Table of Contents

Abstract 1

Object-Oriented Programming 2

Statistical Methodology 3

API Throttling and Rate Limiting 5

Paper Trading and Backtesting 6

Data Visualizastion 7

Conclusion 8

Abstract

This project focuses on developing an algorithmic trading bot that implements the Pairs Trading strategy, a statistical arbitrage approach that capitalizes on the mean-reverting relationship between two correlated stocks. Built in Python, the bot will retrieve real-time and historical financial data, identify cointegrated stock pairs using the Engle-Granger test, and generate trading signals based on Z-score deviations.

Key features include efficient API interactions with rate-limit handling, secure authentication, and back-testing capabilities to evaluate performance through metrics such as ROI, Sharpe ratio, and drawdown. A paper trading environment will be incorporated to simulate live trading conditions without financial risk. Additionally, comprehensive logging and monitoring will track trade execution, portfolio updates, and performance insights to refine and optimize the strategy.

Designed for scalability and efficiency, this project provides a structured and data-driven approach to algorithmic trading, making it a valuable tool for traders seeking to automate mean-reversion strategies.

Object-Oriented Programming

Our Pairs Trading bot uses structured Object-Oriented programming in order to complete its task. The core functionality is encapsulated within a single class named PairsTradingBot, which operates as an autonomous trading agent that fetches data, performs statistical analysis, executes backtests, and simulates paper trades.

This class encapsulates all relevant data and behavior related to the trading strategy. The internal state of the bot is described by attributes such as self.data, self.log_returns, self.z_scores, and self.portfolio_value. Methods like fetch_data(), test_cointegration(), calculate_spread_and_zscore(), and backtest() expose well-defined interfaces to perform specific tasks. Encapsulation ensures the internal workings of the strategy are hidden from the user allowing them to interact with the bot at a high level without needing to understand or modify internal logic.

Abstraction is also achieved through method definitions that represent a logic flow of the pairs trading strategy. The user of the class does not need to manage raw data handling or regression calculations manually; instead, they can interact with the bot through high-level method calls like bot.backtest() or bot.paper_trade. This also hides the complexity of the implementation and presents a simplified interface for running the strategy.

By designing the bot as a class, all trading logic is centralized and can be easily reused or extended. For example, the same class can be used to analyze any pair of stocks simply by changing the pair_symbols input during initialization.

The class also maintains state throughout its lifecycle. Variables such as self.trade_log and self.positions evolve as methods are called, allowing for a continuous record of trading activity and portfolio performance. This persistent state enables seamless transitions between different phases of strategy execution, such as transitioning from backtesting to paper trading.

Statistical Methodology

The algorithmic trading approach implemented in our code gathers data from Yahoo Finance and then follows the three steps below:

- Cointegration Testing
- Z-score calculations
- Statistical Arbitrage Thresholds

Cointegration Testing

Cointegrated pairs are two financial assets whose prices are strongly correlated over the long term despite short-term fluctuations. As a result, their linear combination remains stationary over time due to the mean-reversion quality that the two pairs exhibit. Given this stationarity, we use the Engle-Granger Two-Step Method approach to find cointegrated pairs. The Engle-Granger method consists of the following steps:

1. Run a linear regression on the closing prices of two financial assets.

$$Y_t = \alpha + \beta X_t + \epsilon_t$$

2. From the linear model, we extracted the residuals and performed the Augmented Dickey-Fuller (ADF) Test, where we test:

H_o: The residuals are not stationary
H_1: The residuals are stationary

Using the p-value, we make statistical inferences about the stationarity of the process between the pairs. Testing at a 5% significance level, if the p-value is less than α=0.05, we reject H_o; if it's greater, we fail to reject H_o.

Therefore, if the ADF test concludes that the residuals of the linear regression model are stationary, the program determines that the pairs are cointegrated.

In the code provided, we create a function called cointegreation_test(). This function iterates through the pairs of assets that the user inputs using a nested loop and applies the Engle-Granger Test from the statsmodel library. Then, we use the coint() function from the module statsmodels.tsa.stattools. This function performs linear regression internally, extracts the residuals, and automatically applies the ADF test following the above decision rule. After testing the possible pairs, the function returns if they are cointegrated or not.

Z-score Calculations

Following the Engle-Granger Two-Step Method, we compute the Z-score of the spread between the two assets using the following formula:

$$Z = \frac{X - \mu}{\sigma}$$

Where:

- X=spread between the two assets(series)
- μ=historical average of the series(series.mean())
- σ=historical deviation of the series(series.std())

This normalization allows us to determine how unusual the current value is relative to historical data. The program calculates Z-scores using a z-score function using the formula and variable names defined above.

Statistical Arbitrage Thresholds

After calculating the Z-scores, we define a function generate_trading_signals () that computes the spread between the two pairs and compares the spread to the z-score using the previously defined z-score formula. Then, the following trading signals are generated:

- If the Z-score is greater than +2, the spread is too wide, so the program shorts the overperforming asset and buys the underperforming one.
- If the Z-score is less than -2, the spread is determined to be too narrow. Thus, we buy the overperforming asset and sell the underperforming one.
- If the Z-score is zero, the spread has reverted to the mean, so we close the position.

The Z-score thresholds defined above were determined based on the empirical rule (68-95-99.7 rule), where 95% of the data points fall within two standard deviations of the mean. Hence, values beyond +2 and -2 are rare and signal an extreme deviation from the historical mean.

API Throttling and Rate Limiting

The program provided implements API throttling and rate limiting within the fetch_stock_data() function to ensure compliance with Yahoo Finance's request constraints. By incorporating time.sleep(2) after each API request, the program introduces a two-second delay between consecutive calls, reducing the risk of exceeding API rate limits. This mechanism prevents excessive requests that could trigger temporary bans or data retrieval failures, ensuring smoother and more sustainable access to financial data. Additionally, this delay can be adjusted to accommodate more restrictive rate limits if necessary, making the function adaptable to different API usage policies.

Beyond throttling, fetch_stock_data() also incorporates robust error handling to maintain stable execution, even when API issues arise. The try-except block ensures that if a request fails due to network disruptions, invalid tickers, or API timeouts, the program does not crash but instead logs an error message with details about the failure. Furthermore, the function checks if the returned stock data is empty, issuing a warning rather than assuming validity. This proactive approach prevents incorrect calculations downstream and alerts the user to potential issues with data retrieval. Together, these implementations enhance the program's reliability, allowing it to handle API constraints while ensuring uninterrupted data flow for cointegration analysis and trading signal generation.

Paper Trading and Backtesting

We built a back-testing system that simulates trading using historical stock data to understand how well our pairs trading strategy might perform in the real world.

We started by collecting five years of price data for the user's chosen stock pairs like Apple and Microsoft, or Coca-Cola and Pepsi. Our bot carefully checked each possible pair to find those that tend to move together over time. Once it found the strongest match, it calculated how far apart the two stocks drift from each other and used this to spot trading opportunities.

Whenever one stock became unusually expensive compared to its partner (based on the Z-score), the bot would simulate trades, buying the cheaper one and selling the more expensive one, expecting them to return to their usual relationship.

We kept track of each trade and measured how the portfolio grew or shrank over time. The paper trading simulation mimicked how the bot would behave in real-time, tracking available cash, updating positions, and logging each trade decision based on incoming signals. This gave us an additional layer of validation, showing how the strategy would have managed capital and reacted to market dynamics under live conditions.

Throughout the process, we also implemented logging to monitor the bot's activity at each step—whether it was fetching data, detecting signals, executing trades, or evaluating performance. These logs were essential for debugging, auditing decisions, and ensuring the strategy behaved as expected. In the end, we looked at key results like:

- Return on Investment (ROI): How much profit the bot would have made.
- Sharpe Ratio: The returns per unit of risk.
- Maximum Drawdown: How deep the worst dips in value were.

These results gave us a clear and realistic picture of how our strategy would've behaved in past markets, during both stable periods and more chaotic times like economic shifts or inflation surges.

Data Visualization

Data visualization is an integral component of our Pairs Trading bot, serving as a powerful tool for interpreting the strategy's behavior and performance over time. In our implementation, the plot_results() method leverages the matplotlib library to produce two key visualizations: the evolution of the Z-score and the cumulative portfolio value. These plots not only provide an intuitive understanding of the strategy's inner workings but also support performance evaluation and strategy validation.

The first plot displays the Z-score of the spread between the two assets. By visualizing the Z-score over time and overlaying threshold lines at -2 and +2, we are able to clearly see when the strategy would initiate a long or short trade. This plot offers immediate insight into the frequency and timing of signal triggers and allows us to verify whether the spread tends to revert to its mean following extreme deviations—an essential assumption in mean-reversion-based trading strategies like pairs trading.

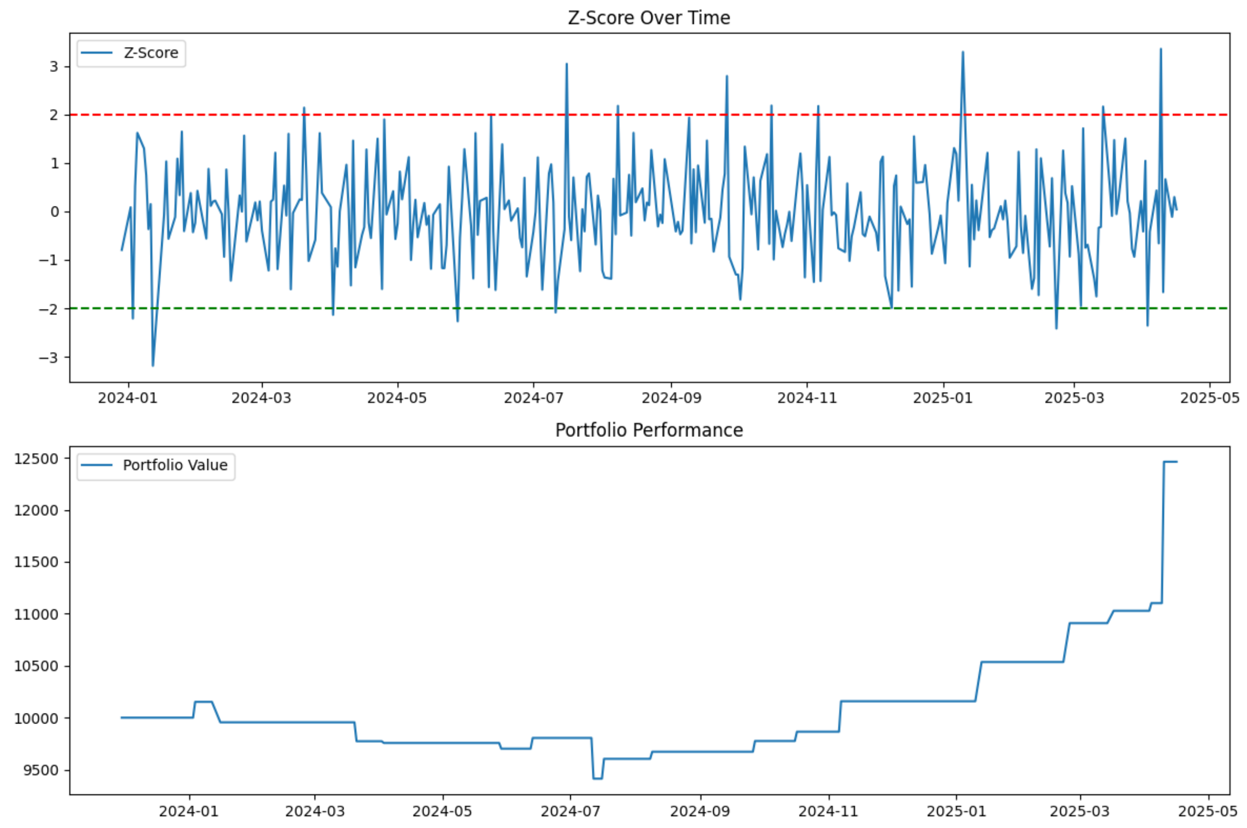
The second plot illustrates the cumulative portfolio value throughout the trading period, starting from the initial capital base. This visualization captures the compounding effects of the trading strategy, showcasing how the portfolio's value fluctuates in response to market conditions and trading decisions. It allows us to easily identify periods of strong performance, drawdowns, and recoveries. Moreover, it provides a tangible sense of the strategy's return profile and risk exposure, complementing statistical metrics such as ROI, Sharpe Ratio, and maximum drawdown.

Beyond performance evaluation, these visualizations enhance the transparency and communicability of the trading strategy. They make it easier for analysts, stakeholders, or potential investors to grasp the dynamics of the model without delving into the raw data or code. By transforming quantitative outputs into visual narratives, we not only validate the integrity of the strategy but also improve its accessibility and interpretability. Ultimately, the inclusion of clear, informative plots is a vital component of both the analytical and presentation aspects of the Pairs Trading bot.

Conclusion

Building this trading bot has been both a technical challenge and a fascinating dive into how markets behave. Using data and statistics, we created a strategy to make informed trading decisions based on patterns we found between pairs of stocks. The back testing gave us confidence that the bot isn't just an idea; it holds up when tested against real-world scenarios. In many cases, especially during market volatility, the strategy identified meaningful opportunities and reacted profitably. What's especially exciting is that this project combines everything from programming and data science to economics and market psychology. It shows how powerful algorithmic trading can be when grounded in theory and evidence.

Graphs



After running our Paris Trading Strategy code and inputting DAL (Delta Air Lines Inc.) and KO (Coca-Cola Co.) as our tickers, we see that the pair are cointegrated. The first graph shows the Z-Score over the backtested period of time. The bottom graph shows the portfolio performance using the pairs trading strategy on DAL and KO.

Paper Trading Simulation Log

```
2025-04-17 12:53:11,692 - INFO - Fetching data for ['DAL', 'KO']
2025-04-17 12:53:12,186 - INFO - Data fetched for DAL
2025-04-17 12:53:24,567 - INFO - Data fetched for KO
2025-04-17 12:53:36,578 - INFO - Testing cointegration...
2025-04-17 12:53:36,621 - INFO - Cointegration p-value: 5.905008471417287e-05
2025-04-17 12:53:36,621 - INFO - Pair is cointegrated. Proceeding with strategy.
2025-04-17 12:53:36,629 - INFO - Starting backtest...
C:\Users\tvenu\AppData\Local\Temp\ipykernel_42720\443181854.py:129: FutureWarning:
Series.__getitem__ treating keys as positions is deprecated. In a future version, integer keys will always
be treated as labels (consistent with DataFrame behavior). To access a value by position, use
`ser.iloc[pos]`
  roi = (cumulative_returns[-1] / self.portfolio_value - 1) * 100
2025-04-17 12:53:36,646 - INFO - ROI: 24.62%
2025-04-17 12:53:36,646 - INFO - Sharpe Ratio: 1.33
2025-04-17 12:53:36,646 - INFO - Max Drawdown: 7.28%
2025-04-17 12:53:36,646 - INFO - Starting paper trading simulation...
2025-04-17 12:53:36,646 - INFO - 2023-12-29 00:00:00-05:00: Portfolio Value: 10000.00
2025-04-17 12:53:36,646 - INFO - 2024-01-02 00:00:00-05:00: Portfolio Value: 10000.00
2025-04-17 12:53:36,646 - INFO - 2024-01-03 00:00:00-05:00: Portfolio Value: 10000.00
2025-04-17 12:53:36,654 - INFO - 2024-01-04 00:00:00-05:00: Portfolio Value: 10000.65
2025-04-17 12:53:36,655 - INFO - 2024-01-05 00:00:00-05:00: Portfolio Value: 10002.06
2025-04-17 12:53:36,655 - INFO - 2024-01-08 00:00:00-05:00: Portfolio Value: 10002.71
2025-04-17 12:53:36,657 - INFO - 2024-01-09 00:00:00-05:00: Portfolio Value: 10003.49
2025-04-17 12:53:36,657 - INFO - 2024-01-10 00:00:00-05:00: Portfolio Value: 10003.09
2025-04-17 12:53:36,658 - INFO - 2024-01-11 00:00:00-05:00: Portfolio Value: 10003.62
2025-04-17 12:53:36,659 - INFO - 2024-01-12 00:00:00-05:00: Portfolio Value: 9999.32
2025-04-17 12:53:36,661 - INFO - 2024-01-16 00:00:00-05:00: Portfolio Value: 9998.12
2025-04-17 12:53:36,661 - INFO - 2024-01-17 00:00:00-05:00: Portfolio Value: 9997.46
2025-04-17 12:53:36,662 - INFO - 2024-01-18 00:00:00-05:00: Portfolio Value: 9999.01
2025-04-17 12:53:36,662 - INFO - 2024-01-19 00:00:00-05:00: Portfolio Value: 9998.05
2025-04-17 12:53:36,662 - INFO - 2024-01-22 00:00:00-05:00: Portfolio Value: 9997.96
2025-04-17 12:53:36,662 - INFO - 2024-01-23 00:00:00-05:00: Portfolio Value: 9999.45
2025-04-17 12:53:36,664 - INFO - 2024-01-24 00:00:00-05:00: Portfolio Value: 10001.62
2025-04-17 12:53:36,665 - INFO - 2024-01-25 00:00:00-05:00: Portfolio Value: 10005.06
2025-04-17 12:53:36,665 - INFO - 2024-01-26 00:00:00-05:00: Portfolio Value: 10003.57
2025-04-17 12:53:36,667 - INFO - 2024-01-29 00:00:00-05:00: Portfolio Value: 10003.79
2025-04-17 12:53:36,667 - INFO - 2024-01-30 00:00:00-05:00: Portfolio Value: 10002.34
2025-04-17 12:53:36,667 - INFO - 2024-01-31 00:00:00-05:00: Portfolio Value: 10002.38
2025-04-17 12:53:36,669 - INFO - 2024-02-01 00:00:00-05:00: Portfolio Value: 10000.73
2025-04-17 12:53:36,669 - INFO - 2024-02-02 00:00:00-05:00: Portfolio Value: 10002.01
2025-04-17 12:53:36,669 - INFO - 2024-02-05 00:00:00-05:00: Portfolio Value: 10001.51
2025-04-17 12:53:36,670 - INFO - 2024-02-06 00:00:00-05:00: Portfolio Value: 10003.56
```

```
2025-04-17 12:53:36,674 - INFO - 2024-02-07 00:00:00-05:00: Portfolio Value: 10003.52
2025-04-17 12:53:36,676 - INFO - 2024-02-08 00:00:00-05:00: Portfolio Value: 10004.11
2025-04-17 12:53:36,676 - INFO - 2024-02-09 00:00:00-05:00: Portfolio Value: 10004.95
2025-04-17 12:53:36,679 - INFO - 2024-02-12 00:00:00-05:00: Portfolio Value: 10004.81
2025-04-17 12:53:36,679 - INFO - 2024-02-13 00:00:00-05:00: Portfolio Value: 10004.36
2025-04-17 12:53:36,679 - INFO - 2024-02-14 00:00:00-05:00: Portfolio Value: 10006.12
2025-04-17 12:53:36,687 - INFO - 2024-02-15 00:00:00-05:00: Portfolio Value: 10006.22
2025-04-17 12:53:36,695 - INFO - 2024-02-16 00:00:00-05:00: Portfolio Value: 10004.50
2025-04-17 12:53:36,695 - INFO - 2024-02-20 00:00:00-05:00: Portfolio Value: 10002.93
2025-04-17 12:53:36,695 - INFO - 2024-02-21 00:00:00-05:00: Portfolio Value: 10002.20
2025-04-17 12:53:36,695 - INFO - 2024-02-22 00:00:00-05:00: Portfolio Value: 10005.06
2025-04-17 12:53:36,695 - INFO - 2024-02-23 00:00:00-05:00: Portfolio Value: 10004.37
2025-04-17 12:53:36,695 - INFO - 2024-02-26 00:00:00-05:00: Portfolio Value: 10005.45
2025-04-17 12:53:36,695 - INFO - 2024-02-27 00:00:00-05:00: Portfolio Value: 10006.54
2025-04-17 12:53:36,695 - INFO - 2024-02-28 00:00:00-05:00: Portfolio Value: 10006.48
2025-04-17 12:53:36,695 - INFO - 2024-02-29 00:00:00-05:00: Portfolio Value: 10007.73
2025-04-17 12:53:36,703 - INFO - 2024-03-01 00:00:00-05:00: Portfolio Value: 10008.46
2025-04-17 12:53:36,703 - INFO - 2024-03-04 00:00:00-05:00: Portfolio Value: 10006.75
2025-04-17 12:53:36,703 - INFO - 2024-03-05 00:00:00-05:00: Portfolio Value: 10007.76
2025-04-17 12:53:36,711 - INFO - 2024-03-06 00:00:00-05:00: Portfolio Value: 10008.14
2025-04-17 12:53:36,711 - INFO - 2024-03-07 00:00:00-05:00: Portfolio Value: 10009.86
2025-04-17 12:53:36,711 - INFO - 2024-03-08 00:00:00-05:00: Portfolio Value: 10008.64
2025-04-17 12:53:36,711 - INFO - 2024-03-11 00:00:00-04:00: Portfolio Value: 10008.12
2025-04-17 12:53:36,711 - INFO - 2024-03-12 00:00:00-04:00: Portfolio Value: 10007.78
2025-04-17 12:53:36,711 - INFO - 2024-03-13 00:00:00-04:00: Portfolio Value: 10008.86
2025-04-17 12:53:36,711 - INFO - 2024-03-14 00:00:00-04:00: Portfolio Value: 10007.44
2025-04-17 12:53:36,711 - INFO - 2024-03-15 00:00:00-04:00: Portfolio Value: 10008.76
2025-04-17 12:53:36,719 - INFO - 2024-03-18 00:00:00-04:00: Portfolio Value: 10008.91
2025-04-17 12:53:36,719 - INFO - 2024-03-19 00:00:00-04:00: Portfolio Value: 10009.31
2025-04-17 12:53:36,719 - INFO - 2024-03-20 00:00:00-04:00: Portfolio Value: 10011.54
2025-04-17 12:53:36,719 - INFO - 2024-03-21 00:00:00-04:00: Portfolio Value: 10012.43
2025-04-17 12:53:36,719 - INFO - 2024-03-22 00:00:00-04:00: Portfolio Value: 10012.00
2025-04-17 12:53:36,719 - INFO - 2024-03-25 00:00:00-04:00: Portfolio Value: 10011.92
2025-04-17 12:53:36,719 - INFO - 2024-03-26 00:00:00-04:00: Portfolio Value: 10012.29
2025-04-17 12:53:36,727 - INFO - 2024-03-27 00:00:00-04:00: Portfolio Value: 10013.18
2025-04-17 12:53:36,727 - INFO - 2024-03-28 00:00:00-04:00: Portfolio Value: 10013.59
2025-04-17 12:53:36,727 - INFO - 2024-04-01 00:00:00-04:00: Portfolio Value: 10014.42
2025-04-17 12:53:36,727 - INFO - 2024-04-02 00:00:00-04:00: Portfolio Value: 10013.63
2025-04-17 12:53:36,727 - INFO - 2024-04-03 00:00:00-04:00: Portfolio Value: 10013.60
2025-04-17 12:53:36,727 - INFO - 2024-04-04 00:00:00-04:00: Portfolio Value: 10013.26
2025-04-17 12:53:36,727 - INFO - 2024-04-05 00:00:00-04:00: Portfolio Value: 10013.21
2025-04-17 12:53:36,727 - INFO - 2024-04-08 00:00:00-04:00: Portfolio Value: 10015.56
2025-04-17 12:53:36,727 - INFO - 2024-04-09 00:00:00-04:00: Portfolio Value: 10015.29
2025-04-17 12:53:36,727 - INFO - 2024-04-10 00:00:00-04:00: Portfolio Value: 10014.71
```

```
2025-04-17 12:53:36,727 - INFO - 2024-04-11 00:00:00-04:00: Portfolio Value: 10017.21
2025-04-17 12:53:36,735 - INFO - 2024-04-12 00:00:00-04:00: Portfolio Value: 10017.18
2025-04-17 12:53:36,735 - INFO - 2024-04-15 00:00:00-04:00: Portfolio Value: 10017.04
2025-04-17 12:53:36,735 - INFO - 2024-04-16 00:00:00-04:00: Portfolio Value: 10016.99
2025-04-17 12:53:36,735 - INFO - 2024-04-17 00:00:00-04:00: Portfolio Value: 10018.75
2025-04-17 12:53:36,735 - INFO - 2024-04-18 00:00:00-04:00: Portfolio Value: 10017.92
2025-04-17 12:53:36,735 - INFO - 2024-04-19 00:00:00-04:00: Portfolio Value: 10014.92
2025-04-17 12:53:36,735 - INFO - 2024-04-22 00:00:00-04:00: Portfolio Value: 10017.05
2025-04-17 12:53:36,735 - INFO - 2024-04-23 00:00:00-04:00: Portfolio Value: 10017.29
2025-04-17 12:53:36,735 - INFO - 2024-04-24 00:00:00-04:00: Portfolio Value: 10012.97
2025-04-17 12:53:36,735 - INFO - 2024-04-25 00:00:00-04:00: Portfolio Value: 10016.44
2025-04-17 12:53:36,744 - INFO - 2024-04-26 00:00:00-04:00: Portfolio Value: 10016.52
2025-04-17 12:53:36,744 - INFO - 2024-04-29 00:00:00-04:00: Portfolio Value: 10016.98
2025-04-17 12:53:36,744 - INFO - 2024-04-30 00:00:00-04:00: Portfolio Value: 10016.75
2025-04-17 12:53:36,744 - INFO - 2024-05-01 00:00:00-04:00: Portfolio Value: 10016.35
2025-04-17 12:53:36,744 - INFO - 2024-05-02 00:00:00-04:00: Portfolio Value: 10018.23
2025-04-17 12:53:36,744 - INFO - 2024-05-03 00:00:00-04:00: Portfolio Value: 10018.91
2025-04-17 12:53:36,744 - INFO - 2024-05-06 00:00:00-04:00: Portfolio Value: 10021.29
2025-04-17 12:53:36,744 - INFO - 2024-05-07 00:00:00-04:00: Portfolio Value: 10019.32
2025-04-17 12:53:36,744 - INFO - 2024-05-08 00:00:00-04:00: Portfolio Value: 10019.03
2025-04-17 12:53:36,752 - INFO - 2024-05-09 00:00:00-04:00: Portfolio Value: 10019.90
2025-04-17 12:53:36,752 - INFO - 2024-05-10 00:00:00-04:00: Portfolio Value: 10018.87
2025-04-17 12:53:36,752 - INFO - 2024-05-13 00:00:00-04:00: Portfolio Value: 10019.22
2025-04-17 12:53:36,755 - INFO - 2024-05-14 00:00:00-04:00: Portfolio Value: 10020.29
2025-04-17 12:53:36,755 - INFO - 2024-05-15 00:00:00-04:00: Portfolio Value: 10020.63
2025-04-17 12:53:36,755 - INFO - 2024-05-16 00:00:00-04:00: Portfolio Value: 10018.77
2025-04-17 12:53:36,755 - INFO - 2024-05-17 00:00:00-04:00: Portfolio Value: 10019.71
2025-04-17 12:53:36,755 - INFO - 2024-05-20 00:00:00-04:00: Portfolio Value: 10021.18
2025-04-17 12:53:36,755 - INFO - 2024-05-21 00:00:00-04:00: Portfolio Value: 10019.01
2025-04-17 12:53:36,755 - INFO - 2024-05-22 00:00:00-04:00: Portfolio Value: 10017.55
2025-04-17 12:53:36,760 - INFO - 2024-05-23 00:00:00-04:00: Portfolio Value: 10018.56
2025-04-17 12:53:36,760 - INFO - 2024-05-24 00:00:00-04:00: Portfolio Value: 10020.01
2025-04-17 12:53:36,760 - INFO - 2024-05-28 00:00:00-04:00: Portfolio Value: 10017.26
2025-04-17 12:53:36,760 - INFO - 2024-05-29 00:00:00-04:00: Portfolio Value: 10016.48
2025-04-17 12:53:36,760 - INFO - 2024-05-30 00:00:00-04:00: Portfolio Value: 10016.32
2025-04-17 12:53:36,760 - INFO - 2024-05-31 00:00:00-04:00: Portfolio Value: 10016.23
2025-04-17 12:53:36,760 - INFO - 2024-06-03 00:00:00-04:00: Portfolio Value: 10015.55
2025-04-17 12:53:36,760 - INFO - 2024-06-04 00:00:00-04:00: Portfolio Value: 10009.84
2025-04-17 12:53:36,760 - INFO - 2024-06-05 00:00:00-04:00: Portfolio Value: 10012.75
2025-04-17 12:53:36,760 - INFO - 2024-06-06 00:00:00-04:00: Portfolio Value: 10010.92
2025-04-17 12:53:36,760 - INFO - 2024-06-07 00:00:00-04:00: Portfolio Value: 10011.65
2025-04-17 12:53:36,768 - INFO - 2024-06-10 00:00:00-04:00: Portfolio Value: 10012.76
2025-04-17 12:53:36,768 - INFO - 2024-06-11 00:00:00-04:00: Portfolio Value: 10009.28
2025-04-17 12:53:36,768 - INFO - 2024-06-12 00:00:00-04:00: Portfolio Value: 10015.19
```

```
2025-04-17 12:53:36,768 - INFO - 2024-06-13 00:00:00-04:00: Portfolio Value: 10014.08
2025-04-17 12:53:36,768 - INFO - 2024-06-14 00:00:00-04:00: Portfolio Value: 10011.12
2025-04-17 12:53:36,768 - INFO - 2024-06-17 00:00:00-04:00: Portfolio Value: 10012.94
2025-04-17 12:53:36,768 - INFO - 2024-06-18 00:00:00-04:00: Portfolio Value: 10012.67
2025-04-17 12:53:36,768 - INFO - 2024-06-20 00:00:00-04:00: Portfolio Value: 10013.61
2025-04-17 12:53:36,768 - INFO - 2024-06-21 00:00:00-04:00: Portfolio Value: 10012.00
2025-04-17 12:53:36,776 - INFO - 2024-06-24 00:00:00-04:00: Portfolio Value: 10009.69
2025-04-17 12:53:36,776 - INFO - 2024-06-25 00:00:00-04:00: Portfolio Value: 10008.77
2025-04-17 12:53:36,777 - INFO - 2024-06-26 00:00:00-04:00: Portfolio Value: 10007.11
2025-04-17 12:53:36,777 - INFO - 2024-06-27 00:00:00-04:00: Portfolio Value: 10008.22
2025-04-17 12:53:36,778 - INFO - 2024-06-28 00:00:00-04:00: Portfolio Value: 10006.43
2025-04-17 12:53:36,779 - INFO - 2024-07-01 00:00:00-04:00: Portfolio Value: 10006.12
2025-04-17 12:53:36,780 - INFO - 2024-07-02 00:00:00-04:00: Portfolio Value: 10005.96
2025-04-17 12:53:36,780 - INFO - 2024-07-03 00:00:00-04:00: Portfolio Value: 10006.96
2025-04-17 12:53:36,780 - INFO - 2024-07-05 00:00:00-04:00: Portfolio Value: 10003.40
2025-04-17 12:53:36,784 - INFO - 2024-07-08 00:00:00-04:00: Portfolio Value: 10005.62
2025-04-17 12:53:36,788 - INFO - 2024-07-09 00:00:00-04:00: Portfolio Value: 10007.22
2025-04-17 12:53:36,788 - INFO - 2024-07-10 00:00:00-04:00: Portfolio Value: 10006.88
2025-04-17 12:53:36,788 - INFO - 2024-07-11 00:00:00-04:00: Portfolio Value: 10002.65
2025-04-17 12:53:36,788 - INFO - 2024-07-12 00:00:00-04:00: Portfolio Value: 9996.81
2025-04-17 12:53:36,790 - INFO - 2024-07-15 00:00:00-04:00: Portfolio Value: 9995.82
2025-04-17 12:53:36,790 - INFO - 2024-07-16 00:00:00-04:00: Portfolio Value: 10002.28
2025-04-17 12:53:36,790 - INFO - 2024-07-17 00:00:00-04:00: Portfolio Value: 9999.92
2025-04-17 12:53:36,792 - INFO - 2024-07-18 00:00:00-04:00: Portfolio Value: 9998.29
2025-04-17 12:53:36,792 - INFO - 2024-07-19 00:00:00-04:00: Portfolio Value: 9999.13
2025-04-17 12:53:36,792 - INFO - 2024-07-22 00:00:00-04:00: Portfolio Value: 9996.96
2025-04-17 12:53:36,794 - INFO - 2024-07-23 00:00:00-04:00: Portfolio Value: 9996.15
2025-04-17 12:53:36,794 - INFO - 2024-07-24 00:00:00-04:00: Portfolio Value: 9993.16
2025-04-17 12:53:36,796 - INFO - 2024-07-25 00:00:00-04:00: Portfolio Value: 9993.70
2025-04-17 12:53:36,796 - INFO - 2024-07-26 00:00:00-04:00: Portfolio Value: 9993.09
2025-04-17 12:53:36,796 - INFO - 2024-07-29 00:00:00-04:00: Portfolio Value: 9991.64
2025-04-17 12:53:36,797 - INFO - 2024-07-30 00:00:00-04:00: Portfolio Value: 9990.37
2025-04-17 12:53:36,797 - INFO - 2024-07-31 00:00:00-04:00: Portfolio Value: 9991.79
2025-04-17 12:53:36,797 - INFO - 2024-08-01 00:00:00-04:00: Portfolio Value: 9986.38
2025-04-17 12:53:36,797 - INFO - 2024-08-02 00:00:00-04:00: Portfolio Value: 9980.38
2025-04-17 12:53:36,797 - INFO - 2024-08-05 00:00:00-04:00: Portfolio Value: 9979.01
2025-04-17 12:53:36,799 - INFO - 2024-08-06 00:00:00-04:00: Portfolio Value: 9979.68
2025-04-17 12:53:36,800 - INFO - 2024-08-07 00:00:00-04:00: Portfolio Value: 9977.09
2025-04-17 12:53:36,800 - INFO - 2024-08-08 00:00:00-04:00: Portfolio Value: 9981.11
2025-04-17 12:53:36,800 - INFO - 2024-08-09 00:00:00-04:00: Portfolio Value: 9980.85
2025-04-17 12:53:36,800 - INFO - 2024-08-12 00:00:00-04:00: Portfolio Value: 9981.12
2025-04-17 12:53:36,800 - INFO - 2024-08-13 00:00:00-04:00: Portfolio Value: 9981.27
2025-04-17 12:53:36,800 - INFO - 2024-08-14 00:00:00-04:00: Portfolio Value: 9980.35
2025-04-17 12:53:36,800 - INFO - 2024-08-15 00:00:00-04:00: Portfolio Value: 9981.76
```

```
2025-04-17 12:53:36,800 - INFO - 2024-08-16 00:00:00-04:00: Portfolio Value: 9981.21
2025-04-17 12:53:36,808 - INFO - 2024-08-19 00:00:00-04:00: Portfolio Value: 9981.73
2025-04-17 12:53:36,808 - INFO - 2024-08-20 00:00:00-04:00: Portfolio Value: 9980.99
2025-04-17 12:53:36,808 - INFO - 2024-08-21 00:00:00-04:00: Portfolio Value: 9980.86
2025-04-17 12:53:36,808 - INFO - 2024-08-22 00:00:00-04:00: Portfolio Value: 9981.05
2025-04-17 12:53:36,808 - INFO - 2024-08-23 00:00:00-04:00: Portfolio Value: 9981.84
2025-04-17 12:53:36,808 - INFO - 2024-08-26 00:00:00-04:00: Portfolio Value: 9980.41
2025-04-17 12:53:36,808 - INFO - 2024-08-27 00:00:00-04:00: Portfolio Value: 9979.60
2025-04-17 12:53:36,808 - INFO - 2024-08-28 00:00:00-04:00: Portfolio Value: 9978.97
2025-04-17 12:53:36,808 - INFO - 2024-08-29 00:00:00-04:00: Portfolio Value: 9979.81
2025-04-17 12:53:36,808 - INFO - 2024-08-30 00:00:00-04:00: Portfolio Value: 9980.32
2025-04-17 12:53:36,808 - INFO - 2024-09-03 00:00:00-04:00: Portfolio Value: 9979.67
2025-04-17 12:53:36,808 - INFO - 2024-09-04 00:00:00-04:00: Portfolio Value: 9980.11
2025-04-17 12:53:36,808 - INFO - 2024-09-05 00:00:00-04:00: Portfolio Value: 9981.27
2025-04-17 12:53:36,817 - INFO - 2024-09-06 00:00:00-04:00: Portfolio Value: 9981.16
2025-04-17 12:53:36,817 - INFO - 2024-09-09 00:00:00-04:00: Portfolio Value: 9982.11
2025-04-17 12:53:36,817 - INFO - 2024-09-10 00:00:00-04:00: Portfolio Value: 9982.12
2025-04-17 12:53:36,817 - INFO - 2024-09-11 00:00:00-04:00: Portfolio Value: 9983.46
2025-04-17 12:53:36,817 - INFO - 2024-09-12 00:00:00-04:00: Portfolio Value: 9983.30
2025-04-17 12:53:36,817 - INFO - 2024-09-13 00:00:00-04:00: Portfolio Value: 9983.55
2025-04-17 12:53:36,817 - INFO - 2024-09-16 00:00:00-04:00: Portfolio Value: 9983.01
2025-04-17 12:53:36,817 - INFO - 2024-09-17 00:00:00-04:00: Portfolio Value: 9984.66
2025-04-17 12:53:36,817 - INFO - 2024-09-18 00:00:00-04:00: Portfolio Value: 9984.95
2025-04-17 12:53:36,817 - INFO - 2024-09-19 00:00:00-04:00: Portfolio Value: 9986.27
2025-04-17 12:53:36,825 - INFO - 2024-09-20 00:00:00-04:00: Portfolio Value: 9985.08
2025-04-17 12:53:36,825 - INFO - 2024-09-23 00:00:00-04:00: Portfolio Value: 9985.21
2025-04-17 12:53:36,825 - INFO - 2024-09-24 00:00:00-04:00: Portfolio Value: 9986.24
2025-04-17 12:53:36,825 - INFO - 2024-09-25 00:00:00-04:00: Portfolio Value: 9987.06
2025-04-17 12:53:36,825 - INFO - 2024-09-26 00:00:00-04:00: Portfolio Value: 9990.16
2025-04-17 12:53:36,825 - INFO - 2024-09-27 00:00:00-04:00: Portfolio Value: 9990.16
2025-04-17 12:53:36,825 - INFO - 2024-09-30 00:00:00-04:00: Portfolio Value: 9990.16
2025-04-17 12:53:36,825 - INFO - 2024-10-01 00:00:00-04:00: Portfolio Value: 9990.16
2025-04-17 12:53:36,825 - INFO - 2024-10-02 00:00:00-04:00: Portfolio Value: 9990.16
2025-04-17 12:53:36,825 - INFO - 2024-10-03 00:00:00-04:00: Portfolio Value: 9990.16
2025-04-17 12:53:36,825 - INFO - 2024-10-04 00:00:00-04:00: Portfolio Value: 9990.16
2025-04-17 12:53:36,825 - INFO - 2024-10-07 00:00:00-04:00: Portfolio Value: 9990.16
2025-04-17 12:53:36,833 - INFO - 2024-10-08 00:00:00-04:00: Portfolio Value: 9990.16
2025-04-17 12:53:36,833 - INFO - 2024-10-09 00:00:00-04:00: Portfolio Value: 9990.16
2025-04-17 12:53:36,833 - INFO - 2024-10-10 00:00:00-04:00: Portfolio Value: 9990.16
2025-04-17 12:53:36,833 - INFO - 2024-10-11 00:00:00-04:00: Portfolio Value: 9990.16
2025-04-17 12:53:36,833 - INFO - 2024-10-14 00:00:00-04:00: Portfolio Value: 9990.16
2025-04-17 12:53:36,833 - INFO - 2024-10-15 00:00:00-04:00: Portfolio Value: 9990.16
2025-04-17 12:53:36,833 - INFO - 2024-10-16 00:00:00-04:00: Portfolio Value: 9990.16
2025-04-17 12:53:36,833 - INFO - 2024-10-17 00:00:00-04:00: Portfolio Value: 9990.54
```

```
2025-04-17 12:53:36,833 - INFO - 2024-10-18 00:00:00-04:00: Portfolio Value: 9990.54
2025-04-17 12:53:36,833 - INFO - 2024-10-21 00:00:00-04:00: Portfolio Value: 9990.26
2025-04-17 12:53:36,833 - INFO - 2024-10-22 00:00:00-04:00: Portfolio Value: 9990.56
2025-04-17 12:53:36,833 - INFO - 2024-10-23 00:00:00-04:00: Portfolio Value: 9989.29
2025-04-17 12:53:36,833 - INFO - 2024-10-24 00:00:00-04:00: Portfolio Value: 9988.47
2025-04-17 12:53:36,841 - INFO - 2024-10-25 00:00:00-04:00: Portfolio Value: 9988.75
2025-04-17 12:53:36,841 - INFO - 2024-10-28 00:00:00-04:00: Portfolio Value: 9987.24
2025-04-17 12:53:36,841 - INFO - 2024-10-29 00:00:00-04:00: Portfolio Value: 9984.19
2025-04-17 12:53:36,841 - INFO - 2024-10-30 00:00:00-04:00: Portfolio Value: 9983.43
2025-04-17 12:53:36,841 - INFO - 2024-10-31 00:00:00-04:00: Portfolio Value: 9984.07
2025-04-17 12:53:36,841 - INFO - 2024-11-01 00:00:00-04:00: Portfolio Value: 9982.60
2025-04-17 12:53:36,841 - INFO - 2024-11-04 00:00:00-05:00: Portfolio Value: 9984.20
2025-04-17 12:53:36,841 - INFO - 2024-11-05 00:00:00-05:00: Portfolio Value: 9983.06
2025-04-17 12:53:36,841 - INFO - 2024-11-06 00:00:00-05:00: Portfolio Value: 9977.39
2025-04-17 12:53:36,841 - INFO - 2024-11-07 00:00:00-05:00: Portfolio Value: 9981.09
2025-04-17 12:53:36,841 - INFO - 2024-11-08 00:00:00-05:00: Portfolio Value: 9980.36
2025-04-17 12:53:36,841 - INFO - 2024-11-11 00:00:00-05:00: Portfolio Value: 9974.25
2025-04-17 12:53:36,841 - INFO - 2024-11-12 00:00:00-05:00: Portfolio Value: 9972.96
2025-04-17 12:53:36,841 - INFO - 2024-11-13 00:00:00-05:00: Portfolio Value: 9971.74
2025-04-17 12:53:36,841 - INFO - 2024-11-14 00:00:00-05:00: Portfolio Value: 9970.08
2025-04-17 12:53:36,841 - INFO - 2024-11-15 00:00:00-05:00: Portfolio Value: 9970.04
2025-04-17 12:53:36,849 - INFO - 2024-11-18 00:00:00-05:00: Portfolio Value: 9971.93
2025-04-17 12:53:36,849 - INFO - 2024-11-19 00:00:00-05:00: Portfolio Value: 9970.36
2025-04-17 12:53:36,849 - INFO - 2024-11-20 00:00:00-05:00: Portfolio Value: 9973.36
2025-04-17 12:53:36,849 - INFO - 2024-11-21 00:00:00-05:00: Portfolio Value: 9975.48
2025-04-17 12:53:36,849 - INFO - 2024-11-22 00:00:00-05:00: Portfolio Value: 9975.79
2025-04-17 12:53:36,849 - INFO - 2024-11-25 00:00:00-05:00: Portfolio Value: 9974.40
2025-04-17 12:53:36,849 - INFO - 2024-11-26 00:00:00-05:00: Portfolio Value: 9975.44
2025-04-17 12:53:36,855 - INFO - 2024-11-27 00:00:00-05:00: Portfolio Value: 9976.24
2025-04-17 12:53:36,855 - INFO - 2024-11-29 00:00:00-05:00: Portfolio Value: 9976.11
2025-04-17 12:53:36,855 - INFO - 2024-12-02 00:00:00-05:00: Portfolio Value: 9976.07
2025-04-17 12:53:36,855 - INFO - 2024-12-03 00:00:00-05:00: Portfolio Value: 9977.49
2025-04-17 12:53:36,857 - INFO - 2024-12-04 00:00:00-05:00: Portfolio Value: 9971.52
2025-04-17 12:53:36,857 - INFO - 2024-12-05 00:00:00-05:00: Portfolio Value: 9969.91
2025-04-17 12:53:36,857 - INFO - 2024-12-06 00:00:00-05:00: Portfolio Value: 9971.61
2025-04-17 12:53:36,857 - INFO - 2024-12-09 00:00:00-05:00: Portfolio Value: 9976.28
2025-04-17 12:53:36,857 - INFO - 2024-12-10 00:00:00-05:00: Portfolio Value: 9975.88
2025-04-17 12:53:36,857 - INFO - 2024-12-11 00:00:00-05:00: Portfolio Value: 9973.93
2025-04-17 12:53:36,857 - INFO - 2024-12-12 00:00:00-05:00: Portfolio Value: 9980.00
2025-04-17 12:53:36,857 - INFO - 2024-12-13 00:00:00-05:00: Portfolio Value: 9978.79
2025-04-17 12:53:36,857 - INFO - 2024-12-16 00:00:00-05:00: Portfolio Value: 9978.60
2025-04-17 12:53:36,857 - INFO - 2024-12-17 00:00:00-05:00: Portfolio Value: 9980.76
2025-04-17 12:53:36,857 - INFO - 2024-12-18 00:00:00-05:00: Portfolio Value: 9983.52
2025-04-17 12:53:36,857 - INFO - 2024-12-19 00:00:00-05:00: Portfolio Value: 9979.73
```

```
2025-04-17 12:53:36,857 - INFO - 2024-12-20 00:00:00-05:00: Portfolio Value: 9978.84
2025-04-17 12:53:36,857 - INFO - 2024-12-23 00:00:00-05:00: Portfolio Value: 9977.32
2025-04-17 12:53:36,865 - INFO - 2024-12-24 00:00:00-05:00: Portfolio Value: 9976.16
2025-04-17 12:53:36,865 - INFO - 2024-12-26 00:00:00-05:00: Portfolio Value: 9975.94
2025-04-17 12:53:36,865 - INFO - 2024-12-27 00:00:00-05:00: Portfolio Value: 9977.98
2025-04-17 12:53:36,865 - INFO - 2024-12-30 00:00:00-05:00: Portfolio Value: 9978.22
2025-04-17 12:53:36,865 - INFO - 2024-12-31 00:00:00-05:00: Portfolio Value: 9979.12
2025-04-17 12:53:36,865 - INFO - 2025-01-02 00:00:00-05:00: Portfolio Value: 9981.14
2025-04-17 12:53:36,865 - INFO - 2025-01-03 00:00:00-05:00: Portfolio Value: 9981.10
2025-04-17 12:53:36,865 - INFO - 2025-01-06 00:00:00-05:00: Portfolio Value: 9977.16
2025-04-17 12:53:36,865 - INFO - 2025-01-07 00:00:00-05:00: Portfolio Value: 9974.94
2025-04-17 12:53:36,865 - INFO - 2025-01-08 00:00:00-05:00: Portfolio Value: 9976.19
2025-04-17 12:53:36,865 - INFO - 2025-01-10 00:00:00-05:00: Portfolio Value: 9963.89
2025-04-17 12:53:36,865 - INFO - 2025-01-13 00:00:00-05:00: Portfolio Value: 9971.15
2025-04-17 12:53:36,873 - INFO - 2025-01-14 00:00:00-05:00: Portfolio Value: 9968.75
2025-04-17 12:53:36,873 - INFO - 2025-01-15 00:00:00-05:00: Portfolio Value: 9970.31
2025-04-17 12:53:36,873 - INFO - 2025-01-16 00:00:00-05:00: Portfolio Value: 9969.71
2025-04-17 12:53:36,873 - INFO - 2025-01-17 00:00:00-05:00: Portfolio Value: 9972.15
2025-04-17 12:53:36,873 - INFO - 2025-01-21 00:00:00-05:00: Portfolio Value: 9963.75
2025-04-17 12:53:36,873 - INFO - 2025-01-22 00:00:00-05:00: Portfolio Value: 9963.52
2025-04-17 12:53:36,873 - INFO - 2025-01-23 00:00:00-05:00: Portfolio Value: 9963.76
2025-04-17 12:53:36,873 - INFO - 2025-01-24 00:00:00-05:00: Portfolio Value: 9965.73
2025-04-17 12:53:36,873 - INFO - 2025-01-27 00:00:00-05:00: Portfolio Value: 9969.77
2025-04-17 12:53:36,873 - INFO - 2025-01-28 00:00:00-05:00: Portfolio Value: 9965.63
2025-04-17 12:53:36,873 - INFO - 2025-01-29 00:00:00-05:00: Portfolio Value: 9964.70
2025-04-17 12:53:36,873 - INFO - 2025-01-30 00:00:00-05:00: Portfolio Value: 9967.91
2025-04-17 12:53:36,873 - INFO - 2025-01-31 00:00:00-05:00: Portfolio Value: 9970.11
2025-04-17 12:53:36,873 - INFO - 2025-02-03 00:00:00-05:00: Portfolio Value: 9972.08
2025-04-17 12:53:36,873 - INFO - 2025-02-04 00:00:00-05:00: Portfolio Value: 9962.34
2025-04-17 12:53:36,882 - INFO - 2025-02-05 00:00:00-05:00: Portfolio Value: 9963.68
2025-04-17 12:53:36,882 - INFO - 2025-02-06 00:00:00-05:00: Portfolio Value: 9967.80
2025-04-17 12:53:36,882 - INFO - 2025-02-07 00:00:00-05:00: Portfolio Value: 9968.57
2025-04-17 12:53:36,882 - INFO - 2025-02-10 00:00:00-05:00: Portfolio Value: 9976.52
2025-04-17 12:53:36,882 - INFO - 2025-02-11 00:00:00-05:00: Portfolio Value: 9989.92
2025-04-17 12:53:36,882 - INFO - 2025-02-12 00:00:00-05:00: Portfolio Value: 9988.49
2025-04-17 12:53:36,882 - INFO - 2025-02-13 00:00:00-05:00: Portfolio Value: 9997.64
2025-04-17 12:53:36,882 - INFO - 2025-02-14 00:00:00-05:00: Portfolio Value: 9991.78
2025-04-17 12:53:36,882 - INFO - 2025-02-18 00:00:00-05:00: Portfolio Value: 9995.46
2025-04-17 12:53:36,882 - INFO - 2025-02-19 00:00:00-05:00: Portfolio Value: 9996.40
2025-04-17 12:53:36,882 - INFO - 2025-02-20 00:00:00-05:00: Portfolio Value: 10000.18
2025-04-17 12:53:36,890 - INFO - 2025-02-21 00:00:00-05:00: Portfolio Value: 10015.30
2025-04-17 12:53:36,890 - INFO - 2025-02-24 00:00:00-05:00: Portfolio Value: 10010.82
2025-04-17 12:53:36,890 - INFO - 2025-02-25 00:00:00-05:00: Portfolio Value: 10012.01
2025-04-17 12:53:36,890 - INFO - 2025-02-26 00:00:00-05:00: Portfolio Value: 10010.82
```

```
2025-04-17 12:53:36,890 - INFO - 2025-02-27 00:00:00-05:00: Portfolio Value: 10014.56
2025-04-17 12:53:36,890 - INFO - 2025-02-28 00:00:00-05:00: Portfolio Value: 10014.49
2025-04-17 12:53:36,890 - INFO - 2025-03-03 00:00:00-05:00: Portfolio Value: 10020.03
2025-04-17 12:53:36,890 - INFO - 2025-03-04 00:00:00-05:00: Portfolio Value: 10023.33
2025-04-17 12:53:36,890 - INFO - 2025-03-05 00:00:00-05:00: Portfolio Value: 10018.95
2025-04-17 12:53:36,890 - INFO - 2025-03-06 00:00:00-05:00: Portfolio Value: 10023.32
2025-04-17 12:53:36,890 - INFO - 2025-03-07 00:00:00-05:00: Portfolio Value: 10028.61
2025-04-17 12:53:36,890 - INFO - 2025-03-10 00:00:00-04:00: Portfolio Value: 10034.55
2025-04-17 12:53:36,890 - INFO - 2025-03-11 00:00:00-04:00: Portfolio Value: 10041.03
2025-04-17 12:53:36,890 - INFO - 2025-03-12 00:00:00-04:00: Portfolio Value: 10041.61
2025-04-17 12:53:36,890 - INFO - 2025-03-13 00:00:00-04:00: Portfolio Value: 10043.73
2025-04-17 12:53:36,890 - INFO - 2025-03-14 00:00:00-04:00: Portfolio Value: 10038.17
2025-04-17 12:53:36,898 - INFO - 2025-03-17 00:00:00-04:00: Portfolio Value: 10040.63
2025-04-17 12:53:36,898 - INFO - 2025-03-18 00:00:00-04:00: Portfolio Value: 10041.32
2025-04-17 12:53:36,898 - INFO - 2025-03-19 00:00:00-04:00: Portfolio Value: 10035.53
2025-04-17 12:53:36,898 - INFO - 2025-03-20 00:00:00-04:00: Portfolio Value: 10039.22
2025-04-17 12:53:36,898 - INFO - 2025-03-21 00:00:00-04:00: Portfolio Value: 10036.64
2025-04-17 12:53:36,898 - INFO - 2025-03-24 00:00:00-04:00: Portfolio Value: 10031.15
2025-04-17 12:53:36,898 - INFO - 2025-03-25 00:00:00-04:00: Portfolio Value: 10031.36
2025-04-17 12:53:36,898 - INFO - 2025-03-26 00:00:00-04:00: Portfolio Value: 10036.79
2025-04-17 12:53:36,898 - INFO - 2025-03-27 00:00:00-04:00: Portfolio Value: 10044.71
2025-04-17 12:53:36,906 - INFO - 2025-03-28 00:00:00-04:00: Portfolio Value: 10050.53
2025-04-17 12:53:36,906 - INFO - 2025-03-31 00:00:00-04:00: Portfolio Value: 10055.00
2025-04-17 12:53:36,906 - INFO - 2025-04-01 00:00:00-04:00: Portfolio Value: 10059.29
2025-04-17 12:53:36,906 - INFO - 2025-04-02 00:00:00-04:00: Portfolio Value: 10054.82
2025-04-17 12:53:36,906 - INFO - 2025-04-03 00:00:00-04:00: Portfolio Value: 10074.35
2025-04-17 12:53:36,906 - INFO - 2025-04-04 00:00:00-04:00: Portfolio Value: 10070.77
2025-04-17 12:53:36,906 - INFO - 2025-04-07 00:00:00-04:00: Portfolio Value: 10067.57
2025-04-17 12:53:36,906 - INFO - 2025-04-08 00:00:00-04:00: Portfolio Value: 10070.49
2025-04-17 12:53:36,906 - INFO - 2025-04-09 00:00:00-04:00: Portfolio Value: 10056.77
2025-04-17 12:53:36,906 - INFO - 2025-04-10 00:00:00-04:00: Portfolio Value: 10073.93
2025-04-17 12:53:36,906 - INFO - 2025-04-11 00:00:00-04:00: Portfolio Value: 10071.38
2025-04-17 12:53:36,906 - INFO - 2025-04-14 00:00:00-04:00: Portfolio Value: 10076.18
2025-04-17 12:53:36,906 - INFO - 2025-04-15 00:00:00-04:00: Portfolio Value: 10072.82
2025-04-17 12:53:36,906 - INFO - 2025-04-16 00:00:00-04:00: Portfolio Value: 10073.00
2025-04-17 12:53:36,906 - INFO - Paper trading completed.
2025-04-17 12:53:36,906 - INFO - Final Portfolio Value: 10073.00
2025-04-17 12:53:36,906 - INFO - 2025-01-10 00:00:00-05:00: Sell DAL, Buy KO
2025-04-17 12:53:36,906 - INFO - 2025-02-21 00:00:00-05:00: Buy DAL, Sell KO
2025-04-17 12:53:36,914 - INFO - 2025-03-14 00:00:00-04:00: Sell DAL, Buy KO
2025-04-17 12:53:36,914 - INFO - 2025-04-03 00:00:00-04:00: Buy DAL, Sell KO
2025-04-17 12:53:36,914 - INFO - 2025-04-09 00:00:00-04:00: Sell DAL, Buy KO
```

Code

```
import yfinance as yf
import pandas as pd
import numpy as np
import statsmodels.api as sm
from statsmodels.tsa.stattools import coint
import matplotlib.pyplot as plt
import time
import logging
from datetime import datetime, timedelta
import threading
import queue

logging.basicConfig(level=logging.INFO, format='%(asctime)s - %(levelname)s -
%(message)s')
logger = logging.getLogger(__name__)

API_RATE_LIMIT = 5
Z_SCORE_THRESHOLD_BUY = -2.0
Z_SCORE_THRESHOLD_SELL = 2.0
LOOKBACK_PERIOD = 20

class PairsTradingBot:
    def __init__(self, pair_symbols, start_date, end_date):
        self.pair_symbols = pair_symbols
        self.start_date = start_date
        self.end_date = end_date
        self.data = None
        self.log_returns = None
        self.spread = None
        self.z_scores = None
        self.positions = pd.DataFrame()
        self.portfolio_value = 10000
        self.trade_log = []
        self.request_queue = queue.Queue()
        self.lock = threading.Lock()
```

```
def fetch_data(self):
    """Fetch historical data from Yahoo Finance with rate limiting."""
    logger.info(f"Fetching data for {self.pair_symbols}")
    for symbol in self.pair_symbols:
        self.request_queue.put(symbol)

    def process_queue():
        while not self.request_queue.empty():
            with self.lock:
                symbol = self.request_queue.get()
                ticker = yf.Ticker(symbol)
                df = ticker.history(start=self.start_date, end=self.end_date)
                if self.data is None:
                    self.data = pd.DataFrame(index=df.index)
                self.data[symbol] = df['Close']
                logger.info(f"Data fetched for {symbol}")
                time.sleep(60 / API_RATE_LIMIT)

    thread = threading.Thread(target=process_queue)
    thread.start()
    thread.join()

    if self.data is None or self.data.empty:
        raise ValueError("No data fetched. Check symbols or API connectivity.")
    self.data.dropna(inplace=True)

    self.log_returns = np.log(self.data / self.data.shift(1))
    self.log_returns.dropna(inplace=True)

def test_cointegration(self):
    """Perform Engle-Granger cointegration test on log returns."""
    logger.info("Testing cointegration...")
    stock1, stock2 = self.pair_symbols
    score, p_value, _ = coint(self.log_returns[stock1], self.log_returns[stock2])
    logger.info(f"Cointegration p-value: {p_value}")
    return p_value < 0.05
    def calculate_spread_and_zscore(self):
        """Calculate spread and Z-scores using log returns."""
        stock1, stock2 = self.pair_symbols
        model = sm.OLS(self.log_returns[stock1],
sm.add_constant(self.log_returns[stock2])).fit()
        hedge_ratio = model.params[stock2]


        self.spread = self.log_returns[stock1] - hedge_ratio * self.log_returns[stock2]

        spread_mean = self.spread.rolling(window=LOOKBACK_PERIOD).mean()
        spread_std = self.spread.rolling(window=LOOKBACK_PERIOD).std()
        self.z_scores = (self.spread - spread_mean) / spread_std
        self.z_scores.dropna(inplace=True)

    def generate_signals(self):
        """Generate trading signals based on Z-scores."""
        signals = pd.DataFrame(index=self.z_scores.index)
        signals['Z-Score'] = self.z_scores
        signals['Signal'] = 0

        signals.loc[signals['Z-Score'] < Z_SCORE_THRESHOLD_BUY, 'Signal'] = 1
        signals.loc[signals['Z-Score'] > Z_SCORE_THRESHOLD_SELL, 'Signal'] = -1

        return signals

    def backtest(self):
        """Backtest the strategy."""
        logger.info("Starting backtest...")
        signals = self.generate_signals()
        self.positions = pd.DataFrame(index=signals.index, columns=self.pair_symbols)

        for i in range(1, len(signals)):
            if signals['Signal'].iloc[i] == 1:
                self.positions.iloc[i, 0] = 1
                self.positions.iloc[i, 1] = -1
            elif signals['Signal'].iloc[i] == -1:
                self.positions.iloc[i, 0] = -1
                self.positions.iloc[i, 1] = 1
            else:
                self.positions.iloc[i] = [0, 0]

        returns = self.positions.shift(1) * self.data.pct_change()
        portfolio_returns = returns.sum(axis=1)
        cumulative_returns = (1 + portfolio_returns).cumprod() * self.portfolio_value

        roi = (cumulative_returns[-1] / self.portfolio_value - 1) * 100
        sharpe_ratio = np.sqrt(252) * portfolio_returns.mean() / portfolio_returns.std()
        drawdown = (cumulative_returns.cummax() - cumulative_returns) /
cumulative_returns.cummax()
        max_drawdown = drawdown.max() * 100

        logger.info(f"ROI: {roi:.2f}%")
        logger.info(f"Sharpe Ratio: {sharpe_ratio:.2f}")
        logger.info(f"Max Drawdown: {max_drawdown:.2f}%")

        return cumulative_returns, roi, sharpe_ratio, max_drawdown

    def paper_trade(self):
        """Simulate paper trading in real-time (mocked here with historical data)."""
        logger.info("Starting paper trading simulation...")
        signals = self.generate_signals()
        cash = self.portfolio_value
        holdings = {symbol: 0 for symbol in self.pair_symbols}

        for date, row in signals.iterrows():
            stock1, stock2 = self.pair_symbols
            price1, price2 = self.data[stock1].loc[date], self.data[stock2].loc[date]

            if row['Signal'] == 1:
                if cash >= price1:
                    holdings[stock1] += 1
                    holdings[stock2] -= 1
                    cash -= price1
                    cash += price2
                    self.trade_log.append(f"{date}: Buy {stock1}, Sell {stock2}")
            elif row['Signal'] == -1:
                if cash >= price2:
                    holdings[stock1] -= 1
                    holdings[stock2] += 1
                    cash += price1
                    cash -= price2
                    self.trade_log.append(f"{date}: Sell {stock1}, Buy {stock2}")

            portfolio_value = cash + holdings[stock1] * price1 + holdings[stock2] * price2
            logger.info(f"{date}: Portfolio Value: {portfolio_value:.2f}")

        logger.info("Paper trading completed.")
        logger.info(f"Final Portfolio Value: {portfolio_value:.2f}")
        for log in self.trade_log[-5:]:
            logger.info(log)

    def plot_results(self, cumulative_returns):
        """Plot Z-scores and portfolio performance."""
        plt.figure(figsize=(12, 8))

        plt.subplot(2, 1, 1)
        plt.plot(self.z_scores, label='Z-Score')
        plt.axhline(Z_SCORE_THRESHOLD_BUY, color='g', linestyle='--')
        plt.axhline(Z_SCORE_THRESHOLD_SELL, color='r', linestyle='--')
        plt.legend()
        plt.title('Z-Score Over Time')

        plt.subplot(2, 1, 2)
        plt.plot(cumulative_returns, label='Portfolio Value')
        plt.legend()
        plt.title('Portfolio Performance')

        plt.tight_layout()
        plt.show()
```

```
if __name__ == "__main__":

    print("Enter two stock tickers for the pairs trading strategy.")
    ticker1 = input("Enter first ticker (e.g., AAPL): ").strip().upper()
    ticker2 = input("Enter second ticker (e.g., MSFT): ").strip().upper()

    if ticker1 == ticker2:
        raise ValueError("The two tickers must be different.")

    pair_symbols = [ticker1, ticker2]

    end_date = datetime.now()
    start_date = end_date - timedelta(days=504)

    bot = PairsTradingBot(pair_symbols, start_date, end_date)
    bot.fetch_data()
    if bot.test_cointegration():
        logger.info("Pair is cointegrated. Proceeding with strategy.")
        bot.calculate_spread_and_zscore()
        cumulative_returns, roi, sharpe, drawdown = bot.backtest()
        bot.paper_trade()
        bot.plot_results(cumulative_returns)
    else:
        logger.warning("Pair is not cointegrated. Strategy aborted.")
```