Harnessing Machine Learning for Global Stock Price Forecasting: An Analysis Based on the Kaggle Dataset

Data Science Institute - DATA1030 Final Report - Hansen Wang
GitHub: https://github.com/hansen11/Data1030_Final_Project.git

1. Introduction

The fluctuating nature of the stock market has always been a focal point in financial research. Accurate prediction of stock prices is crucial for investors, as it assists in making informed investment decisions. Despite the complexity and unpredictability of stock market dynamics, machine learning offers new avenues for forecasting stock prices. This study aims to employ machine learning techniques to predict the future stock prices of companies worldwide, exploring the efficacy of different algorithms in practical scenarios.

This research utilizes the "World Stock Prices" dataset from Kaggle, provided and continuously updated by N. Elgiriyewithana [1]. The dataset comprises 279,753 entries, encompassing stock data for major companies globally. Each entry includes 12 features: date, opening price, highest price, lowest price, closing price, trading volume, dividend, stock split, brand name, stock code, industry label and country, where the closing price is the target feature. These features provide a comprehensive perspective for predicting stock prices.

Previous studies have attempted to address similar stock price prediction problems using machine learning techniques. These studies often employ various regression models, considering a range of features and market factors [2]. However, the high degree of uncertainty in the stock market and the variability of external influencing factors limit the predictive power of these models. The aim of this research is to explore and compare the performance of different machine learning models on this dataset, seeking to enhance the accuracy of predictions.

2. Exploratory Data Analysis (EDA)

In this section, we embark on an in-depth Exploratory Data Analysis (EDA) of the dataset, with a particular focus on the 'Close' stock prices. The stock market represents a complex and dynamic system influenced by a multitude of factors, ranging from economic conditions to investor sentiment. Our EDA endeavors to provide a comprehensive understanding of the dataset by employing statistical measures, data visualizations, and transformations.

Analyzing the 'Close' stock prices within the dataset reveals a wide range of values, stretching from as low as 0.20 to as high as 2153.20. This variation highlights the high volatility present in certain stocks. More specifically, the average closing price stands at 65.17, with a median of 32.33, suggesting that many stock prices are on the lower end. The high standard deviation of 117.19 further accentuates the extent of price fluctuations.

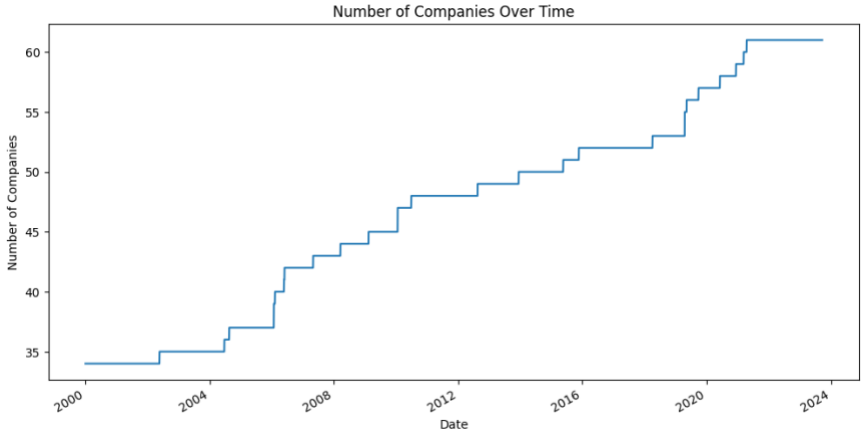

Figure 1. Counts for Companies over Time

Figure 1. A time series analysis shows a significant increase in the number of companies recorded in the dataset over time. Initially, only about 30 companies were included. As time progressed, more companies went public, culminating in approximately 63 companies being listed by the end of the study period. This trend reflects the expansion of the stock market and the active entry of new companies.

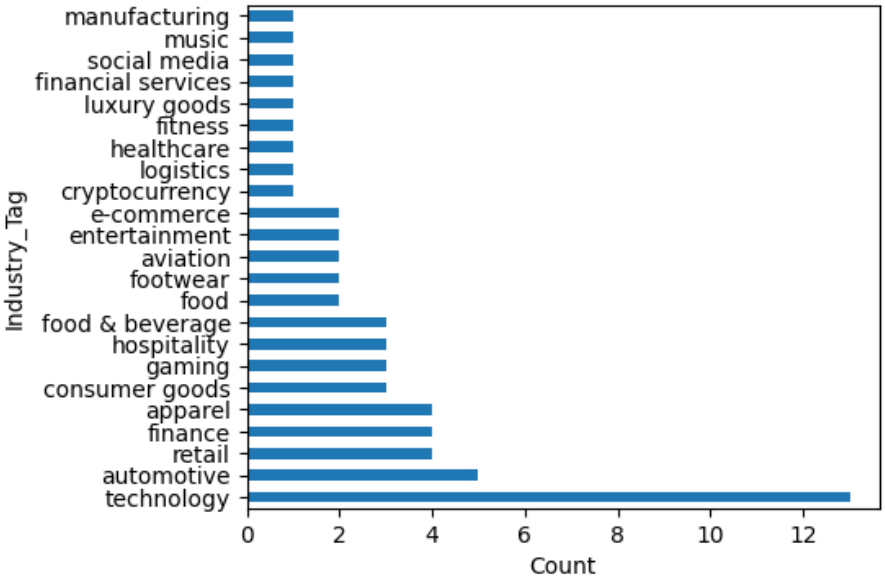

Figure 2. Bar Plot for Counts Based on Industry

Figure 2. Focusing on the industry categorization for a specific date (September 20, 2023), we observed a dominant presence of the technology sector in the number of public companies. This is evident in the bar plot, which clearly shows the tech industry outpacing other sectors in terms of public listings, indicative of the growing influence of technology in the global economy.

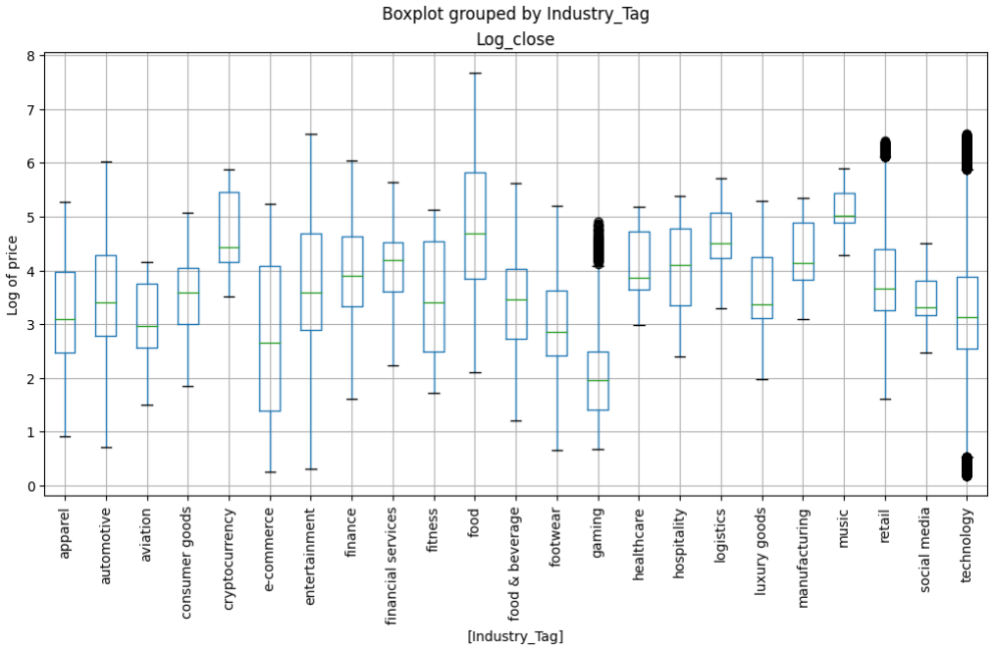

Figure 3. Boxplot for industry tag and log of price

Figure 3. To better comprehend the distribution and variations in stock prices, we applied a logarithmic transformation to the closing prices, creating the 'Log_close' variable. Box plot analyses across different industries post-transformation revealed that the log transformation effectively reduces the impact of outliers, resulting in a more concentrated distribution. This transformation is particularly relevant for financial investors, focusing more on the rate of return rather than absolute price changes.

3. Methods

3.1 Data Splitting

Firstly, we sort the financial time series data by date. This is important because in finance, the order of events over time matters significantly, and we need to maintain the chronological order of the data to reflect how the market evolves. We select the closing price of stocks as our target because it's a crucial indicator in financial analysis.

After that, we split the data into two parts: a training set and a test set. However, we ensure that the order of the data is not shuffled because in finance, the sequence of events matters a lot. Shuffling the data could introduce future information into the model, which would make the model's performance unrealistic.

3.2 Grouping and Sampling data

To facilitate the analysis of stock data using autoregression, a critical preprocessing step involved the grouping of the dataset by company. This step is of paramount importance as it ensures that the historical data (lagged values) are correctly aligned with the respective company's data. By aggregating data based on the 'Brand_Name' or company attribute, we maintain the temporal structure of the dataset while preserving the individual characteristics of each company's stock data.

With nearly three million data points in database, direct random sampling would risk disrupting the temporal integrity of the time-series data. Instead, a stratified sampling approach was adopted, wherein subsets of data were selected from each company group, preserving the representation of each company in the reduced dataset. This method successfully mitigates the risk of data distortion while providing a manageable dataset that reflects the diverse behaviors exhibited by different companies in the stock market.

3.3 Lagged Feature Engineering for Autoregression

We employ an autoregressive model to analyze stock market data, focusing specifically on stock closing prices. The model incorporates three key lagged variables: the closing prices of one day prior (lag1), two days prior (lag2), and three days prior (lag3). These lagged values are integrated into the features of both the training and testing sets to predict future trends in stock prices.

3.4 Preprocessing and Feature Transformation

To effectively process the stock market data for input into the autoregressive model, a comprehensive preprocessing routine was implemented. This preprocessing step involves appropriate transformations of different feature types to ensure consistency in data format and range, thereby enhancing the model's performance and reliability.

Firstly, for categorical features such as 'Brand_Name', 'Ticker', 'Industry_Tag', and 'Country', we employed One-Hot Encoding. This transforms these non-numerical features into a format

interpretable by machine learning models, allowing the model to differentiate between various categories.

Next, features like 'Volume', 'Dividends', and 'Stock Splits' were subjected to Min-Max Scaling, ensuring that these features fall within a range of 0 to 1. This helps maintain consistency in the scale of different features while preventing any single feature from dominating during the model training process.

Lastly, for continuous numerical features, including 'Open', 'High', 'Low', and the lagged features 'lag 3 days', 'lag 2 days', and 'lag 1 day', Standard Scaling was applied. This method transforms the data into a distribution with a mean of 0 and a standard deviation of 1, aiding the model in better understanding and comparing values of varying magnitudes.

3.5 Model Selection

In our study, we initially divided the data using train_test_split, followed by employing TimeSeriesSplit with 5 splits (n_splits = 5) for validation. This cross-validation technique preserves the temporal sequence of the data, thereby preventing the misinterpretation often caused by traditional random split methods in time-series analysis. Simultaneously, we optimized the hyperparameters of four different machine learning models using the GridSearchCV function, ensuring the identification of the most effective parameter combinations for each model.

Moving to the model evaluation phase, we adopted the Expand Window Method for Prediction. This approach is particularly apt for time-series models where generating multiple datasets with random splits is not feasible. By sequentially expanding the training dataset and retraining the model at each step, we could predict the next time points more accurately. This methodology enabled us to calculate multiple Mean Squared Errors (mean_mse and STD_mse) for a robust assessment of each model's performance. The parameters tuned and the respective values tested for each model are detailed in Table.

MODELS	PARAMETERS
Ridge (L2)	**Alpha**: 0.1, 2, 4
Random Forestry	**n_estimators:** 50, 100, 200 **max_depth:** 3, 5, 10
KNN	**n_neighbors:** 1, 3, 5, 7, 9 **weights:** distance
SVR	**C:** 0.1, 1 **gamma:** 0.001, 0.01

Table 1. Tuned parameters for each model.

4. Result

We use test MSE score and win rate to evaluate the performances of all values, the mean scores and standard deviations are represented in the following table.

Model	Test MSE	Win Rate	Mean MSE	STD of MSE
Ridge(L2)	1.70	0.79	1.70	2.98
Random Forestry	128.15	0.74	130.15	384.18
KNN	144.76	0.59	144.84	368.4
SVR	320.40	0.657	320.57	913.25
Baseline	20.518			

Table 2. Evaluation scores for each model

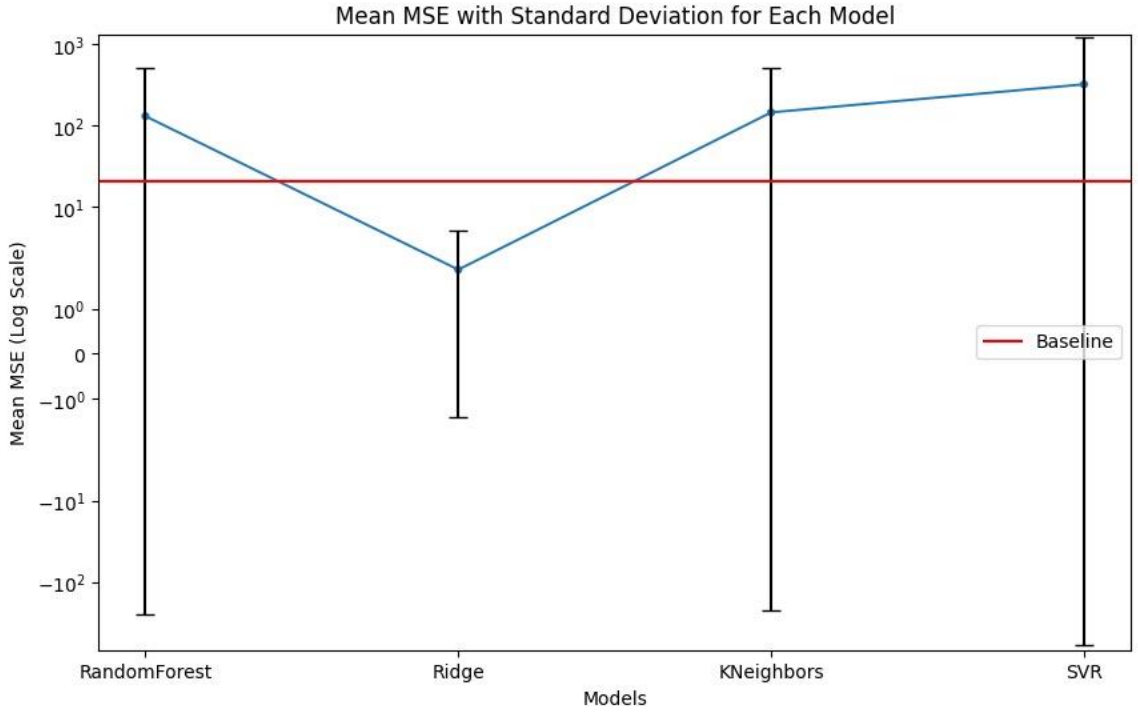

Figure 4. Mean MSE with STD for each model.

The evaluation involved the use of several key metrics, including Test Mean Squared Error (MSE), Win Rate (indicative of classification accuracy), and a comparison to a baseline MSE score. The Baseline, represented by a Test MSE of 20.518, serves as a reference point, reflecting the error one would incur by naively predicting the mean of the target variable for all data points. Our models, including Ridge(L2), Random Forest, KNN (K-Nearest Neighbors), and SVR (Support Vector Regression), were scrutinized in terms of their Test MSE. However, the results unveiled a notable trend among the models assessed. Except for the Ridge(L2) model, all other

models, including Random Forest, KNN, and SVR, exhibited Test MSE values that were considerably higher than the Baseline. This divergence from the Baseline underscores the challenges encountered in achieving improved predictive performance for the stock price regression task. Consequently, the Ridge(L2) model emerged as the most promising, demonstrating a lower Test MSE compared to the Baseline. While this may appear surprising, several factors may contribute to this performance. It's possible that the model possesses a unique ability to capture subtle patterns in financial time series data or that meticulous data preprocessing and regularization techniques have enhanced its effectiveness.

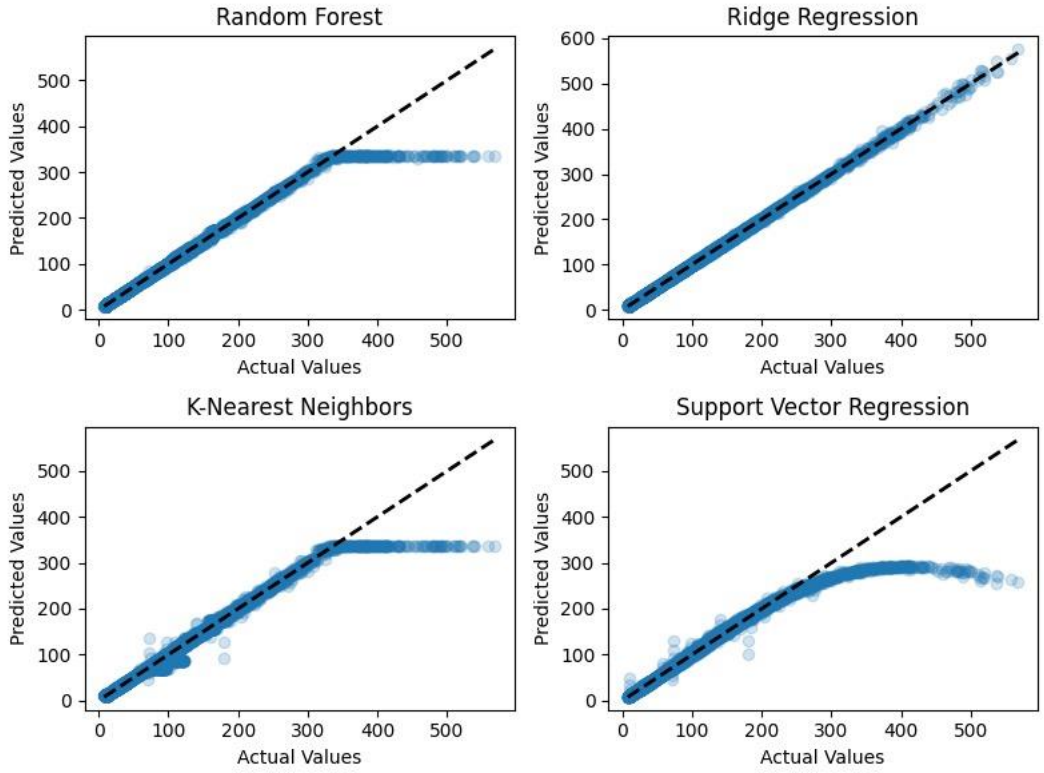

Figure5. Scatter plot for predicated values and actual values.

Examining the scatter plot, we observe that the Ridge Regression model consistently maintains a close alignment between predicted and actual values, signifying its robust performance. In contrast, Random Forest, KNN, and SVR models occasionally exhibit instances where actual values exceed predictions, particularly around 300.

To assess the global feature importance of the model, we employed three distinct methods: analysis of model coefficients, permutation importance, and SHAP values. Presented below are the outcomes of these comprehensive evaluations:

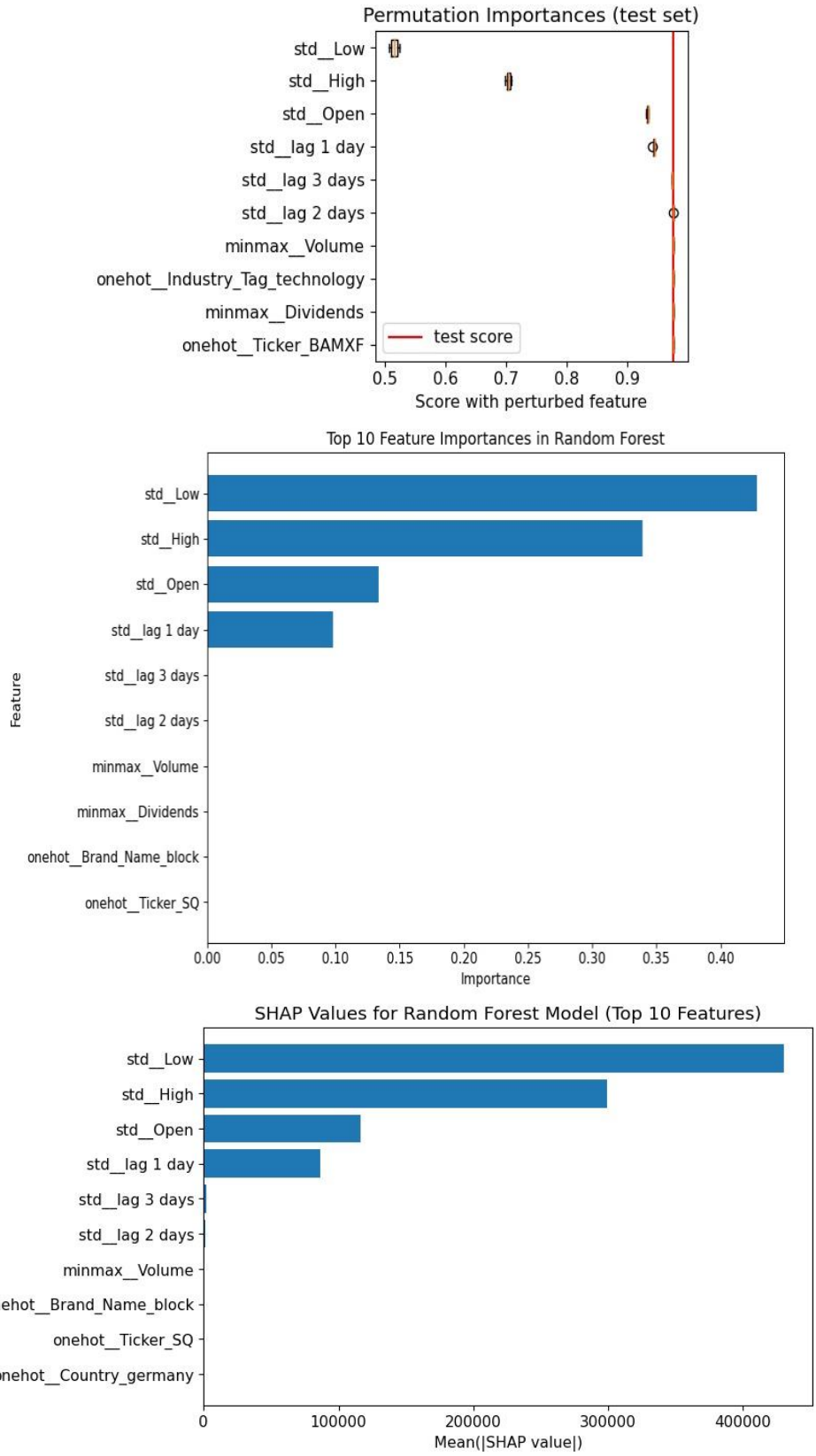

Figure 6. Global feature importance.

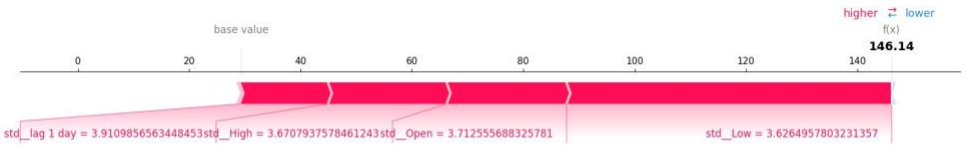

Figure 7. Local feature importance using SHAP.

The analysis reveals that for accurate stock price predictions, a focus on recent market data, especially 'lag 1 day', 'High', 'Open', and 'Low', is crucial. The SHAP analysis further emphasizes the local importance of these features. In summary, the most accurate predictions of future stock prices are likely when emphasizing recent market data, particularly the previous day's trends and price ranges. This approach underscores the critical impact of short-term market dynamics and investor sentiments on stock movement.

5. Outlook

Through this project, we compare several machine learning models to predict future global market stock prices. The best-performing model emerged as Ridge Regression with optimized hyperparameters. While this result may seem surprising, several factors could contribute to this performance. It's possible that Ridge Regression uniquely captures subtle patterns in financial time series data, or that our meticulous data preprocessing and regularization techniques have significantly enhanced its effectiveness.

However, to reinforce our confidence in these findings and ensure the robustness of our conclusions, it is prudent to explore alternative methods. We could experiment with more complex models, such as neural networks, known for their ability to capture intricate data patterns, or ensemble methods that combine predictions from multiple models to potentially improve accuracy. Additionally, incorporating external factors like economic indicators or market sentiment analysis might offer a more comprehensive understanding of the variables influencing stock prices.

Reference:

[1] Elgiriyewithana, N. (2023, September 21). *World stock prices (daily updating)*. Kaggle. https://www.kaggle.com/datasets/nelgiriyewithana/world-stock-prices-daily-updating

[2] Shen, J., & Shafiq, M. O. (2020, August 28). *Short-term stock market price trend prediction using a comprehensive deep learning system - journal of big data*. SpringerOpen. https://journalofbigdata.springeropen.com/articles/10.1186/s40537-020-00333-6