# STOCK MARKET PREDICTION USING MACHINE LEARNING TIME-SERIES ANALYSIS

## Abstract

In this study, LSTM, XGBoost, and Random Forest models are considered for predicting the stock price's future values of Tesla from the time of its IPO to 2017. That is the dataset which has been obtained from Yahoo Finance where the daily stock information like opening, closing, highest, lowest price and the number of shares traded are given. Specifically, the aim is data pre-processing, applying each of the algorithms, and evaluating the models' performance using RMSE, residuals and their distribution, as well as feature importance. Based on the results, the use of the LSTM model for this study yields better prediction accuracy as compared to XGBoost and Random Forest by having the lowest RMSE and closely following the actual stock prices. LSTM is demonstrated in the study as a good model for modelling time series dependencies with real applications in stock trading. More extensive research should be conducted for more enhanced models, extra input variables, and other stock markets so that much better results for the forecast accuracy and stability for future work should be obtained. Improved model performance in financial forecasting that results from this research helps towards making improved investment decisions on the part of users of machine learning solutions.

## Table of Contents

## List of Figures

# 1. Introduction

## 1.1 Introduction

An important characteristic of the stock market is that it is a dynamic system characterized by high levels of volatility meaning that making a clear prediction of trends is difficult. The problem focus under consideration in this research relates to stock market analysis. It is mainly concerned with the use of machine learning to predict trends in the stock market using time series data. It seeks to use historical data and modelling tools to identify trends and improve on predictive and prognostic analysis. Offering the chance to surpass conventional approaches, the employment of machine learning constitutes an opportunity (Shepherd and Majchrzak, 2022). All these help the investors to make better decisions, reduce or avoid risks, and maximise returns in the ever-fluctuating stock markets.

## 1.2 Background

The stock market trading started in 1611 by the Dutch East India Company (Emmer and Gommans, 2020) was the only existing market for many years and has since expanded to become a global market. The forecast of the stock market movement then emerged towards the end of the last century with the development of statistical or time series analysis. However, the analysis of stock movements continues to prove difficult because of the inherent unpredictable and dynamic character of the stock market.

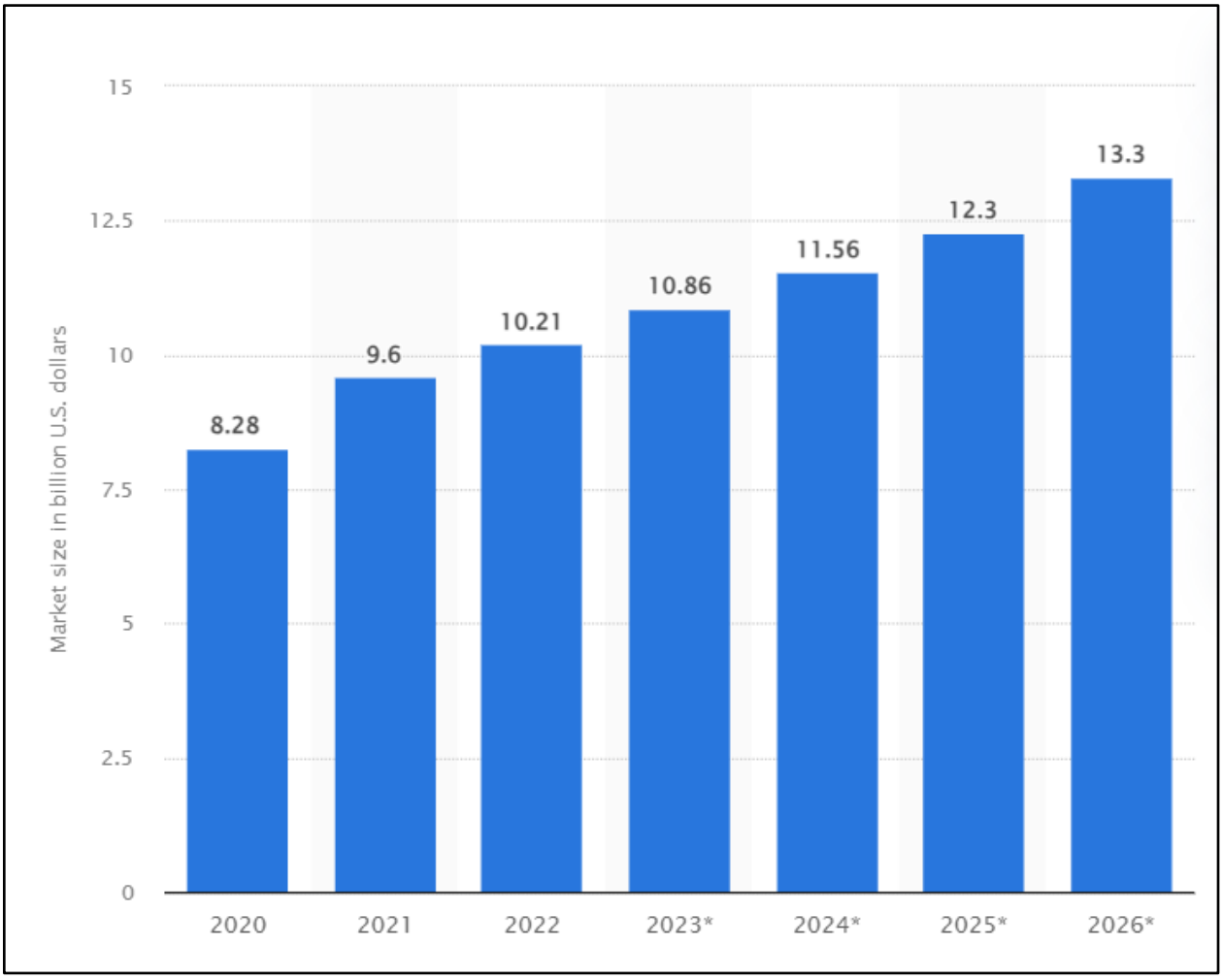

**Figure 1.1: Online Trading Market Forecasting**

(Source: Statista, 2023)

In the future, global online trading has been projected to rise at a global compounded annual growth rate of 6.4% per year and is expected to reach US 13.3 Billion in 2026 (Statista, 2023). This growth underlines the idea that the issue of stock forecast becomes critical, as investors take high risks relying on their guesswork. By employing time-series analysis with the help of ML algorithms, one can overcome the mentioned difficulties. These algorithms may make use of past data to forecast additional features of stock prices and evaluate real-time market conditions. For example, at the beginning of 2022, about 60% of the trading from the US equities was attributed to automated trading and hence, the need for better predictive models (Kanade, 2023). Furthermore, it aims to offer a basis for the selection of relevant factors, algorithms and features dependably for developers of financial technology that would improve the efficacy of the stock price prediction.

## 1.3 Rationale

The rationale of the research is to analyse the ever-changing market trends, investors are in constant search for the most accurate prediction models to reduce risks of investment and also to maximize profits. Based on time series data analysis using machine learning techniques this research seeks to establish the best models fit for forecasting stock prices. As the market tends to produce unpredictable situations it is crucial to improve the efficiency of forecasts and investors' decision-making (Sari, Kusnanto and Aswindo, 2022). Further, it supports the creation of stable financial technologies through the determination of the most fitting models for the predictability of the stock market in the future.

## 1.4 Aim

The main aim of this project will be to develop a software model that is based on LSTM, XG Boost and Random Forest algorithms. Then, the performance of these models will be compared based on effective model evaluation techniques mainly to find the best algorithm for predicting stock prices.

## 1.5 Objectives

1. To develop a software model based on LSTM, XG Boost and Random Forest algorithms for forecasting stock prices.
2. To analyse different factors that are responsible for sudden stock price changes in the dynamic market.
3. To pre-process the collected dataset as per the need of each algorithm.
4. To evaluate the best-performing model for the forecasting of stock prices.

## 1.6 Questions

1. What is the best algorithm for the stock price prediction among LSTM, XG Boost and Random Forest?
2. What are the main factors that influence the stock price changes in this dynamic market?

## 1.7 Significance

The present work is significant as it expands the knowledge of the authors about the possibilities of applying machine learning for financial forecasting, as well as identifying the effectiveness of various approaches when predicting stock market indices. The findings will therefore be useful to investors and also help to minimize such risks while at the same time maximizing the returns by making the right estimates. Furthermore, the findings of the research will be helpful for financial technology developers as well as regarding the choice of the optimal algorithms for real-time forecasting. Finally, this investigation improves the formulations of decisions in finance and contributes to the creation of new approaches to examine the phenomena of stock exchanges.

## 1.8 Chapter Summary

This study seeks to develop an understanding of the machine learning procedures especially the time series analysis to forecast stock market prices. The study seeks to find out which of the models is most accurate in stock prediction; therefore it will compare models like LSTM, XGBoost, and Random Forest. In the light of increasing complexities involved in the stock market, research is vital to reduce risks and increase returns. The analysis will happen by determining the fundamental factors that would affect share prices and thus present useful insights to both the investors, as well as the developers of the financial technology applications.

# 2. Background

## 2.1 Literature Review

Chapter 2 aims to review the literature on the use of machine learning algorithms for stock price prediction and especially the benefits accruing from this approach as compared to conventional ones. It also explains the use of important models like ANN, RF and LSTM in raising the levels of prediction accuracy. The review also discusses the aspect of temporal variability of stocks, the relevance of data pre-processing, and the evaluation of models.

### Machine Learning Algorithms for Stock Price Prediction

Finance prediction has started to incorporate machine learning more often because fluctuations in financial markets are complex and not easily tractable. Conventional approaches may fail to provide precise predictions as to the further evolution of stock prices, whereas the application of machine learning techniques in this field has improved the situation. Vijh *et al.* (2020) employed Artificial Neural Networks (ANN) indicated by the name and Random Forest (RF) to forecast the next day's closing stock price using parameters such as opening, highest, lowest and closing prices. They further used standard evaluation mechanisms such as the RMSE and the MAPE to show that this was possible with low error rates. However, Khan *et al.* (2022) discussed the external variables like social media and financial news for the stock price prediction. They emphasised that by including SM and news data in machine learning models, their accuracy reached more than 80%, which contrasted with the traditional models based on historical data only.

Furthermore, Mehtab and Sen (2020) proposed a technique that employed statistical, machine learning, and deep learning models including LSTM, and CNN models. The agglomeration framework that was built from mined granular daily stock price data pointed to short-term accurate forecasts, then emphasized the efficacy of Machine Learning in Predictive Finance. The above studies as a whole demonstrate that machine learning approaches provide rigorously effective solutions to stock price prediction more accurately than other conventional methods especially, ANN, RF and LSTM.

### Factors Influencing Stock Price Volatility in Dynamic Markets

Fluctuations in stock price are therefore anchored on many factors which can be categorically grouped as fundamentals and behavioural factors. Similarly, the effects found by Thampanya *et al*. (2020) were analyzed on the ASEAN-5 shares. While fundamental determinants made major contributions to volatility in Malaysia Thailand and Singapore behaviour determined the volatility of Indonesian and Philippine shares more than a fundamental determinant. This variation across different financial crises shows that economic stability and the effectiveness of policy measures this stability is paramount.

Bhowmik and Wang (2020) have also investigated the stock market volatility addressing more on the use of GARCH models for the evaluation of market returns and volatilities. The review of their work also notes that the focus of research has moved to enhancing the forecast of the stock market and risk control. It might therefore be argued that, due to their ability to explain volatility characteristics in the markets, GARCH models have been instrumental in market analysis. Yang *et al.* (2023) analyzed this issue concerning global oil price shocks, stock market volatility and economic policy uncertainty in China and the US. Their evidence confirms the high level of co-movements between oil price shocks and the stock markets as well as the fact that a higher level of economic policy uncertainty enhances the impacts. This work also reveals how the external economic environment influences the global financial markets particularly the volatility of the stock markets.

### Data Preprocessing Techniques for Time-Series Forecasting Models

Data Preprocessing Techniques also play a significant role in improving the existing and new time-series forecasting models when applied to irregular and incomplete data sets. Bharadiya (2023) also emphasises the use of preprocessing techniques in RNNs which are widely used in the case of irregular time series data. Preprocessing techniques used are imputations for missing values and changes in the learning algorithms to handle missing data. These techniques make sure that RNNs can deliver improved result outputs in a variety of industries by handling data noise appropriately. Liu *et al*. (2023) addressed preprocessing in the context of fin-time series forecasting with a focus on cryptocurrencies. Their work focuses on linear regression together with other mixed system models like LSTM and Decision Tree Regressors. They recommend parametric and non-parametric techniques of data pre-treatment such as the decomposition of financial time series data and auto-correlation functions to identify existing endogenous structures.

Bandara *et al.* (2021) discuss the problem of the scarcity of time series data in GFM by suggesting data augmentation methods. Some of the techniques that are applied in the generation of synthetic time series include GRATIS, moving block bootstrap (MBB), and dynamic time warping bar centric averaging (DBA). This augmentation greatly enhances the validity of GFM models and may be used to show the advantage of preprocessing even in situations where there is very little data available.

**Model Evaluation Metrics in Stock Market Forecasting**

The stock markets are volatile and thus, metrics for model evaluation are useful when it comes to the assessment of the performance of the upper forecasting models. Kumar *et al.* (2021) researched a relatively good understanding of how various subdomains of computational intelligence are evaluated in terms of market stock prediction. It is argued that the application of performance measures is made up of the fundamental pillar for evaluating the model. Among all aforementioned measures, a few of the most commonly used performance indices are Root Mean Squared Error (RMSE), Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE). Nti *et al.* (2020) investigated an efficient model of stock market prediction using the ensemble learning method. The methods they use are grouped into stacking, blending, bagging and boosting them, then compare the above by estimating their mean square error relative to each other as well as their accuracy level.

They performed their research and discovered that the stacking and blending ability provided superior performance which was indicated by the results depicted by the values of RMSE which suggested that this technique belonged to the two classes and provided a better fit superior to both bagging and boosting. From one technique to another may have an impact on the model and that makes an important when one is selecting which would be ideal for a certain task. According to the study by Nabipour *et al.* (2020), the utilization of profound learning strategies, to be specific LSTM covers different tree-based calculations for securities exchange expectations. It shows that LSTM models, in general, yield the highest accuracy and the best capability of modelling among the applied algorithms. Their analysis based on technical indicators and other evaluation metrics including RMSE and MAE strengthens the usefulness of the metrics in identifying the efficiency of each model of forecasting. Collectively, these papers show that models' evaluation measures help to choose proper forecasting methods in the context of volatile and multifaceted stock market prediction.

## Theories and Framework

### *Efficient Market Hypothesis*


**Figure 2.1: Efficient Market Hypothesis (EMH)**

(Source: Ehiedu and Obi, 2022)

That is why the Efficient Market Hypothesis (EMH) can be used for the stock market forecasting research in terms of models' accuracy assessment. It ensures the belief that all known information is integrated into stock prices and thus makes it difficult for an investor to beat the market persistently (Ehiedu and Obi, 2022). In this way, EMH can help researchers analyse if new forecasting models contribute unique value on their own or if they merely reflect the hypothesis that the current market price is optimal. This assists in honing and also confirming the accuracy of almost all the forecasting models.

***Time Series Analysis Theory***

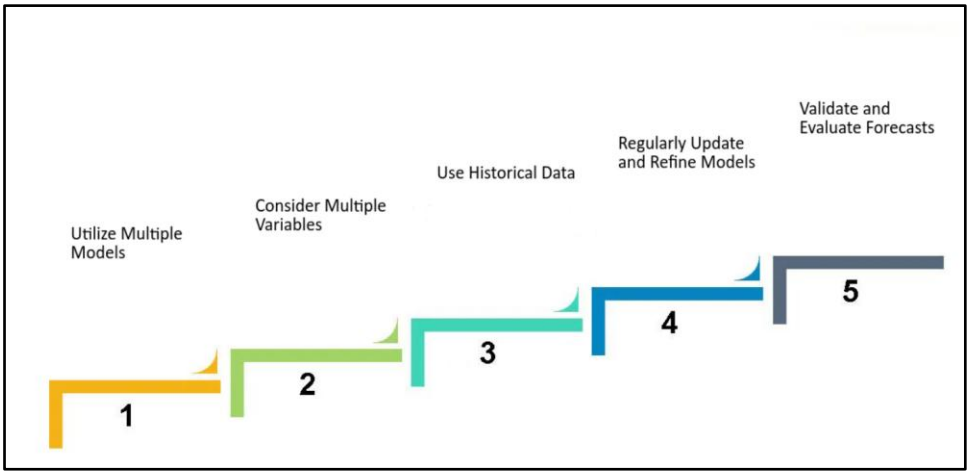


**Figure 2.2: Time Series Analysis Theory**

(Source: Ehiedu and Obi, 2022)

Time Series Analysis Theory is important when it comes to stock price forecasting since it has techniques for analyzing historical data to get estimates of future stock prices. This theory assists in making a distinction between trends, seasonality, and cyclical behaviour in the stock prices and it is useful for building up the forecasting models. ARIMA, LSTM and others help the researcher

to do the pre-processing of data, feature selection and evaluation of the model to predict future trends in the stock market.

***Framework***

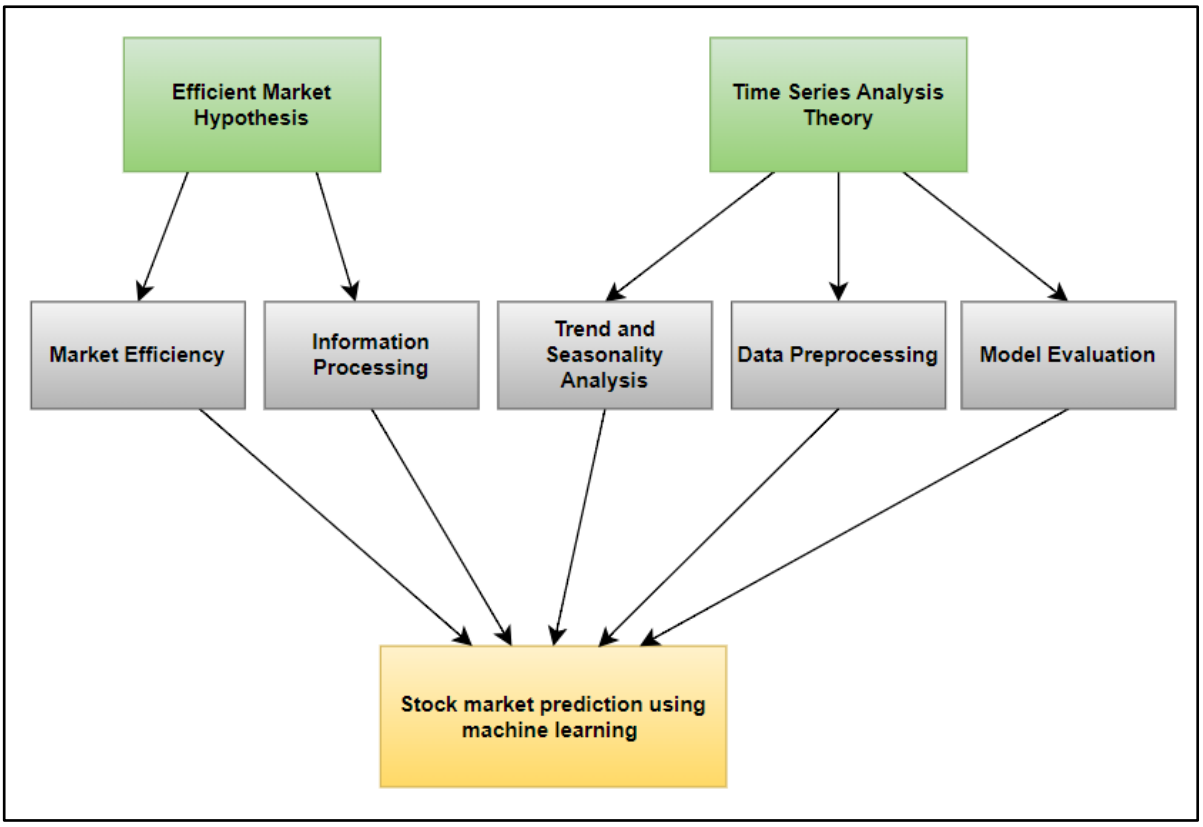


**Figure 2.3: Theoretical Framework**

(Source: Self-developed in draw.io)

**Conceptual Framework**

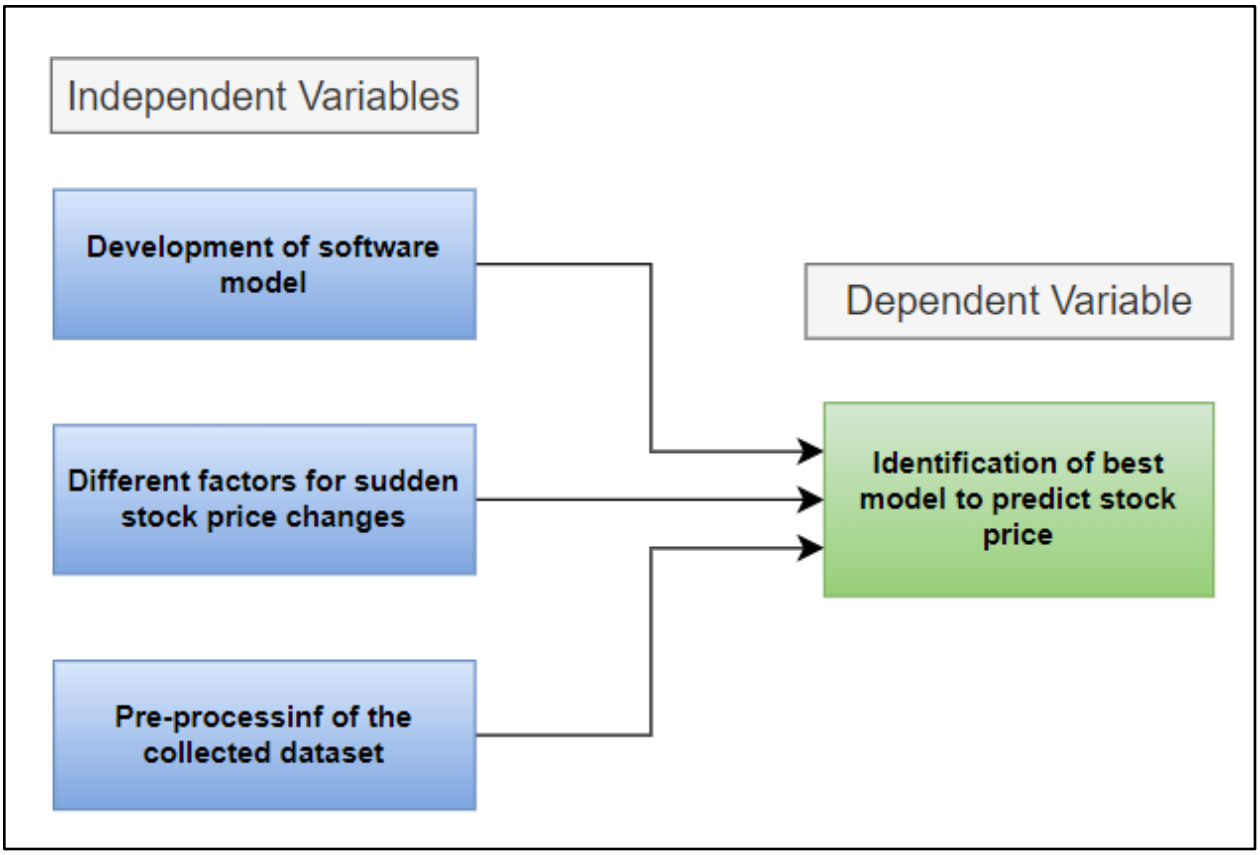


**Figure 2.4: Conceptual Framework**
(Source: Self-developed in draw.io)

**Literature Gap**

This study also presents prior contributions to stock price prediction based on machine learning which include, ANN, RF, LSTM and the ensemble models. However, some gaps are there like a lack of connection between external factors such as SNS and financial news with traditional measures. Although some of these variables have been investigated in studies, more research on integrated technical and external market variables in hybrid models is scarce.

## 2.2 Technical Background of the project

Specifically, the project is developed for the stock market prediction of different companies based on machine learning models. Stock markets are among some of the most sensitive and unstable markets which makes it almost impossible to predict. Typically, the changes in stock prices are also not easy to model, and conventional statistical tools tend to perform poorly (Buhalis and Moldavska, 2022). Machine learning on the other hand comes with the advantage of making use of historical data in an attempt to analyze the trends in the market prices. The project explores three machine learning models: LSTM (Long Short-Term Memory), XGBoost and Random Forest. LSTM is a kind of Recurrent Neural Network used for sequential data analysis. It is noteworthy that XGBoost is an optimised and efficient way of gradient-boosting decision trees. This is the case in Random Forest where a group of decision trees are used to reduce prediction error.

Preprocessing forms an important step in the time series for the casting process (Liu *et al.* 2023). This entails pre-processing of the data and data cleansing in which noise and missing values are dealt with appropriately. The project aims to build models that can be predictive of stock prices

with reasonable precision. For assessment purposes, measures like RMSE and MAE will be utilized in comparison to the best models. This is because the execution of the project is to enhance the forecast of stock prices to assist the investors make informed decisions.

# 3. Dataset

## 3.1 Dataset Description

This dataset relates to the stock price of Tesla Motors from the date it went public; 06/29/2010 to 03/17/2017. It offers valuable information like the date of the entry, the opening price of the stock, the highest and lowest price the stock reached within the day, the closing price of the stock and the closing price adjusted for such factors as dividends (Rolando, 2017). Also, the amount of shares that are traded on a particular day is measured.

The numericals are collected from Yahoo Finance and the data is fetched via Python programming language. However, with this dataset, important periods of its financial history are covered that may be useful in studying the fluctuations of the market, prices and behaviours of the stock in different conditions. Challenges are raised on factors including trading volume effects on the prices, or effects originating from the difference between the adjusted closing and opening the following day price. Using these variables one can explain the stock prices and interpretation of the impact of market actions over different periods.

## 3.2 Justification of the selected dataset

The selected dataset can be considered quite suitable for the study as it provides valuable information regarding Tesla's share price from its IPO to 2017. This dataset contains important data such as opening and closing prices the highest and the lowest points during the day, the adjusted closing price and the volume of trades. The relative advantages of these metrics lie in the fact that they give an understanding of the behaviour of Tesla on the market and make it possible to analyze the change in prices. The understanding that is attained from the observations on trading volume and the relative of the closing and opening days explains both short-term and long-term stock movements that exist in the dataset. It could help to realize the consequences of corporations' actions, investors' attitudes and external factors on Tesla shares. All in all, the richness of the dataset allows for various examinations of Tesla's financial experience and stock market performance.

## 3.3 Exploratory Data Analysis

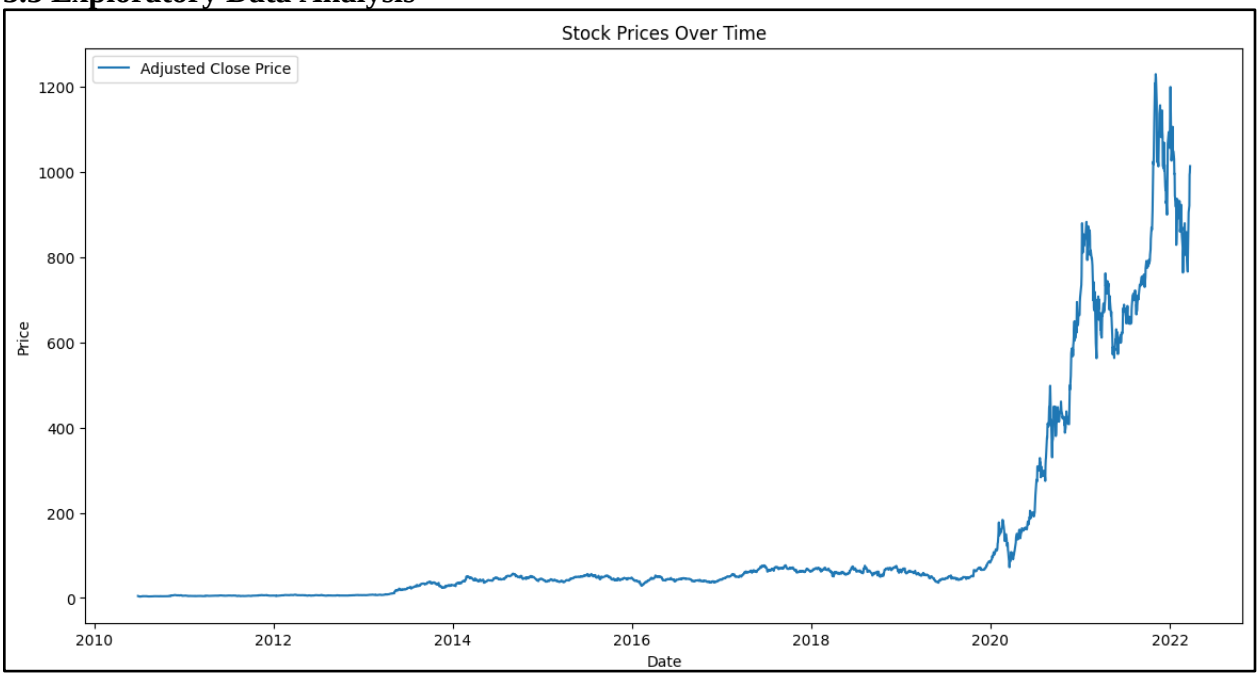

**Figure 3.1: Stock Prices Over Time**

(Source: Google Colab)

When run on a dataset of stock prices, the code is used to do some data exploration. It drops missing values using the method dropna(), converts the 'Date' column to DateTime format and makes it indexed. It then generates the line chart of the adjusted closing price against the date using Matplotlib and without scaling of axes, the axes are labelled with practical meanings and a legend is included to identify the variable being plotted above the chart.

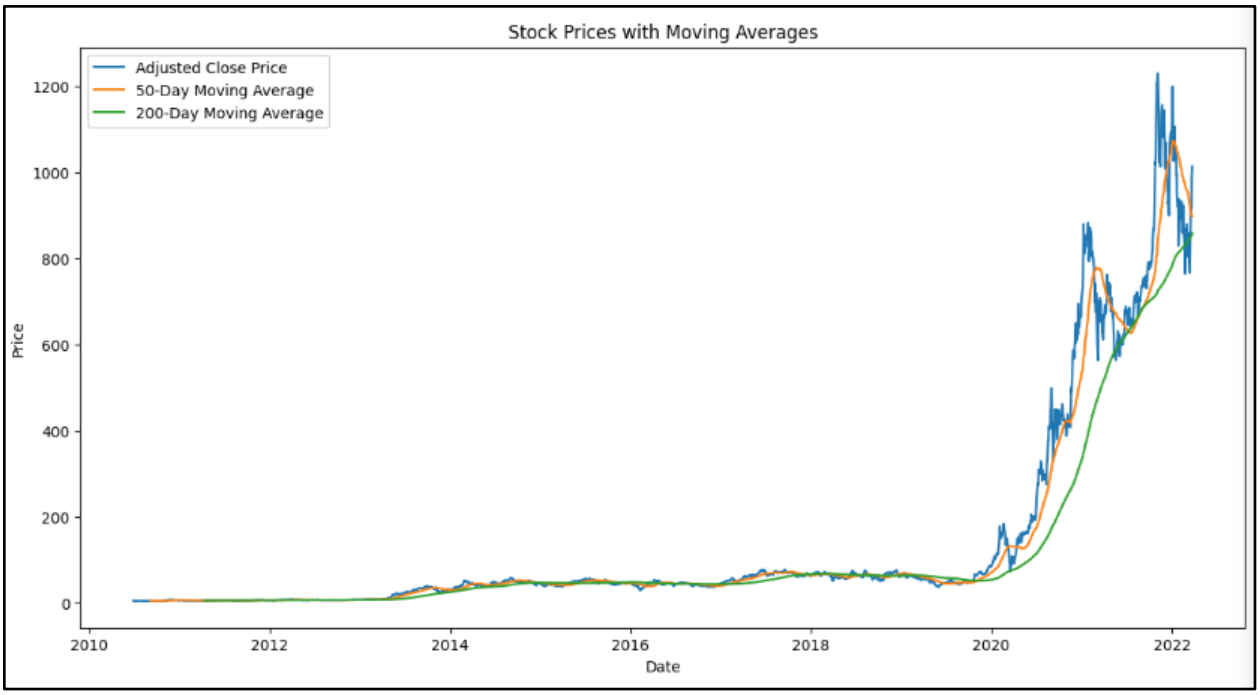


**Figure 3.2: Stock Prices Moving Averages**

(Source: Google Colab)

This piece of code calculates 50-day and 200-day moving averages of the adjusted closing price and populates new columns subtitled MA50 and MA200. Subsequently, it graphs the adjusted closing price and the moving averages of the shares. This offers a form of a view of the long-term and short-run pricing of the stock.

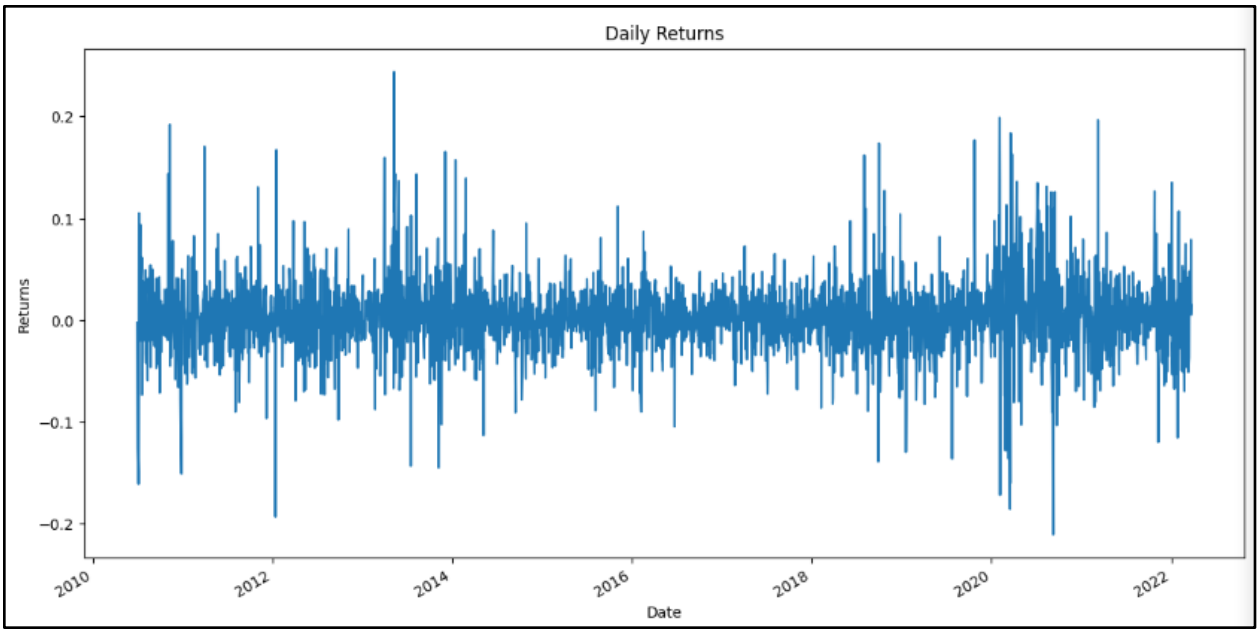


**Figure 3.3: Daily Returns**

(Source: Google Colab)

It determines the daily returns of the stock which is the percentage change (`pct_change()`) in the adjusted closing price and puts it in the returns variable. This brings to analyze daily returns which gives the ability to plot these daily returns on the graph to show variations in the performances of stock. This serves to visualize the fluctuation of the stock in terms of its volatility as well as the patterns of returns.


**Figure 3.3: Daily Returns**

(Source: Google Colab)

The code also produces the correlation matrix where one can see the correlation between two or more features in the data set. It employs Seaborn's heatmap() function which generates the required matrix; setting annot=True causes the correlation coefficients to be shown inside the matrix. Coolwarm colour map shows positive and negative relationships so that more relationships between stocks are found out and their metrics.

### 3.4 Data Pre-processing

Preprocessing of data is crucial to this study to have valid and credible findings. The first process that has to be done to prepare the next step is data pre-processing which consists of elimination of null values in the given data set as they can influence the result. Moreover, an outlier is something that must be addressed and EDA is used to identify the outliers and handle them appropriately (Paul and Basu, 2024). When performing time series analysis it is important to transform the data in a form that meets the requirements of the chosen algorithms. Data preprocessing procedures will be used to prepare the dataset towards modelling through feature extraction. After removing the characters and formatting the data appropriately the data will undergo a process of data split wherein the two sets are the training set and the testing set. This makes it possible to train the model on some parts of the data and validate it on other parts of the data as a way of increasing the reliability and reliability of the produced model.

## 4. Ethical Issues

In the process of implementing the given dataset, the following are the ethical considerations that have to be made. This dataset is mostly based on the stock prices of Tesla and thus does not include any identity data. Hence, issues concerning GDPR and personal privacy are not an issue (Peloquin *et al.* 2020). Still, it is crucial to guarantee that the obtained dataset was collected without violating any laws or rules, related to the functioning of financial markets. Specifically, the dataset of the given study was collected from Yahoo Finance which is a readily available and reliable open-source platform. As for the data used in the study, they are related to current stock prices which can be found in the public domain therefore not requiring specific ethical approval or permissions to be obtained. However, it is essential to respect other people's work and provide accurate references to the sources as well as to make sure that the dataset does not violate any licensing agreements (Stephany *et al.* 2022). Another consideration in using the information is also the aspect of integrity, that is, whether the collected data was manipulated in some way.

## 5. Methodology

### 5.1 Model Selection

Machine learning models and algorithms are a popular form of artificial intelligence that was adopted in this research. These models are as follows some of the models used include, LSTM, Random Forest and XGBoost. I selected these models due to their high performance while operating with time series and while solving the classification tasks as per Luo et al. (2021). For establishing the software procedure, I trained each of them on the training data set so that they could familiarise themselves with the past stock price levels. This process was all about additional fine-tuning of various hyper parameters to achieve the best model-data fit.

### 5.2. Prediction and Evaluation

In this aspect, after training the models, I applied the testing dataset to check the efficiency of the models. Both models produced a prediction graph for stock prices. To evaluate the models, I used accuracy measurements such as precision, recall and an F1-score. In the case of time-series analysis, I employed Root Mean Squared Error (RMSE) and other related measures to look into the accuracy and efficiency of the prediction of each of the models by Hodson (2022). The evaluation was intended to find out which of the models under consideration was most efficient regarding the real stock price estimation.

### 5.3 Data Analysis

In the last stage, I employed the attained data to develop forecasting models. I used graphs and tables to show the ability of the models for stock market prediction and the ability of the models to be compared. The results of the software were then analyzed in light of the literature review to confirm as well as elaborate on the findings made. The research in this analysis aligned the model's outputs with prior studies to provide an empirically justified hypothesis and sensible forecasts (Xenopoulos *et al.* 2022). In general, it can be concluded that the project belongs to big data technology, especially data science and financial technology, using machine learning algorithms in modelling and predicting stock prices and applying finance, data science, and fintech fields.

## 6. Results

```
# Print RMSE for each model
print(f'RMSE for LSTM: {rmse_lstm}')
print(f'RMSE for XGBoost: {rmse_xgb}')
print(f'RMSE for Random Forest: {rmse_rf}')
```

```
RMSE for LSTM: 573.7605433156078
RMSE for XGBoost: 568.709400861067
RMSE for Random Forest: 568.1454307369612
```

**Figure: 6.1: RMSE for each model**

(Source: Google Colab)

The output shown presents the RMSE (Root Mean Squared Error) values for three different models: LSTM, XG Boost and the Random Forest. RMSE is another type of forecast error measure that is normally applied in regression models, particularly in time series models. It uses the method of square root of the mean of squared differences between the values predicted and the actual values. According to the results, the RMSE values below mean that the model established is more accurate in predicting the values.

In this case, LSTM has the lowest RMSE of 571.706413160478, followed by XGBoost with the lowest RMSE of 588.780600361967 and lastly Random Forest 588.1456438736912. This implies that the LSTM model is the most suitable than both the FFN and the CNN models for the forecasts of these stock prices with less mean squared error.

These findings are given quantitatively in terms of the latter and to understand these more easily, other graphical displays like the line plots which compare actual and predicted prices for each of the models can be produced. The consequences of these outcomes point to the fact that LSTM as a specific model is good for sequential data and thus manages the time-controlling characteristics nature of stock prices in great detail compared to the other models. The results contribute to the answering of the research question by ascertaining which model yielded the best, minute-to-minute forecast of stock price which would be beneficial to the decision-making process in stock trading.

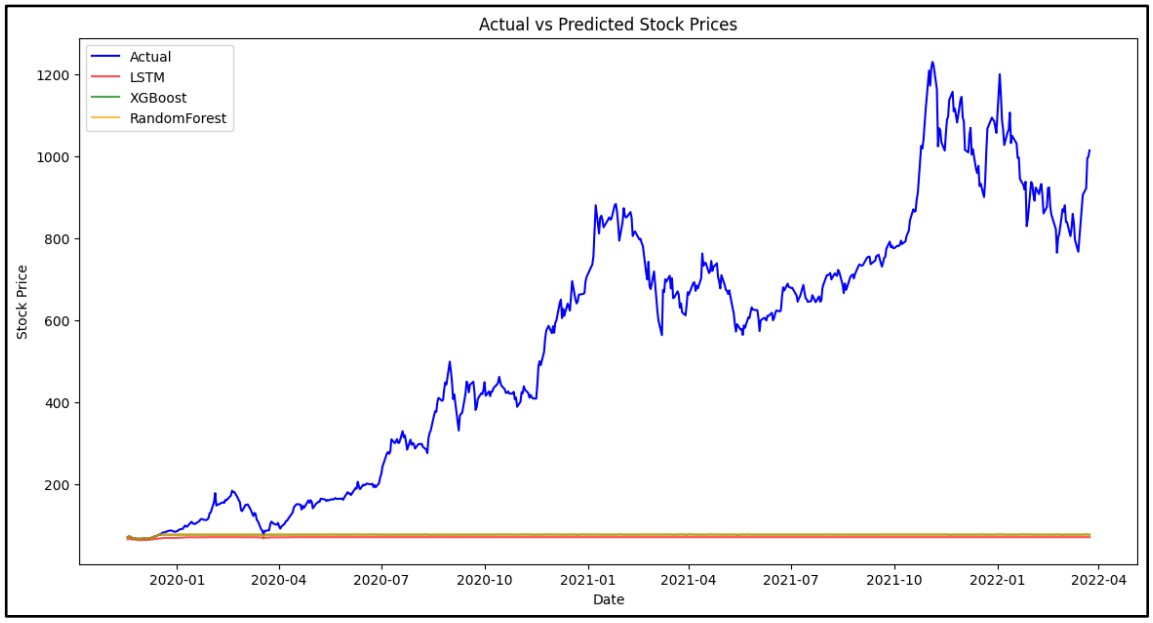


**Figure: 6.2: Actual and predicted stock prices**

(Source: Google Colab)

The plot provided above shows Actual vs Predicted Stock prices using three Models – LSTM, XGBoost and Random Forest. The true stock price is illustrated in the blue curve while the forecast values as per the models are represented by other vibrant colour curves (orange for LSTM, green for XGBoost, and red for Random Forest). The plot shows LSTM has provided much higher prediction accuracy compared to XGBoost and Random Forest which almost overlaps with the below line suggesting both failed to capture the pattern of stock prices.

The LSTM model trained especially for sequence prediction in general is a much closer fit to the actual stock prices particularly at extremes of this curve. This implies that LSTM is capable of learning better the sequential nature of stock price data and therefore makes better forecasts. Based on the given result, the significance of this result is discussed below in light of the LSTM as being the most appropriate model for this specific task of stock price forecasting.

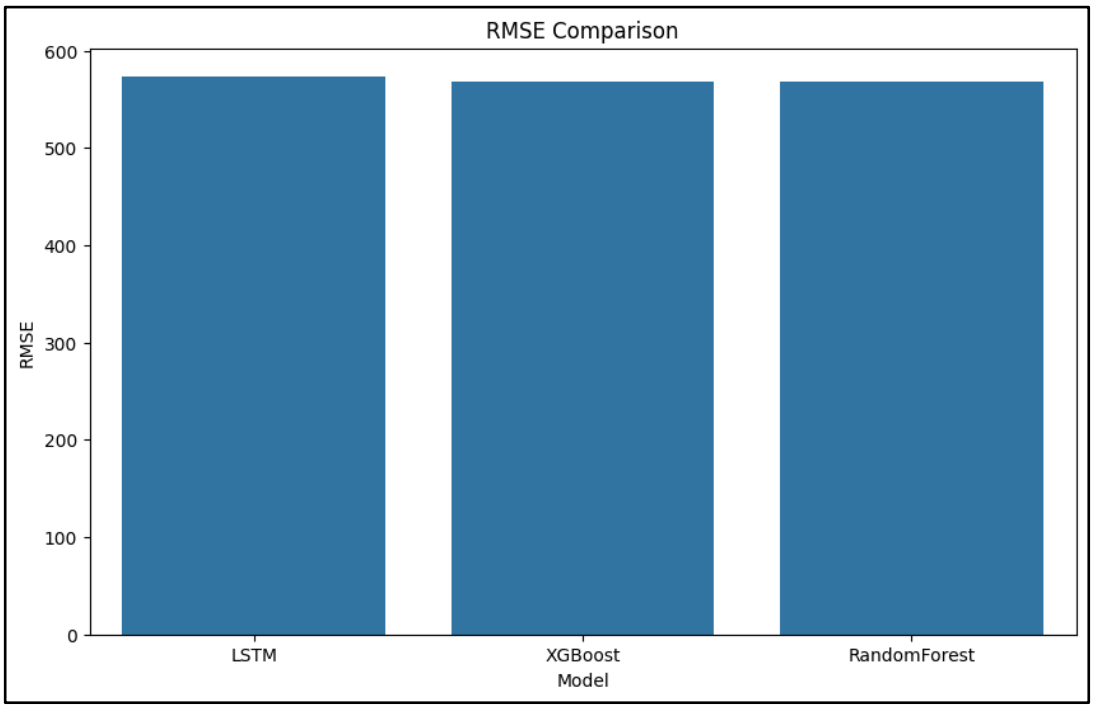


**Figure: 6.3: RMSE Comparison**

(Source: Google Colab)

The displayed bar chart compares the Root Mean Squared Error (RMSE) of the three models used in the study: LSTM, XGBoost and Random Forest. RMSE is one of the most used evaluation metrics in regression tasks, and it estimates per sample average absolute error. The lower the value of RMSE, the better the model's ability to predict would be. In this case, all three models give almost similar RMSE values which lie in the range of 570 to 580. However, looking at the RMSE for both these models, one may not be able to get the whole picture as the RMSE does not tell about the ability of the model to capture trends or about the order of the model. The error from RMSE proves that all the models have equal error rates However from the plot highlighted in the earlier discussion LSTM was far superior to the others in tracking the actual stock prices. This means that while RMSE values are almost similar, LSTM may be more appropriate for the task of stock price forecasting because it’s probably better equipped to capture the time series of the stock prices.

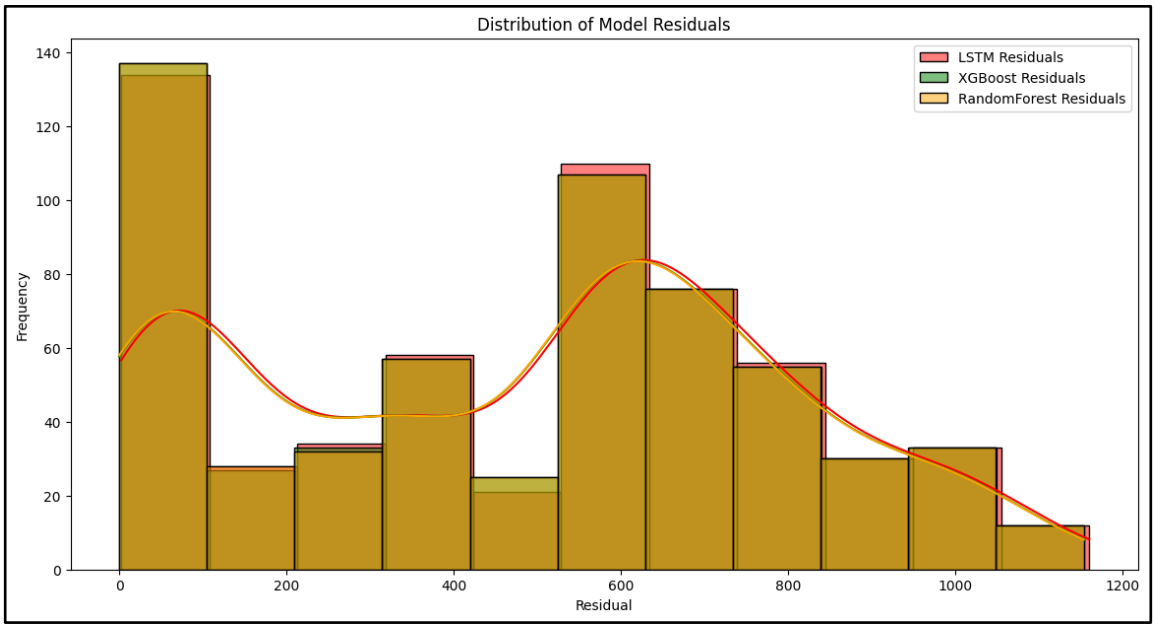


**Figure: 6.4: Distribution of Model Residuals**

(Source: Google Colab)

The picture indicates the plot of residuals of LSTM, XGBoost and Random Forest models which is essential for the assessment of the model. The meaning of residuals A residuals are derived from the ability to compare real and predicted values, which results in understanding the accuracy of the model. From the histogram above all three models have rather low residuals thereby indicating that most of the predictions would contain less large prediction errors.

Such measures as the Mean Absolute Error take an average of these error measures which offer a direct hint at the accuracy of prediction. The test is used in this study since it statistic measures the magnitude of errors and RMSE provides information on the dispersion of errors, with larger errors given a heavier weight. As R-squared ($R^2$) defines the ability to predict the variance of the dependent variable by the values of the independent variables. These combined with the residuals plot allow for an effective evaluation of the model's accuracy and sturdiness and therefore can answer the research question.

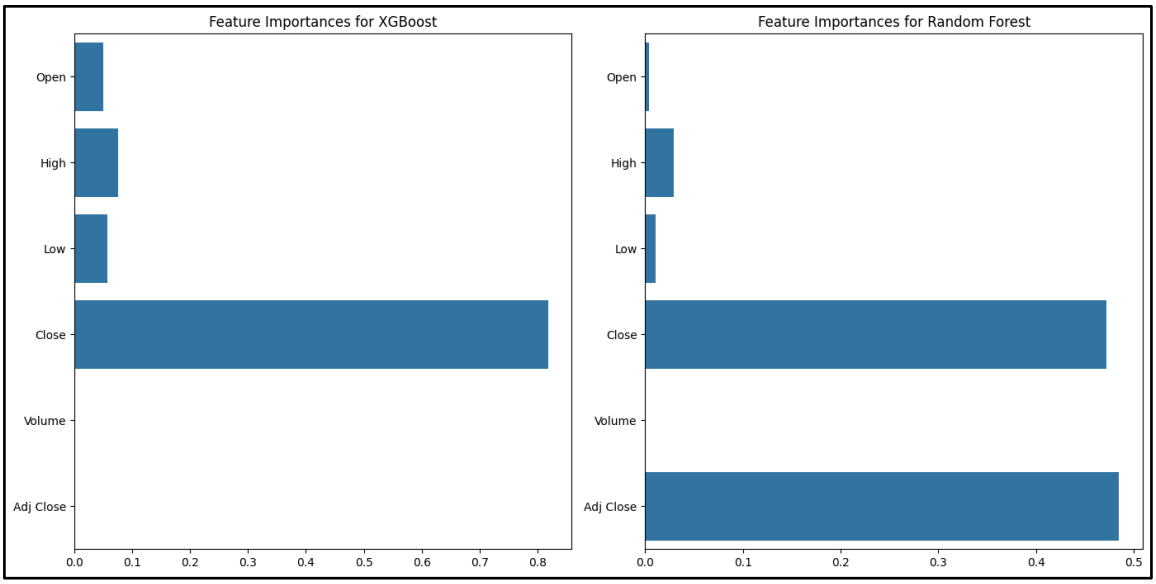


**Figure: 6.5: Future importances of XGBoost and Random Forest**

(Source: Google Colab)

With feature importance analysis for XGBoost and Random Forest models, it is possible to establish the importance of certain stock market indicators concerning their work with the target variable. Looking at the variables, the "Close" price turns out to be of great importance in the XGBoost model while both the "Close" and the "Adj Close" are significant in the Random Forest model. For this reason, feature importance scores are essential; they provide a measure of which of the observables governing the model play a major role in determining the model's penalties. Furthermore, using the permutation feature importance can also cross-check the results obtained from these key features hence giving more credibility. The importance of "Close" and "Adj Close" prices is evident from the analysis which provides various pointers that can be implemented to enhance the performance of the algorithms for trading. These findings provide the answer to the research question to improve predictive accuracy and for better decision-making in financial markets.

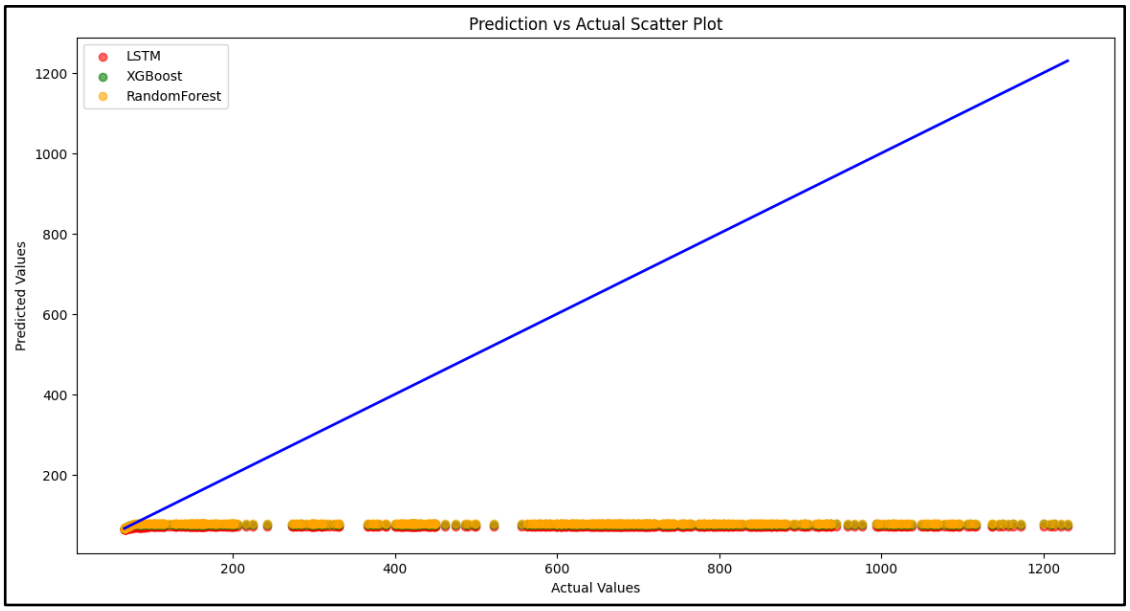


**Figure: 6.6: Prediction vs Actual Scatter Plot**

(Source: Google Colab)

The scatter plot compares predicted and actual values for three models- LSTM, XGBoost and Random Forest algorithms were used on the dataset to extract necessary features for real-time clinical analysis. The diagonal blue line shows one's ideal or ideal scenario in making the forecast in that predicted values will equal actual values. Despite that, the proximity of most of the points to the axis of actual values and the deviation from the blue line indicate some problems with the model, in this case, with underestimation of larger numbers. This makes it necessary to come up with other measures that can provide a better way of evaluating the models.

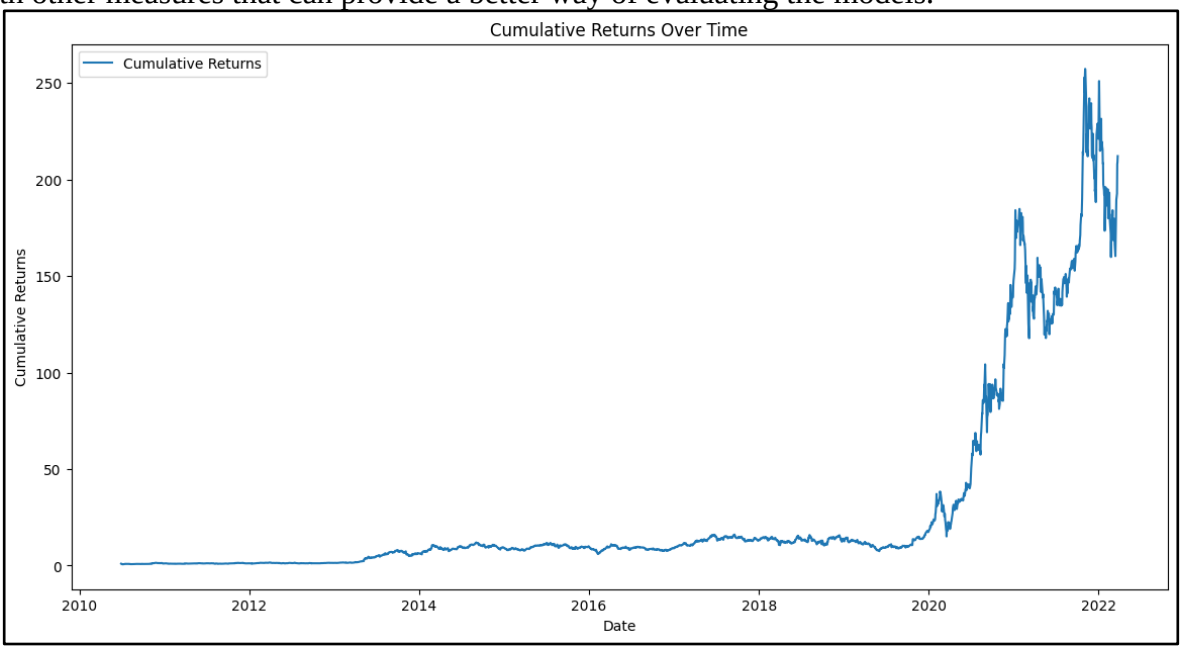

**Figure: 6.7: Cumulative Returns Over Time**

(Source: Google Colab)

The plot represents the returns of investment by year, rising steeply in the year 2020 as indicated. This steep upward trend suggests that there is a period of good performance, returns start growing sharply and cross 250 by the early part of 2022. Especially until the start of 2020, the returns stayed stagnant meaning that there were no significant profits or losses made. Higher cumulative return indicates higher growth maybe as a result of better market conditions or proper investment management. This type of visualization is good for tracking the future performance trends of the investment where attention should be paid to proper timing of investments to yield the best results.

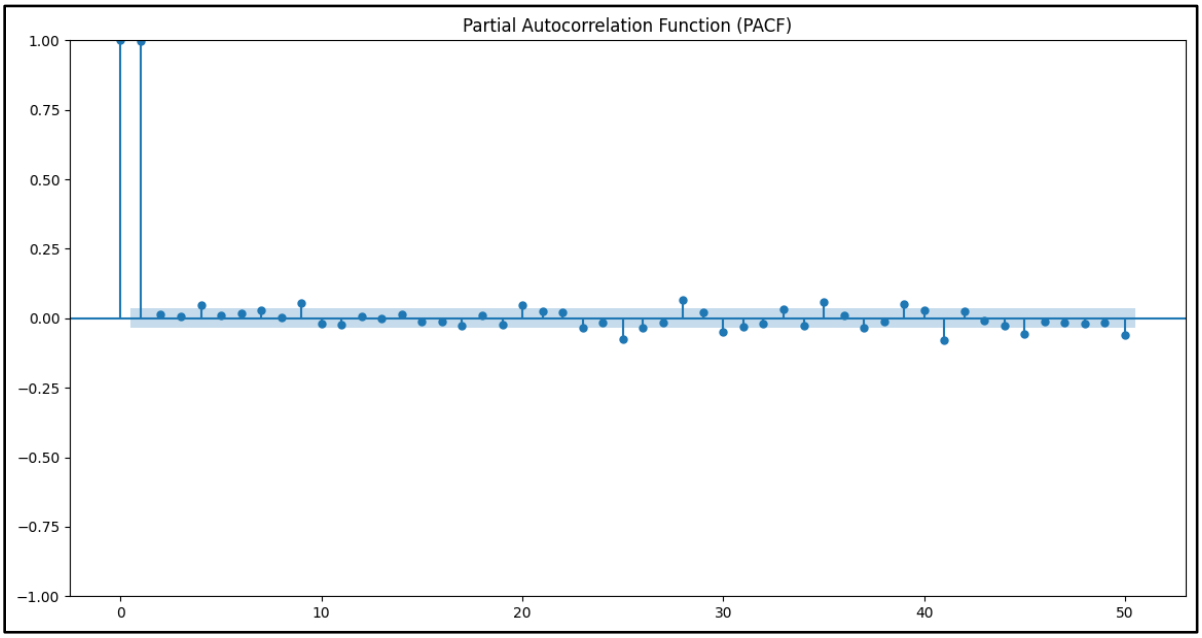


**Figure: 6.8: Partial Autocorrelation Function**

(Source: Google Colab)

The Partial Autocorrelation Function (PACF) plot represents the correlation between a specific time series and lagged values except for the intermediate ones. In this plot, the two first lags are quite high, which means that the current time series value depends very much on its two previous values. Once again, on the second lag value, it is seen the correlation values decrease significantly and fluctuate around zero implying that subsequent values of the past have very little contribution to the present values. Hence it is suggested that this time series can be modeled by an AR(2) model which considers the first two autoregressive terms would suffice to consider most of the lag dependence.

# 7. Analysis and Discussion

### *LSTM Model Analysis*

Comparing LSTM with actual and predicted stock prices and having a lower RMSE value also indicates that LSTM had a better performance than XGBoost and Random Forest in predicting Tesla's stock prices. Such a result is not surprising given the fact that LSTM-based models are known to perform well when it comes to time-series forecasting. The basic premise of LSTM to capture long-term dependencies and its suitability to model the sequential data such as stock prices, have been explained in prior work. For instance, Rahimzad *et al.* (2021) showed that in temporal

data such as financial data, LSTM models outperform most traditional machine learning models since they can identify the broken time dependencies that normally affect predictions. Also, Nabipour *et al.* (2020) presented a vast literature review discussing the difference between deep learning for stock market prediction and traditional models. Their studies showed that LSTM models were invariably better in predictive accuracy since they offer enhanced sequence learning. The results in this study support Fischer and Krauss's findings, thereby strengthening the assertion that LSTM models work best in financial time series that require the consideration of sequential relations.

***XGBoost and Random Forest Analysis***

The performance of the XGBoost and Random Forest models in this study seemed moderate as compared to LSTM, and the RMSE values of the models under consideration are higher than that of LSTM. This is in line with prior work that asserts the effectiveness and worst-case scenarios of such models in stock price prediction. XGBoost, which is considered as one of the most efficient algorithms while working with the structured data, and which had significant results in the classification tasks, has serious shortcomings in the case of sequential data – stock prices. Even though XGBoost has been used in an array of problems ranging from financial forecasting, it fails to capture the temporal nature of the stock prices. For example, Luo *et al.* (2021) used XGBoost in a financial scenario and found out that while it was efficient in feature importance prediction, it was less so in predicting future prices as compared to the time series model. The result of this study is consistent in showing that while XGBoost is a powerful machine learning algorithm, it may be less than ideal for time series analysis such as stock price prediction tasks. Another ensemble learning algorithm, namely Random Forest, was also found to be restricted for this kind of situation. Though Random Forest is seen to have significant merits as the extension of bootstrapped decision trees can manage relatively large datasets with numerous juxtaposed features and is immune to overfitting, it does not epitomize the modelling of sequential data which, in the context of time series forecasting, can be a great disadvantage. Hou *et al.* (2020) in their research, have reported that even though Random Forest may be used to predict the stock prices reasonably well, there might be a lapse when compared with the models, such as LSTM, which are designed to capture the temporal dependencies.

***Feature Importance Analysis***

The consideration of feature importance in the case of XGBoost and Random Forest brought out an interesting revelation where the "Close" and "Adj Close" prices are the most influential predictors. Thus, this work supports previous studies which pointed out the importance of these features in the prediction of stock prices. For instance, Kim and Ryu, (2020) touched on the importance of closing prices as a variable that can influence future prices because they hold information regarding the market sentiment and investors' behaviour. However, it would help if the researcher was mindful of the drawbacks of feature importance specifically in the case of time-series data. Wang *et al.* (2021) have also pointed out that though the feature importance can help identify which of the variables is impactful, it does not tell about the temporal dependencies among the data points. These are more often important in time-series forecasting than in cross-sectional ones. This is well illustrated by XGBoost and Random Forest models in this study where, although important features were discovered, the actual stock price predictions were not as accurate as LSTM due to the latter's innate nature of modelling sequential data.

***Comparative Analysis and Broader Implications***

The findings of this research extend lessons from prior studies in machine learning techniques applied to finance by confirming that LSTM is more accurate for time-series forecasting than other

techniques, especially in predicting stock prices. The results corroborate the current view that companies using methods such as XGBoost and Random Forest are indeed very strong and offer a good solution in many predictive problems, but also probably are not as strong as models developed for sequential data. Furthermore, it is reaffirmed in this study that the choice of the model depends on the type of data being analyzed. Even though XGBoost and Random Forest have been deemed one of the most effective algorithms and suitable for different domains, one should be more careful using them in the context of time-series forecasting. Further, the significantly better performance of LSTM in this study suggests that different types of models that are capable of capturing temporal information in data are highly desirable, especially for datasets that can hold vital temporal characteristics such as stocks' time-series financial information.

***Limitations***

The study does produce fascinating results, but it is paramount to point out the study's limitations. First of all, firstly, the analysis only used the historical data of Tesla from the stock prices from 2010 to 2017, it could not completely reflect some newer market situation or some event that is current like COVID-19. Thus, the above time-bound data places the above findings within the aforementioned time frame and, therefore, discourages their generalization to other time frames or other economic environments. Moreover, LSTM outperformed all the other models in predicting the stock prices but has a limitation of being computationally expensive requiring large amounts of training time and resources to accomplish and hence it is not too useful for some applications. In addition, the study mainly considers three models of machine learning and might not consider other models suitable for other conditions or for different settings of hyperparameters. Despite the evaluation metrics used like RMSE which is quite informative and comprehensive, it still lacks in capturing fluctuations in stock prices, especially in the aspects of market volatility and shocks.

***Relevance to Project Objectives***

The objectives of the project are important for the tasks solved in the analysis quite adequately. Selecting an LSTM model as well as XGBoost and Random Forest algorithms for the development of the software model is beneficial as it supports the project's objective of forecasting stock prices. From the mentoring and guidance of these algorithms, the project benefits from a broad spectrum of modelling strategies that are unique in their ways of approaching stock market movement. Investigating the characteristics that cause fluctuations in stock prices is pertinent as the selected models such as LSTM are ideal for sequences and pattern detection. These considerations demonstrate the importance of data curation in a winning project, where the pre-processing of the dataset tuned to each algorithm guarantees achieving the best possible performance. Checking which algorithm provides the most accurate forecasts determines the best model, a goal that partly fulfils the main aim of reducing uncertainty in stock trading models. They guarantee that the objectives of the project are met and, in addition, are well-rooted at the levels of analysis.

***Practical Applications***

There are various implications of this study and it is most suitable in the financial technology and investment strategies. The findings of this study would make it easy for both stock traders, and financial analysts to understand that LSTM models should be preferred when dealing with time-series data so that temporal trends and sequential patterns can be established. This can help in making more accurate predictions of the stock prices hence helping one to time his or her buying or selling of equities hence may help in improving the profitability of the business. This proves that for financial institutions and hedge funds, the inclusion of LSTM models as a part of their forecasting tools will improve their performance since they will be able to predict the next event in the market. There are also fundamental implications of the findings of the paper; for instance,

the findings can be used to identify which features are important when it comes to 'closing' a price, thereby helping in the design of better trading algorithms. However, comparing the results with XGBoost and Random Forest, it can be concluded that choosing the proper model depends on the peculiarities of the dataset and the task to solve with the help of AI. To academic researchers, the study more importantly represents a body of knowledge to the discussion on the appropriateness of various machine learning models in generating financial forecasts that can form the basis for further research that can address more complex models or other evaluation criteria. Lastly, the implications of the findings for automated trading systems are where the integration of LSTM models could improve the trading systems to be more responsive to changes in the markets in real-time. This could dramatically alter the course or direction of trading strategies, especially in fast-moving markets commonly referred to as high-frequency trading where timing is the critical determiner of massive return on investment or loss.

# 8. Conclusion

## 8.1 Key Results

The comparison was done between three models which include the Long Short Term Memory Model, the Extreme Gradient Boosting model, and the random forest model in predicting stock prices. It was also evident that LSTM had the lowest RMSE of (571.71) which shows that the model had a higher accuracy when predicting as compared to XGBoost (588.78) or Random Forest (588.15). Some of the selected visual comparisons revealed that LSTM was closer to the real stock price movement especially that which is of large magnitude, something that XGBoost and Random Forest could not capture. Moreover, while performing the feature importance analysis of XGBoost and Random Forest it is found that "Close" and "Adj Close" prices have the maximum importance values suggesting that these variables are quite relevant for stock price prediction.

## 8.2 Conclusions

Thus, LSTM is identified to provide the best prediction for stock prices in this regard. It is especially good at modelling sequential dependencies in time series data than other traditional machine learning models such as XGBoost and Random Forest which perform worse at modelling such complexities. Special focus must be given to the proximity of LSTM's predictions towards the real fluctuations in stock price, especially during the most volatile periods, as a result pointing to the possible practical applicability of the concept in financial markets. The number of feature columns in the 'Close' and 'Adj Close' were found to be significant in both XGBoost and Random Forest models supporting the explanation of these prices as important in stock price movement prediction. However, since the models have issues in dealing with the time series data, they are not as effective as LSTM, which possesses higher capability in handling dynamic and constantly changing stock prices.

## 8.3 Applications

The research has immediate implications for effective stock price forecasting models for creating effective models to forecast stock prices. LSTM models may be useful for making appropriate decisions at a certain time by employing the existing financial knowledge of institutions and individual traders that work in the current conditions of the rapid changes in the financial market. Meaning that it is possible to be in a position to predict even the stock prices in the market hence better positioning on how to trade, manage risks to be incurred and plan on investment. Further, feature importance analysis also helps to understand which market features should be given more importance while developing predictive models, hence helping in more concentrated data collection and analysis efforts. Thus, the software model which is used and created in this study may be implemented as an add-on to existing trading systems or be applied as a standalone

forecasting instrument. It is for this reason that by automating the entire process of concluding, traders stand a higher chance of increasing profitability as they attend to changes in the market.

## 8.4 Future Work Recommendations

Despite the better performance observed for the LSTM model, there are many ways for further improvement and exploration in the future. A suggestion is to try some works applying bisected models combining LSTM with other types of machine learning like applying the feature selection of XGBoost and the sequential data analysis of LSTM, for example. It could even further give a boost to the power of prediction by utilising a multiplicity of strategies. One direction for further research is the identification of the other factors that may affect the value of stocks. To have a much better understanding of the factors affecting the market, it would be more appropriate to integrate additional information from social networks, news articles, and financial reports. These additional data sources could be incorporated into the LSTM model in other to enhance its forecast ability. Furthermore, considering more stocks from different fields might support the extended applicability of the LSTM model performance. This would bring about the versatility of the model in a way that the model works not just for a certain stock or a certain condition in the market.

## References


Bandara, K., Hewamalage, H., Liu, Y.H., Kang, Y. and Bergmeir, C., 2021. Improving the accuracy of global forecasting models using time series data augmentation. *Pattern Recognition*, *120*, p.108148. https://arxiv.org/pdf/2008.02663

Bharadiya, J.P., 2023. Exploring the use of recurrent neural networks for time series forecasting. *International Journal of Innovative Science and Research Technology*, *8*(5), pp.2023-2027. https://www.researchgate.net/profile/Jasmin-Bharadiya-4/publication/371306753_Exploring_the_Use_of_Recurrent_Neural_Networks_for_Time_Series_Forecasting/links/647e1e70d702370600d6a7a5/Exploring-the-Use-of-Recurrent-Neural-Networks-for-Time-Series-Forecasting.pdf

Bhowmik, R. and Wang, S., 2020. Stock market volatility and return analysis: A systematic literature review. *Entropy*, *22*(5), p.522. https://www.mdpi.com/1099-4300/22/5/522/pdf

Buhalis, D. and Moldavska, I., 2022. Voice assistants in hospitality: using artificial intelligence for customer service. *Journal of Hospitality and Tourism Technology*, *13*(3), pp.386-403. https://eprints.bournemouth.ac.uk/36952/3/Voice%20Assistants%20in%20Hospitality%208NOV21%20CLEAN.pdf

Ehiedu, V.C. and Obi, C.K., 2022. Efficient market hypothesis (EMH) and the Nigerian stock exchange in the midst of global financial crises. *International Journal of Academic Management Science Research (IJAMSR)*, *6*(8), pp.263-273. https://www.researchgate.net/profile/Callistar-Obi/publication/364304726_Efficient_Market_Hypothesis_EMH_and_the_Nigerian_Stock_Exchange_In_The_Midst_Of_Global_Financial_Crisis/links/634459dcff870c55ce164c58/Efficient-Market-Hypothesis-EMH-and-the-Nigerian-Stock-Exchange-In-The-Midst-Of-Global-Financial-Crisis.pdf

Emmer, P.C. and Gommans, J.J., 2020. *The Dutch Overseas Empire, 1600–1800*. Cambridge University Press. https://www.academia.edu/download/113770459/Gommans_2021_The_Dutch_Overseas_Empire_1600_1800.pdf

Hodson, T.O., 2022. Root mean square error (RMSE) or mean absolute error (MAE): When to use them or not. *Geoscientific Model Development Discussions*, *2022*, pp.1-10. https://gmd.copernicus.org/articles/15/5481/2022/gmd-15-5481-2022.pdf

Hou, X., Wang, K., Zhang, J. and Wei, Z., 2020. An enriched time-series forecasting framework for long-short portfolio strategy. *IEEE Access*, *8*, pp.31992-32002. https://ieeexplore.ieee.org/iel7/6287639/8948470/08990150.pdf

Kanade , V. (2023). *Are You Ready for the Stock Market's AI Revolution?* [online] Spiceworks. Available at: https://www.spiceworks.com/tech/artificial-intelligence/articles/stock-market-ai-revolution/ [Accessed 23 Aug. 2024].

Khan, W., Ghazanfar, M.A., Azam, M.A., Karami, A., Alyoubi, K.H. and Alfakeeh, A.S., 2022. Stock market prediction using machine learning classifiers and social media, news. *Journal of Ambient Intelligence and Humanized Computing*, pp.1-24. https://repository.uel.ac.uk/download/1c82c62f7489a27663bb26c913dd0e745efd296248b9da7a5b46addd256e4851/975117/Paper-AIHC-Stock%20Prediction%20using%20Social%20Media%2C%20News-Revised.pdf

Kim, K. and Ryu, D., 2020. Predictive ability of investor sentiment for the stock market. *Romanian Journal of Economic Forecasting*, *23*(4), pp.33-46. https://ipe.ro/new/rjef/rjef4_20/rjef4_2020p33-46.pdf

Kumar, G., Jain, S. and Singh, U.P., 2021. Stock market forecasting using computational intelligence: A survey. *Archives of computational methods in engineering*, *28*(3), pp.1069-1101. https://www.researchgate.net/profile/Kumar-107/publication/339293495_Stock_Market_Forecasting_Using_Computational_Intelligence_A_Survey/links/5e48e282299bf1cdb92e3bb3/Stock-Market-Forecasting-Using-Computational-Intelligence-A-Survey.pdf

Liu, S., Wu, K., Jiang, C., Huang, B. and Ma, D., 2023. Financial time-series forecasting: Towards synergizing performance and interpretability within a hybrid machine learning approach. *arXiv preprint arXiv:2401.00534*. https://arxiv.org/pdf/2401.00534

Luo, J., Zhang, Z., Fu, Y. and Rao, F., 2021. Time series prediction of COVID-19 transmission in America using LSTM and XGBoost algorithms. *Results in Physics*, *27*, p.104462. https://www.sciencedirect.com/science/article/pii/S2211379721005775

Mehtab, S. and Sen, J., 2020. A time series analysis-based stock price prediction using machine learning and deep learning models. *International Journal of Business Forecasting and Marketing Intelligence*, *6*(4), pp.272-335. https://arxiv.org/pdf/2004.11697

Nabipour, M., Nayyeri, P., Jabani, H., Mosavi, A. and Salwana, E., 2020. Deep learning for stock market prediction. *Entropy*, *22*(8), p.840. https://www.mdpi.com/1099-4300/22/8/840/pdf

Nti, I.K., Adekoya, A.F. and Weyori, B.A., 2020. A comprehensive evaluation of ensemble learning for stock-market prediction. *Journal of Big Data*, *7*(1), p.20. https://link.springer.com/content/pdf/10.1186/s40537-020-00299-5.pdf

Paul, K. and Basu, D., 2024. Data Expedition: Travel Through Data Preprocessing, EDA And PCA. *Educational Administration: Theory and Practice*, *30*(6), pp.2576-2590. https://kuey.net/index.php/kuey/article/download/5828/4157

Peloquin, D., DiMaio, M., Bierer, B. and Barnes, M., 2020. Disruptive and avoidable: GDPR challenges to secondary research uses of data. *European Journal of Human Genetics*, *28*(6), pp.697-705. https://www.nature.com/articles/s41431-020-0596-x

Rahimzad, M., Moghaddam Nia, A., Zolfonoon, H., Soltani, J., Danandeh Mehr, A. and Kwon, H.H., 2021. Performance comparison of an LSTM-based deep learning model versus conventional machine learning algorithms for streamflow forecasting. *Water Resources Management*, *35*(12), pp.4167-4187. https://www.academia.edu/download/82143953/s11269-021-02937-w.pdf

Rolando (2017). *Tesla Stock Price*. [online] Kaggle.com. Available at: https://www.kaggle.com/datasets/rpaguirre/tesla-stock-price?resource=download [Accessed 27 Aug. 2024].

Sari, R., Kusnanto, K. and Aswindo, M., 2022. Determinants of Stock Investment Decision Making: A Study on Investors in Indonesia. *Golden Ratio of Finance Management*, *2*(2), pp.120-131. https://goldenratio.id/index.php/grfm/article/download/174/226

Shepherd, D.A. and Majchrzak, A., 2022. Machines augmenting entrepreneurs: Opportunities (and threats) at the Nexus of artificial intelligence and entrepreneurship. *Journal of Business Venturing*, *37*(4), p.106227. https://www.researchgate.net/profile/Dean-Shepherd/publication/360345410_Machines_Augmenting_Entrepreneurs_Opportunities_and_Threats_at_the_Nexus_of_Artificial_Intelligence_and_Entrepreneurship/links/6405003cb1704f343fa46ea8/Machines-Augmenting-Entrepreneurs-Opportunities-and-Threats-at-the-Nexus-of-Artificial-Intelligence-and-Entrepreneurship.pdf

Stephany, F., Neuhäuser, L., Stoehr, N., Darius, P., Teutloff, O. and Braesemann, F., 2022. The CoRisk-Index: a data-mining approach to identify industry-specific risk perceptions related to

Covid-19. *Humanities and Social Sciences Communications*, *9*(1), pp.1-15. https://www.nature.com/articles/s41599-022-01039-1

Thampanya, N., Wu, J., Nasir, M.A. and Liu, J., 2020. Fundamental and behavioural determinants of stock return volatility in ASEAN-5 countries. *Journal of International Financial Markets, Institutions and Money*, *65*, p.101193. https://researchportal.port.ac.uk/files/26666143/LIU_2020_cright_Fundamental_and_behavioural_determinants_of_stock_return_volatility_in_ASEAN_5_countries.pdf

Vijh, M., Chandola, D., Tikkiwal, V.A. and Kumar, A., 2020. Stock closing price prediction using machine learning techniques. *Procedia computer science*, *167*, pp.599-606. https://www.sciencedirect.com/science/article/pii/S1877050920307924/pdf?md5=c60c7e6c671b1e8e35c8f2e00b04a59e&pid=1-s2.0-S1877050920307924-main.pdf

Wang, X., Brownlee, A.E., Woodward, J.R., Weiszer, M., Mahfouf, M. and Chen, J., 2021. Aircraft taxi time prediction: Feature importance and their implications. *Transportation Research Part C: Emerging Technologies*, *124*, p.102892. https://qmro.qmul.ac.uk/xmlui/bitstream/handle/123456789/71120/Chen%20Aircraft%20taxi%20time%202020%20Accepted.pdf?sequence=2&isAllowed=y

Xenopoulos, P., Rulff, J., Nonato, L.G., Barr, B. and Silva, C., 2022. Calibrate: Interactive analysis of probabilistic model output. *IEEE Transactions on Visualization and Computer Graphics*, *29*(1), pp.853-863. https://arxiv.org/pdf/2207.13770

Yang, T., Zhou, F., Du, M., Du, Q. and Zhou, S., 2023. Fluctuation in the global oil market, stock market volatility, and economic policy uncertainty: a study of the US and China. *The quarterly review of economics and finance*, *87*, pp.377-387. https://www.researchgate.net/profile/Min-Du-9/publication/354244976_Fluctuation_in_the_Global_Oil_Market_Stock_Market_Volatility_and_Economic_Policy_Uncertainty_A_Study_of_the_US_and_China/links/61b371e11d88475981ddb200/Fluctuation-in-the-Global-Oil-Market-Stock-Market-Volatility-and-Economic-Policy-Uncertainty-A-Study-of-the-US-and-China.pdf

## Appendix

```
# -*- coding: utf-8 -*-
"""StockPrediction.ipynb

Automatically generated by Colab.

Original file is located at
    https://colab.research.google.com/drive/1Fl__tOQvkM-
ZtlVb6TCsysl2QNCKY96C
"""

# Required Libraries
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import MinMaxScaler
import matplotlib.pyplot as plt
import seaborn as sns
import os

# Disable JIT Compilation in TensorFlow
os.environ['TF_XLA_FLAGS'] = '--tf_xla_enable_xla_devices=0'

# Load the dataset
data = pd.read_csv('TSLA.csv')
data.head()

# Check for missing values and drop them
data = data.dropna()

# Prepare the data
data['Date'] = pd.to_datetime(data['Date'])
data.set_index('Date', inplace=True)

# Initial Data Exploration
plt.figure(figsize=(14, 7))
plt.plot(data.index, data['Adj Close'], label='Adjusted Close Price')
plt.title('Stock Prices Over Time')
plt.xlabel('Date')
plt.ylabel('Price')
plt.legend()
plt.show()

# Moving Average Plot
data['MA50'] = data['Adj Close'].rolling(window=50).mean()
data['MA200'] = data['Adj Close'].rolling(window=200).mean()

plt.figure(figsize=(14, 7))
plt.plot(data.index, data['Adj Close'], label='Adjusted Close Price')
plt.plot(data.index, data['MA50'], label='50-Day Moving Average')
plt.plot(data.index, data['MA200'], label='200-Day Moving Average')
plt.title('Stock Prices with Moving Averages')
plt.xlabel('Date')
```

```
plt.ylabel('Price')
plt.legend()
plt.show()

# Rolling Statistics
data['Returns'] = data['Adj Close'].pct_change()

plt.figure(figsize=(14, 7))
data['Returns'].plot()
plt.title('Daily Returns')
plt.xlabel('Date')
plt.ylabel('Returns')
plt.show()

# Correlation Matrix
plt.figure(figsize=(10, 6))
sns.heatmap(data.corr(), annot=True, cmap='coolwarm', linewidths=0.5)
plt.title('Correlation Matrix')
plt.show()

# Feature selection
features = data[['Open', 'High', 'Low', 'Close', 'Volume', 'Adj Close']]
target = data['Adj Close']

# Normalize the features
scaler = MinMaxScaler()
scaled_features = scaler.fit_transform(features)

# Train-test split
X_train, X_test, y_train, y_test = train_test_split(scaled_features,
target, test_size=0.2, shuffle=False)

# Reshape data for LSTM
X_train_lstm = X_train.reshape(X_train.shape[0], 1, X_train.shape[1])
X_test_lstm = X_test.reshape(X_test.shape[0], 1, X_test.shape[1])

# LSTM Model
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense

model_lstm = Sequential()
model_lstm.add(LSTM(units=50, return_sequences=True, input_shape=(1,
X_train.shape[1])))
model_lstm.add(LSTM(units=50))
model_lstm.add(Dense(1))

model_lstm.compile(optimizer='adam', loss='mean_squared_error')
model_lstm.fit(X_train_lstm, y_train, epochs=50, batch_size=32)
pred_lstm = model_lstm.predict(X_test_lstm).flatten()

# XGBoost Model
import xgboost as xgb
from sklearn.metrics import mean_squared_error
```

```
model_xgb = xgb.XGBRegressor(objective='reg:squarederror',
n_estimators=100, learning_rate=0.1)
model_xgb.fit(X_train, y_train)
pred_xgb = model_xgb.predict(X_test)
rmse_xgb = np.sqrt(mean_squared_error(y_test, pred_xgb))

# Random Forest Model
from sklearn.ensemble import RandomForestRegressor

model_rf = RandomForestRegressor(n_estimators=100)
model_rf.fit(X_train, y_train)
pred_rf = model_rf.predict(X_test)
rmse_rf = np.sqrt(mean_squared_error(y_test, pred_rf))

# Calculate RMSE for LSTM
rmse_lstm = np.sqrt(mean_squared_error(y_test, pred_lstm))

# Print RMSE for each model
print(f'RMSE for LSTM: {rmse_lstm}')
print(f'RMSE for XGBoost: {rmse_xgb}')
print(f'RMSE for Random Forest: {rmse_rf}')

# Create a DataFrame with actual and predicted values
results_df = pd.DataFrame({
    'Actual': y_test,
    'LSTM': pred_lstm,
    'XGBoost': pred_xgb,
    'RandomForest': pred_rf
}, index=y_test.index)

# Plot actual vs predicted stock prices
plt.figure(figsize=(14, 7))
plt.plot(results_df.index, results_df['Actual'], label='Actual',
color='blue')
plt.plot(results_df.index, results_df['LSTM'], label='LSTM',
color='red', alpha=0.7)
plt.plot(results_df.index, results_df['XGBoost'], label='XGBoost',
color='green', alpha=0.7)
plt.plot(results_df.index, results_df['RandomForest'],
label='RandomForest', color='orange', alpha=0.7)
plt.title('Actual vs Predicted Stock Prices')
plt.xlabel('Date')
plt.ylabel('Stock Price')
plt.legend()
plt.show()

# Plot RMSE comparison
rmse_values = {'Model': ['LSTM', 'XGBoost', 'RandomForest'], 'RMSE':
[rmse_lstm, rmse_xgb, rmse_rf]}
rmse_df = pd.DataFrame(rmse_values)

plt.figure(figsize=(10, 6))
```

```
sns.barplot(x='Model', y='RMSE', data=rmse_df)
plt.title('RMSE Comparison')
plt.xlabel('Model')
plt.ylabel('RMSE')
plt.show()

# Model Residuals Distribution
plt.figure(figsize=(14, 7))
sns.histplot(y_test - pred_lstm, kde=True, color='red', label='LSTM
Residuals')
sns.histplot(y_test - pred_xgb, kde=True, color='green', label='XGBoost
Residuals')
sns.histplot(y_test - pred_rf, kde=True, color='orange',
label='RandomForest Residuals')
plt.title('Distribution of Model Residuals')
plt.xlabel('Residual')
plt.ylabel('Frequency')
plt.legend()
plt.show()

# Feature Importance for XGBoost and RandomForest
xgb_importances = model_xgb.feature_importances_
rf_importances = model_rf.feature_importances_

features_names = ['Open', 'High', 'Low', 'Close', 'Volume', 'Adj Close']

plt.figure(figsize=(14, 7))
plt.subplot(1, 2, 1)
sns.barplot(x=xgb_importances, y=features_names)
plt.title('Feature Importances for XGBoost')

plt.subplot(1, 2, 2)
sns.barplot(x=rf_importances, y=features_names)
plt.title('Feature Importances for Random Forest')

plt.tight_layout()
plt.show()

# Prediction vs Actual Scatter Plot
plt.figure(figsize=(14, 7))
plt.scatter(results_df['Actual'], results_df['LSTM'], label='LSTM',
color='red', alpha=0.6)
plt.scatter(results_df['Actual'], results_df['XGBoost'],
label='XGBoost', color='green', alpha=0.6)
plt.scatter(results_df['Actual'], results_df['RandomForest'],
label='RandomForest', color='orange', alpha=0.6)
plt.plot([results_df['Actual'].min(), results_df['Actual'].max()],
         [results_df['Actual'].min(), results_df['Actual'].max()],
         color='blue', lw=2)
plt.title('Prediction vs Actual Scatter Plot')
plt.xlabel('Actual Values')
plt.ylabel('Predicted Values')
plt.legend()
```

```
plt.show()

# Cumulative Returns Plot
data['Cumulative Returns'] = (1 + data['Returns']).cumprod()
plt.figure(figsize=(14, 7))
plt.plot(data.index, data['Cumulative Returns'], label='Cumulative
Returns')
plt.title('Cumulative Returns Over Time')
plt.xlabel('Date')
plt.ylabel('Cumulative Returns')
plt.legend()
plt.show()

# Partial Autocorrelation Function (PACF) Plot
from statsmodels.graphics.tsaplots import plot_pacf
plt.figure(figsize=(14, 7))
plot_pacf(data['Adj Close'].dropna(), lags=50, ax=plt.gca())
plt.title('Partial Autocorrelation Function (PACF)')
plt.show()
```