A Market Regime and Sentiment-Mitigated Approach to Algorithmic Trading *

Andres Caicedo, Anuva Agarwal, Cindy Chiu, Dev Patel, Nathan Luksik
Duke University

April 25, 2023

Abstract

We sought to combine several different concepts from peer-reviewed literature into one algorithmic trading strategy for US equities. Our amalgamation comprises cointegration analysis, OPTICS clustering, pairs trading, hidden Markov Models, regime-based asset allocation, fundamentals valuations, and Natural Language Processing. We managed risk by constraining the proportion of the portfolio used in each strategy, constraining our stock universe to larger companies, choosing one pair from each Morningstar sector, accounting for pair price volatility, considering three exit barriers, choosing stocks based on fundamentals, examining the regime for best suited fundamentals, and employing Natural Language Processing on news streams. We implemented our algorithm in QuantConnect coded in Python and tested on several different time periods from 2016-2023. We found that the computational limitations of the platform prevented us from making any headway in tuning hyperparameters. We did however find that the algorithm performed exceptionally well in 2022, which was a bearish year for the Standard and Poor's 500 Index (S&P 500). We provide recommendations on future research with similar goals.

*Acknowledgement

We would like to thank our instructor Professor David Ye, and our teaching assistant, Pranay Jain, both at Duke University, for their encouragement and valuable guidance at every step of the way in the graduate class on Algorithmic Trading at Duke, this Spring 2023.

Contents

1 Introduction

Markets have long been known to exhibit certain statistical properties that persist over a period of days, weeks, months, or even years due to reasons such as, but not limited to, macroeconomic conditions, governmental regulations, and political events. A constant challenge for market participants is detecting property changes in the market and responding accordingly. Ammann explored this very idea and showed how these prolonged market regimes could impact various investing styles differently ((Ammann / Verhofen, 2006)). Various researchers and financial players have attempted to classify these regimes in many different ways and have sought to maximize profits by tailoring their trading and investing strategies to each regime.

The coining of the terms bull and bear, for instance, are just one such attempt by people to provide a general classification of these market properties. Many, such as (Wang / Lin / Mikhelson, 2020), choose to determine regime classifications in this manner and then attempt to divide the market into these predetermined categories. In another example, (Kim / Jeong / Lee, 2019) explored the use of the hidden Markov model for regime detection to identify specific asset classes to invest in based on the current dominant market regime. Others, such as (Bao / Botte, 2021), instead employ forms of unsupervised machine learning models to divide the market into different regimes as the model sees fit and then try to understand and label the resulting classifications

With the latter methodology, taking advantage of the differences between regimes with respect to trading is much less straightforward but may offer novel insights. We employed the former methodology for the sake of clarity and inspired by (Wang / Lin / Mikhelson, 2020), in this paper, we explore the use of hidden Markov models to classify the market into 'bear' and 'bull' regimes. We also incorporate a market-neutral strategy, which is the pairs trading strategy, to mitigate the risk and diversify our strategy.

One thing that holds true is to never put all your eggs in one basket since it is difficult to know the future predictive power of a quantified trading strategy. Pairs trading is essentially trading on the spread, whereas factor model investing is based on investing in individualized stocks. This further showcases the importance of incorporating diversification even at the strategy level.

2 Background

2.1 Hidden Markov Models

A hidden markov model is a statistical model which assumes the system being modeled acts as a markov chain. A markov chain is a stochastic model describing a sequence of possible events, or states, each of which can take on values from some set. The fundamental assumption of a markov chain is that the system being modeled is memoryless. That is, the states prior to the current state have no impact on future states. Hence, the probability of moving to each possible next state depends solely on the current state.

Formally, a markov chain has three main components (Jurafsky / Martin, 2023):

1. A set of N states
$$Q = q_1, q_2, ..., q_N$$

2. A transition probability matrix A, in which each a_{ij} represents the probability of moving from state i to state j such that $\sum_{j=1}^{n} a_{ij} = 1 \ \forall i$
$$A = a_{11}, a_{12}, ..., a_{n1}, ..., a_{n1}$$

3. An initial probability distribution over the set of states, where π_i is the probability that the markov chain will start in state i (note that $\sum_{i=1}^{N} \pi_i = 1$)
$$\pi = \pi_1, \pi_2, ..., \pi_N$$

In the case where states cannot be observed directly, they are called "hidden", and we add two components to the original markov model, resulting in a hidden markov model:

4. A sequence of T observations, each drawn from a vocabulary $V = v_1, v_2, ..., v_V$
$$O = o_1, o_2, ..., o_T$$

5. A sequence of observation likelihoods, called emission probabilities, each expressing the probability of an observation o_t being generated from a state i
$$B = b_i(o_t)$$

Here, we have two assumptions. The first of which is the initial markov assumption from before, that the probability of a state only depends on the previous state: $P(q_i|q_i, ..., q_{i-1}) = P(q_i|q_{i-1})$ The second is that the probability of output observation o_i depends solely on the state which produces the observation qi and not on any other observations or states: $P(o_i|q_1, ..., q_i, ..., q_T, o_1, ..., o_i, ..., o_T) = P(o_i|q_i)$.

Fitting hidden markov models can be characterized by 3 fundamental subproblems (Rabiner, 1989):

1. Likelihood – given a hidden markov model $\lambda = (A, B)$ and observation sequence O, determine likelihood $P(O|\lambda)$

2. Decoding – given a hidden markov model $\lambda = (A, B)$ and observation sequence O, determine the best hidden state sequence Q

3. Learning – given observation sequencee O and a set of states in the hidden markov model, learn the parameters A and B of the model.

The first problem can be solved by applying the forward algorithm. This is a dynamic programming algorithm that recursively computes the forward probabilities and finds the likelihood of ending in a state given the prior observation sequence O. The algorithm does this by summing the probabilities of all of the various hidden state paths that can potentially generate the observation sequence.

The second problem can be solved by the Viterbi algorithm, which decodes the observation sequence O to find the most probable sequence of hidden states. The Viterbi algorithm recursively computes the most probable path Q through a sequence of states by storing the probability and state sequence of the most probable path at each point in time.

Lastly, for the third problem, the Baum-Welch algorithm, which uses the forward-backward algorithm, is used. This computes the conditional distribution of observations for the hidden states in two passes. In the first pass, the algorithm computes the forward probabilities as described in the forward algorithm. In the second pass, it computes the backward probabilities, giving the probability of observing the rest of the observation sequence given a starting state. As a result of two passes, we can compute the probability of being in a particular state at any given point in time given O, which is then iteratively used in an expectation-maximization fashion to move from our initial estimates of the parameters A and B to more probable estimates (Wang / Lin / Mikhelson, 2020).

2.2 Pairs Trading

Pairs trading is a popular form of statistical arbitrage strategy based on the fundamental concept of mean reversion. The strategy involves selecting a pair of highly correlated securities that exhibit a tendency to return to their historical average prices after periods of deviation. The aim is to exploit this trend by longing the undervalued security and shorting the overvalued security, with the expectation that the undervalued stock's price will increase and the overvalued stock's price will decrease. The prices are expected to converge or revert back to the mean, generating a profit for the investor.

The basis of our pairs trading algorithm originates from (Chan, 2013)'s implementation of a pairs trading strategy involving two pairs of stocks from the alternative energy sector. The strategy assumes that both pairs of stocks are cointegrated, implying that they have a long-term relationship that can be modeled by a linear equation. The cointegration assumption allows the investor to profit by simultaneously longing or shorting the spread according to a two-standard deviation threshold. Statistical arbitrage relies on the spread, which is the difference in log prices, of two securities. This involves longing the undervalued security and shorting the overvalued security. The expectation is that the undervalued stock's price will increase and the overvalued stock's price will decrease, leading to convergence of their prices back to the mean. The strategy employs a long and short position at any moment, making pairs trading a "market neutral" strategy.

To further enhance the pairs trading strategy, we assume that we have no prior knowledge of which pairs of stocks will be profitable in the future. To address this issue, we employ an unsupervised learning model to select potentially profitable stock pairs automatically, using clustering. This allows us to automate the process of selecting stock pairs and identify pairs with potentially higher profit potential.

In summary, pairs trading is a powerful statistical arbitrage strategy that exploits the mean reversion of two highly correlated stocks. By simultaneously longing an undervalued stock and shorting an overvalued stock, the strategy seeks to profit from the convergence of their prices back to their fundamental levels. The strategy employs a long and short position at any moment, making it a "market neutral" strategy. Our algorithm builds upon this strategy by using clustering to select potentially profitable stock pairs automatically, providing a powerful tool for investors.

3 Strategy Description

In our strategy, we divide our funds evenly between two sub-strategies. The first is a pairs trading strategy in which the pairs selection process involves using unsupervised machine learning to regularly cluster together stocks that are highly cointegrated, and therefore are good candidates for pairs trading. In the second sub-strategy, we make a hidden markov model to classify the market into various regimes and employ different trading strategies based on the model's predictions.

3.1 Pairs Trading Strategy

Pairs trading is used to exploit the securities that are out of equilibrium in financial markets. The strategy involves identifying two securities (e.g., stocks, bonds, foreign exchanges) whose prices tend to move together in the long term. When prices are divergent, the cheaper security is bought long, and the more expensive one is sold short. When prices converge back to the equilibrium, the trade is ended, and a profit is obtained.

The inclusion of pairs trading as a market-neutral approach is deemed necessary to mitigate the overall risk of the strategy, in addition to regime classification. The adoption of this approach is independent of market conditions to ensure a consistent risk reduction. Pairs trading is expected to be a profitable technique across various market regimes. Limiting its application to a particular market regime would unnecessarily curtail its effectiveness and limit the contributions to the Value and Fama-French models. Consequently, we intend to allocate 50% of our capital to pairs trading. The amalgamation of both strategies should produce a diversified portfolio and reduce our exposure to risk.

3.1.1 Asset Classes in Pairs Trading

To start our stock selection process, we first apply a filter to our universe of stocks by only including companies with a market capitalization larger than micro-cap. In particular, we eliminate companies with a market capitalization of less than 250 million to ensure that we only consider mature firms. Moreover, we exclude penny stocks from our analysis by removing stocks with a share price below ten dollars. This selection criterion is put in place to eliminate stocks that may be subject to high volatility and low liquidity. Ideally, we want the top 1500 stocks sorted by market cap from the resulting stocks. However, QuantConnect has subscription limitations, and we could only apply our universe selection on 500 stocks in the live trading.

After the initial filtering process, we gather relevant data on the resulting stocks, including their daily returns and various fundamentals such as market capitalization and sector codes. This information serves as crucial input to our subsequent analysis and helps us gain insight into the stocks' performance and underlying characteristics.

3.1.2 PCA

We collect the daily return for 180 days, which means we have at least 180 features, and OPTICS can't take in so many features. Therefore, we leverage Principal Component Analysis for dimensionality reduction. Principal Component Analysis (PCA) is a technique used for dimensionality reduction. It is commonly employed to identify the underlying structure in data by reducing the number of variables in the dataset while retaining as much information as possible.

PCA works by transforming the data into a new set of uncorrelated variables, called principal components, that capture the maximum amount of variation in the original data. The first principal component is the linear combination of the original variables that explains the largest amount of variance in the data. The subsequent principal components are calculated in a similar fashion, with the constraint that each subsequent component is orthogonal to the previous ones.

In this paper, PCA will reduce 180 daily stock prices into 10 variables while trying to keep as much variance as possible. After getting the results from PCA, we add back in market cap to ensure that market cap is one of the features. The stocks will be clustered based on these components.

3.1.3 Stock Clustering

In order to address the potential impact of different sectors on our trading strategy, we conducted clustering based on each sector code. Specifically, we employed the OPTICS algorithm for clustering, which is a density-based clustering approach that can identify clusters of varying densities and shapes. The advantage of OPTICS is its ability to handle noise and outliers in the data. By using a density-based approach, the algorithm is able to identify points that do not belong to any cluster, which can be useful for detecting anomalies or data points that may be spurious. In addition, OPTICS allows users to only specify parameters such as the minimum number of points required to form a cluster, which minimize the hyperparameter tuning process. To ensure that each resulting cluster has a reasonable number of pairs for trading, we set the minimal number of pairs required to form a cluster to be 4.

By clustering the pairs based on their sector codes, we aim to group together pairs that have similar characteristics and tendencies within their respective sectors. This approach can help to improve the effectiveness and robustness of our pairs trading strategy, as it allows us to account for potential sector-specific factors that may impact the co-movement of stock prices.

3.1.4 Pair Selection

In selecting the most optimal pair for our trading strategy, it is essential to ensure that the chosen pair is co-integrated and has stationary residuals between them. Co-integration implies that the pair moves together in the long run, and stationary residuals signify that the difference between the pair's prices is predictable and not prone to wild swings.

To identify the optimal pairs, we use a clustering algorithm that groups stocks with similar characteristics together. Within each cluster, we iterate through all the possible pairs and perform the co-integration test and the Augmented Dickey-Fuller (ADF) test. The co-integration test determines whether there is a long-run relationship between two variables, while the ADF test examines the stationarity of a time series by assessing the existence of a unit root.

If multiple pairs within a cluster pass both tests, we select the pair with the smallest p-value in both tests, as this indicates the highest level of confidence in the statistical significance of the pair. However, if none of the pairs in the cluster passes the tests, we discard the entire cluster.

Overall, this process ensures that we only select pairs that are statistically significant, reducing the risk of including pairs that may have unreliable or unpredictable behavior. As a result, our trading strategy will consist of at most nine pairs, providing us with a focused and optimized approach to market neutral trading.

3.1.5 Pair Weights Allocation

Given that our portfolio will comprise multiple pairs, it is imperative to determine an appropriate capital allocation strategy that considers the relative risk contributions of each pair. Equal allocation represents the simplest method of capital allocation. However, it does not account for differences in the volatility of individual pairs, which can increase the overall risk of the portfolio.

To mitigate the unintended large volatility contribution from any single pair, we employ Equal Contribution to Risk (CTR) (Bailey / Lopez de Prado, 2012) weighting scheme to ensure that each pair contributes to volatility equally. CTR is a measure of the proportion of overall risk in a portfolio that can be attributed to a specific asset or group of assets. In determining CTR, the weight of each asset in the portfolio and its volatility, as measured by its standard deviation, are taken into account.

Instead of relying on traditional return volatility, we opt to use the volatility of the z-score for each pair as our measure of risk. Specifically, we calculate the residual z-scores of each pair's stock return over a period of 180 days and determine the standard deviation of these z-scores. The use of residual z-score volatility enables us to capture the volatility of each pair's deviation from its mean, providing a more accurate measure of the pair's risk contribution.

We then use the CTR weighting approach to allocate capital to each pair based on their relative risk contribution. By assigning weights that account for the differences in each pair's risk, we achieve a more diversified and balanced portfolio, which can mitigate overall risk exposure.

3.1.6 Trading Pairs

Once we have selected the pairs and determined their weights, our trading strategy will involve implementing trades on a daily basis. To initiate a trade, we first check if there is an existing open position for the pair. If there is no open position, we then evaluate the z-score of the pair.

If the z-score is greater than the entry threshold, which is set at 1.3, we short the spread and enter the trade. This means we sell the overvalued stock and buy the undervalued stock in the pair, with the expectation that the prices will converge in the future. On the other hand, if the z-score is smaller than the negative entry threshold, which is set at -1.3, we long the spread and enter the trade. This implies that we buy the undervalued stock and sell the overvalued stock in the pair, with the expectation that the prices will converge in the future.

It is important to note that our entry threshold is set at a conservative level to minimize the possibility of false signals and reduce the likelihood of incurring losses. By only initiating trades when the z-score crosses a certain threshold, we can ensure that our trades are based on statistically significant movements in the market and avoid any premature or impulsive actions.

Overall, this approach ensures that we enter trades when the pair's prices are significantly different from their historical relationship, which provides a higher probability of a profitable outcome. By consistently implementing these trades, we can capture any market inefficiencies and generate returns for our portfolio.
For exiting each trade, we implement a triple barrier approach:

1. Take profit when $abs(Z_{score}) < Z_{take-profit}$
2. Stop-loss when $abs(Z_{score}) > Z_{stop}$
3. Time barrier exit by closing out all positions every Friday

To optimize the profitability of our trading strategy, we employ a take-profit approach to exit our trades. Specifically, we will close out the trade if the mean of each pair starts to revert by crossing zero, which is our exit threshold. This threshold represents the point at which the spread returns to its long-term average, indicating a potential convergence in the prices of the two stocks.
Moreover, to minimize our losses and manage risk, we have also implemented a stop-loss threshold, which is set at 2.3 or -2.3. This threshold represents the maximum deviation that we are willing to tolerate before we liquidate the pair. If the z-score of the pair exceeds this threshold, we will immediately liquidate the pair to avoid any further losses. This approach helps us to limit our downside risk and minimize our losses in case the market moves against us.

Additionally, we have set a time exit barrier, wherein we liquidate all the pairs and reset the pairs trading portfolio every Friday before the market closes. By resetting our portfolio each week, we can ensure that our pairs are based on the most up-to-date information.

Overall, by utilizing these exit strategies, we can effectively manage our risk and maximize our returns by exiting trades at the right time and minimizing losses in case of adverse market movements.

3.2 Regime Switching via Hidden Markov Models

Drawing from Wang et al (Wang / Lin / Mikhelson, 2020), we create a hidden markov model which observes historical daily S&P returns and volatility on a 10-day moving average to identify patterns of market behavior, i.e. regimes, that characterize the market into one of three states: bull, bear, or neutral. Intuitively, we seek to differentiate between periods of when the market is generally healthy and growing, and periods of market downturn and unpredictability, and in each case employ a trading strategy optimal for those conditions. We implement this hidden markov model as described above in the background section, and train it each day using a rolling window of 2600 days, or approximately 10 years of S&P information.

If we determine we are in a bull market, we employ a long-only value/growth strategy, and if we determine we are in a bear market, we instead employ a modified Fama-French strategy. The rationale behind these two choices is as follows. From backtesting many factor models over the last 10 years, Wang et. al. (Wang / Lin / Mikhelson, 2020) find that the former strategy generally performs the best, but is subject to large drawdowns during periods of economic downturn such as the latter half of 2018 and the beginning of the Covid-19 pandemic in March 2020. The strategy that performed the best during such periods was a modified form of the Fama-French 3-factor model. Hence, we chose our factor models based on which we thought would be the most profitable for each regime. The third state, neutral, is meant to provide a classification for when the market should be considered neither a bull nor a bear market. In this case, we will continue to use whichever factor model is currently active.

Being that we alter our trading strategy depending on which market regime the hidden markov model has determined we are in for that day, we will be subject to high overhead fees if the current regime, therefore our strategy and hence portfolio, switches frequently. Furthermore, we seek to uncover broader market trends and not be greatly affected by every single regime determination by our model. Therefore, we implement a high-pass filter to combat these issues. First, we use the Kolmogorov-Smirnov test to determine the most fitting kind of distribution for the aggregate of the aforementioned volatility and return values for each regime in our training window. Then we get the probability distribution function (PDF) of this aggregation. We only let the hidden markov model determine a switch in regimes if the ratios of the PDF of the most recent regime to the sum of the PDFs for all regimes for volatility and for daily returns are at least 0.3 and 0.5, respectively (Wang / Lin / Mikhelson, 2020).

3.2.1 Value Model

The value model we employ is a long-only growth model which selects stock based on highest dividend-per-share, highest book-value-per-share, highest free cash flow yield, and least or most negative stock price change in the past 1 month. Hence, it seeks to find "value stocks". This strategy performs best in bull markets because these regimes are characterized by high investor optimism and a willingness to take on risk. The principle of buying stocks that are undervalued by the market, with the expectation that their price will eventually increase as the market recognizes their true value.

3.2.2 Fama-French Model

When not employing the value model, in our strategy we implement a modified Fama-French model. The Fama-French is an asset pricing model that expands on the capital asset pricing model (CAPM). While CAPM accounts for market risk, this model adds size risk and value risk factors. Hence, this model ranks stocks based on three factors, selecting for stocks with the smallest market cap, lowest price-to-book ratio, and best excess return on the market. Backtesting various factor models showed that modifying the Fama-French by measuring excess return via selecting for low one month momentum instead of looking at the PE ratio results in improved performance, particularly in bear markets (Wang / Lin / Mikhelson, 2020). This model works well in bear markets because the underlying factors included in the model are expected to perform well in periods of economic distress or market downturns. Investors tend to favor smaller and more value-oriented stocks during such periods of distress.

3.2.3 HMM Asset Classes and Selections

First, we use the QuantConnect Course Filter to filter stocks that have fundamental data and prices greater than ten dollars. We then sort the list by dollar volume and take the top 500 stocks. After the Course Filter, we apply a fine filter to filter stocks for each of the factor models, resulting in three lists of stocks: a list of the stocks we have selected to long for the Value model, a list of stocks we have selected to long for the Fama-French model, and a list of stocks we have selected to short for the Fama-French model.

For both factor models, we filter out stocks for which QuantConnect does not have data regarding that model's relevant factors, calculate a score for each stock based on these factors, and then sort them based on this score. The score is calculated by weighing each of the three factors, which can vary from setting the factors to equal weights or varying the weights for each factor. For the Fama-French model, once the stocks are ranked and sorted, we take the 50 stocks with the highest score to long, and the 50 with the lowest score to short. For the value model, after ranking and sorting, we again pick the top 50 stocks to long. Finally, we pass the stocks in all three lists (Fama-French Long, Fama-French Short, and Value Long) through our sentiment analysis filter, described below, to yield the final selection of stocks that will be traded, depending on the outcome of the hidden markov model that day.

3.2.4 Sentiment Analysis

News announcements can have a high impact on the price action of stock and thus alter investor behavior. Automated news analysis has been a component of algorithmic trading which has shorted the time traders take to respond to breaking stories. (Souza / Kolchyna / Treleaven / Aste, 2015) mentioned that there is a correlation between high/low pessimism of media and high market trading volume.

With recent advances in machine learning models like GPT, BERT, and Bard, they can be trained to do sentiment analysis on data streams like Twitter, Tiingo, and other news subscriptions. In our strategy, we utilized the Natural Language Processing Toolkit (NLTK) and the VADER tool, which is a less resource-consuming sentiment analysis model that uses a set of rules to specify a mathematical model without explicitly coding it. Valence Aware Dictionary for Sentiment Reasoning (VADER) performs as well as individual human raters at matching ground truth. Vader labels sentiment as either positive, neutral, or negative in terms of valence scores. The compound VADER score is also computed by summing the valence scores for each word in the lexicon and normalized to be between -1 and 1.

Our strategy implements sentiment analysis by incorporating valence scores as a means to filter stocks, thereby avoiding buying stocks with extremely negative recent public sentiment and avoiding shorting stocks with extremely positive recent public sentiment. If the compound score of any stock in the value model long list or in the Fama-French long list is less than 0.9, or the score of any stock in the Fama-French short list is greater than 0.998, we remove the stock from its respective list. These thresholds were determined from analyzing the distribution of sentiment score results for many potentially-tradeable stocks and finding that the vast majority of stocks score quite highly, frequently within a few hundredths or even thousandths of the maximum value of 1. Furthermore, these threshold values are extreme because we did not want to filter out too many stocks or avoid trading stocks with ambiguous sentiment scores, but rather use sentiment analysis to avoid taking chances on any stocks for which public sentiment is in stark opposition to what quantitative factors indicate.

4 Risk Management

4.1 Triple Barrier

Our pairs trading strategy employs a triple barrier approach to optimize profitability and manage risk. We use a take-profit approach to exit trades when the mean of each pair reverts by crossing zero, and a stop-loss threshold of 2.3 or -2.3 to minimize losses in case of adverse market movements. Additionally, we implement a time exit barrier by resetting our portfolio each week before the market closes to ensure up-to-date information. This approach helps us effectively manage risk and maximize returns.

1. Take profit when $abs(Z_{score}) < Z_{take-profit}$
2. Stop-loss when $abs(Z_{score}) > Z_{stop}$
3. Time barrier exit by closing out all positions every Friday

4.2 Portfolio Diversification

In the context of investment strategies, risk management is a critical aspect that requires careful consideration to optimize returns and reduce the impact of potential losses. In our universe selection process for both strategies, we have implemented several measures to mitigate risk and enhance the stability of our portfolios.

To this end, we have filtered out companies with a market capitalization of less than 250 million. This ensures that we are trading stocks that are relatively stable and have a proven track record of performance. We have also excluded all penny stocks, which are typically low-priced, highly volatile stocks with limited liquidity. By avoiding penny stocks, we reduce the risk of significant losses due to sudden price swings and low trading volumes.

In the regime switching strategy, we have used different factors to select stocks, including book value, price changes, and other relevant factors. By diversifying our stock selections, we reduce the concentration risk associated with investing in a single sector or industry. This approach allows us to spread our investments across a range of companies, reducing the overall risk of our portfolios.

To further mitigate risk, we have implemented the long-short portfolio in both the Fama French model from the regime switching strategy and the pairs trading strategy. This approach allows us to maintain a market-neutral position, reducing the impact of market movements when regime switches on our portfolio. The long-short portfolio also helps us to balance our portfolio by reducing overall investing risk. By taking a long position in one security and a short position in another, we can potentially profit from both upward and downward price movements in the market.

4.3 Equal Contribution to Risk

We attempted to distribute the volatility contributed by each of the pairs in our portfolio. Contribution to risk refers to the proportion of overall risk in a portfolio that can be attributed to a specific asset or group of assets. Thus, we found the most appropriate method to be

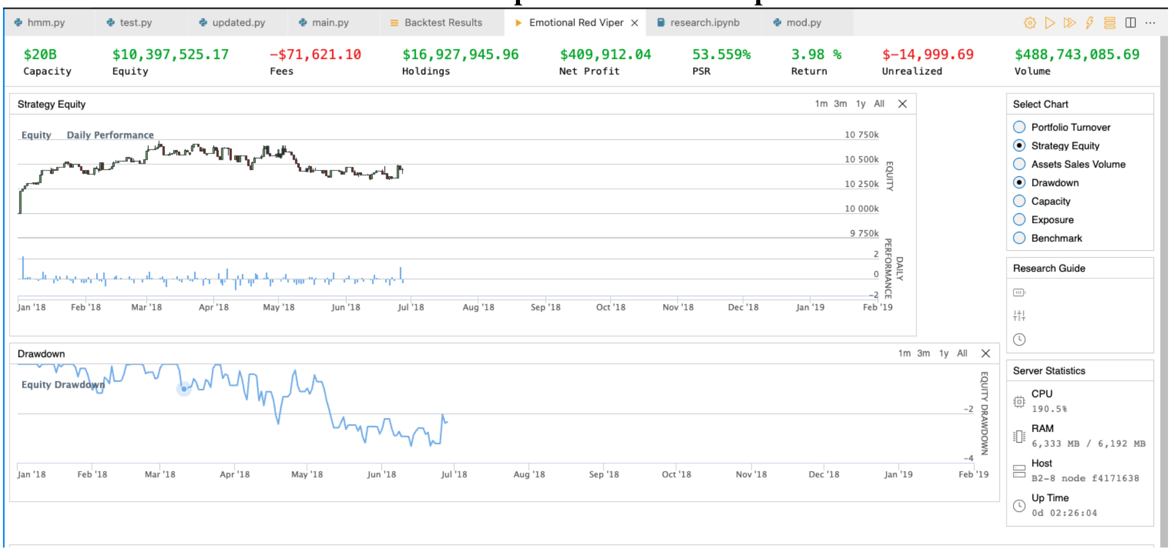
Equal Contribution to Risk. The contribution to risk of an asset is determined by its weight in the portfolio and its volatility, as measured by its standard deviation. We had to develop a method to convert the price divergence of the pair into a volatility measure. As a result, we found the z-score of the pair over a time window to provide the best analogy.

5 Backtesting Results

We tried to run the whole 4 year backtesting period for in-sample. However, due to QuantConnect limit, our algorithm has maxed out the memory several times while running the full period. We reached out to the QuantConnect team and asked for advice. We modified our code accordingly but it still maxed out the memory resources. Therefore, we have ran smaller periods in the in-sample period to test out instead.

5.1 In Sample training: Jan 1, 2017 to June 1, 2017

Results from small period from in-sample



Parameter	Value*
Profit	
Sharpe Ratio	
Maximum Drawdown	
Maximum Drawdown Length (days)	
Maximum One Day Loss	

*Values are still being generated. Awaited from ongoing backtest.

5.2 Out of Sample A: Jan 1, 2022 to Jan 1, 2023

Results from Out of Sample A

Parameter	Value
Profit	31.99%
Sharpe Ratio	1.476
Maximum Drawdown	9%
Maximum Drawdown Length (days)	103
Maximum One Day Loss	3.94%

5.3 Out of Sample B: Jan 1, 2016 to Jan 1, 2017

Parameter	Value
Profit	-14.47%
Sharpe Ratio	-0.654
Maximum Drawdown	20.9%
Maximum Drawdown Length (days)	116
Maximum One Day Loss	4.3%

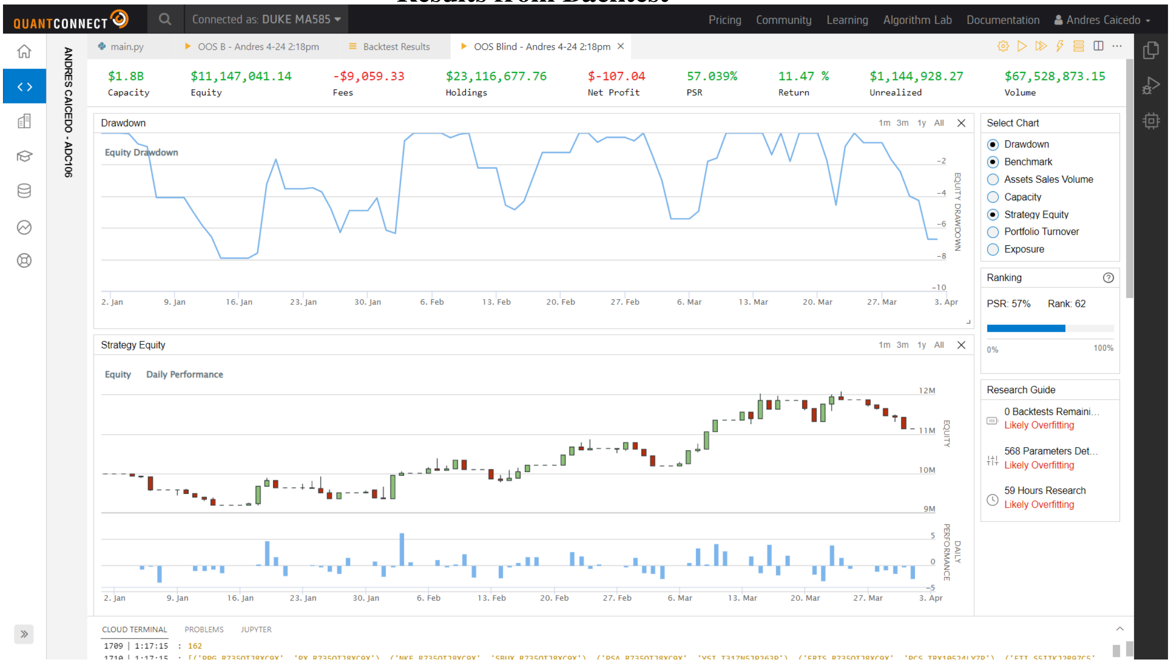
5.4 Stress Test - March 2020

Results from Live Trading

Parameter	Value
Profit	1.64%
Sharpe Ratio	.937
Maximum Drawdown	4.5%
Maximum Drawdown Length (days)	7
Maximum One Day Loss	2.0%

5.5 Blind OOS - Jan 1,2023 - Apr 1, 2023

Results from Backtest



Parameter	Value
Profit	11.47%
Sharpe Ratio	1.529
Maximum Drawdown	8.2%
Maximum Drawdown Length (days)	31
Maximum One Day Loss	3.25%

5.6 Live Paper Trading Result

Results from Live Trading

Parameter	Value
Profit	0.05%
Maximum Drawdown	0.00%
Maximum Drawdown Length (days)	1
Maximum One Day Loss	0.00%

6 Conclusion

In this research endeavor, we sought to combine several different concepts from peer-reviewed literature into one algorithmic trading strategy for US equities, allocating half of our portfolio to pairs trading and half of our portfolio to market regime classification, to diversify our strategy and reduce the risk. Summarily, the undertaking a strategy with a large number of variables, concepts, lines of code would be best suited to a larger time horizon to allow for improvement. However, the authors found it largely beneficial to have attempted a novel concept that proved to be extremely rigorous over having attempted a simpler set of ideas. We switch the model used for different market regimes with a K-S filter.

Through scouring literature, we found an array of risk-mitigation techniques including Natural Language processing, Triple Barrier, and Equal Contribution to Risk. We additionally, found at least one python implementation of a regime-switching strategy, which is as disputable as the existence of anything other than Geometric Brownian Motion in Equities Markets.

We noted the importance of computational resource management while backtesting the amalgamation of code. Quantconnect's system is aggressively computationally limiting. The total backtesting time period encompassed about 7 years of data. However, backtesting one year lasts 80 minutes, and exceeding 18 months was prohibited by maximum memory capacity per node. Lastly, the student organization completing similar projects shared 18 nodes for approximately 40 students. Frequently, no nodes were available, preventing commencing a backtest.

Market regime classification switches between a strategy that alternates been long-only (Value Model), and long-short positions (Fama-french model), based on ranking of stocks on fundamentals and factor performance. In addition to regime classification, the inclusion of pairs trading as a market neutral approach is deemed necessary to mitigate the overall risk of the strategy, and is expected to be profitable across various market regimes. It's basis is the concept of mean reversion which states that deviation of asset prices from their long term historical average will converge back to these historical aerages. Picking two stocks - one that is overvalued and one that is undervalued, and taking a short and long position in the two, respectively, allows one to profit from 'statistical arbitrage'.

7 Limitations and Future Improvements

Optimizing the parameters of the factor models is something we would like to explore going forward. Even a perfect regime classifier is useless if the factor models employed are unable to take advantage of the market conditions. Moreover, we believe there is a lot of potential to the idea of expanding the hidden markov model to include more than just 3 states, and along with it, additional factor models.

We hypothesize we could expand on this concept by attempting to classify the market in a more nuanced way than just bear and bull; that is, we would like to train a model to incorporate indicators specific to predicting the likelihood of success for various individual strategies, and then determine which, of several, would be optimal under combinations of those indicators. We believe this idea would allow us to incorporate multiple strategies in a more obvious way, as we could decide the number of HMM states from the strategies we employ. Instead of having regime classifications as either bull or bear, our regime classifications would, instead, become "conditions best for strategy x" or "best for strategy y". As a cautionary statement, we must be very wary of overfitting parameters to the training period for both optimizing and adding factor models. (Johnson-Skinner / Liang / Yu / Morariu, 2021)

Another limitation we need to consider is the computing resources available. Within the testing of our strategy, there were many instances where the memory maxed out. This impacted both our live trading and backtesting. We reached out to QuantConnect and revised our code upon their recommendations. However, even after replacing our code with the rolling window they suggested we do, the memory still maxed out when we were running the full in-sample test. In addition, since our regime switching requires 10 years of SPY data, we had to wait long periods for a full backtest to be completed. Furthermore, QuantConnect has a limitation on how many stocks we can subscribe to in live trading. Therefore, instead of looking at 1000 stocks, we could only look at 500 of them and it limited the cluster performance. We believe that if given more computing resources, we would be able to fully understand the limitations of our strategy and hence improve the performance. In addition, we believe if given more support on the QuantConnect platform, we would be able to understand the platform and built-in library to utilize it to its fullest potential.

References

Rabiner, L.(1989): *A tutorial on hidden Markov models and selected applications in speech recognition*, 2: 257-286.

Ammann, Manuel / Verhofen, Michael(2006): *The effect of market regimes on style allocation,*309-337.

Bailey, David H / Prado, Marcos Lopez de (2012): *Balanced baskets: A new approach to trading and hedging risks*, 4: .

Souza, Thársis Tuani Pinto / Kolchyna, Olga / Treleaven, Philip C / Aste, Tomaso(2015): *Twitter sentiment analysis applied to finance: A case study in the retail industry*.

Kim, Eun chong / Jeong, Han wook / Lee, Nak young(2019): *Global asset allocation strategy using a hidden Markov model*, 4: 168.

Wang, Matthew / Lin, Yi Hong / Mikhelson, Ilya(2020): *Regime-switching factor investing with hidden Markov models*, 12: 311.

Bao, Doris / Botte, Alex(2021): *A Machine Learning Approach to Regime Modeling*.

Jurafsky, Dan / Martin, James(2023): *Hidden Markov Models*Appendix A.

Chan, Ernie (2013): *Algorithmic Trading: Winning Strategies and Their Rationale*. , Wiley Publishing.

Johnson Skinner, Ethan / Liang, You / Yu, Na / Morariu, Alin (2021): *A Novel Algorithmic Trading Strategy using Hidden Markov Model for Kalman Filtering Innovations*. : 1766–1771.