PAIR TRADING: STATISTICAL ARBITRAGE STRATEGY
DUKE COMPUTER SCIENCE 216: EVERYTHING DATA

Anisa Tapia, Apoorv Jha, Eleane Ye, Claire Hutchinson, Samhitha Sunkara
https://microgel.github.io/CS-216-Project/

I. Introduction and Research Questions

Our project is based on the concept of pairs trading, a market neutral trading strategy enabling traders to profit from virtually any market conditions: uptrend, downtrend, or sideways movement. This method is especially intriguing in today's market—the extreme volatility of which we saw recently with the GameStop short squeeze. Trading and generating profits in an increasingly saturated market is difficult, so we want to use data science and algorithmic techniques to see if we can generate profits without exposure to market risks. Our goal is thus to algorithmically implement a pairs trading strategy and test if it is profitable.

Pairs trading uses pairs of securities that have some sort of underlying economic link. An example could be two companies that manufacture the same type of product, Like Microsoft and Apple. Because of this underlying link, we would expect the spread (ratio or difference in prices) between these two to remain constant with time. However, on occasion there might be a divergence in the spread between these two pairs caused by temporary supply/demand changes, large buy/sell orders for one security, reaction for important news about one of the companies, etc. When there is a temporary divergence between the two securities, i.e. one stock moves up while the other moves down, the pairs trade would be to short the outperforming stock and to long the underperforming one, betting that the "spread" between the two would eventually converge. The idea is that in the long run, the price ratio between two stocks fluctuates less than stocks themselves. So when the price of a particular stock deviates too much from the calculated mean, there is an opportunity for profits as the price will eventually go back to the mean ratio.

Our project has deviated significantly from our initial proposal, as our research question is now: How to generate profits without exposure to market risks? Earlier, we were planning to examine different factors affecting stock price data in general and would have compared the performance of various ML models. We chose to update our project idea because this problem is more substantial, feasible, and relevant considering today's markets, as described above.

II. Data Sources

We have collected an initial list of all tech industry stocks (US) from Investopedia. For the 10 stocks we chose from the list, we have fetched price data from 2016 to 2020 using Yahoo Finance and Pandas Datareader. Detailed descriptions along with code and visualizations can be found here. The data sources are appropriate since Yahoo Finance and Investopedia have both deep-rooted trust established within the quantitative finance industry. Furthermore, we verified

the quality of data ourselves. A unique problem with historic stock price data is that stock prices deviate due to stock splits, dividends, etc. In order to mitigate these challenges, we have conducted our in sample testing on close prices of stocks which have been adjusted for stock splits, dividends, mergers, etc.

III. Preliminary Results and Methods

Detailed descriptions along with code and visualizations can be found here.

We have only used our training data for all steps taken till now. We first check the stock pairs for correlation and rank the pairs based on the pearson method of correlation. Next, we check for cointegrated pairs and find two pairs to be particularly cointegrated. We choose the pair (MSFT, MU) for developing our strategy based on statistical significance of historical mean reversion.

Next, we generate trading signals using arbitrarily chosen entry and exit parameters. We develop a backtesting module and test the strategy's profitability in the training period. We get the Compound Annual Growth Rate to be 44.339%, indicating that even for arbitrary parameters, our strategy outperforms the market for the same period (S&P 500 index), and the strategy of just buying and holding the respective stocks from the chosen pair for the same period.

This is highly promising as we plan to optimize the parameters in the next phase of our project and then aim to test the strategy on out of sample (test) data and evaluate performance.

IV. Reflection and Next Steps

We have successfully created the backbone for our project. Since the backtesting module is ready, we will only need to input parameters and test data to evaluate our strategy's performance on out of sample data (when we get to it).

Understanding the statistics and math behind the pair trading strategy was challenging at first. Implementing it algorithmically seemed daunting, but by dividing the work, meeting regularly, and using the help of our TA, we have been successful in reaching the project prototype milestone. We have outlined all steps in our project in the linked webpage. In brief, we have not yet done parameter tuning or any out of sample testing, which we will finish by the final submission deadline.

Next steps detailed in the linked webpage. In brief, conduct parameter tuning, test strategy, make better visualizations, evaluate performance/risk, polish as end-to-end algorithmic trading strategy.

Note: We would request you to view the linked webpage, where we have published our jupyter notebook. It would give you a comprehensive picture of our project and what we have achieved till now. It contains descriptions of all steps taken, visualizations, results, and a step-by-step plan of what we will finish before the final submission.