

Pairs Trading Strategy: A Cointegration-Based Approach

IEOR 198 Final Project

Abstract

This project implements, documents, and evaluates a market-neutral pairs trading strategy based on statistical cointegration, with all steps explicitly grounded in a reproducible Python codebase. Using daily adjusted close prices for a curated subset of SP 500 equities, we identify statistically cointegrated stock pairs via the Engle–Granger two-step methodology implemented through the `statsmodels.tsa.stattools.coint` function. Trading signals are generated from deviations of an estimated spread from its rolling historical mean, measured using z-scores computed directly from the constructed spread series.

The strategy is evaluated through a full backtesting framework that mirrors realistic trading assumptions, including transaction costs, delayed signal execution, equal-dollar long/short exposure, and a strict train–test split to prevent look-ahead bias. Performance is analyzed both in-sample and out-of-sample using cumulative returns, Sharpe ratio, volatility, drawdowns, and win rates computed programmatically. While the strategy demonstrates moderate profitability during the training period, out-of-sample results reveal a significant degradation in performance, underscoring the instability of cointegration relationships over time and highlighting key challenges in deploying statistical arbitrage strategies in practice.

1 Introduction

Pairs trading is a classic relative-value strategy that seeks to exploit temporary deviations from a long-run equilibrium relationship between two assets. Unlike directional trading strategies, pairs trading is designed to be market-neutral by simultaneously taking long and short positions in related securities. The theoretical motivation relies on the concept of cointegration: while individual asset prices may be non-stationary, a linear combination of two cointegrated assets can be stationary and mean-reverting.

In this project, we implement a cointegration-based pairs trading strategy using U.S. equities. The goal is to evaluate whether statistically identified cointegrated pairs can generate profitable mean-reversion signals when subjected to realistic backtesting constraints. Particular emphasis is placed on avoiding common pitfalls such as look-ahead bias and overfitting by performing pair selection exclusively on training data and evaluating performance out-of-sample.

2 Dataset

We use daily adjusted close prices for 40 SP 500 stocks spanning four sectors: Technology, Financials, Consumer Discretionary, and Energy. Sector-level filtering is applied to ensure economic plausibility of candidate pairs. The data is obtained via the `yfinance` API and covers the period from January 3, 2022 to November 29, 2024, yielding 732 trading days.

Missing values are handled via forward- and backward-filling, and assets with more than 5
The dataset is split into an 80

3 Methods

3.1 Cointegration-Based Pair Selection

To identify candidate pairs, we apply the Engle–Granger two-step cointegration test to all possible stock pairs within the same sector. For each pair (X_t, Y_t) , we estimate the regression

$$Y_t = \alpha + \beta X_t + \varepsilon_t, \quad (1)$$

where ε_t represents the residual spread. We then test the residuals for stationarity using the Augmented Dickey–Fuller test. Pairs with cointegration test p-values below 0.05 are retained.

Using this procedure, 21 statistically significant cointegrated pairs are identified in the training sample. The most significant pair is **AMD–CRM** from the Technology sector, with a p-value of 0.00032. This pair is selected for detailed strategy implementation and backtesting.

3.2 Spread Modeling and Signal Generation

For the selected pair, the hedge ratio β is estimated via ordinary least squares on the training data. The resulting spread is defined as

$$S_t = P_t^{(1)} - \beta P_t^{(2)} - \alpha. \quad (2)$$

To generate trading signals, we compute a rolling z-score of the spread using a 20-day lookback window:

$$Z_t = \frac{S_t - \mu_t}{\sigma_t}, \quad (3)$$

where μ_t and σ_t are the rolling mean and standard deviation of the spread. Trading rules are as follows:

- Enter a long-spread position when $Z_t < -2$
- Enter a short-spread position when $Z_t > 2$
- Exit the position when Z_t crosses zero

3.3 Backtesting Framework

The strategy is backtested under the assumption of equal dollar allocation to each leg, resulting in approximately zero net market exposure. Positions are updated daily based on the previous day’s signal to avoid look-ahead bias. Transaction costs are set to 0.02% per trade per leg, consistent with course guidelines.

Strategy returns are computed as the signal-weighted difference in log returns between the two stocks, with transaction costs deducted whenever the position changes. Performance metrics are evaluated separately for the training and testing periods.

4 Results

In-sample performance for the AMD/CRM pair shows a total return of 20.6% with an annualized Sharpe ratio of 0.28. While profitable, the strategy experiences substantial volatility and a maximum drawdown exceeding 28%.

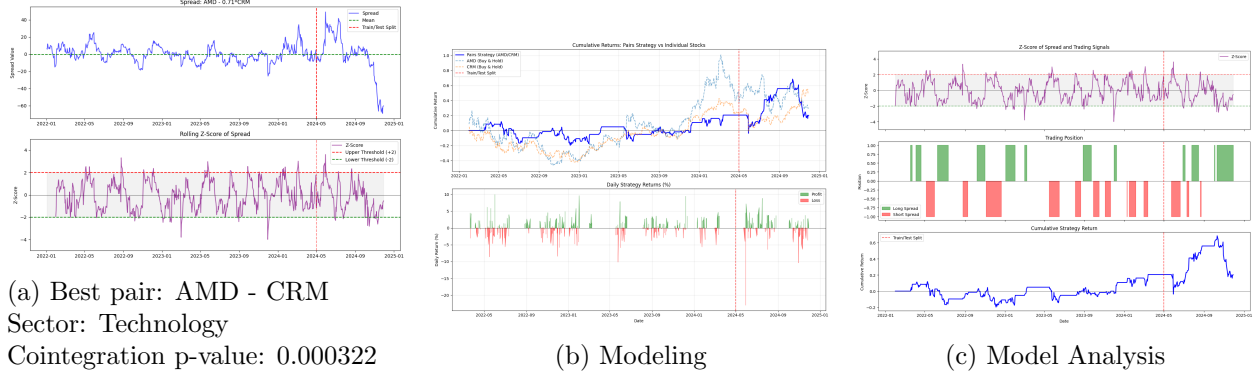


Figure 1: Visual summary of pair selection, spread modeling, and analysis.

Out-of-sample performance deteriorates significantly. During the testing period, the strategy produces a total return of -0.28% and a Sharpe ratio of -0.01. Volatility increases and drawdowns deepen, indicating that the historical cointegration relationship weakens or breaks during the test period.

Across the full sample, the strategy yields a total return of 20.2% with a Sharpe ratio of 0.19. Approximately 49 trades (entries and exits) are executed, with an average holding period of 29 days. The win rate remains close to 50%, consistent with a mean-reversion strategy but insufficient to overcome volatility and transaction costs out-of-sample.

As seen in Figure 1, the selected AMD-CRM pair exhibits clear mean-reversion in the spread, though volatility remains significant. The modeling and analysis plots highlight both the effectiveness of the z-score trading signals and the limitations of relying on historical cointegration relationships.

5 Conclusion

This project demonstrates both the appeal and limitations of cointegration-based pairs trading. While statistically significant cointegrated pairs can be identified and exploited in-sample, out-of-sample performance remains fragile. Market regimes, structural breaks, and parameter instability pose significant challenges to real-world implementation.

Future improvements could include dynamic hedge ratio estimation, regime detection, portfolio-level diversification across multiple pairs, and more robust statistical filtering. Overall, the results underscore the importance of rigorous out-of-sample testing and cautious interpretation of in-sample profitability in quantitative trading strategies.