Data Mining Project Proposal - Stock Market Analysis + Prediction

Grace Ryoo, Henry Lue, Joshua Kim

Project Overview

This project aims to investigate the efficacy of various machine learning models in predicting the directional movement of stock prices. The primary objective is to determine which classification model performs best at this task. A central component of this research is to evaluate the predictive power of common technical indicators. We will compare the performance of models trained on a dataset containing only basic price data (Open, High, Low, Close, Volume) against models trained on an enriched dataset that includes engineered features like moving averages and momentum oscillators. This comparison will help us ascertain whether these computed indicators provide tangible value and improve predictive accuracy beyond what raw price data can offer.

Dataset and Feature Engineering

The foundation of this project will be two custom-built datasets derived from historical stock price information for a selection of tickers within the S&P 500 index. The initial data will be sourced using the yfinance library, providing daily **OHLCV** (Open, High, Low, Close, Volume) values. The target variable, representing the daily price direction, will be binary. It will be labeled as 1 if the closing price of the next day is higher than the current day's closing price (close(t+1)>close(t)) and 0 otherwise. To ensure a robust evaluation, the data will be split

chronologically into a training set and a testing set (e.g., Training: 2010-2021, Testing: 2022-2024), which simulates a real-world scenario where a model predicts future, unseen data.

From this base dataset, a second, feature-rich dataset will be generated by calculating and appending several widely-used technical indicators. These engineered features include:

- **Moving Averages (MA)**: Simple moving averages (SMA) will be calculated over various time windows (e.g., 10-day, 50-day, 200-day) to smooth price data, identify underlying trends, and detect potential support or resistance levels.
- **Relative Strength Index (RSI)**: This momentum oscillator measures the speed and magnitude of recent price changes to identify overbought or oversold conditions in a stock, typically on a scale of 0 to 100.
- **Moving Average Convergence Divergence (MACD)**: This trend-following momentum indicator illustrates the relationship between two exponential moving averages (EMAs) of a stock's price. The MACD can signal potential buy and sell opportunities through crossovers and divergences.

The creation of these two distinct datasets—one with raw data and one with engineered features—is crucial for our comparative analysis.

Methodology and Evaluation

Our methodology involves training and evaluating several distinct machine learning models on both the basic and the feature-engineered datasets. The selected models represent a range of techniques, including linear models, tree-based ensembles, and neural networks. Specifically, we

will compare **Logistic Regression, Decision Trees, Random Forests, Support Vector Machines (SVM)**, and a **Multi-Layer Perceptron (MLP)**. This approach will allow us to determine not only if technical indicators add value but also which type of algorithm is best suited for this classification task.

The performance of each model will be rigorously assessed using a combination of metrics:

- **Primary Metric (Accuracy)**: The main yardstick for success will be classification accuracy. Our fundamental goal is to develop a model that can predict the stock's direction more effectively than a random guess.
- **Secondary Metrics (Precision, Recall, F1-Score)**: To gain a more nuanced understanding of model behavior, we will also analyze Precision, Recall, and the F1-Score. These are particularly important for identifying the reliability of specific predictions. For example, high precision for "Up" predictions would signify a trustworthy "buy" signal, minimizing false positives. These metrics are essential for evaluating performance, especially if the dataset has an imbalance between "up" and "down" days.
- **Baseline:** All model performances will be benchmarked against a simple baseline, such as the natural frequency of "up" days in the historical data or a 50% random chance model. A model is only considered useful if it significantly outperforms this baseline.

Final Objectives and Hypotheses

This project is guided by two primary hypotheses that we seek to validate. The final objective is to deliver a comprehensive analysis that either supports or refutes these claims based on empirical evidence from our experiments.

1. **Feature Impact Hypothesis**: We hypothesize that a classification model trained on the dataset with engineered technical indicators (RSI, MACD, etc.) will achieve a significantly higher predictive accuracy than a model trained solely on raw OHLCV data.
2. **Model Performance Hypothesis**: We predict that an ensemble method, specifically Random Forest, will outperform a single Decision Tree classifier. This is expected because ensemble models typically reduce variance and are less prone to overfitting, leading to more robust and generalizable predictions on unseen test data.

Resources

https://www.geeksforgeeks.org/python/web-scraping-for-stock-prices-in-python/

- yfinance python library
    - https://github.com/ranaroussi/yfinance?tab=readme-ov-file
    - https://pypi.org/project/yfinance/
    - https://nebigdatahub.org/yahoo-finance-project/
    - https://www.kaggle.com/code/faressayah/stock-market-analysis-prediction-using-lstm
- Data mining models
    - https://www.geeksforgeeks.org/machine-learning/random-forest-regression-in-python/
    - https://www.geeksforgeeks.org/machine-learning/decision-tree-implementation-python/
    - https://www.datacamp.com/tutorial/svm-classification-scikit-learn-python
    - https://www.geeksforgeeks.org/deep-learning/multi-layer-perceptron-learning-in-tensorflow/
    - https://www.datacamp.com/tutorial/understanding-logistic-regression-python