Aung Si
September 18th, 2023

Capstone Proposal: Automated Portfolio Optimization using Machine Learning

Business Understanding

This project aims to augment current portfolio management techniques by employing machine learning algorithms. Traditional methods, such as mean-variance optimization, offer a static model that lacks the adaptability conferred by predictive analytics. By integrating machine learning into portfolio optimization, this project aims to add a layer of time-sensitive adaptability. The target audience includes institutional investors and hedge funds within the finance industry. The project aligns with my intrinsic interest in tackling the intricate and often unpredictable nature of the stock market. It serves as a quant-level project that feeds into my overarching goal: to leverage data science in making financial strategies more accessible and nuanced.

Data Understanding

The data comes from stock market prices (transformed into stock market returns), retrieved via the *yfinance* package. Sector-level aggregated returns are computed off individual stock returns and are the targets for the models. Features include rolling Sharpe ratios, the returns of various market indices (Bond, Stock, Commodities, and Volatility indices), and the rest are lagged versions of these features. The feature set aim to capture different attributes such as past returns and sector averages. PCA will potentially be used depending on the nature of multicollinearity within the feature set. The project expands on prior work by combining multiple machine learning models and portfolio optimization strategies.

Data Preparation

For Data Preparation, the dataset resides in Pandas dataframes and predominantly comprises numerical variables such as returns and volatility. Preliminary statistics and frequency counts are computed to gain insights into data distribution. The preprocessing steps include making the time-series data stationary by applying logarithmic transformations or differencing. Features are also shifted to align with the forecast horizon of 126 days, and rows with NA values are eliminated. A minimum of 4230 rows will be utilized for the analysis. Various forms of data visualizations, like bar graphs and line plots, serve to elucidate important data characteristics in the exploratory phase.

Modeling

In the Modeling phase, the primary focus is regression. The suite of models includes ElasticNet, Support Vector Regressor, RandomForestRegressor, GradientBoostingRegressor, XGBoostRegressor, and SARIMAX. These models undergo preprocessing steps that include PCA and standard scaling. Training and testing are carried out using two distinct approaches for robust validation:

1) A rolling window of a fixed 126-day interval for both validation and prediction
2) An expanding rolling window for the same purposes.

Time-series cross-validation is employed to ensure model integrity. The target variable is future stock returns. Naïve moving average and linear regression serve as baseline models. The range of machine learning algorithms is chosen to account for the nonlinear dynamics inherent in stock market data.

Evaluation

The project primarily deploys supervised learning techniques to forecast future stock returns. For model validation, we utilize time-series cross-validation along with Over-Under Loss (OUL), a tailored metric that imposes penalties on overpredictions, an essential aspect for a long-only investment strategy. Beyond these evaluative measures, the project also mandates backtesting across multiple portfolio optimization strategies such as Equal-Weight, Maximum Sharpe, Minimum Variance, and Risk Parity. This backtesting phase is not merely adjunctive but indispensable in the financial context, providing a temporal lens for performance evaluation. The minimum viable product (MVP) will incorporate simpler models, specifically moving average and linear regression, to establish baseline performance.

Deployment

The final interface will be deployed via Streamlit to offer a real-time dashboard. This will enable users to visualize the chosen portfolio's structure and backtest results. A 'Recommend Stocks' button will also be available, allowing users to generate a list of algorithmically recommended stocks.

Tools/Methodologies

Python libraries such as Pandas, NumPy, Matplotlib, Seaborn, and scikit-learn are used. Custom modules will also be employed for specific functionalities like portfolio optimization. The analysis is performed locally, and data is stored on the machine.

By marrying machine learning with traditional finance, this project aims to put forth a nuanced approach to how portfolios can be managed, making the process not only automated but also more model-driven and data-centric.