COLUMBIA UNIVERSITY

IN THE CITY OF NEW YORK

Cost: Machine Learning techniques to predict stock prices and regime changes in financial markets

Dominique Morris dm3532@columbia.edu

Jie Chen jc5890@columbia.edu

Saloni D Jaitly sdj2129@columbia.edu

Supervised by:

Prof. Dr. Jörg Osterrieder (ZHAW School of Engineering) and Prof. Ali Hirsa (Columbia University)

Acknowledgement

We would like to express our sincere gratitude and appreciation to the following individuals and institutions for their invaluable support and contribution to this project.

First and foremost, we extend our gratitude to COST (Cooperation in Science and Technology) and Columbia University for their generous support, which made this research possible.

We would like to extend special thanks to Professor Hirsa, Professor Osterrieder, Professor Hadzi Misheva, and teaching assistant Miao Wang for their guidance, expertise, and continuous support throughout the duration of this project. Their valuable insights and advice were instrumental in shaping our research and enhancing its quality. We would also like to acknowledge the significant contributions of Wassim Boutabratine, Ruben Illouz, and Marc Al Haj. Their prior work in this research project laid a strong foundation and provided valuable insights that paved the way for our current study.

Abstract 4
Introduction 4
1.1 Background 4
Importance of stock price prediction 4
Previous Work 5
Purpose of project 5
1.2 Assumptions 5
Previous Work 6
2.1 Overview of Prior Projects 6
2.2 Contributions and Scope of Our Work 7
Literature Review 8
3.1 Definitions and Buzzwords 8
3.2 Review of Relevant Research and Studies 9
Methodology 10
4.1 Approach and Methodological Framework 10
4.2 Data Collection and Analysis Techniques 10
Data Description 11
5.1 Data Sources and Acquisition 11
5.2 Data Preprocessing and Cleaning Procedures 11
Performance Metrics 12
6.1 Selection and Justification of Metrics 12
6.2 Evaluation Criteria 13
Findings and Observations 15
7.1 Key Findings and Results 15
7.2 Failed Attempts and Rerouting: 15
Conclusion 16
8.1 Summary of Findings and Observations 16
8.2 Implications and Insights 17
Future Steps 19
9.1 Recommendations for Further Research or Improvements 19
9.2 Next Steps for the Project 19
Appendix A: Detailed Methodology 20
10.1 Additional Technical Information or Algorithms 20
Reflections on the Project 20
12.1 Lessons Learned and Areas for Improvement 20
12.2 Challenging Parts of the Project 21
References 21
11.1 List of Citations and Sources 21

Abstract

Machine learning techniques have gained significant attention in finance, offering a range of methods such as linear regression, decision trees, support vector machines, and artificial neural networks. These models aim to uncover patterns in historical data to predict future outcomes. In the context of stock price prediction, machine learning algorithms can be trained on historical stock prices, financial news, and other relevant data to identify patterns for making future stock price predictions. This study focuses on evaluating the effectiveness of various machine learning models for forecasting stock prices and detecting regime changes in the S&P 500 E-mini future trades. The objective is to generate profitable trades by incorporating regime classification and optimizing the models using deep learning trading techniques. Building upon previous research, a well-structured and simplified function-based code was developed, with additional emphasis on leveraging the capabilities of ChatGPT for preparation and analysis. The study aims to explore the feasibility of using machine learning models in real-time trading and investment systems, with the ultimate goal of achieving more accurate and profitable trades.

Introduction

1.1 Background

Importance of stock price prediction

In financial markets, stock price prediction plays a crucial role due to its numerous benefits and implications. By accurately forecasting stock prices, investors can make informed decisions regarding buying and selling stocks, allowing them to optimize their investment strategies and allocate their resources effectively. Additionally, stock price prediction aids in managing risk by enabling proactive measures to mitigate potential losses. It contributes to market analysis by providing valuable insights into trends, patterns, and correlations that influence price movements. The ability to make accurate predictions also supports financial planning and wealth management, as it assists in estimating investment returns and informing long-term plans. Moreover, stock price prediction serves as an economic indicator, reflecting the overall economic conditions and assisting policymakers and economists in making informed decisions. Furthermore, it plays a vital role in algorithmic trading, facilitating the development of trading strategies and enabling high-frequency trades. However, it is important to acknowledge the inherent challenges in stock price prediction, given the complexities and volatility of the market. Therefore, stock price prediction should be considered alongside other factors, and the principles of risk management and diversification remain crucial in investment decision-making.

Previous Work

According to previous calculations, they achieved out-of-sample accuracies ranging from 0.50 to 0.52 and Sharpe ratios ranging from -0.004 to 0.005 for all of the machine learning models they tested, with and without regime classification.

Purpose of project

The purpose of this project is to leverage machine learning techniques to predict stock prices and identify regime changes in financial markets. These techniques are chosen for their ability to recognize patterns, capture non-linear relationships, and adapt to changing market conditions. By analyzing large amounts of historical data, machine learning models can uncover complex patterns and relationships, leading to more accurate stock price predictions. These models have the capacity to adapt to shifting market dynamics, select relevant features, and extract valuable information from raw data. Real-time analysis using machine learning aids in timely decision-making, particularly in high-frequency trading scenarios. Additionally, these models contribute to risk management by estimating the probability of extreme events and assisting in portfolio optimization. Overall, the project aims to harness the computational power and adaptability of machine learning techniques to enhance accuracy, capture non-linear dynamics, and support decision-making in the realm of financial markets.

1.2 Assumptions

- We assume that the data used for stock price prediction is stationary, meaning that the patterns and relationships observed in historical data will continue to hold in the future. However, it is important to note that financial markets are known to exhibit non-stationarity, where market conditions can change over time. Therefore, adapting models to handle non-stationarity is crucial for accurate predictions.
- We assume that the observations in the dataset are independent of each other. However, in financial markets, consecutive stock prices may exhibit dependencies, such as autocorrelation, where the current price is influenced by previous prices. To accurately capture the temporal nature of market data, it is necessary to account for these dependencies and incorporate them into the modeling process.
- Another assumption is that financial markets are efficient, meaning that stock prices reflect all available information and are not easily predictable. While the efficient market hypothesis has been widely debated, incorporating additional factors and information beyond historical prices can enhance the predictive capabilities of machine learning models.
- Finally, we assume that the selected features used in the model capture the essential information and relationships influencing stock prices. The identification and inclusion of appropriate features are critical to the effectiveness of the model in making accurate predictions.

Previous Work

2.1 Overview of Prior Projects

Our work builds upon the research conducted by Ziyun Lu, Senhao Wang, and Yi Zhang, as well as Wassim Boutabratine, Ruben Illouz, and Marc Al Haj. Lu, Wang, and Zhang published a paper titled "Predicting Directional Movements of E-mini S&P 500 Futures for Intraday Trading" in May 2022, which laid the foundation for this project. They focused on preparing the data for analysis and developing prediction models using logistic regression, decision trees, random forests, and regime classification techniques such as frequentist, k-means, and LSTM-predicted approaches.

The dataset used in their research was divided into 5-minute intervals, allowing them to predict the directional movements within this time frame. The training and testing data spanned from March 1, 2021, to March 25, 2022, and included futures data for crude oil and US 10-year T-bills from the same period. To evaluate the models' performance, they utilized metrics such as out-of-sample accuracy, Sharpe ratio, and maximum drawdown.

Their contributions to prior research were notable in several ways. Firstly, they created simulated data that replicated the features of real data, enabling them to explore the impact of changing market dynamics on the models. They employed Geometric Brownian Motion (GBM) to generate simulated paths with varying drift and increased noise parameters. By repeating the previous group's analysis of the simulated data and utilizing grid search, they identified the best-performing models based on out-of-sample accuracy.

Additionally, they focused on understanding the signal-to-noise ratio by training logistic regression models on the simulated data. They ran the regression multiple times for each pair of (noise, drift) values and computed the mean of the return and Sharpe ratios. This analysis provided insights into the relationship between noise, drift, and model performance. In the end, they generated a matrix of Sharpe ratios and returns that depended on the noise and drift parameters.

Overall, these prior projects laid the groundwork for our research by establishing data preparation techniques, developing prediction models using various machine learning algorithms, and exploring the impact of changing market dynamics on model performance through simulated data analysis

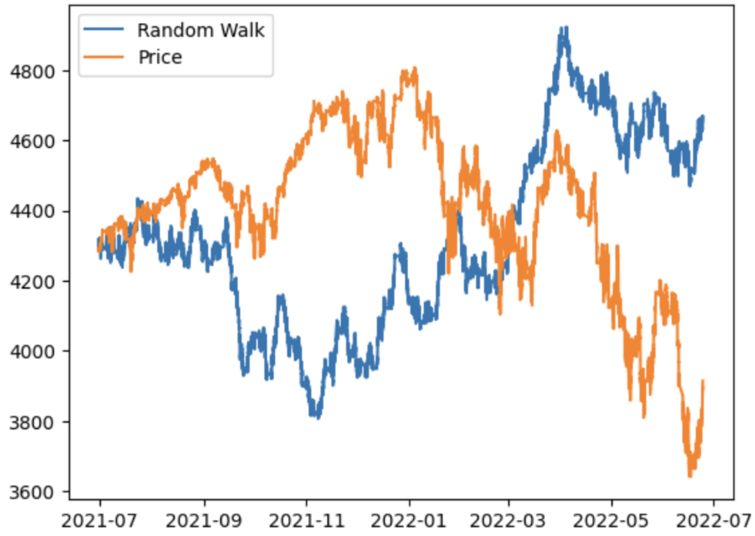


Fig A: Random Walk Time Series

2.2 Contributions and Scope of Our Work

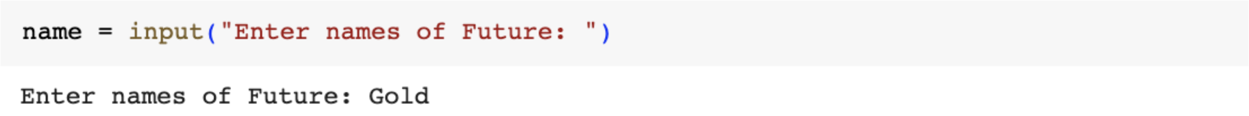
Our work makes several contributions to the existing research. Firstly, we have transformed the complex code into user-friendly utility functions, which are presented at the beginning of our project. These functions allow for easier implementation and direct usage in the construction of the run_classifier function. This function offers the flexibility to utilize different parameters, such as the decision tree classifier or random forest classifier.

Additionally, we have replicated the random walk time series simulation conducted by the previous group. By employing Geometric Brownian Motion (GBM) for stock prices, we have simulated data using all the available features. Furthermore, we have enhanced the system's versatility by incorporating a feature selection component. This enables users to analyze data with specific feature combinations based on their input.

In the realm of regime classification, we have developed a one-dimensional volatility regime classification approach. This methodology provides a framework for identifying different volatility regimes in the dataset.

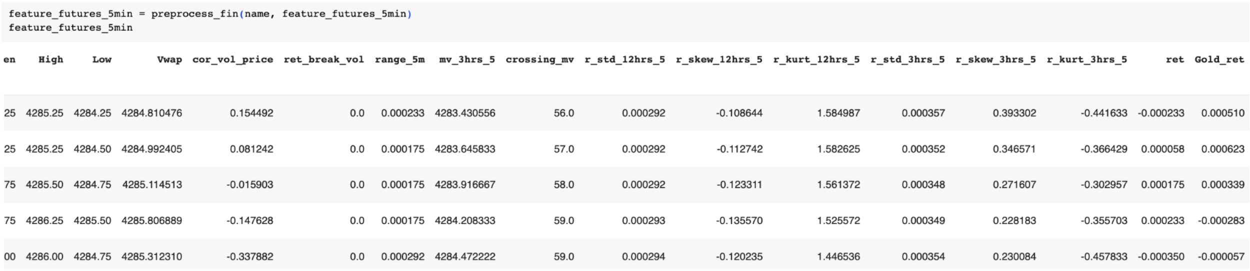
In terms of prediction, we have streamlined the code and focused on utilizing random forest classifiers. We have dedicated efforts to improving prediction accuracy by adjusting the classifier parameters and refining the dataset's training and testing split methodology.

Overall, our work enhances the usability and efficiency of the codebase, introduces new simulation capabilities with varied feature combinations, incorporates a volatility regime classification approach, and improves prediction accuracy through algorithmic adjustments.



```
name = input("Enter names of Future: ")
```

```
Enter names of Future: Gold
```



```
feature_futures_5min = preprocess_fin(name, feature_futures_5min)
feature_futures_5min
```

en	High	Low	Vwap	cor_vol_price	ret_break_vol	range_5m	mv_3hrs_5	crossing_mv	r_std_12hrs_5	r_skew_12hrs_5	r_kurt_12hrs_5	r_std_3hrs_5	r_skew_3hrs_5	r_kurt_3hrs_5	ret	Gold_ret
25	4285.25	4284.25	4284.810476	0.154492	0.0	0.000233	4283.430556	56.0	0.000292	-0.108644	1.584987	0.000357	0.393302	-0.441633	-0.000233	0.000510
25	4285.25	4284.50	4284.992405	0.081242	0.0	0.000175	4283.645833	57.0	0.000292	-0.112742	1.582625	0.000352	0.346571	-0.366429	0.000058	0.000623
75	4285.50	4284.75	4285.114513	-0.015903	0.0	0.000175	4283.916667	58.0	0.000292	-0.123311	1.561372	0.000348	0.271607	-0.302957	0.000175	0.000339
75	4286.25	4285.50	4285.806889	-0.147628	0.0	0.000175	4284.208333	59.0	0.000293	-0.135570	1.525572	0.000349	0.228183	-0.355703	0.000233	-0.000283
00	4286.00	4284.75	4285.312310	-0.337882	0.0	0.000292	4284.472222	59.0	0.000294	-0.120235	1.446536	0.000354	0.230084	-0.457833	-0.000350	-0.000057

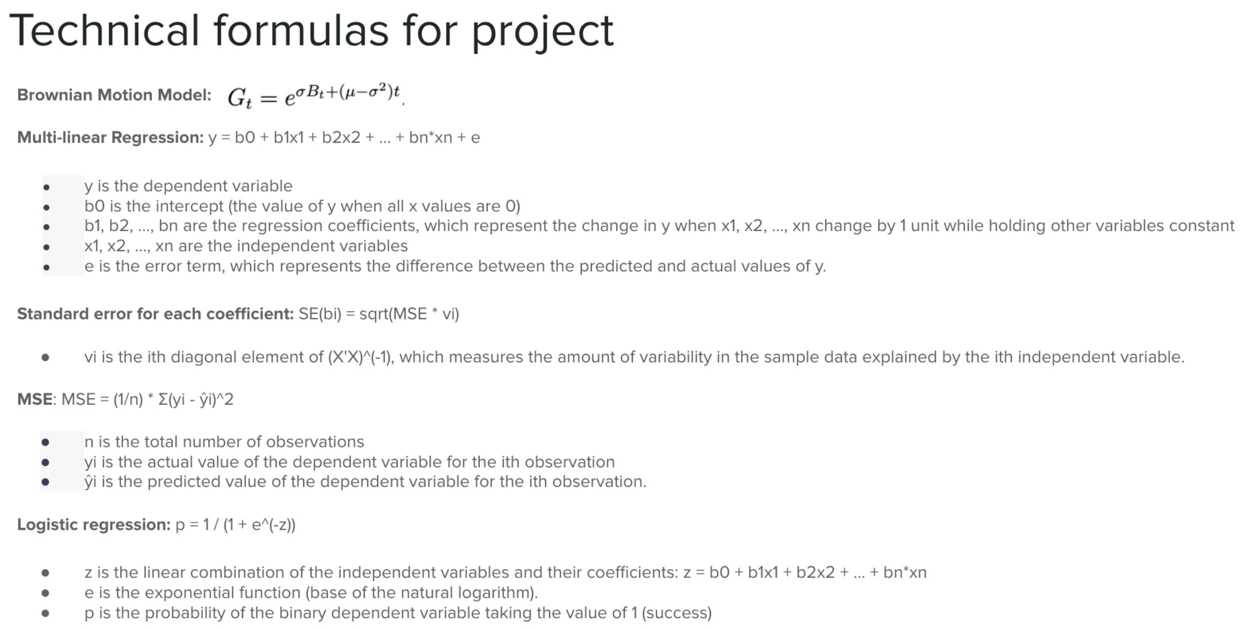
Fig B: feature_futures_5min data- Preprocesses financial data for futures trading by merging data frames, dropping invalid data, and downloading the prepared data for future use.

Literature Review

3.1 Definitions and Buzzwords

- Intraday trading: Intraday trading involves the buying and selling of financial instruments like stocks or futures contracts within a single trading day. It requires quick decision-making to capitalize on short-term price changes.
- Volatility: Volatility measures the extent of variation and unpredictability in the price or value of a financial instrument. Higher volatility indicates larger price fluctuations, while lower volatility suggests more stable prices.
- Trading signals: Trading signals are alerts or indicators generated by algorithms or technical analysis tools. These signals provide guidance on when to buy or sell a financial instrument based on identified patterns, trends, or other criteria derived from data analysis.
- Machine learning algorithms: Machine learning algorithms are computational methods that autonomously learn patterns and relationships within data, without explicit programming. These algorithms can process large amounts of data and generate predictions or insights based on the discovered patterns.
- Forecasting: Forecasting involves making predictions or estimates about future events or outcomes by analyzing historical data and patterns. In the context of this project, it specifically refers to predicting the future direction of S&P 500 E-mini futures.

Intraday trading is a complex and challenging task due to the volatility and intricacies of financial markets. However, machine learning algorithms have emerged as promising tools for generating intraday trading signals. These algorithms possess the capability to analyze vast amounts of data and uncover patterns that may not be easily recognizable to human traders. The objective of this project is to forecast the direction of S&P 500 E-mini futures. Although there is limited research specifically on this topic, the previous team has identified a collection of relevant studies that contribute to the understanding of this area.



Technical formulas for project

Brownian Motion Model: $G_t = e^{\sigma B_t + (\mu - \sigma^2)t}$.

Multi-linear Regression: y = b0 + b1x1 + b2x2 + ... + bn*xn + e

- y is the dependent variable
- b0 is the intercept (the value of y when all x values are 0)
- b1, b2, ..., bn are the regression coefficients, which represent the change in y when x1, x2, ..., xn change by 1 unit while holding other variables constant
- x1, x2, ..., xn are the independent variables
- e is the error term, which represents the difference between the predicted and actual values of y.

Standard error for each coefficient: SE(bi) = sqrt(MSE * vi)

- vi is the ith diagonal element of (X'X)^(-1), which measures the amount of variability in the sample data explained by the ith independent variable.

MSE: MSE = (1/n) * Σ(yi - ŷi)^2

- n is the total number of observations
- yi is the actual value of the dependent variable for the ith observation
- ŷi is the predicted value of the dependent variable for the ith observation.

Logistic regression: p = 1 / (1 + e^(-z))

- z is the linear combination of the independent variables and their coefficients: z = b0 + b1x1 + b2x2 + ... + bn*xn
- e is the exponential function (base of the natural logarithm).
- p is the probability of the binary dependent variable taking the value of 1 (success)

3.2 Review of Relevant Research and Studies

Over the past few years, there has been a growing interest in the application of machine learning techniques in the financial market. Through our research, we have identified ten academic references that highlight the potential of machine learning in various financial domains. These studies encompass areas such as stock price prediction, portfolio optimization, risk management, and algorithmic trading.

Comparing machine learning algorithms with traditional statistical models, these studies consistently demonstrate the superior performance of machine learning in terms of accuracy, stability, and risk-adjusted returns. The findings indicate that machine learning models have the capability to outperform traditional methods when applied to financial tasks.

Moreover, the review papers we examined offer a comprehensive overview of different machine-learning techniques and their specific applications within the finance field. They delve into the challenges and limitations associated with utilizing machine learning in finance, providing valuable insights for future researchers and practitioners.

In conclusion, the existing literature strongly suggests that machine learning can play a significant role in transforming the financial market. However, further research is necessary to fully grasp and harness the potential of machine learning in finance.

Methodology

4.1 Approach and Methodological Framework

To predict stock prices and detect regime changes in financial markets using machine learning techniques, our approach involves a structured process that encompasses data preprocessing, model selection, and evaluation.

The first step is data collection and preprocessing, where we gather relevant data and clean it to ensure accuracy and consistency. This stage may involve handling missing data, normalizing variables, and removing outliers.

Next, we focus on feature extraction, extracting meaningful and informative features from the preprocessed data. These features can include moving averages, volatility measures, and other indicators that capture important patterns and relationships within the data.

Once the features are extracted, the model selection phase begins. For stock price prediction, we consider various models such as linear regression, random forests, and Geometric Brownian Motion (GBM). These models are chosen based on their suitability for the specific data and features at hand. For regime detection, clustering algorithms and Gaussian Mixture Models (GMM) are commonly employed methods for analyzing market regimes.

Finally, the evaluation stage allows us to assess the performance and insights gained from the models. We examine the outputs of the models and evaluate their accuracy and predictive power. This evaluation helps us gain a deeper understanding of stock price movements and regime changes, providing valuable insights for decision-making and investment strategies

4.2 Data Collection and Analysis Techniques

Data collection and analysis play a crucial role in identifying regime changes and predicting stock prices in financial markets. In this study, various data collection techniques are employed, including gathering historical stock data such as daily and intraday price and volume data for the target stocks. Market indicators, such as futures data for specific resources, are also incorporated into the analysis.

To ensure data integrity and accuracy, data cleaning techniques are applied. This process involves strengthening the consistency of the data by addressing missing values, outliers, and other data quality issues. Additionally, time-series analysis is conducted to examine temporal patterns in the stock price data, enabling the identification of trends, patterns, and seasonality. Statistical analysis techniques are utilized to identify correlations between variables, providing insights into their relationships.

Machine learning models are utilized to predict stock prices and detect regime changes. Unsupervised learning techniques, including Gaussian Mixture Models and clustering algorithms, are particularly useful for identifying changes in volatility, trends, or market states, aiding in the evaluation of market conditions.

The function 'volatility_regime_analysis()' is implemented to perform regime analysis specifically for the futures dataset. It calculates the standard deviation of 5-minute returns of SP500 futures within a 3-hour period, serving as a proxy for volatility. The volatility regimes are then divided based on the 33% and 67% quantiles of the standard deviation. The function generates plots displaying the returns and standard deviations of SP500 futures in different volatility regimes. It also computes the number of ups and downs in returns for each regime, presenting the count in a bar chart. The function provides relevant information such as volatility regime threshold values and the count of ups and downs.

The provided code includes a custom implementation of the train-test split function. This function creates a binary target variable and performs the train-test split using the 'train_test_split' function from scikit-learn. The resulting splits are returned as separate data structures, including the feature matrix and the target variable for the training and testing data.

The 'run_classifier' function trains a classifier on the provided training data and uses it to make predictions on both the training and testing data. Accuracy scores are calculated by comparing the predicted labels with the actual labels. DataFrames are created to store the predicted signals and accuracy scores. The function also performs a portfolio backtest, which evaluates the performance of the predicted signals. The resulting accuracy scores and signal DataFrames are returned for further analysis.

The 'plot_feature_importance' function visualizes the feature importance of the classifier, aiding in the identification of influential features and enhancing the interpretation of the classifier's behavior.

Data Description

5.1 Data Sources and Acquisition

Prior work has been conducted on data preprocessing, feature extraction, and data cleaning. This work includes collecting historical financial data as well as cleaning the data by handling missing values and outliers. Potential data sources and acquisition include historical stock data from financial data providers, stock exchanges, or data vendors and APIs.

5.2 Data Preprocessing and Cleaning Procedures

The data preprocessing and cleaning procedures prepare the data for further analysis or modeling by formatting columns, handling missing or invalid values, and extracting features for futures trading at a 5-minute interval. The procedures format date and time columns according to Python’s datetime, drops unnecessary columns, splits the data based on symbols, filters out invalid observations, resamples the data to 5-minute intervals, and extracts relevant features. Future data to be used is taken as input from user.

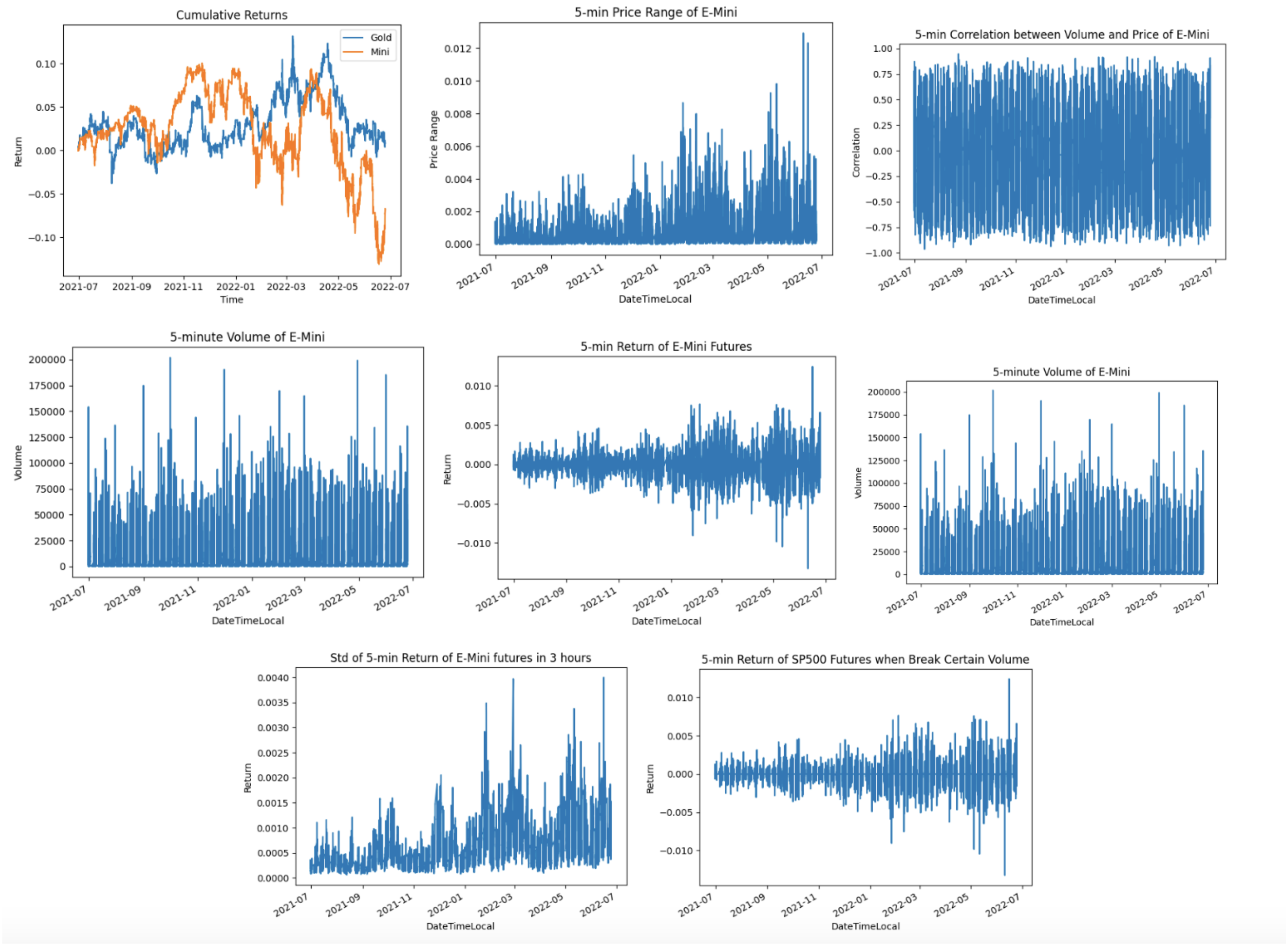


Fig C: Time Series plots for feature-futures-data-5min

Performance Metrics

6.1 Selection and Justification of Metrics

To evaluate the effectiveness of different trading strategies, we consider a range of performance metrics, including accuracy, Sharpe ratio, maximum drawdowns, and cumulative wealth, as previously used by precedent groups.

For volatility regime analysis the following are justifications for the selection of metrics:

- The standard deviation (SD) of 5-minute returns is a commonly used metric to measure volatility in financial markets. It quantifies the variability of returns. By calculating the standard deviation of 5-minute returns of SP500 futures, the code aims to capture the level of volatility in the market.

- The 33% and 67% quantiles of SD of 5-minute returns is used to divide the data into three volatility regimes: low, medium and high volatility. This division allows for categorization of volatility levels based on the observed distribution of standard deviations.
- Counting the number of ups and downs within each volatility regime provides a quantitative measure of the frequency of positive and negative returns. This metric allows for a comparison of the relative occurrences of ups and downs in different volatility regimes, providing insights into the behavior of prices under different volatility conditions.

6.2 Evaluation Criteria

One evaluation criterion is the visualization of returns and standard deviations in different volatility regimes. The code generates plots to show the patterns in returns and volatility across the identified regimes. The code also calculates threshold values for dividing the data into volatility regimes. These threshold values are printed to provide insight into the separation of regimes. The counts of ups and downs are plotted on a bar chart for the different regimes.

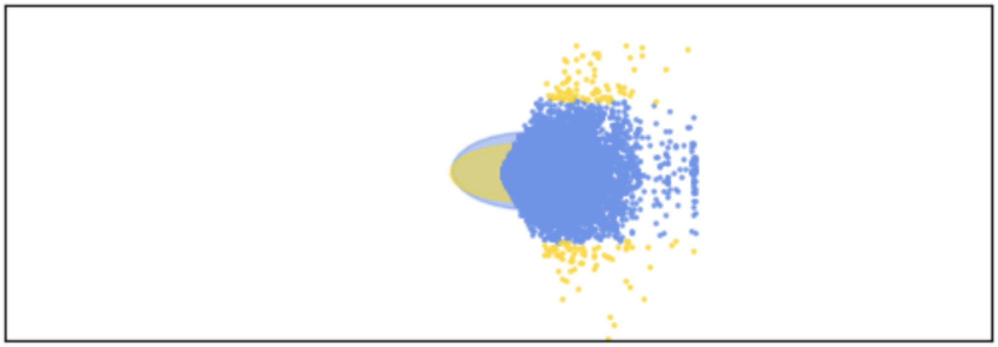
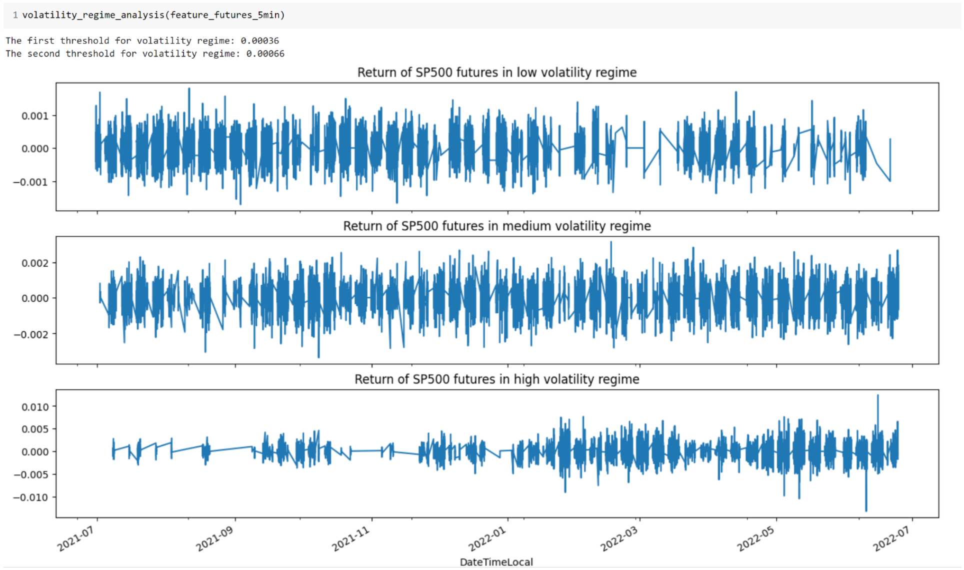


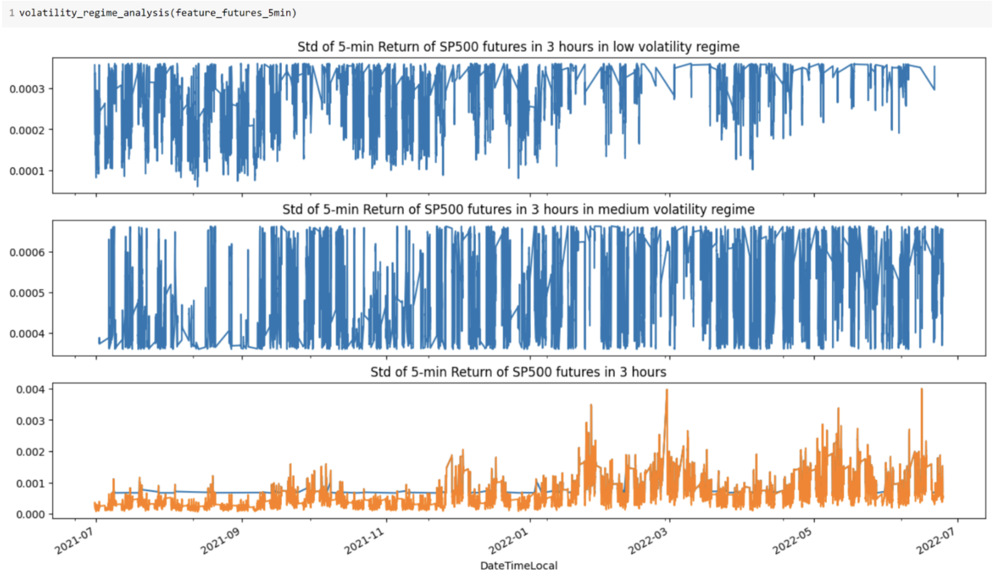
Fig D - Gaussian Mixture Model for Regime Classification



```
volatility_regime_analysis(feature_futures_5min)
```

```
The first threshold for volatility regime: 0.00036
The second threshold for volatility regime: 0.00066
```

Return of SP500 futures in low volatility regime
0.001
0.000
−0.001
Return of SP500 futures in medium volatility regime
0.002
0.000
−0.002
Return of SP500 futures in high volatility regime
0.010
0.005
0.000
−0.005
−0.010
2021-07
2021-09
2021-11
2022-01
2022-03
2022-05
2022-07
DateTimeLocal



```
volatility_regime_analysis(feature_futures_5min)
```

Std of 5-min Return of SP500 futures in 3 hours in low volatility regime
0.0003
0.0002
0.0001
Std of 5-min Return of SP500 futures in 3 hours in medium volatility regime
0.0006
0.0005
0.0004
Std of 5-min Return of SP500 futures in 3 hours
0.004
0.003
0.002
0.001
0.000
2021-07
2021-09
2021-11
2022-01
2022-03
2022-05
2022-07
DateTimeLocal

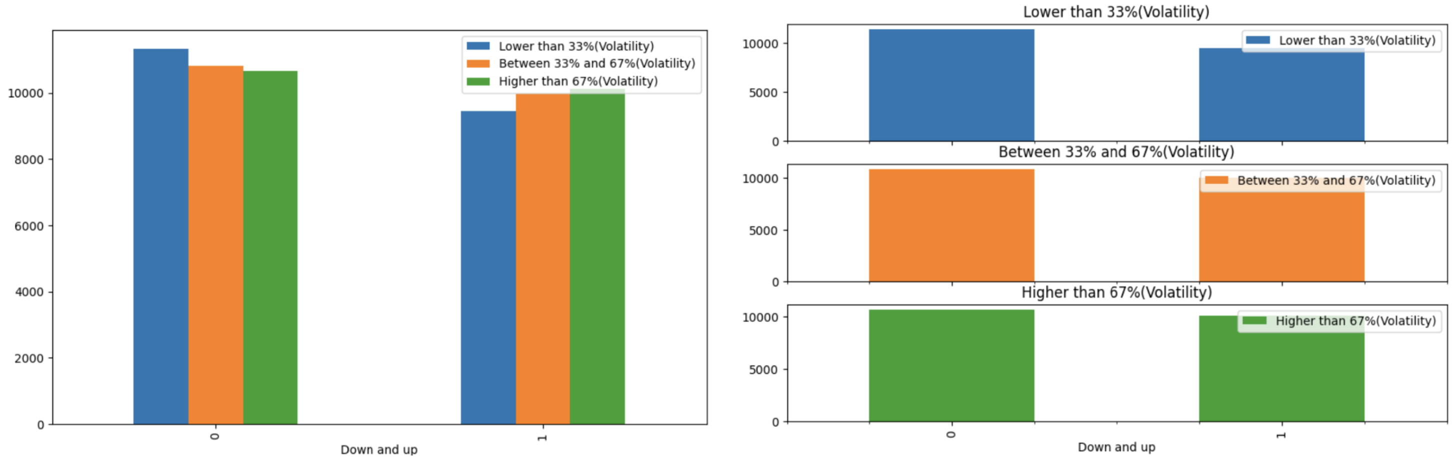


Fig E: Regime Classification - plots the returns and standard deviations of SP500 futures

Findings and Observations

7.1 Key Findings and Results

- Accuracy: The previous groups used decision trees and random forest models to achieve an in-sample accuracy of 0.7. To improve the accuracy, the leaf nodes and max depth were increased, resulting in an increased accuracy of 0.97. Despite the improved in-sample accuracy, the out-of-sample accuracy remains in the range of 0.5, indicating that the model does not generalize well to unseen data.
- Simplified classification models: The simplified classification models provide a detailed output, allowing for a better understanding of the model's predictions. Additionally, the feature importance can be plotted, providing insights into the significance of different features in the model.
- Volatility regime analysis: The function "volatility_regime_analysis()" performs regime analysis on the futures dataset, specifically focusing on the standard deviation of 5-minute returns for S&P500 futures over a 3-hour period. Regime classification using quantiles: By applying regime analysis, different volatility regimes or market conditions can be identified. The analysis utilizes quantiles, such as the 33rd and 67th percentiles, to define thresholds for distinguishing between low, moderate, and high volatility regimes. Categorizing and analyzing volatility regimes provides insights into market dynamics. This information can be used to adjust trading or investment strategies accordingly. During high-volatility regimes, more cautious approaches and risk management measures may be employed, while low-volatility regimes may present opportunities for more aggressive strategies.

7.2 Failed Attempts and Rerouting:

- Evolving goal: Initially, our goal for this project was to structure the elaborate code from previous codes by understanding its nuances and depths. However, as we progressed, we realized the need for a simplified version that would be more suitable for our purposes.
- Quantile-based regimes: We initially attempted to classify regimes based on volatility or volume using 1-dimensional quantile analysis. However, we found that the patterns of futures returns across different regimes were not helpful in predicting directional movements. The distributions of ups and downs were similar across regimes, indicating a lack of distinct predictive patterns. Advanced classification methods: Due to the limitations of 1-dimensional quantile analysis, the earlier team explored using volatility, volume, and trend to classify regimes using an approach called "M N K." However, as the overall complexity increased, we decided to reroute and explore alternative methods. Gaussian mixture model: After considering different options, we decided to explore the use of a Gaussian mixture model. This approach seemed more promising for our classification needs, providing a better balance between accuracy and complexity.
- Overfitting: As we applied classification models to the training data, we encountered issues of overfitting. This occurs when the model performs well on the training data but

fails to generalize to unseen data. We had to address this issue by carefully adjusting the model parameters and regularization techniques to prevent overfitting.

- Manual input of futures data: Initially, the code had preprocessed futures data, but we attempted to modify it to accept manual input from the user. This change allowed for more flexibility in testing different datasets and scenarios.
- Improving output processing: While structuring the code and putting it into functions, we encountered challenges in processing the outputs effectively. We had to refine the output processing methods to ensure meaningful and useful results.
- Simplification of decision tree and random forest models: To align the decision tree and random forest models with the purpose of our project, we had to simplify them. This involved adjusting parameters such as leaf nodes and max depth to improve the models' performance for our specific objectives.
- Change in dataset splitting method: We also decided to change the method of splitting the dataset into training and testing sets. This change was made to ensure a more reliable evaluation of model performance and to avoid any data leakage issues.

Conclusion

8.1 Summary of Findings and Observations

The decision tree and random forest models achieved an in-sample accuracy of 0.7, which was improved to 0.97 by increasing leaf nodes and max depth. However, the out-of-sample accuracy remained around 0.5, indicating poor generalization to unseen data. Further improvements are needed to enhance the models' ability to make accurate predictions beyond the training dataset.

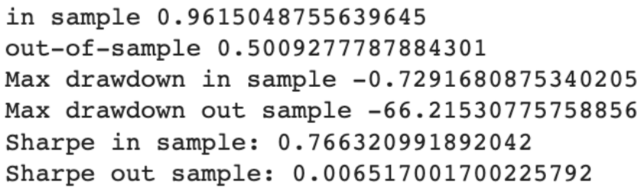
The simplified classification models provided detailed outputs, allowing for a better understanding of the model's predictions. Additionally, feature importances could be plotted, providing insights into the significance of different features in the model. This simplification process facilitated the interpretation and analysis of the models' results.

By applying regime analysis and utilizing quantiles such as the 33rd and 67th percentiles, different volatility regimes or market conditions were identified. Categorizing and analyzing volatility regimes provided valuable insights into market dynamics, enabling the adjustment of trading or investment strategies.

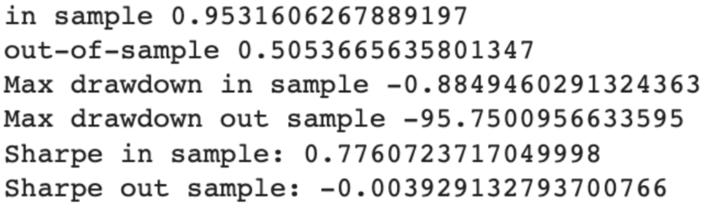
Although the project encountered failed attempts and required rerouting, valuable lessons were learned. Initially, the goal was to structure existing code, but it was realized that a simplified version was necessary. Quantile-based regimes and more advanced classification methods were explored but a Gaussian mixture model was adopted as a more promising approach in predicting directional movements. Challenges were faced in addressing overfitting, implementing manual input of data, improving output processing, simplifying decision tree and random forest models. These challenges prompted adjustments and refinements to overcome limitations and enhance the project's outcomes.

Decision Tree(depth=30)

Random Forest(node=2500)



```
in sample 0.9615048755639645
out-of-sample 0.5009277787884301
Max drawdown in sample -0.7291680875340205
Max drawdown out sample -66.21530775758856
Sharpe in sample: 0.766320991892042
Sharpe out sample: 0.006517001700225792
```



```
in sample 0.9531606267889197
out-of-sample 0.5053665635801347
Max drawdown in sample -0.8849460291324363
Max drawdown out sample -95.7500956633595
Sharpe in sample: 0.7760723717049998
Sharpe out sample: -0.003929132793700766
```

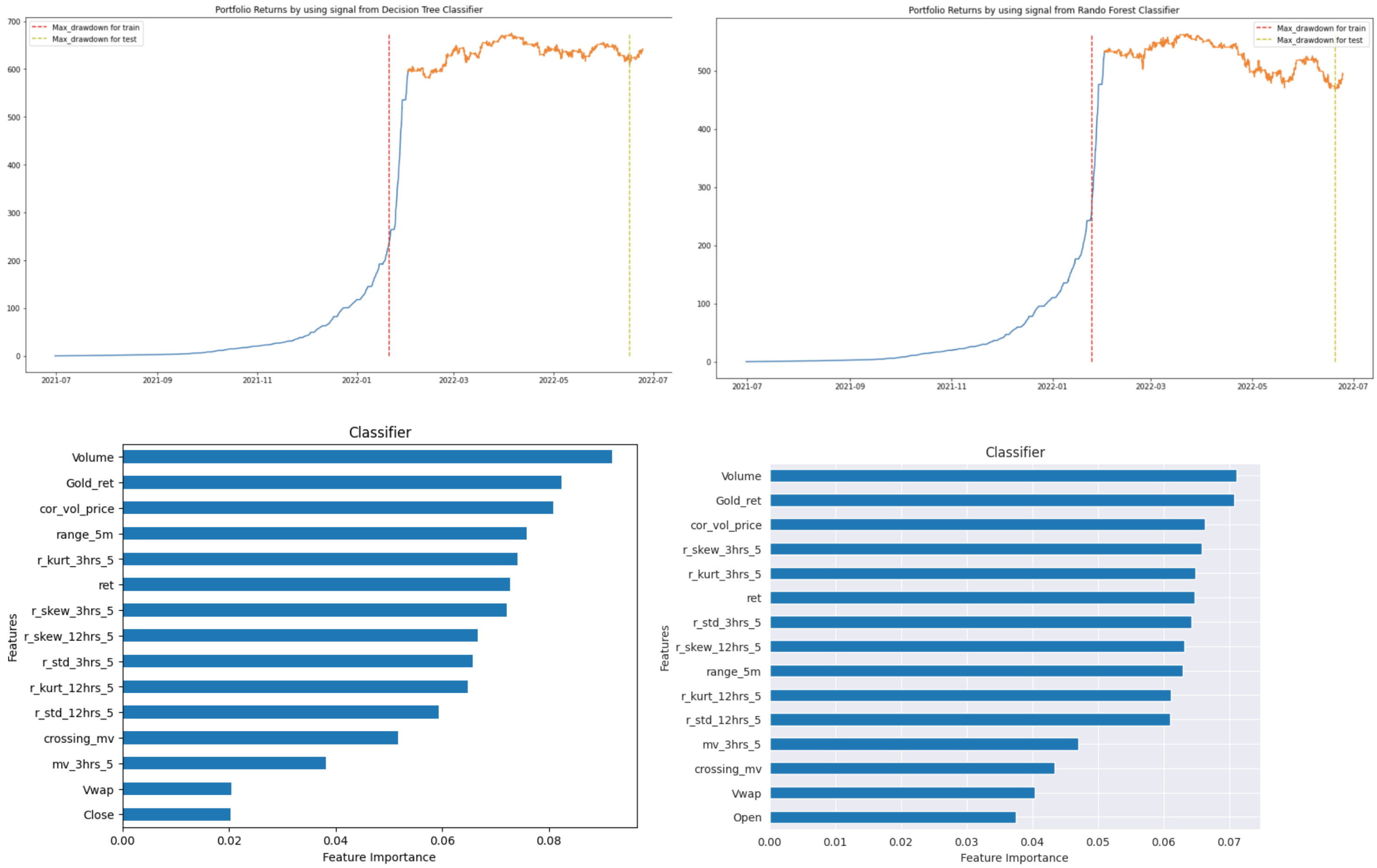


Fig F: Portfolio returns and Feature importance using Decision Tree and Random Forest Classifier

8.2 Implications and Insights

- Predictive Power: The use of machine learning techniques in predicting stock prices and identifying regime changes in financial markets offers significant potential for improving decision-making and enhancing investment strategies. These techniques can capture complex patterns and relationships in the data that may not be easily identifiable through traditional methods. The ability to accurately predict stock prices and detect regime changes can provide investors and traders with a competitive edge in understanding market dynamics and making informed decisions.

- Data Quality and Preprocessing: The quality of the data used for training machine learning models is crucial for obtaining reliable predictions. Rigorous data preprocessing techniques, such as cleaning, normalization, and handling missing values, are essential to ensure accurate and consistent results. Additionally, careful feature selection and engineering can improve the model's ability to capture relevant information and enhance its predictive power.
- Model Selection and Evaluation: Choosing the appropriate machine learning models for stock price prediction and regime analysis is critical. Different models have varying strengths and weaknesses, and selecting the most suitable one depends on the specific characteristics of the dataset and research objectives. A thorough evaluation of the models is essential to assess their performance, including measures such as accuracy, precision, recall, and F1 score. This evaluation provides insights into the model's strengths and limitations, enabling researchers and practitioners to make informed choices.
- Interpretability and Explainability: While machine learning models can offer accurate predictions, the lack of interpretability and explainability may limit their practical applications. Understanding the reasons behind the predictions and being able to interpret the model's behavior is crucial, especially in financial markets where decision-makers need to justify their actions. Developing models with interpretable features and using techniques such as feature importance analysis and visualization can help gain insights and build trust in the predictions.
- Dynamic Nature of Financial Markets: Financial markets are highly dynamic and subject to various external factors and events that can significantly impact stock prices and market regimes. Machine learning models need to be adaptive and capable of capturing changing patterns and behaviors. Continuous monitoring and updating of models are necessary to ensure their effectiveness and relevance in real-time market conditions.
- Risk Management: Accurate predictions of stock prices and regime changes can aid in developing effective risk management strategies. Identifying periods of high volatility or significant regime shifts allows investors to adjust their portfolio allocations, hedge positions, or implement other risk mitigation measures. Machine learning techniques can provide valuable insights into market dynamics and support more informed risk management decisions.
- Ethical Considerations: Applying machine learning in financial markets raises important ethical considerations, such as bias in data, transparency, and fairness. Careful attention should be given to ensuring data integrity, addressing potential biases, and promoting transparency and fairness in model development and deployment. Ethical guidelines and regulations should be followed to prevent unintended consequences and promote responsible use of machine learning in financial decision-making.

Future Steps

9.1 Recommendations for Further Research or Improvements

- Splitting the Dataset: One recommendation is to consider splitting the dataset into training and testing sets, where the testing set consists of the first parts of the data and the training set comprises the remaining larger portion. This approach is known as a "rolling window" or "expanding window" technique. By using the earlier data for testing and the more recent data for training, you can evaluate the model's performance on unseen data that closely resembles real-time scenarios. This approach helps assess the model's ability to generalize and adapt to changing market conditions.
- Feature Engineering and Selection: Another recommendation is to explore additional feature engineering techniques and perform thorough feature selection. Feature engineering involves creating new features or transforming existing ones to capture relevant information and improve the model's predictive power. Feature selection helps identify the most informative features that contribute significantly to the model's performance, reducing noise and enhancing interpretability. Consider incorporating domain knowledge and exploring advanced techniques such as recursive feature elimination or L1 regularization to select the most relevant features for stock price prediction and regime analysis.
- Ensemble Methods and Model Stacking: Ensemble methods can be explored as a recommendation for future steps. Ensemble methods combine multiple models to make predictions, leveraging the strengths of individual models and improving overall performance. Techniques like Random Forest, Gradient Boosting, or stacking models can be considered. Stacking involves training multiple models and combining their predictions using another model as a meta-learner. By combining diverse models, you can potentially achieve higher prediction accuracy and better capture complex patterns in stock prices and regime changes.
- Incorporating External Data and News Sentiment: Consider integrating external data sources and news sentiment analysis into the modeling process. External data, such as macroeconomic indicators, financial news sentiment, or social media sentiment, can provide additional insights and help capture market trends and investor sentiment. By incorporating these data sources, you can potentially enhance the model's ability to capture dynamic market conditions and improve its predictive accuracy.

9.2 Next Steps for the Project

- Continuously refine and optimize the machine learning models used for stock price prediction and regime detection. Experiment with different algorithms, hyperparameter tuning methods, and feature engineering strategies to improve model performance.
- Consider incorporating cutting-edge techniques such as deep learning models or advanced time series forecasting algorithms to uncover more complex patterns in the data. These approaches may offer enhanced capabilities in capturing intricate relationships and improving prediction accuracy.

- Assess the models' performance on both in-sample and out-of-sample data to ensure their ability to generalize well to new and unseen data. This step is crucial for validating the models' robustness and reliability.
- Conduct sensitivity analysis and robustness tests to evaluate the stability and dependability of the models. Assess how the models perform under different scenarios, varying data inputs, and potential changes in market conditions. This analysis will help identify the strengths and limitations of the models.
- Explore the practicality of integrating the developed models into real-time trading or investment systems. Consider the feasibility of implementing the models in live trading environments and assess their potential impact on decision-making processes.

Appendix A: Detailed Methodology

10.1 Additional Technical Information or Algorithms

In the prediction part of our project, we employ established machine-learning techniques and algorithms. The train_test_split function from scikit-learn plays a crucial role by shuffling and splitting the dataset into training and testing sets based on the desired test size. This function also handles the creation of binary target variables, which are essential for our classification task.
To train a classifier and assess its performance, we utilize a classifier object (clf) from scikit-learn. By fitting the model to the training data (X_train, y_train), the classifier learns patterns and relationships in the data. It then applies this knowledge to make predictions on both the training and testing data (X_test). To evaluate the accuracy of our predictions, we calculate accuracy scores using the accuracy_score function from scikit-learn. This metric provides a measure of how well our classifier performs on the given data. In addition to accuracy scores, we generate signals based on the predicted labels. These signals correspond to the indices of the training and testing data, indicating the predicted outcomes for each instance. This information can be valuable for further analysis and decision-making processes.

Throughout these procedures, we leverage well-established libraries such as sci-kit-learn and pandas. These libraries offer efficient functionalities for tasks like data splitting, classifier training, accuracy evaluation, signal generation, and visualization of feature importance.
By employing these techniques and utilizing reliable libraries, we ensure a robust and systematic approach to our prediction process, enabling us to make informed decisions based on the outcomes of our machine-learning models.

Reflections on the Project

12.1 Lessons Learned and Areas for Improvement

Throughout this project, we gained valuable lessons and identified areas for improvement. Initially, we learned the entire prediction process by studying previous work and leveraging the guidance provided by esteemed professors. This allowed us to thoroughly grasp the coding aspect of the project. Applying the knowledge acquired from class content in practical applications proved to be essential.

Moving forward, we recognize the need to focus on reinforcement and seek further improvement. We dedicated efforts to enhance our machine learning models, aiming to increase their accuracy and achieve better results. Additionally, we emphasized streamlining the code execution process for a seamless experience.

12.2 Challenging Parts of the Project

In the initial phase of this project, We faced the challenge of understanding a large amount of code developed by the previous group. Working with such a complex codebase was overwhelming, as it required me to grasp the logic, structure, and overall functionality of the code. It was crucial for us to comprehend it thoroughly in order to make necessary modifications, improvements, and effectively continue the project.

Additionally, our level of expertise and familiarity with the regime classification used in the project posed knowledge gaps and to overcome these gaps, we realized the importance of acquiring the necessary knowledge.

To tackle these challenges, we made sure to allocate sufficient time for code analysis and familiarization. Breaking down the codebase into smaller modules or components proved helpful in making the understanding process more manageable. Engaging with the knowledge base, and consulting the introduction documentation and referencing external resources, by collaborating we were able to collectively tackle the challenges and move forward with the project.

References

11.1 List of Citations and Sources

1. "Stock Market Prediction Using Machine Learning Techniques: A Decade Survey on Methodologies, Recent Developments, and Future Directions" (2021) by Kim et al. https://www.mdpi.com/2079-9292/10/21/2717
2. "Deep Reinforcement Learning for Trading—A Critical Survey" (2021) by Wei et al. https://www.mdpi.com/2306-5729/6/11/119
3. "Analysis of financial pressure impacts on the healthcare industry with an explainable machine learning method: China versus the USA" (2022) by Chen et al. https://www.sciencedirect.com/science/article/pii/S095741742201569X?casa_token=ShWyPjBXkOcAAAAA:7PMtIKiFuIvhTQDUHAQtNpGgzcpyN7v5YLHdodBR842PHLZV-1w8MgcgGuMzc2zUDsm2LCUGiBM
4. "Forecasting the Market with Machine Learning Algorithms: An Application of NMC-BERT-LSTM-DQN-X Algorithm in Quantitative Trading" (2022) by Liu et al. https://dl.acm.org/doi/full/10.1145/3488378?casa_token=s3_Ho5MJoJUAAAAA%3AC5kkvqzdHLwrP5Scy7iLio9O0IJLfFVas7djPmDnBZE5eAtvD0Sdwtqdb9PV0WxrsggjWeS4nFnlcEA

5. "Mean–variance portfolio optimization using machine learning-based stock price prediction" (2021) by Wei et al. https://www.sciencedirect.com/science/article/pii/S1568494620308814?casa_token=_OaMHysaUSoAAAAA:kztPFgelR8J2GlsXkZ4IKk0aNZemgUambbZXyxPOlxEiwHN5d7HSqmUNe0-_-BV8kmPJYNll03I
6. "Machine Learning in Finance: A Metadata-Based Systematic Review of the Literature" (2021) by Zhang et al. https://www.mdpi.com/1911-8074/14/7/302
7. "A Machine-Learning Framework for credit risk assessment of margin lending in the capital market of Iran" (2021) by Reza Tehrani et al. https://www.jcreview.com/admin/Uploads/Files/61bb1d58880c22.63535978.pdf
8. "Artificial intelligence and machine learning in finance: Identifying foundations, themes, and research clusters from bibliometric analysis" (2021) by John W. Goodell et al. https://www.sciencedirect.com/science/article/pii/S2214635021001210?casa_token=ppsHDrAc-08AAAAA:VNigRfmaP3kluiMKN08K_C5jykRrvNlCNZdAVOtXQWLMJtGkQ6X26bCpbpsQVX0F50VaxXYZ7aA
9. "Forecasting monthly copper price: A comparative study of various machine learning-based methods" (2021) by Hong Zhang et al. https://www.sciencedirect.com/science/article/pii/S0301420721002038?casa_token=xQsP70S5YkUAAAAA:9gsYNacAB68UejrC5TN5Ga_zBZ4BZOk_89INKjH_en-Ly5lfLUBtyhKQI9-L4lyPN2FH9v6bJ2U
10. "Time-series forecasting of seasonal items sales using machine learning – A comparative analysis" (2022) by Yasaman Ensafi et al. https://www.sciencedirect.com/science/article/pii/S266709682200002