

In the modern era, buying and selling stocks is important for economic growth, wealth creation and contribution to the economic growth of big businesses. Forecasting stock prices is important for informed investment decision making, risk management, and optimizing trading strategies. It allows investors to anticipate market trends, helps diversify portfolios, and increases profitable investments.

This work examines the effectiveness of predictive models on a multi-decade dataset of Google stock prices. Our initial focus was to conduct exploratory data analysis (EDA) to understand the characteristics of the dataset. We found that the data were not stable and successfully converted them into stable series by differentiation using the SARIMAX model. Then, we use the SARIMAX model that incorporates exogenous variables to predict the final price. The SARIMAX model demonstrated remarkable accuracy, with a Mean Squared Error (MSE) of about 2×10^-18 and an R-squared value close to 1. After the time series prediction, we turned to regression analysis. We selected the object using the connectivity matrix, then trained the Support Vector Machine (SVM) and Random Forest (RF) models. Both models performed well, with the SVM achieving an average MSE of 2.448 and an R^2 of 0.97, while the RF model showed an average MSE of 1.84 and an R^2 of 0.98. Our comparative analysis showed that the SARIMAX model with exogenous variables is a robust method for predicting Google stock price. Additionally, to understand the definition of price ups and downs, we included technical indicators such as simple moving average (SMA), relative strength index (RSI), and Bollinger Band statistics.

Keywords: SARIMAX, SVM, RF, Time Series Analysis, Regression Analysis, Exogenous Variables, SMA, RSI, Bollinger Bands, EDA, Stationarity, Feature Selection.

INDEX

1. **Introduction**
   1.1. Stock prices
   1.2. Seasonality and Residuals in Stock prices
   1.3. Basis of Stock Price Prediction and Research Question
   1.4. Block Diagram of the project

2. **Literature Review & Relevant Theory**
   2.1. Introduction
   2.2. Overview of Time series Models
   2.3. Theory of SARIMAX Model
   2.4. Regression Analysis: SVM and RF

3. **Methodology & Proposed Prediction Models**
   3.1. Description of Dataset
   3.2. EDA and Technical Indicators
   3.3. SARIMAX Model
   3.4. Feature Selection Using Correlation
   3.5. SVM Model
   3.6. RF Model

4. **Results & Discussion**
   4.1. Visualization of predicted prices vs actual prices
   4.2. Evaluation Metrics
   4.3. Comparison

5. **Epilogue**
   5.1. Conclusion
   5.2. Further Scope of the Project

6. **Appendix**
   6.1. Project Code Description

References

CHAPTER-1
INTRODUCTION

1.1 Stock Prices

Stock prices represent the current market price of a share of a company’s stock. They are an integral part of the stock market and are influenced by a variety of factors such as a company’s finances, investor sentiment, market conditions, and broader economic indicators. The price of these shares fluctuates throughout the trading day depending on the level of supply and demand. If more people are willing to buy stock (demand) than sell stock (supply), the price goes up. Conversely, if more people are willing to sell than buy the stock, its price falls. Stock price matters to the company and its investors. A higher stock price of a company can mean higher market valuations, which can be beneficial for raising capital, attracting talent, and expanding their businesses. Stock prices are important to investors as they determine the potential gain or loss on their investments. Investors buy stocks that are expected to increase in value over time, allowing them to sell the shares for more than they are worth, so they realize a profit Stock price is a key indicator of a company’s market value and investors, analysts and the company personally look out for signs of performance and future potential.

1.2 Seasonality and Residuals in Stock prices

In the context of stock price analysis, seasonality and residuals are key components that are often studied to understand and predict market behavior.

1. Seasonality in Stock Prices:

Seasonality refers to predictable and regular movements or patterns in stock prices that occur at certain times of the year. This could be due to various factors like quarterly financial reports, industry-specific cycles, holiday seasons, or tax-related deadlines. For example, retail stocks might show a seasonal pattern by performing better during holiday seasons due to increased consumer spending. Seasonality analysis is important for short-term trading strategies and for understanding periodic fluctuations in stock prices.

2. Residual Analysis in Stock Prices:

In time series analysis, after removing trends and seasonality from a dataset, what remains are the residuals. These are essentially the deviations of the observed values from the predicted values that cannot be explained by the trend or seasonal components. Residual analysis helps in validating the effectiveness of a model. Ideally, residuals should be random and show no specific pattern. Non-random residuals indicate that the model might be missing some explanatory information or variables. Analyzing residuals is crucial for improving model accuracy and ensuring that the model captures all relevant patterns in the data. Understanding and analyzing trends, seasonality, and residuals are fundamental aspects of stock price analysis. They help in building robust predictive models and in making strategic investment decisions.

1.3 Basis of stock price prediction and Research Question

The basis of stock price prediction involves using various analytical methods to forecast future movements of stock prices. This process is complex due to the dynamic and often unpredictable nature of financial markets. Stock price prediction typically relies on the following key approaches:

1. Fundamental Analysis:

This approach focuses on analyzing a company's financial statements, operating efficiencies, business conditions, and macroeconomic factors to examine its financial health and internal standards Key measures include revenue, revenues, profits, and other financial metrics. The purpose of fundamental analysis is to determine whether a stock is undervalued or based on its intrinsic value.

2. Technical Analysis:

Technical analysis involves studying past market data, primarily price and volume, to forecast future stock price movements. It includes the use of charts, patterns, and various technical indicators like moving averages, Relative Strength Index (RSI), Bollinger Bands, etc. The underlying assumption is that historical trading activity and price changes are indicators of future price movements.

3. Quantitative Analysis:

Quantitative analysis uses mathematical and statistical models to predict stock prices. This approach often involves complex algorithms and machine learning techniques. It can include time-series analysis, regression models, neural networks, and other advanced methods. Quantitative models often process large datasets, including historical stock prices, financial indicators, and even alternative data like social media sentiment.

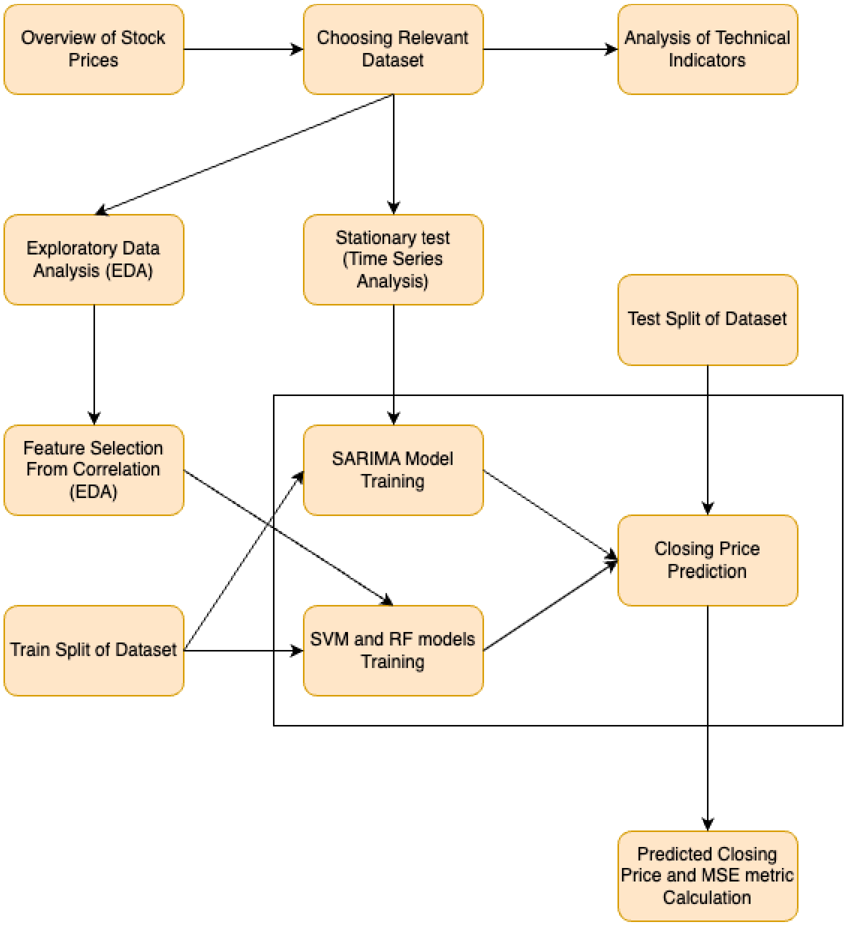
4. Macroeconomic Factors Analysis (Economic indicators):

Global economic conditions, interest rates, inflation, political stability, and other macroeconomic factors can influence stock prices. Investors often monitor these indicators to predict how they will affect different sectors and companies.

Predicting stock prices accurately is challenging due to market volatility and the multitude of factors that can influence prices. Most strategies involve a combination of these approaches to increase the accuracy and reliability of predictions.

Research Question: Our research questions developed during the project. Initially focusing on predicting stock price closings based on historical data, we later refined our questions to explore how different machine learning models perform in this context. These improvements allowed us to go deeper into the comparative analysis of SARIMA, SVM, and Random Forest. We also researched technical indicators of the stock price.

1.4 Block Diagram of the project


Overview of Stock Prices
Choosing Relevant Dataset
Analysis of Technical Indicators
Exploratory Data Analysis (EDA)
Stationary test (Time Series Analysis)
Test Split of Dataset
Feature Selection From Correlation (EDA)
SARIMA Model Training
Closing Price Prediction
Train Split of Dataset
SVM and RF models Training
Predicted Closing Price and MSE metric Calculation

CHAPTER-2
LITERATURE REVIEW AND RELEVANT THEORY

2.1 Introduction

The key concepts from the literature review we have done about stock price prediction:
LSTM (Long Short-Term Memory) models are effective in capturing complex temporal dynamics in stock prices, surpassing traditional linear models like ARIMA in handling nonlinear patterns. ARIMA (AutoRegressive Integrated Moving Average) models, while less capable in non-linear environments, remain robust and efficient for short-term forecasting in stock markets. Feature selection, particularly through methods like SVM-RFE (Support Vector Machine-Recursive Feature Elimination) and RF-RFE (Random Forest-Recursive Feature Elimination), is crucial for improving the accuracy of trend predictions in stock trading. These techniques help in identifying the most relevant features, enhancing model performance. The integration of advanced machine learning techniques in stock price prediction signifies a shift towards more data-driven approaches, capable of adapting to complex market behaviors. These concepts underscore the evolving nature of financial modeling, where machine learning and feature selection play pivotal roles in enhancing predictive accuracy. We also referred to a project which used the LSTM approach in which they got pretty good results with the same dataset that we had chosen. We tried other methods to improve the results of building a robust model for stock price prediction.

2.2 Overview of Time series Models

Time series models are vital in statistical analysis for understanding and forecasting patterns over time, and vary in complexity. The Autoregressive (AR) model predicts based on past values, assuming a linear influence, while the Moving Average (MA) model focuses on the relationship with residuals of past observations. For more complex data, the Autoregressive Moving Average (ARMA) model combines AR and MA, suitable for stationary series. For non-stationary data, the Autoregressive Integrated Moving Average (ARIMA) model, which includes data differencing, is used. The Seasonal ARIMA (SARIMA) adds seasonality handling. Vector Autoregression (VAR) suits multiple interrelated series, analyzing economic variables. Exponential Smoothing models smooth out trends and seasonality. State Space models and the Kalman Filter dynamically approach time series analysis, and are effective in noisy data situations. Each model's applicability depends on data characteristics and analysis requirements. Overall, the choice of a time series model depends on the data's characteristics and the specific requirements of the analysis. Each model offers different advantages and is suited to different kinds of time series data and forecasting needs.

2.3 Theory of SARIMA Model

The Seasonal Autoregressive Integrated Moving Average (SARIMA) model is an advanced statistical approach used for forecasting time series data that exhibits seasonality. It is an extension of the ARIMA model, adding the capability to account for seasonal variations in the data. The SARIMA model captures both the trend and seasonality in data, making it particularly useful for analyses where these elements are pronounced, such as in economic, climatological, or retail sales data. In technical terms, the SARIMA model is denoted as SARIMA(p,d,q)(P,D,Q)[s]. Here, p, d, q are the non-seasonal components of the model, similar to an ARIMA model: p represents the order of the autoregressive part, d is the degree of differencing needed to make the series stationary, and q is the order of the moving average part. The seasonal elements P, D, Q correspond to these but apply to the seasonal components of the time series, with P representing the order of the seasonal autoregressive part, D indicating the degree of seasonal differencing required, and Q being the order of the seasonal moving average part. The s parameter denotes the seasonality period (e.g., 12 for monthly data with annual seasonality). This comprehensive structure allows the SARIMA model to effectively identify, estimate, and incorporate both non-seasonal and seasonal dynamics in the data, providing accurate forecasts for time series with complex patterns.

2.4 Regression Analysis: SVM and RF

Support Vector Regression (SVR) and Random Forest Regression (RFR) are advanced regression techniques widely used in machine learning. SVR, an extension of Support Vector Machines (SVM), is particularly effective for regression tasks. It works by finding a function that best fits the relationship between input variables and continuous output values, aiming for minimal deviation from actual outputs within a defined threshold (ε). SVR is adept at dealing with non-linear relationships through the use of kernel functions like linear, polynomial, and radial basis function (RBF), which transform the input data into a higher-dimensional space. This flexibility, coupled with its ability to manage high-dimensional data, makes SVR a robust choice for complex regression tasks.

Random Forest Regression, on the other hand, is an ensemble method that builds multiple decision trees during training and outputs the mean or median of their predictions. This approach leverages the power of multiple learning models, significantly reducing the risk of overfitting which is common in individual decision trees. Each tree in the forest is constructed using a random subset of features from a random sample of the data, enhancing the model's ability to handle large feature sets and its resilience against outliers and missing values. The aggregation of predictions from diverse trees leads to a more stable and accurate performance, making Random Forest Regression suitable for datasets with complex structures and interactions. While it may lack the interpretability of simpler models, its robustness and effectiveness in capturing intricate patterns make it a valuable tool in regression analysis.

SVR involves finding a linear function (or nonlinear, using kernels) that best fits the data within a certain margin of error, and this is achieved by solving an optimization problem. Random Forest regression, on the other hand, aggregates the predictions of multiple decision trees, each constructed on a random subset of data and features, to make the final prediction.

CHAPTER-3
METHODOLOGY AND PROPOSED PREDICTION MODELS

3.1 Description of Dataset

This data set has 7 columns with all the necessary values such as the date, opening price of the stock, closing price of it, highest in the day, lowest price of it, volume of it and adjusted close of it. It is a comprehensive dataset containing historical daily stock data for Alphabet Inc. (GOOG) spanning from 2013 to January 2023. This type of dataset is valuable for analyzing the performance of the stock over time and conducting various types of financial analysis.

Column Name	Description
Date	The date of the stock price entry, formatted as "YYYY/MM/DD".
Open	The price of Google stock at the opening of the trading day.
High	The highest price of Google stock during the trading day.
Low	The lowest price of Google stock on the trading day.
Close	The price of Google stock at the closing of the trading day.
Adj Close	Adjusted Closing Price, the closing price adjusted for any corporate actions like dividends or stock splits.
Volume	The number of shares of Google stock traded during the day.
Unnamed	An index or serial number for each row, starting from 0.

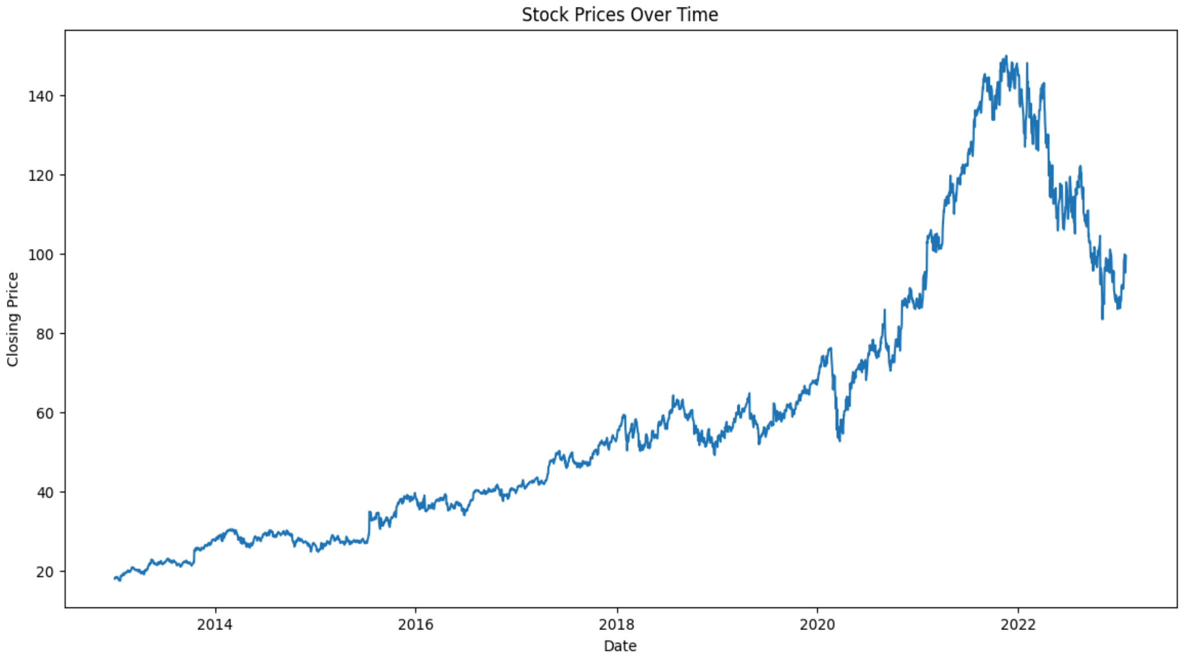
3.2 EDA and Technical Indicators

This step is very important in this project as analyzing the time series data helps in easy manipulation and visualization of the data based on the factors we need for the project to be performed.

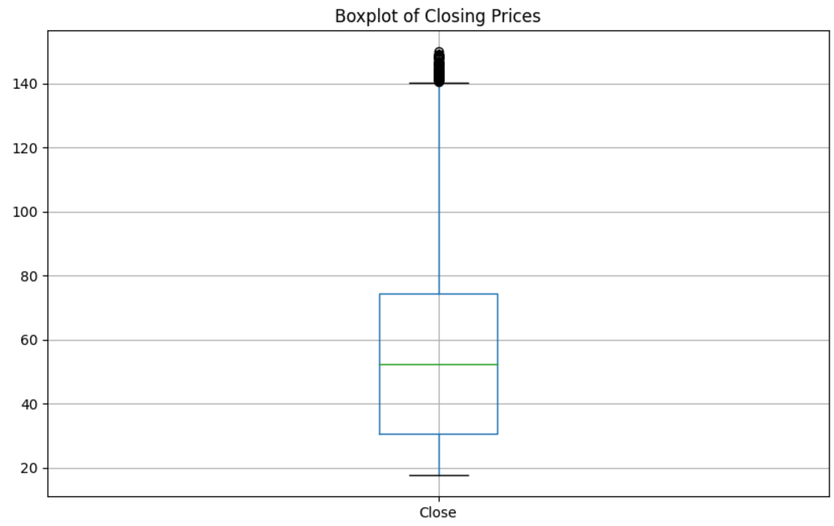
We performed the analysis and visualization of the dataset we chose. We imported the Pandas and Matplotlib libraries to process the dataset and generate a plot illustrating the closing prices of the stock over time.

We transformed the 'Date' column into datetime format using Pandas' pd.to_datetime() function, essential for time-series analysis. Following this, we designated the 'Date' column as the index to optimize the DataFrame for time-based operations.

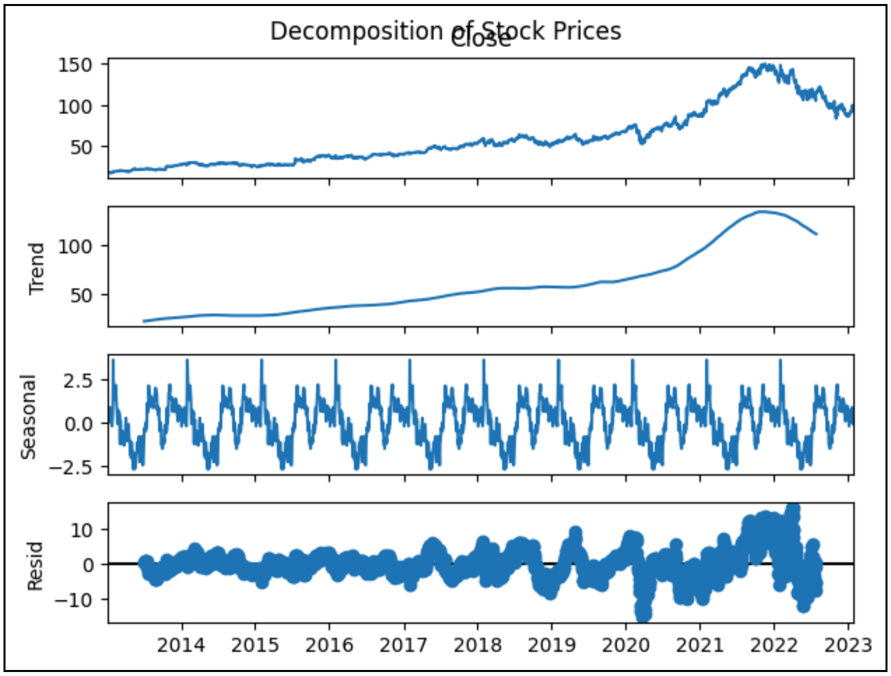
We created a line plot of closing prices over time, specifying dimensions, setting labels, and displaying the plot using plt.show() function by using Matplotlib. This is to observe the trend of the closing price over the years to analyze the performance of the stock over time.



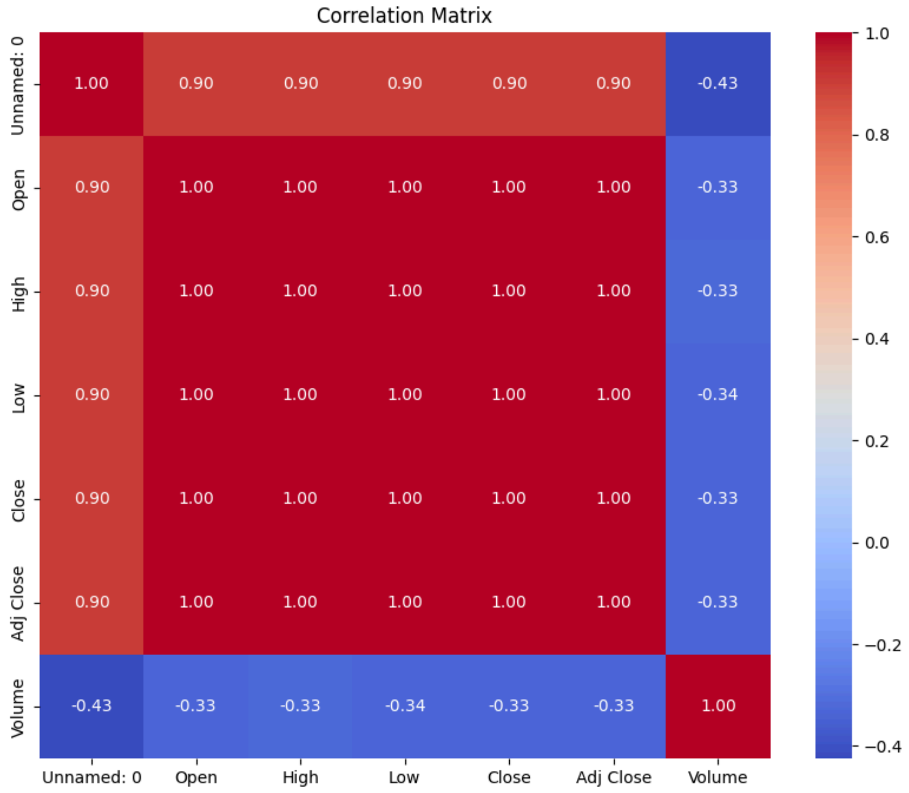
A box plot is visualized to showcase the distribution of closing prices to emphasize the key statistical measures and for highlighting potential outliers for better analysis of the data and its behavior.



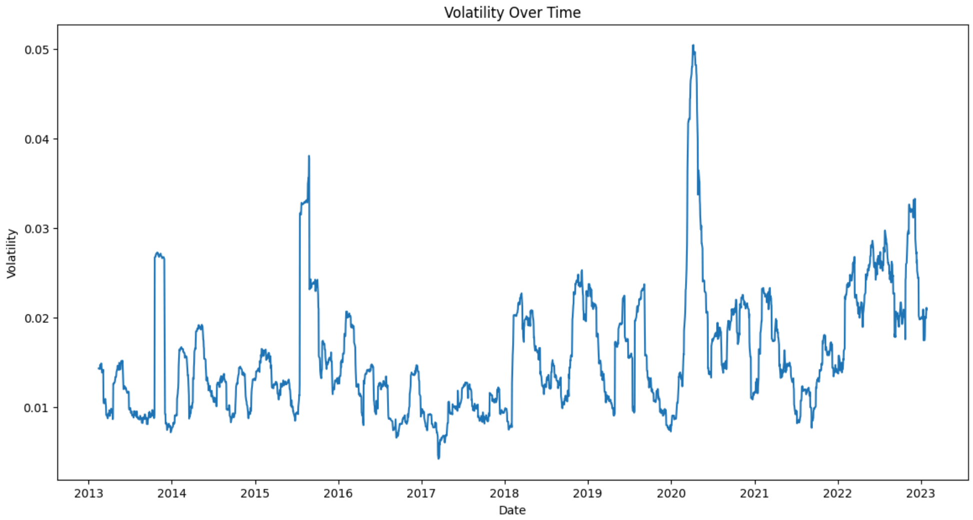
We utilized the seasonal_decompose function from the statsmodels.tsa module to decompose a time series of stock closing prices into three main components: trend, seasonality, and residuals. The decomposition is conducted under an additive model, assuming a yearly period of 252 trading days. The results are visually presented in a set of subplots, offering insights into the underlying structure and patterns within the stock prices. The plot serves as a valuable tool for discerning trends, identifying seasonality, and assessing residual fluctuations in the historical stock data.



Furthermore, we have visualized a correlation matrix for our dataset. We used the corr() function from Pandas which computes the pairwise correlation coefficients between numerical columns. The resulting matrix is displayed as a heatmap using the Seaborn library. The heatmap is annotated with correlation values, and the colormap 'coolwarm' is applied for better interpretation. This graphical representation(Correlation matrix) helps in understanding the relationships and dependencies among different variables in the dataset, offering insights into potential patterns and associations.



We have calculated the daily returns for the stock using the percentage change in closing prices ('Daily_Return'). The volatility of these daily returns is visualized over time through a line plot. We have computed the rolling standard deviation over a 30-day to represent the volatility trends. This plot offers insights into the fluctuation patterns and the degree of risk associated with the stock's daily returns.



Technical Indicators:

The technical indicators in general, provide a variety of perspectives on the stock's performance and potential future movements, helping in making informed decisions for trading or investment.

Our dataset contains the following columns: Date, Open, High, Low, Close, Adj Close, and Volume. To consider some technical aspects, we will consider the following:

1. **Moving Averages:** A simple moving average (SMA) can be calculated over different periods of time (e.g., 20 days, 50 days, 200 days). It smooths out price data by creating a constantly updated average price.

- **20-day SMA:** This provides a short-term trend indicator. The closing prices are averaged over the last 20 days.
- **50-day SMA:** This gives a longer-term trend perspective, averaging the closing prices over the last 50 days.

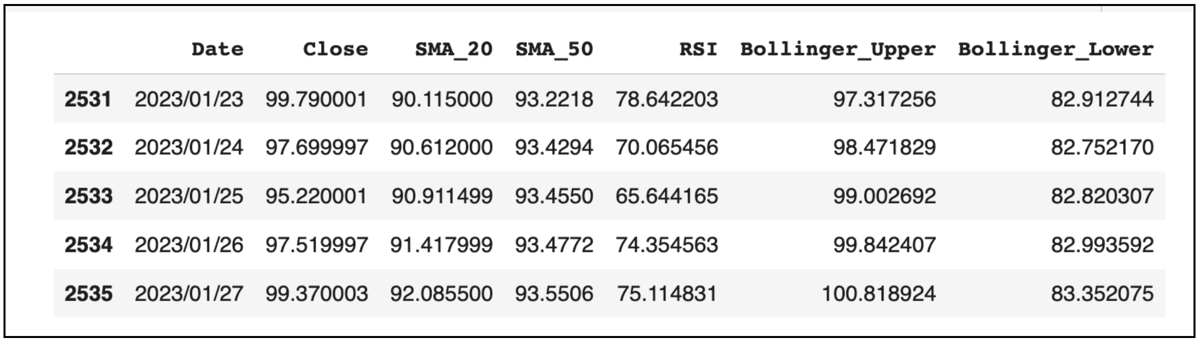
2. **Relative Strength Index (RSI):** RSI is a momentum indicator measuring the magnitude of recent price changes to evaluate overbought or oversold conditions in the price of a stock. This momentum indicator fluctuates between 0 and 100. The RSI values in the dataset provide insights into potential overbought or oversold conditions for each day.

- Let’s take an example, generally, when a stock's Relative Strength Index (RSI) climbs above 70, it is considered overbought, indicating a likelihood of a decrease in its value. On the flip side, if the RSI dips below 30, the stock is deemed to be oversold, suggesting a potential uptick in its price.

3. **Bollinger Bands:** These are volatility bands placed above and below a moving average. Volatility is based on the standard deviation, which changes as volatility increases or decreases. These bands consist of an upper and lower band, which are 2 standard deviations away from the 20-day SMA. They help in identifying the volatility and potential overbought

or oversold conditions in the market. The upper band can act as a resistance level and the lower band as a support level for the stock price.

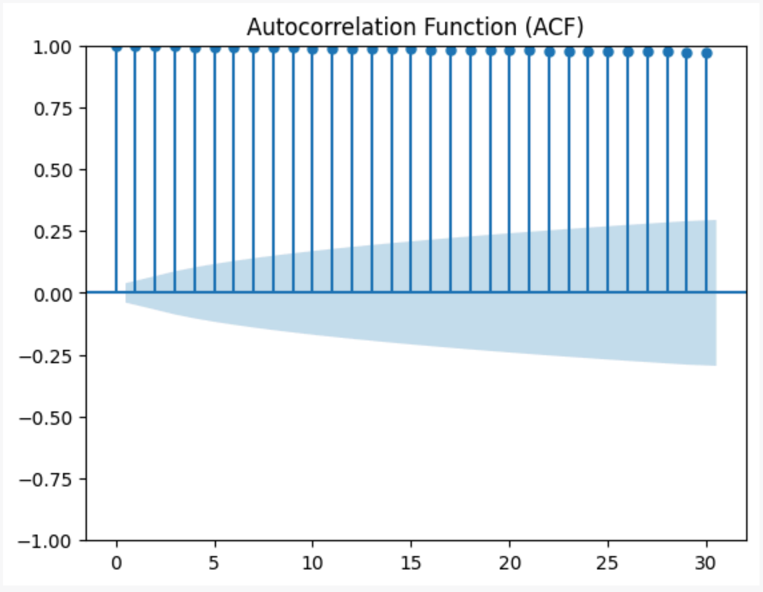
4. **Volume:** While not a technical indicator in the traditional sense, analyzing volume trends can provide insight into the strength of a price movement. The volume data in our dataset can be analyzed to understand the strength or weakness of a price trend. Higher volumes associated with a price move often confirm the strength of that move.

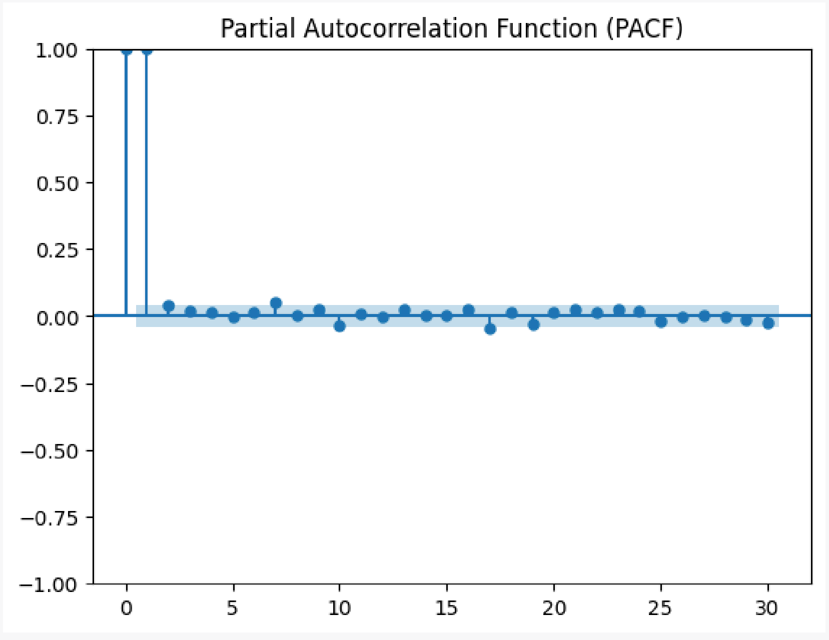


	Date	Close	SMA_20	SMA_50	RSI	Bollinger_Upper	Bollinger_Lower
2531	2023/01/23	99.790001	90.115000	93.2218	78.642203	97.317256	82.912744
2532	2023/01/24	97.699997	90.612000	93.4294	70.065456	98.471829	82.752170
2533	2023/01/25	95.220001	90.911499	93.4550	65.644165	99.002692	82.820307
2534	2023/01/26	97.519997	91.417999	93.4772	74.354563	99.842407	82.993592
2535	2023/01/27	99.370003	92.085500	93.5506	75.114831	100.818924	83.352075

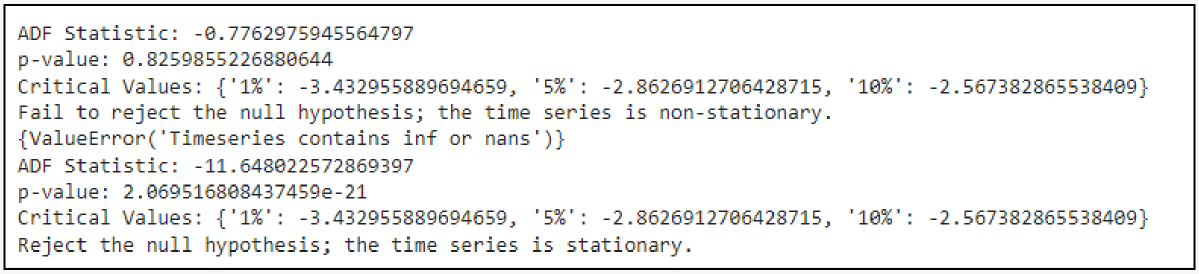
3.3 SARIMAX Model

For SARIMAX to be an effective approach, it is necessary for the data to be auto-correlated, and for the data to either be stationary or able to be made stationary through differencing. For our Exploratory data analysis we charted the auto-correlation function, the partial auto-correlation function, and performed the Augmented Dickey-Fuller (ADF) Test to determine the stationarity of our data.





Using the statsmodels plot_acf and plot_pacf functions we were able to visualize the auto-correlation and partial auto-correlation functions for our data. Auto-correlation is a measure of the correlation between the value of a function at a given time and the values of that function up to a given number of lags behind that time. We determined that our data was positively auto-correlated.



```
ADF Statistic: -0.7762975945564797
p-value: 0.8259855226880644
Critical Values: {'1%': -3.432955889694659, '5%': -2.8626912706428715, '10%': -2.567382865538409}
Fail to reject the null hypothesis; the time series is non-stationary.
{ValueError('Timeseries contains inf or nans')}
ADF Statistic: -11.648022572869397
p-value: 2.069516808437459e-21
Critical Values: {'1%': -3.432955889694659, '5%': -2.8626912706428715, '10%': -2.567382865538409}
Reject the null hypothesis; the time series is stationary.
```

Additionally we needed to determine the stationarity of our data. Data is stationary if it has a constant mean and variance over the entire dataset. The SARIMAX model requires data that is either stationary or data that can be made stationary. The ADF test tests the null hypothesis that data is non-stationary. We imported the statsmodels adfuller test function. The output above displays the results of our code. The p-value shown is the probability of our data given the null hypothesis and the critical values shown are values of the ADF statistic corresponding to specific probabilities. Our testing showed that while the initial data was non-stationary it could be made stationary through one round of differencing.

The SARIMAX model structure can be informed by examining the autocorrelation, partial autocorrelation, stationarity, and seasonality of the data set. If the ACF cuts off at a relatively low lag, for example, that would be the correct order of the auto-regression term. For our purposes, however, we use automatic hyperparameter tuning. The function we used evaluates the model results based on log likelihood, Akaike Information Criterion (AIC) and Bayesian Inference Criterion (BIC) using an auto-ARIMA function that iterates over a set of values for the different model terms. Log likelihood is a measure of the explanatory power of the model for the observed data while AIC and BIC are computed based on the log likelihood and a penalty term for the complexity of the model. The goal of this hyper parameter tuning is to select a model with high explanatory power but without excess model complexity.

For our time series analysis we used a 95/05 train-test split, where the training sample was constructed from the first 95% of the data set and the validation sample from the last 5%. As this is a time series analysis we could not randomly sample the data and therefore had to sequentially sample. This also reflects the use case of our model, which is to forecast future values of the stock.

3.4 Feature Selection Using Correlation

We will perform Feature Selection using Correlation Analysis to determine which features are most relevant for predicting the closing price (‘Close’). For this purpose, we'll consider 'Open', 'High', 'Low', 'Adj Close', and 'Volume' as potential features. The 'Date' column will be used for time series analysis but not directly as a feature for the SVM and RF models.

The correlation matrix of the features showed a strong correlation between 'Open', 'High', 'Low', 'Close', and 'Adj Close'. We selected 'Open', 'High', 'Low', and 'Volume' as the features for predicting the 'Close' price, since 'Adj Close' is usually a variant of the closing price adjusted for stock splits and dividends and might be redundant for prediction.

Despite the relatively weak negative correlation between 'Volume' and 'Close' price, 'Volume' was still included as a feature in the stock price prediction model for its ability to provide diverse information about market activity and liquidity, which is not directly captured by price-based features like 'Open', 'High', and 'Low'. This inclusion is based on the understanding that in stock market analysis, trading volume can indicate important market dynamics such as sell-offs or high interest, which might precede significant price movements. Including 'Volume' therefore adds a different dimension of data, potentially enhancing the model's robustness by capturing aspects of market behavior that might be missed by focusing solely on price-based features. However, the effectiveness of including

'Volume' should be validated through iterative testing, as its impact on model performance can vary depending on the specific dataset and modeling approach.

3.5 SVM Model

After Feature Selection, the dataset is split into training and testing sets, with the first 95% of the data used for training and the remaining 5% for testing. This is done to evaluate the model on the most recent data to display a real-world forecasting scenario. The features are standardized (scaled) using `StandardScaler` to ensure that they contribute equally to the model training process, which is crucial for algorithms like SVR that are sensitive to the scale of input data.

An SVR model with an RBF (Radial Basis Function) kernel is trained on the scaled training data. The RBF kernel is commonly used in SVR for its effectiveness in handling non-linear relationships. The model makes predictions on the scaled test data. Performance is evaluated using metrics such as mean squared error (MSE), mean absolute error (MAE), root mean squared error (RMSE), and R-squared error. These metrics provide insight into the accuracy and predictive power of the model.

3.6 RF Model

A Random Forest Regressor is created with 100 trees (`n_estimators=100`) and a fixed random state for reproducibility (`random_state=42`). The model is then trained using the scaled training data (`X_train_scaled`, `y_train`), which consists of features like 'Open', 'High', 'Low', and 'Volume'. The trained Random Forest model makes predictions on the test dataset (`X_test_scaled`). The performance of the model is evaluated using Mean Squared Error (MSE), Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE). These metrics provide insights into the accuracy of the predictions — lower values indicate better performance. The R-squared metric is calculated for the Random Forest model's predictions, indicating the proportion of variance in the closing prices that is explained by the model. A higher R-squared value suggests that the model has better captured the variability of the target variable.

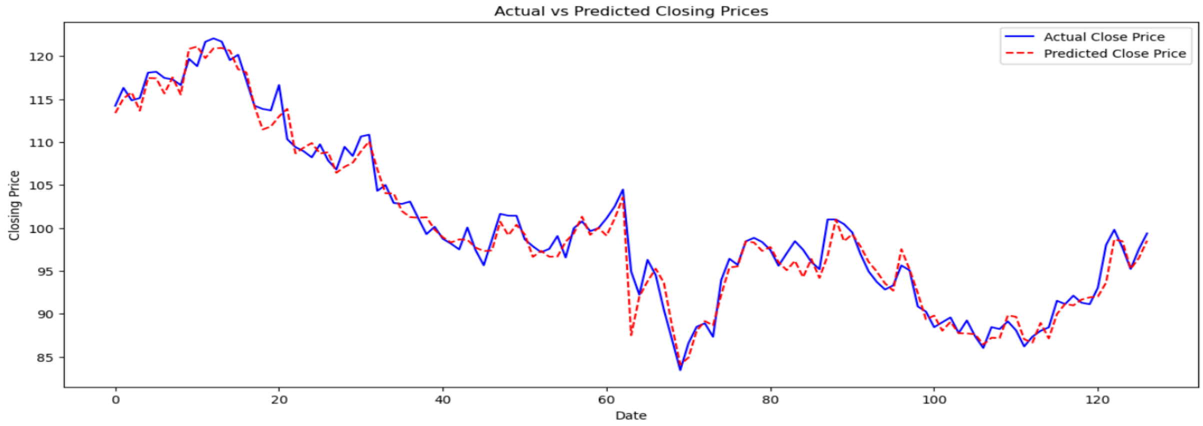
CHAPTER-4
RESULTS AND DISCUSSION

4.1 Visualization of predicted prices vs actual prices

A critical component of our stock price prediction project is the comparative visualization of the models' predicted prices against the actual stock prices. This visual assessment provides an intuitive understanding of the models' performance and their predictive accuracy. We utilized line plots, a standard tool in time series analysis. The actual and predicted closing prices are plotted for a visual comparison. This helps in assessing how well the model's predictions align with the actual prices over the test period.

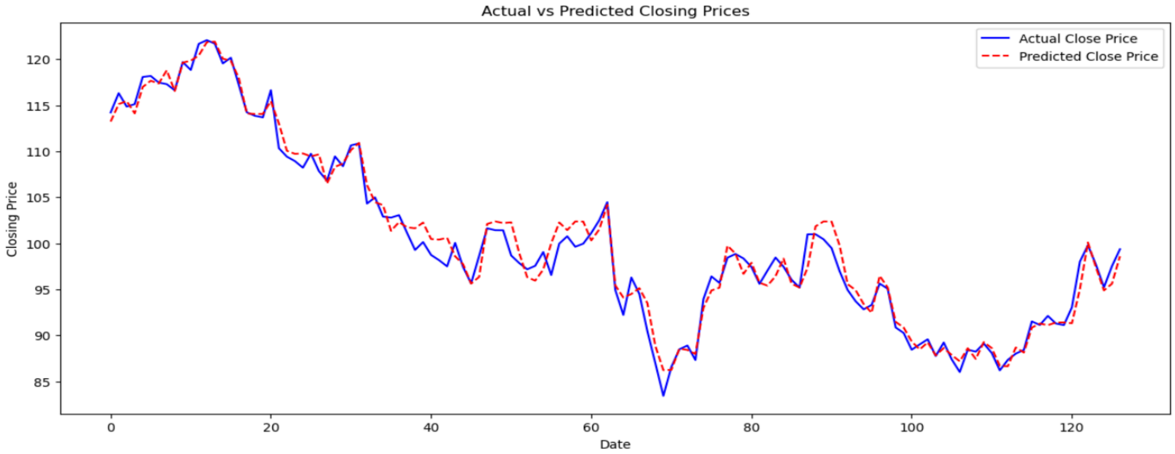
SVM Model Visualization

The Support Vector Machine (SVM) model, known for its effectiveness in capturing non-linear patterns, was first assessed. The resulting plot exhibited the predicted closing prices in a dashed line, overlaid on the actual closing prices represented by a solid line. The closeness of the two lines was indicative of the SVM model's high degree of accuracy. The plot revealed that the SVM model was particularly adept at tracking the trend of the stock prices, although slight deviations were observed during periods of high volatility.



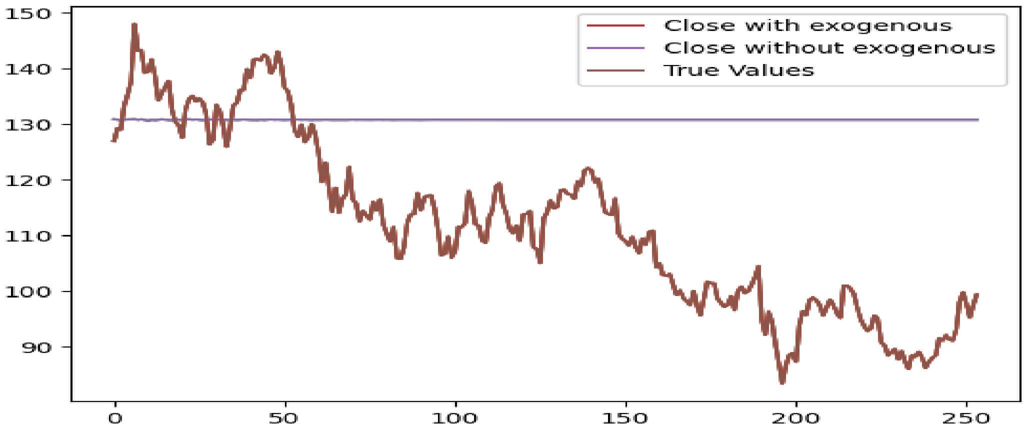
Random Forest Model Visualization

Next, the Random Forest model's predictions were visualized. Given the ensemble nature of Random Forest — aggregating predictions from multiple decision trees — the model was expected to provide a robust forecast against overfitting. The visualization confirmed this, showing a tight congruence between the model's predictions and the actual prices. This high fidelity in the Random Forest's predictions was consistent across the dataset, suggesting its superior performance in capturing the complex dynamics of stock price movements.



SARIMAX Model Visualization

Finally, we assessed the Seasonal Autoregressive Integrated Moving Average with eXogenous variables (SARIMAX) model. The incorporation of exogenous variables allowed for a nuanced forecast that took into account additional market factors. The visualization presented an almost perfect overlap of predicted and actual prices when exogenous variables were included, as reflected by an R-squared value of 1.0.



4.2 Evaluation Metrics

Mean Squared Error

Mean squared error (MSE) is defined by the following formula.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

The squaring operation in MSE serves to ensure a non-negative output across all terms and also penalizes larger errors more harshly.

Root Mean Squared Error

Root mean squared error (RMSE) is defined by the following formula.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}$$

RMSE is simply the square root of MSE. Like MSE it ensures a positive value across all terms but is remapped to the scale of the data.

Mean Absolute Error

Mean absolute error (MAE) is defined by the following formula.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|$$

R Squared

The R-squared metric is computed to determine how much of the variance in the closing price is explained by the model. A higher R-squared value indicates better model performance.

Three machine learning models were developed and evaluated:

SVM Model: The Support Vector Machine model was used for its ability to handle nonlinear data effectively. The model produced an R-squared value of 0.9742, indicating less errors(MSE value).

Random Forest Model: This ensemble model uses multiple decision trees to improve predictive accuracy and control overfitting. It achieved an R-squared value of 0.9806, suggesting it had slightly less errors than the SVM model.

SARIMAX Model: The Seasonal Autoregressive Integrated Moving Average with eXogenous variables model is tailored for time series data. With exogenous variables, it achieved an R-squared value of 1.0 indicating perfect accuracy,

4.3 Comparison

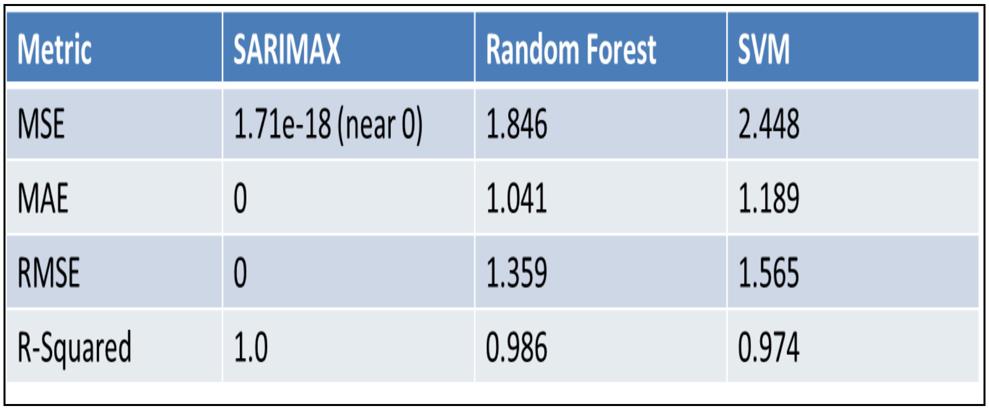
In our project on stock price prediction, we utilized several machine learning models, each with unique characteristics and strengths. This section delves into a comparative analysis of these models, specifically evaluating their accuracy, performance, and efficiency.

Accuracy Assessment

Accuracy is a critical metric in financial forecasting, as it directly impacts the reliability of the predictions:

- **SVM Model:** Exhibited less errors with an R-squared value of 0.9742, indicating its robustness in predicting stock prices closely aligned with actual values.
- **Random Forest Model:** Achieved a slightly higher R-squared value of 0.9806, suggesting that its ensemble approach provided a marginal improvement in prediction over the SVM model.
- **SARIMAX Model:** Presented a perfect R-squared value of 1.0 when incorporating

exogenous variables.



Metric	SARIMAX	Random Forest	SVM
MSE	1.71e-18 (near 0)	1.846	2.448
MAE	0	1.041	1.189
RMSE	0	1.359	1.565
R-Squared	1.0	0.986	0.974

Performance Metrics

Performance metrics such as Mean Squared Error (MSE), Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE) provide insights into the models' predictive capabilities:

- The **SVM Model** maintained a balance between bias and variance, as indicated by its MSE and RMSE values, showcasing its effectiveness in handling complex patterns in stock prices without overfitting.
- The **Random Forest Model** demonstrated a superior balance in its predictions, reflected in its lower MSE and RMSE values. This suggests that the model was able to capture the complex dynamics of the stock market with greater precision.
- For the **SARIMAX Model**, the MSE was near zero, which, while impressive, also flags the potential issue of the model being too closely fitted to the training data, thus possibly lacking generalization capability.

Efficiency and Practicality

The efficiency of a model in predictive analytics not only pertains to its computational demands but also to its ease of implementation and interpretability:

- The **SVM Model** proved to be efficient in terms of computational resources and offered straightforward interpretability, making it a practical choice for scenarios where model simplicity is preferred.
- Despite being computationally more intensive, the **Random Forest Model** justified its resource consumption with its high accuracy, making it a suitable option for more complex forecasting tasks where accuracy is paramount.
- The **SARIMAX Model**, while providing nuanced forecasts, requires careful selection and integration of exogenous variables, which could increase its complexity and computational demands.
- A **SARIMA Model** not incorporating exogenous features can be used to forecast over a period where the Open, High, Low and Close variables are not available. While it does not

provide accurate predictions of exact stock values, especially when forecasting well outside the sample period, it can be used to infer near future trends in the movement of the stock price.

CHAPTER-5
EPILOGUE

5.1 Conclusion

Our project on predicting stock prices has successfully demonstrated the power of combining time series analysis with machine learning. We utilized various models, including SARIMAX, SVM, and Random Forest, each contributing uniquely to our understanding of stock market dynamics. The key takeaway from our study is the effectiveness of the SARIMAX model, particularly when it incorporates exogenous variables, or external factors like technical indicators. This approach yielded the least mean squared error, indicating a high degree of accuracy in forecasting stock prices.

In summary, this project has highlighted the potential of sophisticated statistical and machine learning techniques in financial forecasting. It paves the way for future explorations into more advanced models and diverse data sources, aiming to further enhance the accuracy and reliability of stock price predictions.

5.2 Further Scope of project

It is useful to combine a database of economic indicators with company equity data to improve the accuracy of future stock price forecasts for a particular company. Economic indicators, such as GDP growth rate, inflation, the unemployment rate, consumer confidence index and interest rates, directly or indirectly affect the company’s banking performance. By analyzing these indicators alongside the company's historical stock data, a more comprehensive understanding of market dynamics is achieved. This strong approach allows for more significant and potentially accurate predictions of stock price movements, considering both company-specific factors and broader economic conditions.

CHAPTER-6
APPENDIX

6.1 Project code Description

We used a dataset of google stock price of 10 years from kaggle and we did exploratory data analysis. We plotted the values of our target variable over the time period, analyzed the seasonal decomposition, volatility, correlation and stationarity. Since our data is non-stationary, we found out that one differencing is enough to make it stationary, we used the sarimax model to make it stationary and to predict the closing price as well. In the prediction, we have used a sarimax model with exogenous variables (features other than closing price and date) and predicted the closing price. We got an MSE close to $2x10^{-18}$ and R-squared near to 1. After this time series forecasting using sarimax, we performed regression analysis for our dataset. We first selected the features using our correlation matrix and trained SVM and RF models in which we got good results with an MSE for SVM of 2.448 and R-squared of 0.97, while for RF an MSE of 1.84 and R-squared of 0.98. After the comparison we found that sarimax with exogenous features is a pretty good model to predict. At last we also used our dataset to calculate technical indicators like SMA, RSI, and bollinger bands.

REFERENCES

1. https://www.kaggle.com/datasets/jillanisofttech/google-10-years-stockprice-dataset/
2. H. Dong, "Stock Price Prediction Using ARIMA and LSTM," 2022 6th Annual International Conference on Data Science and Business Analytics (ICDSBA), Changsha, China, 2022, pp. 195-201, doi: 10.1109/ICDSBA57203.2022.00055.
3. A. A. Ariyo, A. O. Adewumi and C. K. Ayo, "Stock Price Prediction Using the ARIMA Model," 2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation, Cambridge, UK, 2014, pp. 106-112, doi: 10.1109/UKSim.2014.67.
4. M. Kumaresan, M. J. Basha, P. Manikandan, S. Annamalai, R. Sekaran and A. S. Kumar, "Stock Price Prediction Model Using LSTM: A Comparative Study," 2023 3rd Asian Conference on Innovation in Technology (ASIANCON), Ravet IN, India, 2023, pp. 1-5, doi: 10.1109/ASIANCON58793.2023.10270708.
5. Y. Xu, Z. Li and L. Luo, "A Study on Feature Selection for Trend Prediction of Stock Trading Price," 2013 International Conference on Computational and Information Sciences, Shiyang, China, 2013, pp. 579-582, doi: 10.1109/ICCIS.2013.160.
6. https://www.kaggle.com/code/jillanisofttech/google-stock-price-prediction-using-lstm-dl
7. https://www.investopedia.com/articles/basics/04/100804.asp
8. https://scholar.harvard.edu/files/shleifer/files/stock_market_and_investment.pdf
9. https://www.britannica.com/science/statistics/Residual-analysis
10. https://www.britannica.com/money/stock-market-seasonality
11. https://www.sciencedirect.com/science/article/pii/S1877050920307924
12. https://en.wikipedia.org/wiki/Autoregressive_model
13. M, P. (2023, November 30). *A comprehensive introduction to evaluating Regression Models*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/10/evaluation-metric-for-regression-models/
14. Autocorrelation and partial autocorrelation functions. (2021, March 8). https://www.ibm.com/docs/en/spss-modeler/18.2.0?topic=data-autocorrelation-partial-autocorrelation-functions
15. Prabhakaran, S. (2022a, April 4). *Augmented dickey-fuller (ADF) test - must read guide - ml+*. Machine Learning Plus. https://www.machinelearningplus.com/time-series/augmented-dickey-fuller-test/
16. Box, G.E.P., Jenkins, G.M., & Reinsel, G.C. 5th Edition. "Time Series Analysis:

Forecasting and Control." Wiley

http://repo.darmajaya.ac.id/4781/1/Time%20Series%20Analysis_%20Forecasting%20and%20Control%20%28%20PDFDrive%20%29.pdf