Correlation Matrix Clustering for Statistical Arbitrage Portfolios

Álvaro Cartea[*†] Mihai Cucuringu[*‡†§] Qi Jin[*‡¶]

September 4, 2023

Abstract

We propose a framework to construct statistical arbitrage portfolios with graph clustering algorithms. First, we use various clustering methods to partition the correlation matrix of market residual returns of stocks into clusters. Next, we construct and evaluate the performance of mean-reverting statistical arbitrage portfolios within each cluster. We explore five clustering algorithms and demonstrate that our proposed framework generates profitable trading strategies with over 10% annualized returns and statistically significant Sharpe ratios above one. The performance of our statistical arbitrage portfolios is neutral to the market and cannot be fully explained by intra-industry mean-reversion effects.

Keywords: Correlation matrix, Clustering, Portfolio management, Random matrix theory, Statistical arbitrage

[*]Oxford-Man Institute of Quantitative Finance, University of Oxford
[†]Mathematical Institute, University of Oxford
[‡]Department of Statistics, University of Oxford
[§]The Alan Turing Institute, London, UK
[¶]Corresponding author; Email: qi.jin@st-annes.ox.ac.uk

1 Introduction

Statistical arbitrage encompasses investment strategies that use statistical and quantitative methods to identify and exploit temporal price deviations among a group of similar assets. An example of a classical statistical arbitrage strategy is pairs trading Elliott et al. (2005); Cartea and Jaimungal (2016); Cartea et al. (2019), which takes a long position in one security and a short position in another security with the expectation that the spread between their prices will revert back to an anticipated level. While works such as Bergault et al. (2022); Bertram (2010) use the Ornstein-Uhlenbeck process to model stock prices, we focus on a model agnostic statistical arbitrage framework that consists of two steps, (1) identify a group of assets that share similarities, and (2) construct an arbitrage portfolio within the group of assets.

In this paper, we propose a framework where we employ graph clustering algorithms to identify groups of correlated stocks that co-move. Then, we construct mean-reverting arbitrage portfolios within each cluster to evaluate if the clustering methods enable statistical arbitrage strategies that deliver economically significant profits.

In the first step, we compute market residual returns, which are given by the difference between the stock returns and the product of the stock's CAPM Fama and French (2004) equity beta and the market return.[1] Next, we compute the correlation matrix of residual returns, interpret it as a weighted signed network, and use graph clustering algorithms to partition the stocks into groups such that on average, the correlation between stocks that are in different groups is low and the correlation between stocks in the same group is high. In this paper, we employ five clustering algorithms to construct statistical arbitrage portfolios. The clustering algorithms include two variants of SPONGE clustering Cucuringu et al. (2019), a modified variant of Spectral clustering Ng et al. (2002), and two variants of the Signed Laplacian clustering Kunegis et al. (2010).

In the second step, we employ a rolling window to identify stocks whose returns are above and are below the mean returns of the cluster, which we label "previous winners" and "previous losers", respectively. Next, we construct a contrarian portfolio that consists of a long position on

[1] In this paper, we use the return of the SPY ETF as a proxy for market returns.

the previous losers and a short position on the previous winners within each cluster. We use this portfolio to evaluate if the stocks in each cluster exhibit mean-reversion patterns, i.e., the returns of stocks in each cluster revert to the mean return of the cluster.

There is an active strand of literature that applies clustering methods in portfolio management. Also, there are two main approaches to construct portfolios after grouping securities into clusters, which we detail below.

One approach uses all stocks in each identified cluster to construct mean-variance Markowitz portfolios Markowitz (1952). For example, León et al. (2017); Tola et al. (2008) first cluster the correlation matrix, then group the stocks according to the corresponding entries in the correlation matrix, and afterwards construct Markowitz portfolios within each cluster. Alternatively, Gatta et al. (2023) uses regression coefficients of asset returns on various factors to cluster securities, and then constructs a variance minimizing portfolio in each cluster.

A second approach in the literature clusters the correlation matrix as in the first approach, and then selects one asset from each of the clusters to construct a single Markowitz portfolio. For example, Tolun Tayalı (2020) constructs a Markowitz portfolio with the medoids of each cluster. Other lines of work that follow this second approach employ various clustering methods and selection mechanisms to identify the representative stock in each cluster. For example, Wang et al. (2022) selects the stock with the lowest volatility within each cluster, and Tang et al. (2021) selects the stock with the highest Sharpe ratio in each cluster.

To the best of our knowledge, ours is the first work that applies clustering algorithms in the design of statistical arbitrage strategies.

Our approach draws from the literature on clustering algorithms. In particular, we use signed clustering algorithms that handle negative weights because the correlation matrix of residual returns we employ consists of both positive and negative entries, so one cannot apply many of the classical graph clustering algorithms proposed in the literature. Previous lines of work, including Aghabozorgi et al. (2015); Focardi (2005); Ziegler et al. (2010); Pavlidis et al. (2006), use signed clustering methods to analyze financial time series such as macroeconomic variables and

time series of large baskets of stock returns.

2 Mathematical Model and Problem Setting

2.1 Signed & Directed Graph Clustering

Clustering is a widely used technique in data analysis. A clustering algorithm identifies groups of nodes in a network that exhibit similar behavior or features. In this paper, we focus on a strand of clustering algorithms that operates on the spectrum of suitably defined matrix operators that are built directly from the data. Such methods, often referred to as "spectral methods", are the subject of a growing body of literature in the last decade, mainly motivated by their computational efficacy, robustness to noise, and amenability to theoretical guarantees that rely on results from the random matrix theory and matrix perturbation literature.

This section introduces the clustering methods we use to construct statistical arbitrage portfolios. In particular, it introduces clustering methods that operate on signed networks (i.e., with adjacency matrices that are symmetric and contain both positive and negative entries).

We partition a signed network into K clusters such that most edges within clusters are positive, and most edges across clusters are negative. To achieve this, we seek a partition that minimizes the number of violations; i.e., negative edges within each cluster and positive edges across clusters, as illustrated in Figure 1.

2.1.1 Spectral Clustering

Spectral clustering Ng et al. (2002) is one of the simplest and most popular spectral methods that clusters a network based on the adjacency matrix of the network. Here, we apply spectral clustering on a correlation matrix of stock market residual returns.

In our approach, the input data is the correlation matrix of stock market residual returns. Each stock corresponds to a node in the network, and the correlation between stock returns represents the co-movement similarity between the nodes. We use the correlation matrix to build the graph

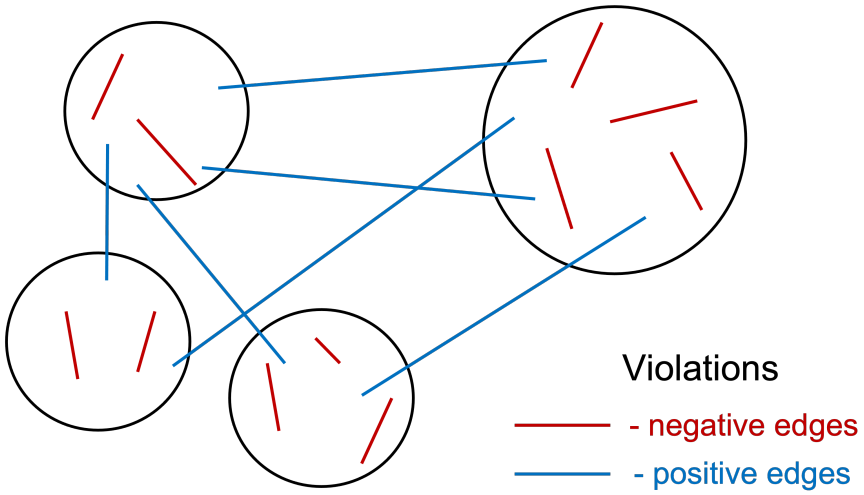

Figure 1: Signed clustering minimizes the number of violations in the constructed partition. A violation, as in this figure, is when there are negative edges in a cluster and positive edges across clusters.

Laplacian matrix $\mathbf{L}$, defined as the difference between the degree matrix $\mathbf{D}$ and the adjacency matrix $\mathbf{A}$ of the similarity graph. The degree matrix $\mathbf{D}$ is a diagonal matrix that captures the degree or total strength of connections for each node in the graph, while the adjacency matrix $\mathbf{A}$ encodes the pairwise similarities between nodes, as determined by the edge weights. The Laplacian matrix measures the difference between the sum of similarities connecting a node to the rest of the network, and it measures the node's total strength of connections. For the Spectral clustering algorithm, the entries of the similarity matrix must be positive; therefore, we take the absolute value of the correlation matrix and use this modified correlation matrix as the adjacency matrix in the Spectral clustering algorithm. In Knyazev (2017), the author uses the standard graph Laplacian matrix to perform Spectral clustering ignoring that some of the edge weights are negative; however, later works, including Cucuringu et al. (2019), report poor performances of this approach. Therefore, in this paper, we employ the unsigned Spectral clustering algorithm which considers the absolute value of the correlation matrix.

Formally, the Laplacian matrix $\mathbf{L}$ is defined as

$$\mathbf{L} = \mathbf{D} - \mathbf{A}\,, \tag{1}$$

and recall that $\mathbf{D}$ is the diagonal degree matrix and $\mathbf{A}$ is the adjacency matrix. The diagonal elements of $\mathbf{D}$ are the sums of the weights (similarities) of the edges that are connected to each node, while the off-diagonal elements of $\mathbf{A}$ represent the pairwise similarities between nodes. In our case, $\mathbf{A}$ is the absolute value correlation matrix and $\mathbf{D}_{ii} = \sum_{i=1}^{n} \mathbf{A}_{ij}$.

Next, we find the K smallest eigenvectors of the Laplacian matrix to obtain a low-dimensional embedding, from which we subsequently extract K clusters. These K eigenvectors, which correspond to a K-dimensional Euclidean space, are the input for k-means++ clustering that partitions the nodes into disjoint clusters.

2.1.2 Signed Laplacian Clustering

The Signed Laplacian clustering algorithm operates on the Signed Laplacian matrix, in contrast to the unsigned Laplacian in the case of Spectral clustering. Mathematically, the graph Signed Laplacian is defined in a similar way to the graph Laplacian. The differences are that the adjacency matrix $\mathbf{A}$ can take negative values and that the degree matrix $\bar{\mathbf{D}}$ is $\bar{\mathbf{D}}_{ii} = \sum_{i=1}^{n} |\mathbf{A}_{ij}|$.

In Kunegis et al. (2010), the authors use the spectrum of the Signed graph Laplacians to perform clustering. Specifically, they extend the ratio cut and normalized cut functions from the unsigned literature to signed graphs and use the Signed Laplacian matrix to perform clustering.

To ensure the Signed Laplacian matrix is symmetric and positive semi-definite, the algorithm normalizes the Signed Laplacian in two alternative ways. One, the random-walk normalized Laplacian is constructed as $\bar{\mathbf{L}}_{\text{rw}} = \mathbf{I} - \bar{\mathbf{D}}^{-1}\,\mathbf{A}$. Two, the symmetric normalized graph Laplacian is defined as $\bar{\mathbf{L}}_{\text{sym}} = \mathbf{I} - \bar{\mathbf{D}}^{-1/2}\,\mathbf{A}\,\bar{\mathbf{D}}^{-1/2}$. We employ both normalization methods in our empirical study.

After normalizing the Signed Laplacian, the algorithm solves an optimization problem on the normalized cut functions. First, it computes the K smallest eigenvectors of the normalized Signed Laplacian and then performs k-means++ clustering on the Euclidean space of eigenvectors.

2.1.3 SPONGE — a generalized eigenproblem

SPONGE (Signed Positive Over Negative Generalized Eigenproblem) is a generalized eigenvalue formulation of the signed clustering problem, which outperforms many benchmarks in the literature Cucuringu et al. (2019). The algorithm is particularly effective when the number of clusters K is large or when the underlying graph is very sparse.

The algorithm first decomposes the adjacency matrix $\mathbf{A} = \mathbf{A}^+ - \mathbf{A}^-$, with $\mathbf{A}^+_{ij} > 0$ and $\mathbf{A}^-_{ij} > 0$. Next, it constructs two Laplacian matrices $\mathbf{L}^+$ and $\mathbf{L}^-$, with corresponding diagonal degree matrices $\mathbf{D}^+$ and $\mathbf{D}^-$ based on $\mathbf{A}^+$ and $\mathbf{A}^-$, respectively. The algorithm minimizes the ratio between the positive cuts and negative cuts, while adding regularization terms to promote clusterizations that avoid small-sized clusters. The approach extends to multiple clusters, and, under appropriate assumptions and changes of variables, it leads to a generalized eigenvalue problem which can be solved efficiently using pre-conditioners Knyazev (2001).

In practice, the SPONGE algorithm finds the K smallest generalized eigenvectors of $(\mathbf{L}^+ + \tau^- \mathbf{D}^-, \mathbf{L}^- + \tau^+ \mathbf{D}^+)$, where τ^+, $\tau^- > 0$ are regularization parameters. The algorithm then performs k-means++ clustering on the induced K-dimensional Euclidean space.

We also employ the variant SPONGE$_{sym}$ of the SPONGE algorithm, which relies on the symmetric graph Laplacian $\bar{\mathbf{L}}_{\text{sym}}$. This variant first finds the smallest K generalized eigenvectors of $(\mathbf{L}^+{}_{sym} + \tau^- \mathbf{I}, \mathbf{L}^-{}_{sym} + \tau^+ \mathbf{I})$, where $\mathbf{L}^+{}_{sym} = (\mathbf{D}^+)^{-1/2} \mathbf{L}^+ (\mathbf{D}^+)^{-1/2}$ is the symmetric Laplacian of $\mathbf{A}^+$ (and similarly for $\mathbf{L}^-{}_{sym}$). This symmetric Laplacian is useful for networks with skewed degree distributions. Under suitably defined signed stochastic block models, Cucuringu et al. (2019, 2021) provide theoretical cluster recovery guarantees (upper bound on the misclustering rate) for the SPONGE family of algorithms as a function of the noise and edge sparsity levels.

2.2 Portfolio Construction

We construct K zero-cost statistical arbitrage portfolios within a universe of N stocks, where K is the number of clusters we identify. There are four steps:

1. Data pre-processing,

2. Group stocks into disjoint clusters,

3. Identify a collection of stocks within each cluster such that a linear combination of them is likely to mean-revert to zero,

4. Assign portfolio weights to the selected stocks in each cluster.

2.2.1 Data Pre-processing

First, we compute the market residual return $R^{res}_{i,t}$ of stock i at time t, which is given by

$$R^{res}_{i,t} = R_{i,t} - \beta_i \, R_{mkt,t} \, , \tag{2}$$

and where $R_{i,t}$ is the raw return of stock i at time t, β_i is the beta coefficient of stock i, which measures its sensitivity to market movements, and $R_{mkt,t}$ is the market return at time t. In our empirical study below, the market is the SPY ETF and we use a 60 trading day rolling window to estimate β and compute the market residual return.

The market residual return represents the component of the stock's return that is not explained by overall market movements, i.e., the idiosyncratic dynamics of each stock. In our case, we use the market residual return to focus on the portion of the stock returns that are specific to the stock themselves and to study the commonalities in their idiosyncratic dynamics.

After computing the market residual return of each stock, we construct the correlation matrix of market residual returns, which we use as the input of the later steps of portfolio construction.

Suppose at time T we want to construct the correlation matrix of market residual returns for N stocks, we first obtain the market residual return of stocks from time $T - W$ to $T - 1$, inclusive. Next, we organize these residual returns into a matrix $\mathbf{R^{res}}$ of dimension w days by N_t stocks. Each element $R^{res}_{t,i}$ in this matrix corresponds to the residual return of stock i on day t.

Then, we compute the entries of the correlation matrix $\mathbf{C}$ as follows

$$C_{i,j} = \frac{\sum_{t=T-w}^{T-1} \left(R_{t,i}^{res} - \bar{R}_i^{res}\right) \left(R_{t,j}^{res} - \bar{R}_j^{res}\right)}{(w-1)\,\sigma_i\,\sigma_j}\,, \tag{3}$$

where $\bar{R}_i^{res}$ denotes the mean of the residual returns of stock i, σ_i and σ_j are the standard deviations of returns for stocks i and j over the w days. The resulting correlation matrix $\mathbf{C}$ is of size $N \times N$ and contains the pairwise correlation coefficients between all stocks in the matrix $\mathbf{R^{res}}$.

2.2.2 Group stocks into clusters

We employ the clustering algorithms outlined in Section 2.1 to partition the correlation matrix of stock market residual returns into K distinct and non-overlapping clusters. Next, we group stocks according to the clusters we obtain from the correlation matrix which groups stocks based on their residual returns while remaining agnostic to market factor moves.

2.2.3 Identify stocks to trade

After computing the clusters, we extract arbitrage signals for each stock. Within each cluster, we compute the mean raw return of stocks over a lookback period of w days, and we measure the cumulative deviation of each stock's raw returns from the cluster mean over the past w days.[2]

Consider the returns of stocks $R_{1,t}, ...R_{j_n,t}$ in cluster j, define the cluster mean return at time T over the lookback period of w days as

$$\bar{R}_{j,t} = \frac{1}{j_n} \sum_{i=1}^{j_n} R_{i,t}\,. \tag{4}$$

Recall that we identify stocks that outperform the cluster mean over the past w days as previous winners, and stocks that underperform the cluster mean over the past w days as previous losers. In particular, we set a threshold p such that stocks whose returns cumulatively deviate by more than a threshold p from the cluster mean are believed to be more likely to revert back to the cluster mean. We expect the previous winners to revert down to the cluster mean and the previous losers to revert

[2]Empirical results show that our portfolio is market-neutral despite using raw returns.

up to the cluster mean over the next T days.

We identify stock j_i with return $R_{j_i,t}$ in cluster j where the deviation $\sum_{t=T-w}^{T-1}(R_{j_i,t} - \bar{R}_{j,t}) > p$ as a previous winner; similarly, if $\sum_{t=T-w}^{T-1}(R_{j_i,t} - \bar{R}_{j,t}) < -p$, we identify stock j_i as a previous loser.

2.2.4 Assign weights to stocks

After identifying the previous winners and the previous losers in each cluster, we assign weights to these stocks and execute a contrarian trading strategy over the next ℓ days.

Within each cluster, we short-sell the previous winners, while simultaneously initiating long positions on previous losers. Portfolio weights are the same for all stocks. Specifically, we normalize the portfolio weights such that the total dollar value of both long and short positions sums up to one within each cluster, which guarantees a zero-cost arbitrage portfolio at inception. For example, if there are two previous winners and four previous losers in a cluster, we construct a portfolio that short-sells fifty cents on each of the previous winners and bets twenty-five cents on each of the previous losers. This approach aligns with our objective of capturing mean-reversion opportunities, while maintaining a cost-neutral trading strategy.

At the end of ℓ days, we re-balance the portfolio. We first re-compute the correlation matrix of market residual returns over the w day lookback window; then re-compute the clusters and re-construct the portfolio.

To manage risk and optimize performance, we introduce a stop-win threshold at q. Should our portfolio realize a return of q before the completion of the ℓ days, we interpret this as evidence of successful mean-reversion. In that case, we immediately re-balance the portfolio in the same way as if the ℓ-day trading period had ended. With this stop-win mechanism, the portfolio is exposed to profitable mean-reversion events, while mitigating downside risk.

Below, we discuss the choice of the number of clusters K we find to construct the portfolios.

2.3 Choosing the Number of Clusters

Determining the number of clusters to partition the network is not straightforward; the literature explores various approaches. Here, we employ methods from random matrix theory and standard statistical analysis to dynamically determine how many clusters to extract every time we construct the arbitrage portfolios.

Consider a universe of N stocks over T days, and store the market residual returns of these stocks in an T by N matrix denoted $\mathbf{X}$. Let $\mathbf{C} = \frac{1}{N}\mathbf{X}^T\mathbf{X}$ be the N-by-N empirical correlation matrix of $\mathbf{X}$. The Marchenko–Pastur theorem characterizes the limiting behavior of the eigenvalues of $\mathbf{C}$. It states that, if the entries of $\mathbf{X}$ are independent identically distributed random variables with mean 0 and finite variance, then as $N, T \to \infty$, with ratio $\rho = N/T$ fixed, the empirical distribution of the eigenvalues of $\mathbf{C}$ converges to the Marchenko–Pastur distribution.

The Marchenko–Pastur distribution characterizes the limiting distribution of eigenvalues of Wishart matrices, and is defined by the density function

$$f(\lambda) = \begin{cases} \frac{\sqrt{(\lambda_+ - \lambda)(\lambda - \lambda_-)}}{2\pi\lambda\sigma^2\rho}, & \text{for } \lambda \in [\lambda_-, \lambda_+], \\ 0, & \text{otherwise,} \end{cases} \tag{5}$$

where $\lambda_- = (1 - \sqrt{\rho})^2$ and $\lambda_+ = (1 + \sqrt{\rho})^2$.

To determine the number k of eigenvalues that provides the dimension of the low-dimensional embedding, we select the eigenvalues of the correlation matrix that exceed the threshold λ^+, which are the eigenvalues associated with dominant factors or patterns in the stock returns. Therefore, k and the number of clusters K are the same, as detailed in Section 2.1.

An alternative, and more classical approach, is to consider the total variance explained by the eigenvalues and select the number of largest eigenvalues needed to account for a specific proportion P of the total variance. Specifically, sort the eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_N$ of the correlation matrix $\mathbf{C}$ in decreasing order, such that λ_1 is the largest eigenvalue and λ_N is the smallest. To determine the number of eigenvalues needed to account for a proportion P of the total variance of $\mathbf{C}$, compute

the cumulative sum of the eigenvalues and divide by the sum of all eigenvalues, i.e.,

$$\frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{N} \lambda_i} \geq P\,. \tag{6}$$

Here, k is the number of eigenvalues required to reach or exceed the threshold P, which is also the number of clusters used to partition the network.

For both methods above, we recompute the desired number of clusters every time we construct the mean-reverting portfolio, and use a twenty-day lookback window on stock returns to determine the number of clusters. As a benchmark, we also include performance results when the number of clusters is fixed to 30.[3]

2.4 Benchmarks and Evaluation Criterion

We compare the performance of each cluster-driven portfolio with two benchmarks. The first benchmark is the SPY ETF. The second benchmark is an arbitrage portfolio based on the Fama–French 12 industry classifications, which is constructed by building statistical arbitrage portfolios within each of the Fama–French 12 industries in the same way as our cluster-driven portfolios. The second benchmark compares intra-cluster mean-reversion effects to the mean-reversion effect discovered by the cluster-driven portfolios.

To evaluate the performance of our portfolios, we use metrics including annualized return, Sharpe ratio, and Sortino ratio Sortino (1994). The Sharpe ratio measures the risk-adjusted return. It considers the market residual return generated per unit of standard deviation of returns as a measure of risk. A higher Sharpe ratio indicates better risk-adjusted performance, i.e., a higher return relative to the amount of risk taken.[4]

On the other hand, the Sortino ratio focuses on the downside risk of the portfolio with the intuition that high upside standard deviation does not negatively impact portfolios and is not a concern

[3]The dynamic algorithms pick 10 to 20 sectors in most days; we choose 30 clusters to differentiate from the number that we choose dynamically.

[4]For convenience, in this paper, the risk-free rate is set to zero.

for investors. It takes into account only the standard deviation of negative returns, providing a measure of risk-adjusted returns targeting the downside volatility, and is defined as

$$\text{Sortino Ratio} = \frac{\text{Portfolio Return} - \text{Risk-Free Rate}}{\text{Downside Deviation}}. \tag{7}$$

Here, the downside deviation represents the standard deviation of negative returns. A higher Sortino ratio implies better risk-adjusted performance because it indicates higher returns relative to the downside volatility of the portfolio.

3 Empirical Results

3.1 Data

Stock price data are from the Center of Research in Security Prices (CRSP) daily returns database WRD (2023). The sample period is from January 2000 to December 2022. We include stocks listed on the NYSE, Amex, and NASDAQ exchanges. For each trading day, to ensure that the trading positions we take are realistic, we only include stocks in the top 25 percentile of market capitalization, which is defined as the product of the price of stock at the end of the day (i.e., close price) and the number of shares outstanding. The stock universe we include consists of around 600 stocks in each trading day. We use close prices adjusted for splits and dividends when computing forward-looking returns.

We compare the performance of our portfolios built with various clustering algorithms with a portfolio built with industry classification data. The industry classification information maps each firm's SIC code to a single, non-overlapping Fama–French 12 industries sector label. The industries are nondurables (1), durables (2), manufacturing (3), energy (4), chemicals (5), business equipment (6), telecommunications (7), utilities (8), shops (9), healthcare (10), finance (11), and other (12).

3.2 Performance of Portfolios

In this subsection, we investigate if the performance of the portfolios constructed with clustering algorithms is economically significant. The number of days used to estimate the number of clusters is 20 days, and the rolling window we use to estimate β and to compute the market residual return is 60 days. We set $w = 5$ days for the number of lookback days to construct the correlation matrix and to compute the cluster mean returns; the rebalance period to re-compute the correlation matrix, re-compute the clusters, and re-balance the portfolios is $\ell = 3$ days. The threshold to identify whether a stock is a previous winner or is a previous loser is $p = 0$, and we set the threshold $q = 5\%$ to consider that the portfolio mean-reverted. When we do not dynamically change the number of clusters, we fix the number of clusters to $K = 30$.

Table 1: Performances of statistical arbitrage portfolios with various clustering algorithms

	MP			90% Eigen			Fixed K		
Model	AR	SR	ST	AR	SR	ST	AR	SR	ST
SPONGE	10.99	1.02	1.81	11.90	1.07	1.89	10.21	1.01	1.80
SPONGE$_{sym}$	**12.05**	**1.11**	**2.01**	**12.20**	**1.10**	**2.01**	10.40	**1.03**	**1.80**
Spec	10.96	1.03	1.82	10.84	0.98	1.75	10.03	0.99	1.72
Lap$_{sym}$	11.19	0.91	1.60	11.24	0.88	1.55	**11.10**	0.97	1.66
Lap$_{rw}$	10.38	0.85	1.47	11.26	0.90	1.56	10.95	0.96	1.64
FF12	-	-	-	-	-	-	10.13	1.08	1.90
SPY	-	-	-	-	-	-	6.59	0.32	0.50

Table 1 presents the performance of the statistical arbitrage portfolios constructed with various clustering algorithms. The tabs "MP", "90% Eigen", and "Fixed K" represent methods for dynamically determining the number of clusters. For "MP", we use the number of eigenvalues that exceed the upper boundary λ^+ of the Marchenko–Pastur distribution. This evaluation is performed on the correlation matrix derived from the returns matrix of dimension T by N. For "90% Eigen", the number of largest eigenvectors of the correlation matrix required to account for 90% of the total variance is the number of clusters we extract. For "Fixed K", we compute 30 clusters. In the table, "AR" is annualized return, "SR" is Sharpe ratio, and "ST" is Sortino ratio.

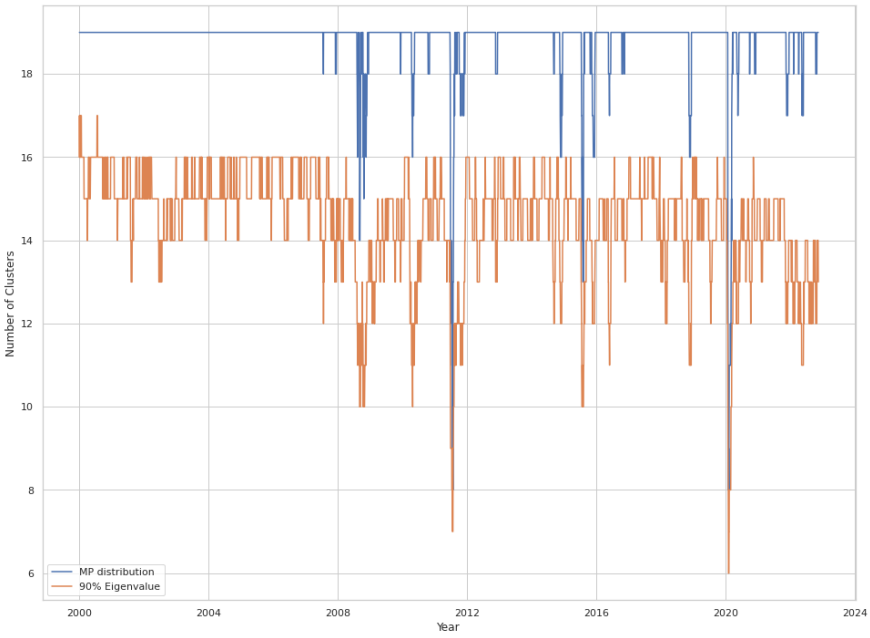

Figure 2: Historical number of clusters chosen by the Marchenko–Pastur distribution and total variance explained methods.

Figure 2 presents the number of clusters chosen by various methods. For "MP distribution", we use the number of eigenvalues that exceed the upper boundary λ^+ of the Marchenko–Pastur distribution. For "90% Eigen", we use the number of largest eigenvectors of the correlation matrix required to account for 90% of the total variance. Both methods provide relatively stable number of clusters, but both methods undergo drops in the number of clusters they find during financial hardships of the United States. For example, both methods experience a large drop in number of clusters they find during the 2008 financial crisis, August 2011 when U.S. credit rating was downgraded for the first time in history, and COVID in 2020. This observation shows that the methods that dynamically determine the number of clusters can capture changes in market dynamics, especially when there is significant downside risks in the market.

The annualized returns of all portfolios are higher than 10%, where the $SPONGE_{sym}$ clustering portfolio delivers the highest overall performance in terms of annualized return, Sharpe ratio, and Sortino ratio. In particular, for the two methods where we dynamically determine the number of clusters, the $SPONGE_{sym}$ clustering portfolio has a Sortino ratio of 2.01. Overall, Our portfolios have similar Sharpe ratio and Sortino ratio as that of the Fama–French benchmark portfolio

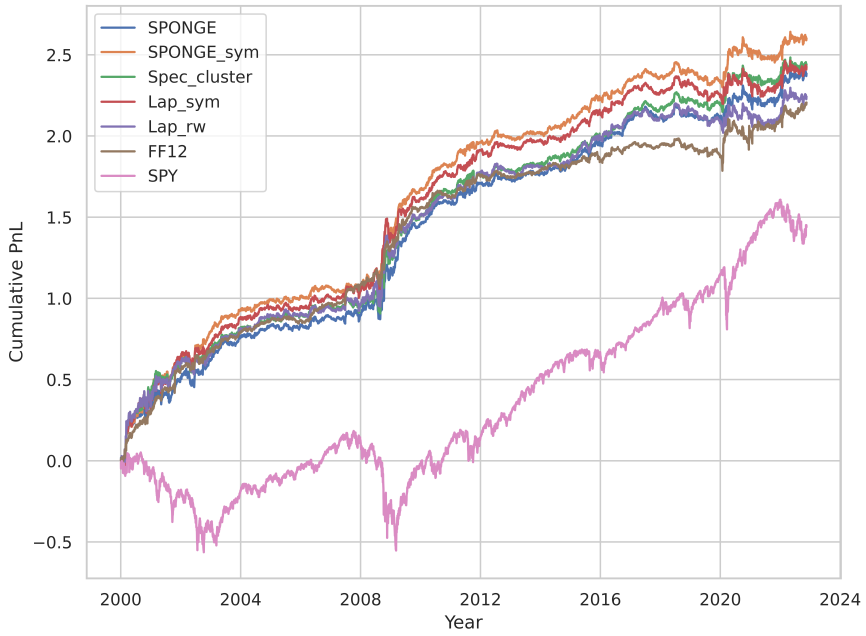

Figure 3: Cumulative returns of various strategies. The number of clusters is determined by the Marchenko–Pastur distribution. The cumulative returns are the sum of the daily returns without compounding.

that uses the Fama–French 12 sectors.

We test the statistical significance of the Sharpe ratios obtained by the strategies, see Bailey and López de Prado (2014). Specifically, the Sharpe ratios of all arbitrage strategies are statistically significant at the 0.01% confidence level, while the Sharpe ratio of the SPY is not statistically significant at the 10% confidence level.

The performance of strategies in Table 1 is similar across various choices of number of clusters; thus, choosing the number of clusters dynamically does not significantly change the performance of portfolios. This observation lends strong support to the idea that the construction of arbitrage portfolios within each cluster is robust to the number of clusters.

Figure 3 shows the cumulative sum of returns of various strategies. In the long term, the statistical arbitrage portfolios tend to perform better when the volatility of the SPY portfolio is large (e.g., during the 2008 financial crisis). The performance of the Fama–French 12 sector portfolio is similar to that of the clustering-driven portfolios up until 2008, and the returns of the clustering-driven portfolios are higher than that of the Fama–French 12 sector portfolio after 2008.

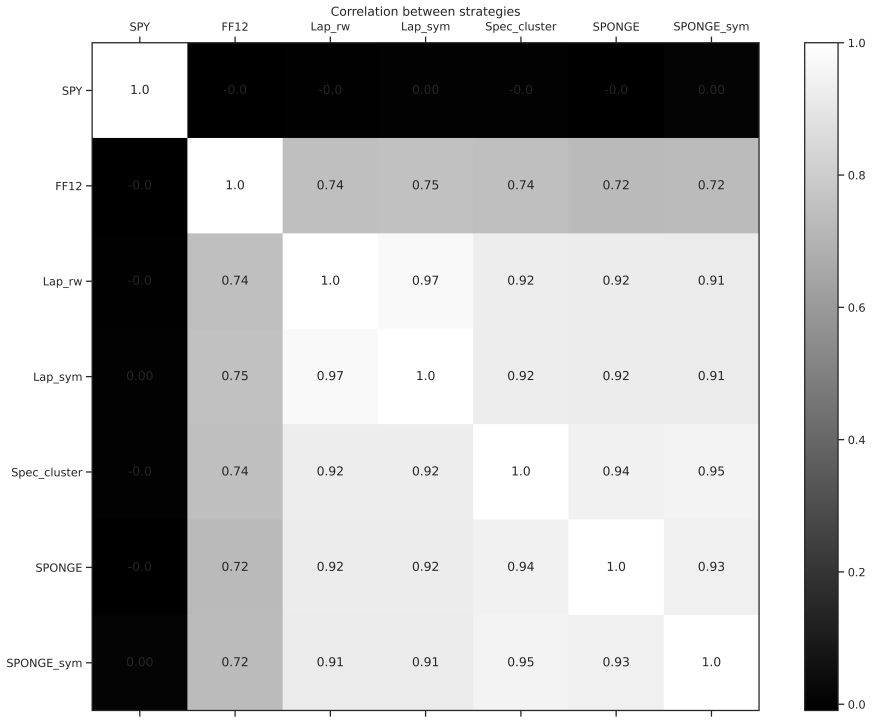

Figure 4: Correlation between returns of various strategies. The number of clusters of the clustering portfolios is determined by the Marchenko–Pastur distribution. Correlation coefficients are the Pearson correlation between the returns of strategies.

Figure 4 shows the correlation between the returns of the strategies we use to construct the portfolios. The correlation between all statistical arbitrage portfolios and the SPY is close to zero, which confirms that the arbitrage portfolios are market neutral. When the clusters are computed with data-driven algorithms, the correlations among strategies are very high; this illustrates that the mean-reverting patterns that the clusters detect are similar. On the other hand, the correlation between the data-driven clustering portfolios and the Fama–French sector portfolio is much lower. The lower correlations with the Fama–French sector portfolio show that the performances of the clustering arbitrage portfolios cannot be fully explained by the underlying intra-sector relationships of the securities.

Figure 5 compares the clusters detected with SPONGE clustering and the stocks of the underlying Fama–French 12 industries. From 2019 to 2022, there are 377 stocks that are traded on every trading day. The SPONGE algorithm detects clusters that have large overlap with the following sectors: Utilities, Energy, Business Equipment, and Healthcare. Other clusters detected by the

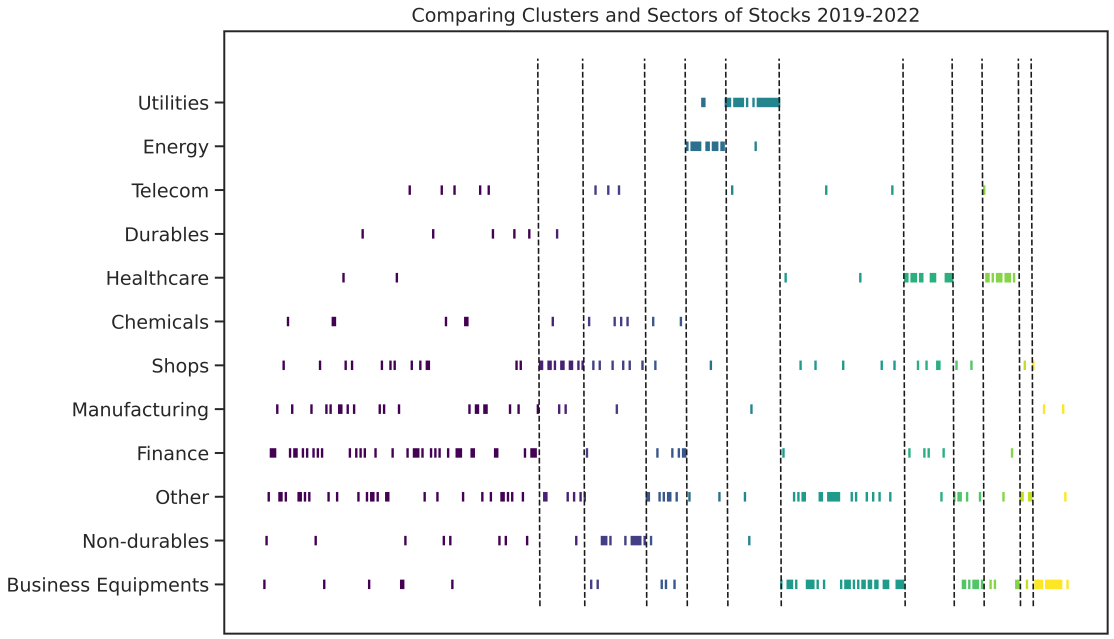

Figure 5: Comparison between the clusters created with the SPONGE algorithm on the correlation matrix of stocks to detect 12 clusters and the underlying Fama–French sector labels from 1 January 2019 to 31 December 2022. The area between black vertical dashes represents each cluster formed with SPONGE clustering. There are 377 stocks that are traded every day in this time period.

SPONGE algorithm do not show strong alignment with any particular sectors; in particular, there is a large cluster that the SPONGE algorithm detects which contains stocks from all sectors except for Utilities and Energy. This observation further supports that the performance of the clustering statistical arbitrage portfolios cannot be fully explained by intra-industry mean-reversion behavior.

Table 2: Adjusted Rand Index between Clusters from various algorithms and the Fama–French 12 Sector labels from 1 January 2019 to 31 December 2022.

	ARI(%)
SPONGE	14.9
SPONGE$_{sym}$	13.5
Spectral	15.2
Laplacian$_{rw}$	14.5
Laplacian$_{sym}$	13.2

Finally, we use the Adjusted Rand Index (ARI) to measure the similarity between the clusters we detect using clustering algorithms and the Fama–French 12 sector labels, see Table 2. The ARI between the clusters of the algorithms and the Fama–French 12 sector labels is low. The similarity

between clusters found by the clustering algorithms and the sector labels is less than 15%. This observation supports the observation that the returns of our statistical arbitrage portfolios cannot be fully explained by industry memberships. Our portfolio discovers new mean-reversion patterns among various partitions of stocks.

4 Conclusion

In this paper, we presented a novel framework for constructing statistical arbitrage portfolios that uses state-of-the-art graph clustering algorithms. Our empirical results demonstrated that our approach generates economically significant, profitable portfolios. In our study with historical data, we also showed that our framework is robust to the choice of number of clusters and the choice of clustering algorithms. Our study fills a gap in the literature by exploring the potential of clustering methods to create profitable statistical arbitrage strategies and by applying signed clustering algorithms to financial time series analysis. Our framework serves as a new evaluation criterion to assess if a clustering algorithm can accurately group stocks into clusters of similar returns.

Several graph clustering algorithms were introduced in the last decades, but the number of downstream tasks which employs the recovered clusters is limited, especially in a financial context. Our work opens further lines of investigation. For example, one can explore clustering and statistical arbitrage for cross-asset correlation matrices, or one can instead use other matrices as inputs to our framework (e.g., matrix of co-movement upon reacting to events). Exploring a variant of our framework on higher-frequency data (e.g., intraday minutely returns) could potentially also lead to interesting findings and profitable trading strategies with significant economic benefits.

References

2023. Wharton Research Data Services. https://wrds-www.wharton.upenn.edu/

Saeed Aghabozorgi, Ali Seyed Shirkhorshidi, and Teh Ying Wah. 2015. Time-series clustering–A decade review. *Information Systems* 53 (2015), 16–38.

David Bailey and Marcos López de Prado. 2014. The Deflated Sharpe Ratio: Correcting for Selection Bias, Backtest Overfitting, and Non-Normality. *The Journal of Portfolio Management* 40 (09 2014), 94–107.

Philippe Bergault, Fayçal Drissi, and Olivier Guéant. 2022. Multi-asset optimal execution and statistical arbitrage strategies under Ornstein-Uhlenbeck dynamics. arXiv:2103.13773 [q-fin.TR]

William K. Bertram. 2010. Analytic solutions for optimal statistical arbitrage trading. *Physica A: Statistical Mechanics and its Applications* 389, 11 (2010), 2234–2243. https://doi.org/10.1016/j.physa.2010.01.045

Álvaro Cartea and Sebastian Jaimungal. 2016. Algorithmic Trading of Co-integrated Assets. *International Journal of Theoretical and Applied Finance* 19, 06 (2016), 1650038. https://doi.org/10.1142/S0219024916500382

Álvaro Cartea, Luhui Gan, and Sebastian Jaimungal. 2019. Trading co-integrated assets with price impact. *Mathematical Finance* 29, 2 (2019), 542–567. https://doi.org/10.1111/mafi.12181 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/mafi.12181

Mihai Cucuringu, Peter Davies, Aldo Glielmo, and Hemant Tyagi. 2019. SPONGE: A generalized eigenproblem for clustering signed networks. *AISTATS* (2019).

Mihai Cucuringu, Apoorv Vikram Singh, Deborah Sulem, and Hemant Tyagi. 2021. Regularized spectral methods for clustering signed networks. *Journal of Machine Learning Research* 22, 264 (2021), 1–79. http://jmlr.org/papers/v22/20-1289.html

Robert J. Elliott, John Van Der Hoek, and William P. Malcolm. 2005. Pairs trading. *Quantitative Finance* 5, 3 (2005), 271–276. https://doi.org/10.1080/14697680500149370 arXiv:https://doi.org/10.1080/14697680500149370

Eugene F. Fama and Kenneth R. French. 2004. The Capital Asset Pricing Model: Theory and Evidence. *Journal of Economic Perspectives* 18, 3 (Summer 2004), 25–46.

Sergio M Focardi. 2005. Clustering economic and financial time series: Exploring the existence of stable correlation conditions. *The Intertek Group* (2005).

Federico Gatta, Carmela Iorio, Diletta Chiaro, Fabio Giampaolo, and Salvatore Cuomo. 2023. Statistical arbitrage in the stock markets by the means of multiple time horizons clustering. *Neural Computing and Applications* 35 (02 2023), 1–19. https://doi.org/10.1007/s00521-023-08313-6

A. Knyazev. 2001. Toward the Optimal Preconditioned Eigensolver: Locally Optimal Block Preconditioned Conjugate Gradient Method. *SIAM Journal on Scientific Computing* 23, 2 (2001), 517–541. https://doi.org/10.1137/S1064827500366124

Andrew Knyazev. 2017. Signed Laplacian for spectral clustering revisited. (01 2017).

Jérôme Kunegis, Stephan Schmidt, Andreas Lommatzsch, Jürgen Lerner, Ernesto W. De Luca, and Sahin Albayrak. 2010. Spectral analysis of signed graphs for clustering, prediction and visualization. *SDM* 10 (2010).

Diego León, Arbey Aragón, Javier Sandoval, Germán Hernández, Andrés Arévalo, and Jaime Niño. 2017. Clustering algorithms for Risk-Adjusted Portfolio Construction. *Procedia Computer Science* 108 (2017), 1334–1343. https://doi.org/10.1016/j.procs.2017.05.185 International Conference on Computational Science, ICCS 2017, 12-14 June 2017, Zurich, Switzerland.

Harry Markowitz. 1952. Portfolio Selection. *The Journal of Finance* 7, 1 (1952), 77–91. https://doi.org/10.1111/j.1540-6261.1952.tb01525.x arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1540-6261.1952.tb01525.x

Andrew Y Ng, Michael I Jordan, Yair Weiss, et al. 2002. On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems* 2 (2002), 849–856.

Nicos G Pavlidis, Vassilis P Plagianakos, Dimitris K Tasoulis, and Michael N Vrahatis. 2006. Fi-

nancial forecasting through unsupervised clustering and neural networks. *Operational Research* 6, 2 (2006), 103–127.

Price Lee N. Sortino, Frank A. 1994. Performance Measurement in a Downside Risk Framework. https://doi.org/10.3905/joi.3.3.59

Wenpin Tang, Xiao Xu, and Xun Yu Zhou. 2021. Asset Selection via Correlation Blockmodel Clustering. arXiv:2103.14506 [q-fin.PM]

Vincenzo Tola, Fabrizio Lillo, Mauro Gallegati, and Rosario N. Mantegna. 2008. Cluster analysis for portfolio optimization. *Journal of Economic Dynamics and Control* 32, 1 (2008), 235–258. https://doi.org/10.1016/j.jedc.2007.01.034 Applications of statistical physics in economics and finance.

Seda Tolun Tayalı. 2020. A novel backtesting methodology for clustering in mean–variance portfolio optimization. *Knowledge-Based Systems* 209 (2020), 106454. https://doi.org/10.1016/j.knosys.2020.106454

Kaizheng Wang, Xiao Xu, and Xun Yu Zhou. 2022. Variable Clustering via Distributionally Robust Nodewise Regression. arXiv:2212.07944 [cs.LG]

Hartmut Ziegler, Marco Jenny, Tino Gruse, and Daniel A Keim. 2010. Visual market sector analysis for financial time series data. In *Visual Analytics Science and Technology (VAST), 2010 IEEE Symposium on*. IEEE, 83–90.