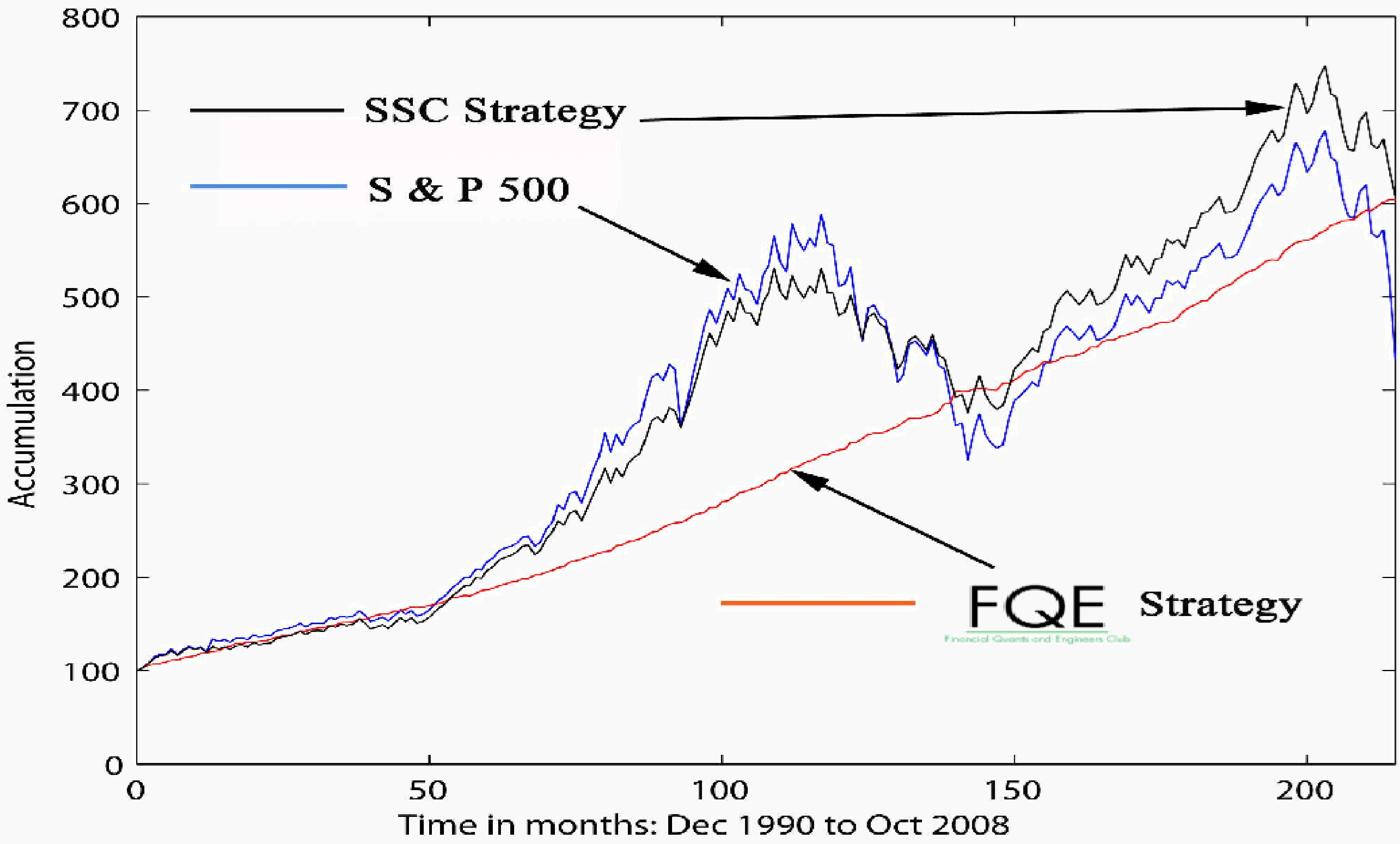
Optimizing Sparse Mean-Reverting Portfolio

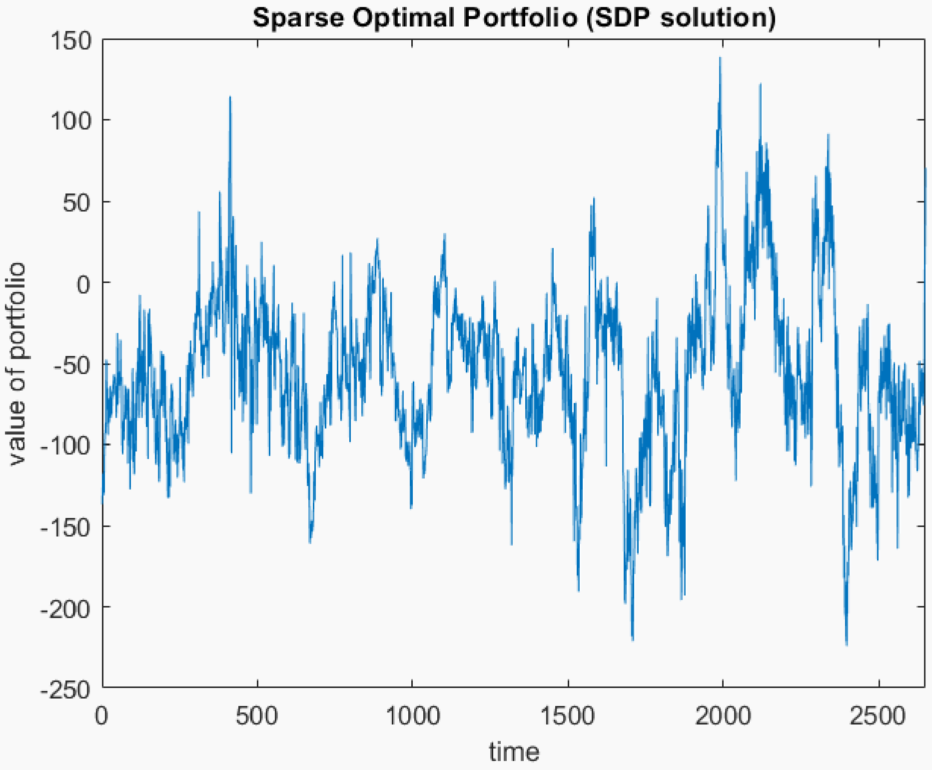
By: Liam, Moussa, Joseph and Daniel

“The sp500 is my favorite stock exchange”



Project Outline

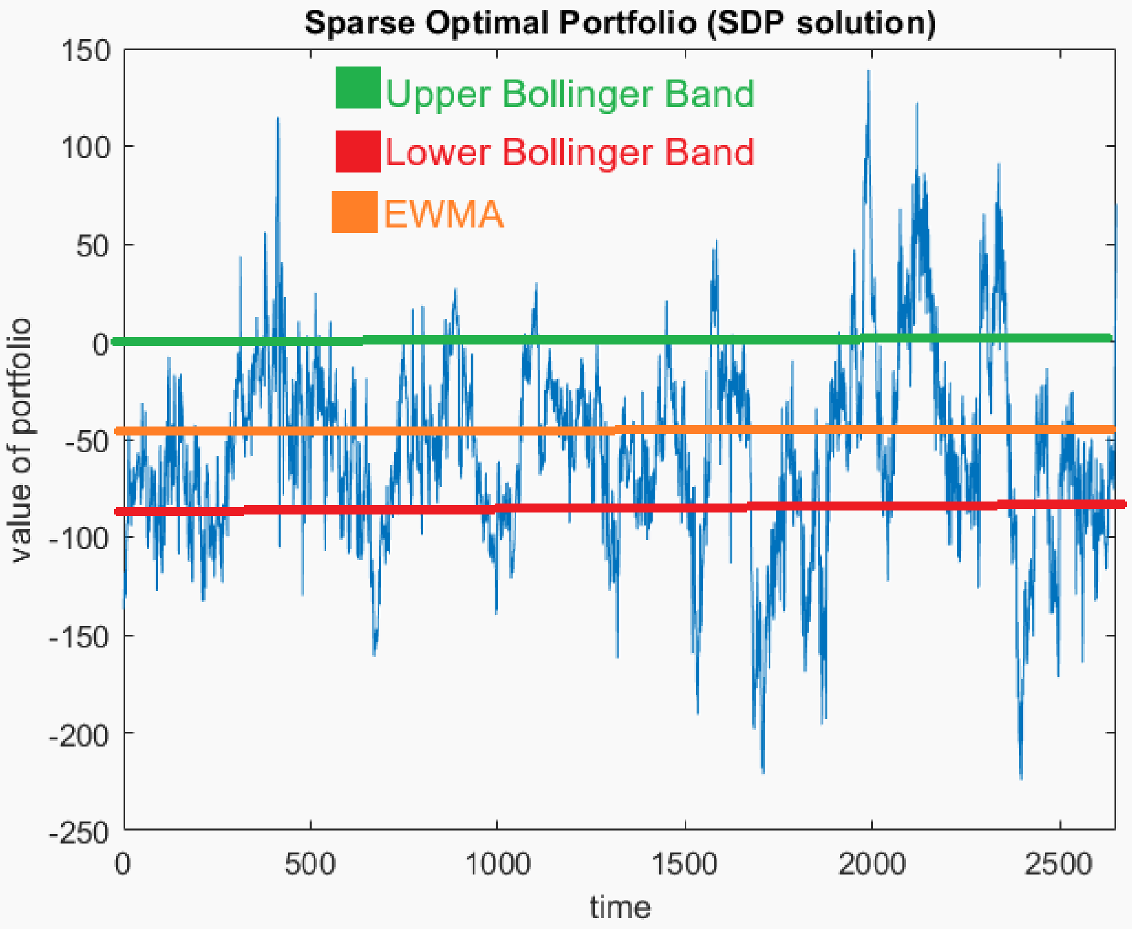
"Optimizing Sparse Mean-Reverting Portfolio" by Sung Min Yoon

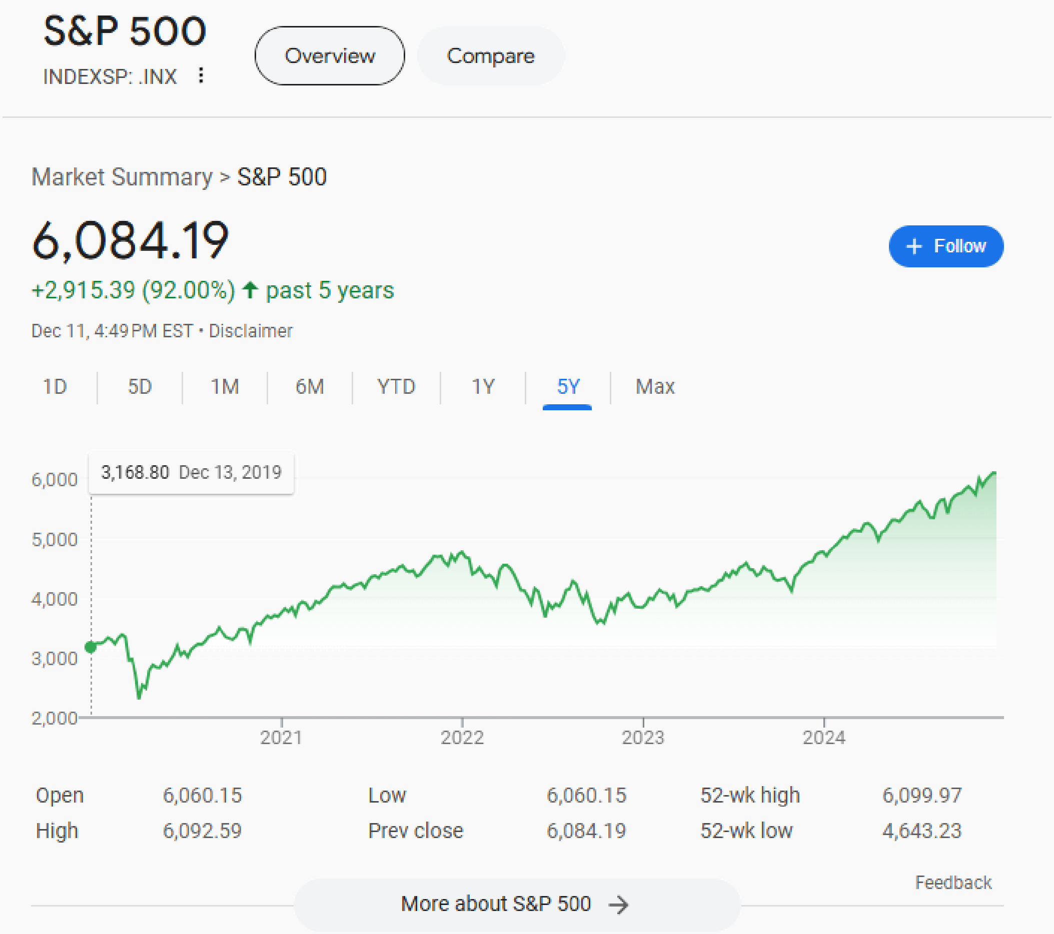


<u>Goals</u>

- Build mean-reverting portfolios with fastest mean-reversion behavior.
- Incorporate constraints: minimum variance and sparsity.
- Use Semidefinite Programming (SDP) for optimization.
- Back-test the results with real financial data.

Brief Overview





The Math

Conversion from convex problem to linear

Mean-Reverting Portfolio and Predictability

1. **Predictability Definition** (Box & Tiao, 1977):

$$\lambda(y) = \frac{y^T A_1 A_0^{-1} A_1^T y}{y^T A_0 y}$$

- A_0: Lag-0 autocovariance matrix.
- A_1: Lag-1 autocovariance matrix.
- y: Portfolio weights vector.

Objective Function:

In the original portfolio optimization, the objective involves minimizing the quadratic form:

$$y^T M y$$

where M is a constant matrix related to predictability.

By introducing a matrix $Y = yy^T$, the quadratic form can be rewritten using the trace:

$$y^T M y = \mathrm{Tr}(MY)$$

Now the objective becomes **linear** with respect to Y, which makes the problem solvable using semidefinite programming (SDP).

$$y = \begin{bmatrix} 0.3\,(\mathrm{AAPL}) \\ 0.5\,(\mathrm{NVDA}) \\ 0.2\,(\mathrm{MSFT}) \end{bmatrix}$$

This weight vector is a sample (not optimized) allocation, where:

- 30% of the portfolio is allocated to AAPL,
- 50% to NVDA,
- 20% to MSFT.

Lag-0 Autocovariance Matrix A_0

The **Lag-0 covariance matrix** captures the variance and covariance between the returns of assets at the same time t:

$$A_0 = \begin{bmatrix} \mathrm{Var}(X_1, X_1) & \mathrm{Cov}(X_1, X_2) & \mathrm{Cov}(X_1, X_3) \\ \mathrm{Cov}(X_2, X_1) & \mathrm{Var}(X_2, X_2) & \mathrm{Cov}(X_2, X_3) \\ \mathrm{Cov}(X_3, X_1) & \mathrm{Cov}(X_3, X_2) & \mathrm{Var}(X_3, X_3) \end{bmatrix}$$

- X_1: Returns of AAPL
- X_2: Returns of NVDA
- X_3: Returns of MSFT

Explanation:

- Diagonal elements represent the **variance** of each asset (e.g., $\mathrm{Var}(X_1, X_1)$ = variance of AAPL).
- Off-diagonal elements represent the **covariance** between pairs of assets (e.g., $\mathrm{Cov}(X_1, X_2)$ = covariance between AAPL and NVDA).

$$x_t = \begin{bmatrix} r_{\mathrm{AAPL},t} \\ r_{\mathrm{NVDA},t} \\ r_{\mathrm{MSFT},t} \end{bmatrix}$$

$$\begin{aligned} \min_{Y}\ & Tr(MY) + \rho\|Y\|_1 \\ \text{s.t.}\ & Tr(A_0 Y) \geq \nu \\ & Tr(Y) = 1 \\ & Y \succeq 0 \end{aligned}$$

$M = A_1 A_0^{-1} A_1^T$ and ν is the minimum variance we want to achieve

Lag-1 Autocovariance Matrix A_1

The **Lag-1 covariance matrix** captures the covariance between returns at time t and time $t-1$:

$$A_1 = \begin{bmatrix} \mathrm{Cov}(X_1^{t-1}, X_1^t) & \mathrm{Cov}(X_1^{t-1}, X_2^t) & \mathrm{Cov}(X_1^{t-1}, X_3^t) \\ \mathrm{Cov}(X_2^{t-1}, X_1^t) & \mathrm{Cov}(X_2^{t-1}, X_2^t) & \mathrm{Cov}(X_2^{t-1}, X_3^t) \\ \mathrm{Cov}(X_3^{t-1}, X_1^t) & \mathrm{Cov}(X_3^{t-1}, X_2^t) & \mathrm{Cov}(X_3^{t-1}, X_3^t) \end{bmatrix}$$

- X_1^{t-1}: Returns of AAPL at time $t-1$.
- X_1^t: Returns of AAPL at time t.

Explanation:

- Lag-1 measures how returns from the previous period $(t-1)$ relate to returns in the current period (t).
- Off-diagonal elements like $\mathrm{Cov}(X_1^{t-1}, X_2^t)$ show how the past returns of one asset (AAPL) affect the current returns of another asset (NVDA).

Back-test - How It Works (1/3):

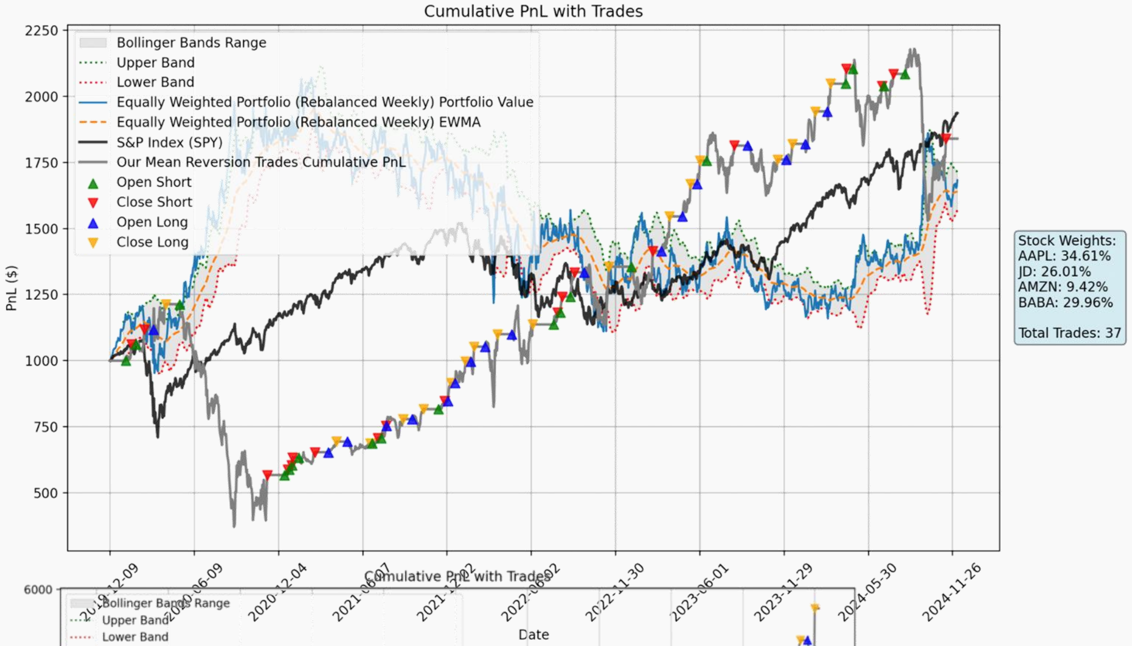
- **Step 1: Data Preparation**
  - Input Stock Selection: Choose specific stocks (e.g., AAPL, AMZN, etc.).
  - Fetch Data: Pull 5 years of historical stock prices for the selected stocks from the S&P 500 using the write_sp500_data function which calls the Robinhood API.
  - Load Data: Read the stock data into a Pandas DataFrame for analysis.
- **Step 2: Portfolio Initialization**
  - Optimal Weights: Use the get_optimal_weights function to calculate weights for each stock in the portfolio.
  - Investment Allocation: Assign the initial investment amount across stocks based on calculated weights.
- **Step 3: Portfolio Value Calculation**
  - With Rebalancing: Reallocate portfolio weights periodically, in our case weekly, to maintain our calculated target weights.

Back-test - How It Works (2/3):

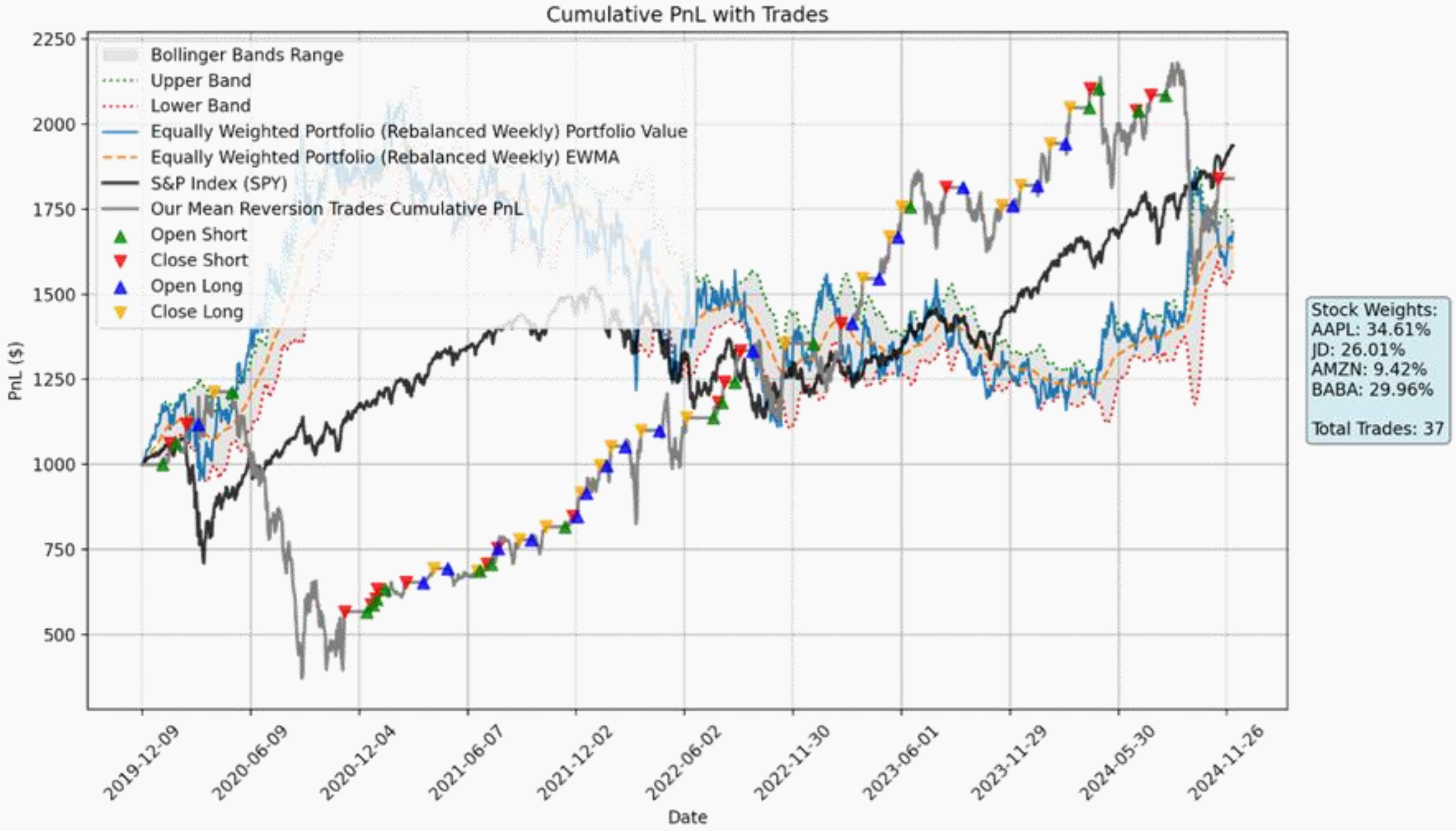
- **Step 4: EWMA and Bollinger Bands**
  - EWMA (Exponentially Weighted Moving Average): Tracks the portfolio's average value over time. We used the 20 day EWMA.
  - Bollinger Bands: Identify overbought/oversold conditions using upper and lower bands based on price volatility. The bands are set to 2 standard deviations from the EWMA.
- **Step 5: Trade Tracking**
  - Entry/Exit Signals:
  - Long Entry: Buy when the portfolio value drops below the lower Bollinger Band.
  - Short Entry: Sell short when the value rises above the upper Bollinger Band.
  - Exit Positions: Exit trades when the value crosses back over the EWMA.

Back-test - How It Works (3/3):

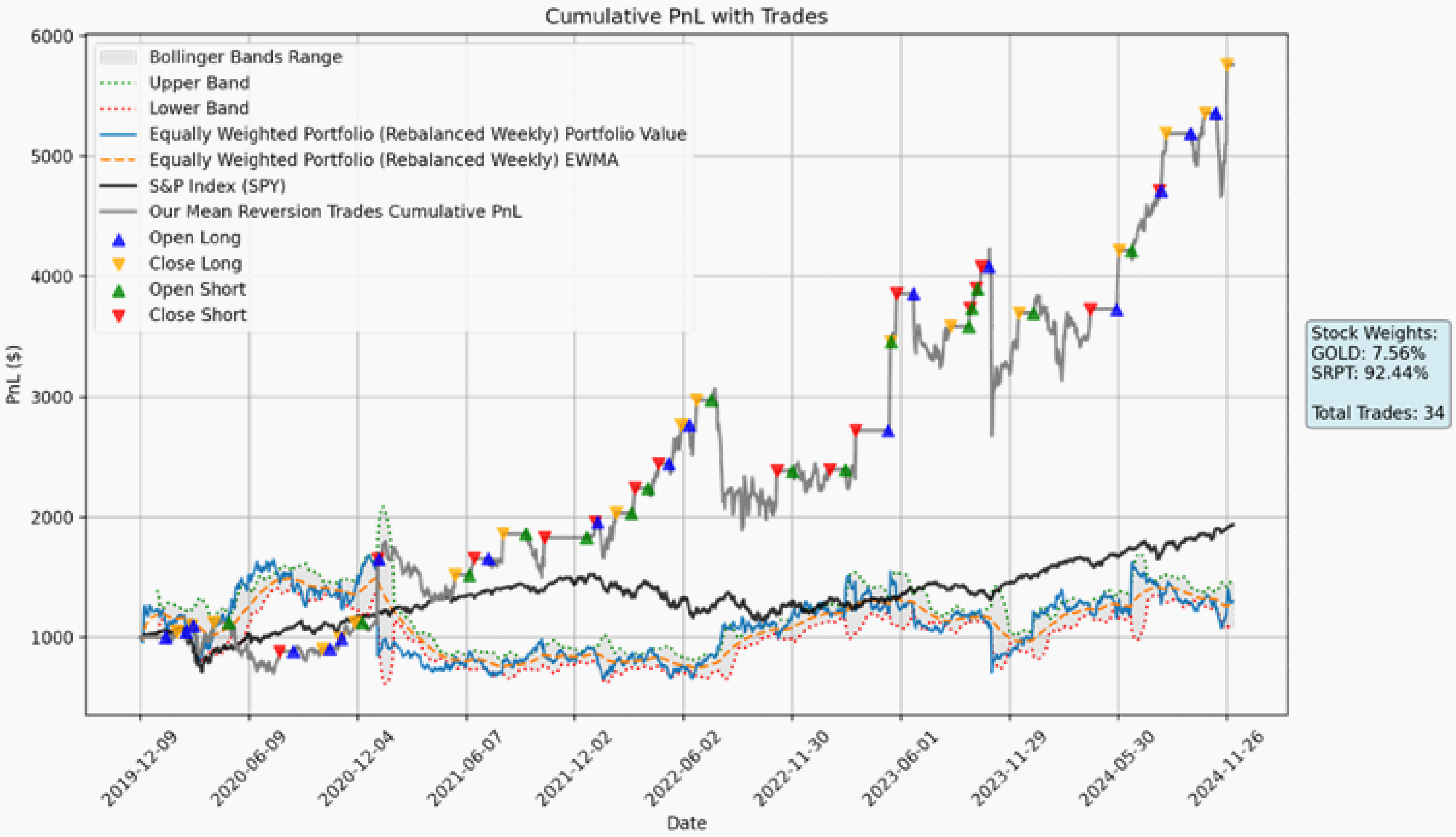
- **Step 6: Visual Analysis**
  - Performance Visualization:
  - Plot portfolio value, EWMA, and Bollinger Bands.
  - Mark trades on the graph (e.g., "Open Long," "Close Long") and track cumulative profit/loss.
  - Comparison to Benchmark (SPY): Add S&P 500 ETF data to evaluate relative performance.



Back-test – Illustration (1/2):



Back-test – Illustration (2/2):



Potential Future Improvements

FQE Financial Quants and Engineers Club

- **Higher Frequency Data**
  - Use hourly or minute-level data for intraday trading.
  - This can capture faster mean-reverting opportunities.
- **Better Rebalancing Period**
  - Daily, Weekly, or Custom Adaptive periods.
- **Better Stock Selection Using PCA**
  - Reduces dimensionality by choosing the most influential stocks.
  - Captures key market movements with fewer assets.
- **Dynamic Regularization**
  - Optimize ρ (sparsity) and ν (minimum variance) dynamically over time.