Project Scoping
Stock Price Prediction

GROUP 10

Team Members

Giri Manohar Vemula
Gopichand Kandikonda
Omkar Vilas Narkar
Oviya Gnanasekar
Samarth Saxena
Vy Nguyen

1. Introduction

Stock price prediction and forecasting have grown in popularity in the financial and investing industries. The process of forecasting the future value of a company's shares or any other financial instrument traded on a financial exchange is known as stock market prediction. The stock market is known for its volatility, dynamics, and nonlinearity, the accurate stock price forecasting is exceedingly difficult due to several (macro and micro) elements such as politics, global economic conditions, unforeseen events, a company's financial performance, and so on, all of this means that there is a large amount of data to search through for patterns. As a result, financial analysts, researchers, and data scientists are always investigating new analytics techniques to spot stock market trends. This gave rise to the concept of algorithmic trading with the use of Machine Learning Operations (MLOps).

This project addresses the challenges faced by traditional stock price forecasting models, which often solely depend on historical price data. These models face difficulty to capture the complex dynamics of financial markets, and other external factors. To overcome this, we integrate diverse datasets, including the **Fama/French 5 Factors daily dataset** and the **Yahoo Finance stock price dataset**, which provide both foundational financial features and real-time stock price information.

In recent years, MLOps have been useful in forecasting and predicting stock prices. The goal of this project is to develop an efficient machine learning operations (MLOps) pipeline that incorporates financial modeling with best feature selection and time series analysis to anticipate stock volatility and optimize the model's performance. The major objective is to forecast price movements in stocks while maintaining an automated pipeline that supports continuous integration and deployment. This infrastructure not only helps predictive models run more efficiently, but it also improves their long-term stability by tackling challenges like data drift, model degradation, and the requirement for real-time updates in an ever-changing financial market. The machine learning models and scalable MLOps infrastructure that can easily adapt new data and market patterns, potentially leads to more accurate projections and smarter investment decisions.

2. Dataset Information

2.1. Dataset Introduction

- **Fama/French 5 Factors daily dataset**: It is a widely used financial dataset designed to explain stock market returns. Developed by Eugene Fama and Kenneth French, the 5 factors shown in the data represent various dimensions of risk and characteristics that impact stock prices. It contains data from 07- 01-1963 till date (data is updated every quarter). This dataset provides the foundational features which are crucial for understanding and forecasting stock price movements.

- **Yahoo Finance Stock price dataset**: The Fama/French 5 factors dataset will be combined with a company's daily stock price dataset, obtained from Yahoo Finance using the yfinance library in Python. We get the complete historical stock price data including the different daily stock prices and number of shares traded. This dataset is used for integrating real-time and historical stock prices of the company for the model. In addition, other macroeconomic, financial indicators and sentiment analysis will be added to the total combined dataset to give better predictions.

2.2. Data Card

- Fama/French 5 Factors:
  Size: 15,375 rows x 7 columns

Feature	Data Type	Description
Date	Date	Date of trading for each day
Mkt-RF	Float	Market Risk Premium - Excess return of the overall market over the risk-free rate
SMB	Float	Small Minus Big - Return difference between small-cap to large-cap companies (size factor)
HML	Float	High Minus Low - Return difference between high (value) and low (growth) book-to-market stocks
RMW	Float	Robust Minus Weak - Return difference between firms with robust vs weak profitability
CMA	Float	Conservative Minus Aggressive - Return difference between firms investing conservatively vs aggressively
RF	Float	Risk Free Rate - Return on a risk free investment

Dataset Link - https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html

- Company stock price data from Yahoo Finance:
  Size: 7 columns (Number of rows depends on start date chosen for company stock prices)

Feature	Data Type	Description
Date	Date	Trading date of each stock price record
Open	Float	Opening price of stock on the day
High	Float	Highest price at which the stock was traded
Low	Float	Lowest price at which the stock was traded
Close	Float	Market Closing price of stock on the day
Adj Close	Float	Adjusted closing price of stock accounting for dividends and other corporate actions.
Volume	Integer	Number of shares traded

Company stock price data is obtained from Yahoo Finance through the yfinance library in Python: https://pypi.org/project/yfinance/

2.3. Data Rights and Privacy

- The Fama/French 5 factors dataset is publicly available through the Kenneth French Data Library. The dataset does not contain any personal information and only includes financial market data, thus meeting the data protection regulations of GDPR and CCPA.

- The company stock price data of Yahoo Finance obtained through yfinance is publicly available to use. It does not contain any personal data and is purely financial and stock-related.

3. Data Planning and Splits

3.1. Data Loading and Preprocessing

Fama/French 5 Factors dataset will be loaded as a csv file into a pandas dataframe. Also, we will use download function of yfinance to obtain stock price data into a pandas dataframe. Handle missing values and outliers (extreme values) that could skew the model. Following feature engineering of yfinance stock data, both datasets along with other macroeconomic factors will be merged into one combined dataset through the 'Date' column.

3.2. Feature Engineering for yfinance dataset

Few additional features added to the dataset are highlighted below:

- Daily returns: Daily percentage change in stock price
- Moving Average over a specific window
- Bollinger bands: Upper and lower ranges of price. Measure of price volatility
- Relative Strength Index: Momentum oscillator to measure speed ,change of price movements
- On-Balance Volume: To observe relationship between volume and price movements

These features would help capture the volatility, trend and momentum of the stock price, leading to improved predictions by the model.

3.3. Data Splitting

Since the data is of a time series nature, it will be split into 3 sections for train, test and validation based on chronological order. The first major section will be used for Training, subsequent smaller section for Validation and the final section for Testing. Thes size of the splits will be determined dynamically, to ensure flexibility in case of any changes in the future.

During training of the model, cross-validation methods like Time Series split will be used to ensure that the time series nature of data is preserved.

4. GitHub Repository

- GitHub repository: Stock-Price-Prediction

- Folder structure:

```
|-- Stock-Price-Prediction/
        |-- .DS_Store
        |-- LICENSE
        |-- requirements.txt
        |-- README.md
        |-- current_tree.txt
        |-- pipeline/
                |-- .DS_Store
                |-- config/
                |-- airlfow/
                        |-- .DS_Store
                |-- dags/
                |-- docker.yaml
        |-- test/
        |-- docs/
        |-- mlruns/
        |-- gcpdeploy/
                |-- gcp.py
        |-- branches/
        |-- data/
                |-- FamaFrench_Data_5_Fact.csv
        |-- assets/
                |-- Initial plan.jpg
        |-- notebooks/
                |-- dataprep.ipynb
                |-- prediction.py
        |-- src/
                |-- dataapi.py
```

The folder structure showcases multiple collaborators for a project which has some version control, machine learning notebooks, python files, and Airflow pipeline, and Google Cloud deployment. This structure demonstrates a well-managed project that includes cloud deployment, process automation, and financial data analysis.

- README file: Readme.md

5. Project Scope

5.1. Problems

The project addresses several key problems in stock price forecasting. First, traditional models often rely solely on historical price data, limiting their ability to capture the broader range of factors influencing stock prices. This project seeks to overcome this by integrating diverse indicators, including macroeconomic variables, to better reflect the complex dynamics of the market. Second, the challenge of real-time data processing from multiple sources such as market news, social sentiment, and economic reports presents significant difficulties in efficiently handling large volumes of information and delivering timely predictions. Third, the volatile nature of financial markets requires continuous model adaptation to maintain prediction accuracy, raising concerns about model stability and responsiveness over time. Additionally, the project must ensure model reproducibility and facilitate iterative improvements, all while adhering to regulatory compliance standards in algorithmic trading. Lastly, the computational demands of processing and analyzing real-time financial data at scale pose significant technical challenges, especially in high-frequency trading environments where speed and efficiency are critical.

5.2. Current Solutions

- Current Approaches for Datasets: Many studies on stock price prediction use financial data, especially historical stock prices from platforms like Yahoo Finance. Some studies also analyze textual data from news articles to gauge sentiment, adding a qualitative aspect to their predictions.
- Current Approaches for Models: Various methods are used for stock price prediction, ranging from traditional to modern techniques. Traditional methods include the widely used ARIMA (AutoRegressive Integrated Moving Average) model for time series forecasting. It involves capturing the relationship between observations and their lagged values (autoregression), ensuring stationarity through differencing, and smoothing the data using moving averages. Additionally, regression-based models like XGBoost are popular for handling a large number of features, minimizing overfitting, and effectively managing missing data. On the modern side, Long Short-Term Memory (LSTM) neural networks have gained traction in time series analysis for their ability to learn and model complex temporal patterns efficiently.

5.3. Proposed Solutions

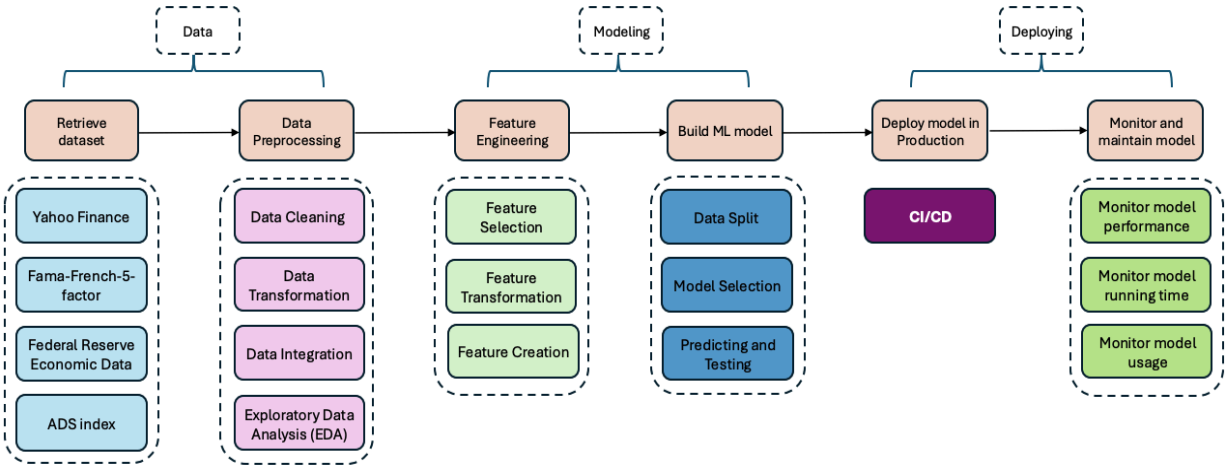
Our proposed solution involves the development of an end-to-end MLOps pipeline designed for real-time stock market price prediction, introducing several key innovations aimed at enhancing predictive accuracy and adaptability. Unlike traditional models that predominantly rely on historical price data, our approach integrates a wider range of quantitative data sources, such as the Fama-French 5 Factor model data. This combination provides a more comprehensive view of the market, allowing for more accurate forecasts.

A potential innovation includes exploring the use of APIs to pull real-time market data, which could offer up-to-the-minute insights into market dynamics and further refine predictions. While this aspect is still under consideration, integrating live data sources would enhance the model's ability to adjust to rapid market changes. By systematically comparing various models and leveraging experiment tracking and model versioning, we ensure the best-performing model is selected for prediction.

Additionally, we emphasize hyperparameter tuning and optimization techniques to further improve model performance, ensuring adaptability to market changes and volatility. Scalability and computational efficiency are key aspects of our design, leveraging cloud platforms to manage large datasets and computational loads efficiently. The pipeline also includes explainability and compliance features to meet regulatory standards, providing transparent and interpretable results for stakeholders. By integrating traditional financial factors with potential real-time data sources, this solution positions itself to outperform conventional forecasting methods and enable more accurate, data-driven investment decisions.

6. Current Approach Flow Chart and Bottleneck Detection

The tasks for this project are depicted in a flowchart to provide a clear overview of the steps required to achieve the project's objectives. However, several potential bottlenecks may arise, as outlined below.



- Data availability: Historical financial data from sources like Yahoo Finance, Fama-French model may be incomplete, or have missing values.
- Data cleaning and preprocessing: Financial data often contains outliers, anomalies, or non-numerical data that complicate preprocessing.
- Feature Engineering Complexity: Identity the right features to use in the model can be time-consuming and requires deep financial knowledge
- Model Selection and Training: Choosing the best algorithm and tuning hyperparameters can be computationally expensive.
- CI/CD pipeline implementation: Building a CI/CD pipeline for machine learning can be complex and time-consuming.
- Monitoring and Maintenance of Deployed model: The performance of deployed models can degrade over time due to changes in market conditions.

7. Business Goals, Project Objectives and Key Metrics

7.1. Business goals

The main business objective of this stock price prediction project is to create a machine learning model that can predict the future performance of a stock. Stock price predictions can help achieve several business goals:

- Improved Investment Strategy: Assisting investors in making more informed decisions when buying or selling stocks
- Operational Efficiency: Using machine learning to automate stock price prediction, leading to quicker responses to market changes and improved operational efficiency in investment management.
- Competitive Advantage: Reliable stock price forecasting can give a business a strategic edge over competitors by effectively capitalizing on market opportunities.

7.2. Project Objectives

- Develop a Stock Price Prediction Model: Build a machine learning model using historical stock price data and relevant features, such as economic indicators, and macroeconomic data, to predict future stock prices. Apply feature engineering and selection techniques to optimize model performance.

- MLOps Integration:
  + Model deployment: Deploy the model into a production environment to enable real-time predictions and scalability.
  + Monitoring: Establish continuous monitoring of the model's performance to detect changes in accuracy or model drift.
  + Automated Retraining: Integrate automated retraining processes triggered by significant drops in model performance or market changes, ensuring the model remains accurate and reliable over time.

7.3. Key metrics

- Model performance: RMSE or MAE will be used to evaluate the model's performance. If the error exceeds a pre-determined threshold, the model will be retrained to improve its accuracy.
- Real-time prediction: Measure the model's running time to ensure it can predict stock prices in real-time, providing timely insights for trading and decision-making.
- Automated CI/CD: Ensure efficient automation of the entire pipeline, including data ingestion, model retraining, evaluation, and deployment, allowing the model to adapt to real-time datasets with minimal manual intervention.
- Continuous Monitoring: Implement robust monitoring systems to track model performance metrics and detect potential data drift, ensuring the model remains aligned with changing market conditions.

8. Failure Analysis

There is a possibility of poor data quality or interruptions in data availability, especially since we are merging datasets from different sources. To address this, we will implement rigorous data validation and cleaning processes to ensure the integrity of our datasets. Errors during data processing and transformation stages such as incorrect data formatting, faulty feature engineering, or integration issues when merging datasets could negatively impact model predictions.

We will check this by applying data validation procedures and by flagging data entries that fall outside expected ranges or patterns. This includes checking for inappropriate negative prices, unusually high or low values, and inconsistencies in data units.

Over time, changes in market conditions may lead to degradation of model performance. To detect and address this, we will implement continuous monitoring of model performance metrics such as RMSE

and MAE. We will also monitor retraining jobs and set up validation checks to ensure successful model updates when necessary. Failures in monitoring and logging systems can hinder our ability to detect and respond to issues promptly. To prevent this, we will implement comprehensive monitoring and logging using GCP tools. By setting up alerts for critical metrics and regularly auditing our monitoring systems, we aim to ensure timely detection and resolution of any pipeline issues.

9. Deployment Infrastructure

We will leverage Google Cloud Platform (GCP) to deploy our machine learning model. Data ingestion will be managed via API calls and integrated into a pipeline, with storage handled by Google Cloud Storage. For orchestrating and managing the data pipelines, we will utilize Cloud Composer (Apache Airflow) or Cloud Dataflow for both batch and stream processing.Model training will be performed using frameworks such as TensorFlow or scikit-learn, with models packaged into containers for deployment via Google AI Platform.

To ensure optimal performance, comprehensive monitoring and logging will be implemented for real-time tracking. Version control of models will allow tracking changes and easy rollbacks when necessary. Additionally, Tableau will be used to visualize predictive results and other relevant data.

Finally, CI/CD pipelines will automate the deployment process, facilitating seamless updates and ongoing model maintenance.

10. Monitoring Plan

The accuracy of our model predictions depends on the quality of input data. We ensure data completeness, validate data accuracy and monitor data drift by tracking the statistical properties of the data over time. This approach allows us to detect significant shifts that could affect model performance. Additionally, by monitoring model drift indicators and observing shifts in feature importance, we can identify changes in the underlying data relationships.

We will monitor pipeline latency to ensure efficiency and collect and analyze error logs to detect exceptions or failures in any pipeline component. To manage and oversee our data ingestion tasks, we utilize Apache Airflow, managed via Google Cloud Composer. We monitor Directed Acyclic Graphs (DAGs) and task statuses through Airflow's user interface, paying special attention for any task failures.

To maintain data integrity, we implement data quality checks at various stages of the pipeline. Any discrepancies or anomalies detected during these checks are logged to Google Cloud Logging for further analysis and troubleshooting. Logs from Airflow, data processing scripts, and other pipeline components are aggregated in Google Cloud Logging, providing a centralized view for monitoring and issue resolution.

For our CI/CD pipeline setup, we implement continuous integration and deployment processes by tracking build and deployment statuses, log durations, and monitor for any failures to ensure smooth and reliable deployments. Key metrics such as prediction latency, throughput, and error rates are tracked to ensure that the model performs optimally in production. Furthermore, we continuously monitor the model retraining process to ensure that our models remain accurate over time, adapting to any changes in market conditions.

11. Success and Acceptance Criteria

The project's successful implementation is determined by efficient stock analysis performance measures such as Uptrend and Resilience, Moving Averages Crossover, Volatility and Recovery values. With a clear view of model performance that predicts different trading strategies and signals of Day, long/short, Buy and Hold.

Acceptance is defined by the resilience of this system in detecting anomalies and maintaining ongoing model performance. The system should ensure that entire tracking and monitoring should be implemented to monitor drift detection, model performance, and failures by meeting the objectives of the project with successful deployment.

12. Timeline Planning

This project will be carried out in stages to enable effective development and deployment. We divide the work into different stages and iterations, the project objectives, success criteria, and data sources are finalized during the first scoping phase. Subsequently, the preprocessing and data collecting stage cleans and gets the historical data ready for model construction. During the proof of concept (POC) stage, early machine learning models are tested, and the cloud development environment is configured on Google Cloud Platform. The project moves on to the setup of the CI/CD pipeline for automated model retraining and deployment after fine-tuning models during the optimization phase.