Cluster-based approaches for cryptocurrency portfolio optimization

Robin Jaccard
EPFL
robin.jaccard@epfl.ch

Jeremy Di Dio
EPFL
jeremy.didio@epfl.ch

January 28, 2024

Abstract

This paper conducts various filtering methods on the cross-correlation matrix of cryptocurrency returns, aiming to construct an optimal investment portfolio in the dynamic landscape of digital assets. Building on Markowitz's work, we focus on improving the estimation of correlation matrices, considering the challenges posed by empirical cross-correlation matrices in real financial markets. To this end, we explore diverse filtering techniques, including Random Matrix Theory, Thresholding, Minimum Spanning Tree, and Planar Maximally Filtered Graph. Additionally, correlation-based clustering is incorporated to identify asset groups with correlated movements, enabling a well-balanced risk exposure. Finally, Principal Component Analysis (PCA) is employed on each cluster to identify *leading coins* that can be used to create a global minimum variance portfolio. Using hourly log-returns of 269 cryptocurrencies from the period 01/01/2021 to 31/12/2022, we assess the stability of identified clusters and conduct a comparative analysis of out-of-sample risks associated with the different filtering methods.

1 Introduction

In recent years, cryptocurrencies as an investment option have witnessed a remarkable surge. Since the introduction of Bitcoin's open-source implementation in 2009, the cryptocurrency investment landscape has been quickly evolving and has been marked by notable rises and falls within short timeframes. The cryptocurrency ecosystem have reached an estimated 500 million users [1]. Instead of investing in conventional investment avenues like stocks and shares, some investors are drawn to the unique technologies or the potential for significant returns offered by the myriad of cryptocurrencies available. Faced with this expansive range of options, an investor must choose wisely to optimize their returns. Throughout this paper, cryptocurrency returns are analyzed with the aim of constructing an optimal investment portfolio.

The challenge of portfolio optimization stands as one of the most important concerns in asset management. Building on the work of Markowitz (1959) [2], which addressed the problem within a set of simplifying assumptions. The present study aims to center its attention on the estimation of the correlation matrix in portfolio optimization.

It is crucial to highlight that empirical cross-correlation matrices inherently face limitations as they assume temporally stationary and linearly interdependent time series. These assumptions are often breached in real financial markets. The process of estimating the correlation matrix itself is unavoidably associated with statistical uncertainties, which are due to the finite length of the asset return time series. However, cross-correlations remain the most widely used measure as filtering procedures can be applied to mitigate those issues. Indeed, the act of filtering can reduce uncertainties surrounding future returns. Our objective is to ensure that the realized risks closely align with the in-sample risk minimized during our calibration window.

In this research, we explore the impact of various filtering techniques on the sensitivity of the portfolio optimization process when applied to the correlation matrix. Our investigation includes filtering methodologies using Random Matrix Theory (RMT), Thresholding, Minimum Spanning Tree, and Planar Maximally Filtered Graph.

Additionally, we incorporate correlation-based clustering procedures into the analysis. The process of clustering aids in the identification of asset groups that exhibit correlated movements or share similar risk profiles. By diversifying within these clusters, the portfolio has the potential to attain a well-balanced risk exposure. It also has the advantage of showcasing greater stability over time when contrasted with the entire matrix.

Using the identified clusters, we conduct Principal Component Analysis (PCA) on each cluster to discern the *leading coins* within each group. Subsequently, we construct a global minimum variance portfolio by exclusively allocating investments to these identified *leading coins*.

To evaluate the effectiveness of the various filtering techniques, we analyze the hourly log-returns of 269 different cryptocurrencies from 2021 to 2023. Our first step involves assessing the stability of the identified clusters to estimate their robustness. Subsequently, we compare the out-of-sample risks associated with each method.

The structure of this paper unfolds as follows: In Section 2, we introduce the dataset. Section 3 outlines the methodologies and details the filtering algorithms employed for portfolio optimization. Subsequently, in Section 4, we evaluate the performance of the different filtering procedures using out-of-sample metrics. Finally, Section 5 summarizes our findings, highlights encountered challenges and proposes potential enhancements.

2 The Dataset

2.1 Data Selection

In this research, the focus was made on the cryptocurrency market, a domain less thoroughly explored compared to other financial markets. The primary step involved gathering data on cryptocurrency prices.

1. **Asset Selection**: The initial task in this study was selecting specific cryptocurrencies. A snapshot of the market capitalization rankings within the cryptocurrency industry was taken as of December 2, 2023. This method yielded a list of the most significant cryptocurrencies at that time. The analysis makes use of the top 500 assets by market capitalization.

2. **Period and Resolution Selection**: The next step in this study involved determining the appropriate period and data resolution. A three-year timeframe was chosen for the analysis. As for the resolution, the focus was on the most granular level available, which is hourly data. Consequently, hourly price data for the top 500 cryptocurrencies was collected from January 2, 2020, to January 2, 2023. This timeframe and resolution were partly determined by data availability.

2.2 Data Collection Process

The data collection phase was done using the CoinGecko API [3]. An important aspect to note is that while the data had an hourly resolution, the exact timestamps of the data entries were somewhat irregular. For instance, the recorded price of a particular asset at what was nominally 8am might be timestamped at 8:24am. As a result, each asset had a slightly different timing for its data points. To address this inconsistency, a rounding to the nearest hour was performed on the data. This method, while practical, did have a potential drawback: the possible loss or merging of some data points where two entries fell within

the same rounded hour. Despite this, rounding was adopted as a necessary compromise to ensure dataset consistency.

2.3 Data Pre-processing

After the collection of raw data, the next phase was data pre-processing, a step essential to prepare the dataset for subsequent analysis. This process involved several key tasks, each aimed at refining and optimizing the data.

1. **Timeframe Refinement**: The initial step in this pre-processing phase was to adjust the timeframe of our data. The period was reduced to a two-year window, influenced by the fact that many selected assets did not exist in 2020. Therefore, the focus shifted to data from January 1, 2021 to December 31, 2022.

2. **Handling Missing Values**: To address missing values in the dataset, a two-step approach was implemented. Any asset with more than 5% of its data points missing was removed from the study. For the remaining assets, a straightforward imputation method was used: replacing missing values with the preceding available data point.

3. **Calculating Hourly Log-Returns**: The dataset was transformed from representing hourly prices to showing hourly returns.

   Initially, the simple return r at time t from the price p was computed using the formula:

   $$r_t = \frac{p_t - p_{t-1}}{p_{t-1}}$$

   In the following, we adopted log-returns, computed as follows:

   $$\ln(r_t + 1)$$

   This shift to log-returns provides certain advantages in terms of mathematical properties and statistical assumptions, improving the analytical robustness of the dataset.

4. **Normalization of Log-Returns**: The final step of this pre-processing phase involved the normalization of the log-returns. The objective of normalization is to adjust the values measured on different scales to a notionally common scale, without distorting the differences in the ranges of values or losing information. This is particularly important in this context, as the selected cryptocurrencies have different price volatilities.

   To normalize the log-returns, a standard scaling method was used. This method involves subtracting the mean log-return of each cryptocurrency and then dividing it by the standard deviation. Mathematically, this can be represented as:

   $$r_t^{\text{normalized}} = \frac{r_t - \mu_r}{\sigma_r}$$

   Where r_t is the log-return at time t, μ_r is the mean log-return, and σ_r is the standard deviation of the log-returns.

2.4 Composition of the Final Dataset

The resulting dataset, shaped by thorough pre-processing, comprises 12,729 hourly log-returns and includes data from 269 different cryptocurrencies. This diverse range captures a broad section of the market and provides a solid basis for further analysis.

3 Methodologies

3.1 Random matrix theory approach

Random Matrix Theory (RMT) emerged from Wigner's pioneering work [4], where he initially applied it to systems of interaction, particularly in the context of studying energy levels in nuclei. Subsequently, Laloux et al. [5] and Plerou et al. [6] extended Wigner's findings to financial markets, employing RMT to analyze cross-correlations within stock market returns. Today, RMT has become a widely adopted technique for cleaning the noise from correlation matrices.

The fundamental idea behind RMT involves comparing the correlation matrix, denoted as C, with the properties of a random matrix, referred to as R. Any deviation in the properties of C from the anticipated characteristics of a random matrix R suggests that the correlation matrix C contains meaningful information about genuine correlations.

Considering the symmetry of the covariance matrix, its eigenvalues are real, and the eigenvectors can be chosen to form an orthonormal basis, expressed as $\Sigma = Q\Lambda Q^T$, where $\Lambda = \text{diag}(\lambda_i)$ contains the eigenvalues, and Q is the square matrix with the i^{th} column as the eigenvector u_i.

According to RMT, when considering time series data of length T from a collection of N assets with returns modeled as independent Gaussian random variables characterized by a zero mean and variance $\sigma^2 = 1$, an interesting observation emerges. As N and T approach infinity with a fixed ratio $Q = T/N \geq 1$, the eigenvalue spectral density of the covariance matrix is given by

$$\rho(\lambda) = \begin{cases} \frac{Q}{2\pi\lambda}\sqrt{(\lambda_+ - \lambda)(\lambda - \lambda_-)}, & \text{if } \lambda_- \leq \lambda \leq \lambda_+ \\ 0, & \text{otherwise} \end{cases}$$

where $\lambda_-^+ = 1 + \frac{1}{Q} \pm 2\sqrt{1/Q}$.

When examining the eigenvalue distribution of the correlation matrix derived from cryptocurrency returns and comparing it with the Marchenko-Pastur distribution, Figure 1's left panel reveals a discrepancy. The empirical correlation matrix's eigenvalue spectrum does not align with the expected Marchenko-Pastur distribution. Possible explanations for this misalignment include non-normality in returns distribution with heavier tails or the presence of non-random structures in the returns data.

To eliminate the hypothesis that it comes from the distribution of the returns, we undertake an independent permutation of each asset's returns. Remarkably, the eigenvalues now conform closely to the Marchenko-Pastur distribution, demonstrating resilience even in the presence of fat-tailed return series. This can be seen in the middle panel of Figure 1.

Moreover, in the right panel of Figure 1, setting the observation period T to 720 (equivalent to one month of data) yields a better alignment between the Marchenko-Pastur distribution and the bulk of the eigenvalue spectrum. This observation suggests that adjusting the time scale refines the fit, emphasizing the influence of the chosen temporal window on the distribution characteristics.

The bulk of the eigenvalues of an empirical correlation matrix that falls in the range $[\lambda_-, \lambda_+]$ can be considered to be mostly due to random noise. Simultaneously, all eigenvalues above λ_+ are supposed to represent meaningful structure in the data. Hence, any empirical correlation matrix C can be decomposed as follows:

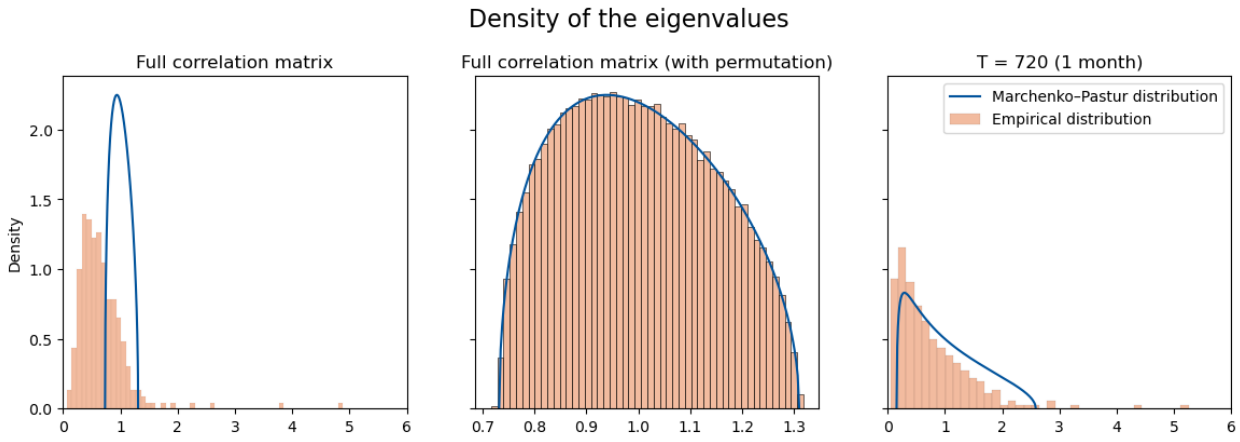

Figure 1: Panel 1 illustrates the empirical correlation matrix's eigenvalue spectrum computed on all the hourly returns from 01/01/2021 to 31/12/2022. Panel 2 showcases the same spectrum but the returns have been independently permuted (100x) for each assets. Finally, Panel 3 display the spectrum when only selecting the first 720 hourly returns. Note that, for clarity, the market mode has been excluded from Panels 1 and 3.

$$C = C^{(r)} + C^{(g)} + C^{(m)}$$

$C^{(r)} = \sum_{k:\lambda_k \leq \lambda_+} \lambda_k u_k u_k^T$ can be seen as the "random" component of the empirical correlation matrix. $C^{(m)} = \lambda_m u_m u_m^T$ where λ_m is the largest observed eigenvalue (much larger than all the others), is often called the "market mode". It represents the way the market is mostly moving. Finally, $C^{(g)} = \sum_{k:\lambda_+ < \lambda_k < \lambda_m} \lambda_k u_k u_k^T$ is what remains from the empirical correlation matrix. The correlations represented in $C^{(g)}$ do not manifest at the individual stock level, akin to uncorrelated noise, nor do they operate across the entirety of the market. Instead, these correlations operate within subsets or sub-groups of stocks within the market [7]. Note that the matrix $C^{(g)}$ is not necessarily a proper correlation matrix but this is not an issue for our problem.

3.2 Creation of Correlation-Based Networks

Community detection requires the construction of a weighted graph that accurately represents the relationships between the assets. This section introduces four distinct methodologies for graph construction, namely **Fully connected network**, **Mininum Spanning Tree (MST)**, **Threshold Network**, and **Planar Maximally Filtered Graph (PMFG)**.

These methods are all derived from the same foundational RMT filtered correlation matrix $C^{(g)}$, denoted by C in the following subsections. Figure 2 shows the resulting network for each of the aforementioned methods.

In the following sections, we briefly describe the three latter approaches for network creation.

3.2.1 Threshold Networks

A commonly employed technique for constructing a network from a correlation matrix involves incorporating a percentile threshold to filter out correlations below a certain strength. An edge is excluded from the network if the correlation between two cryptocurrencies does not rank within the top $p\%$ of the strongest correlations for either of those cryptocurrencies. In other words, only the most substantial

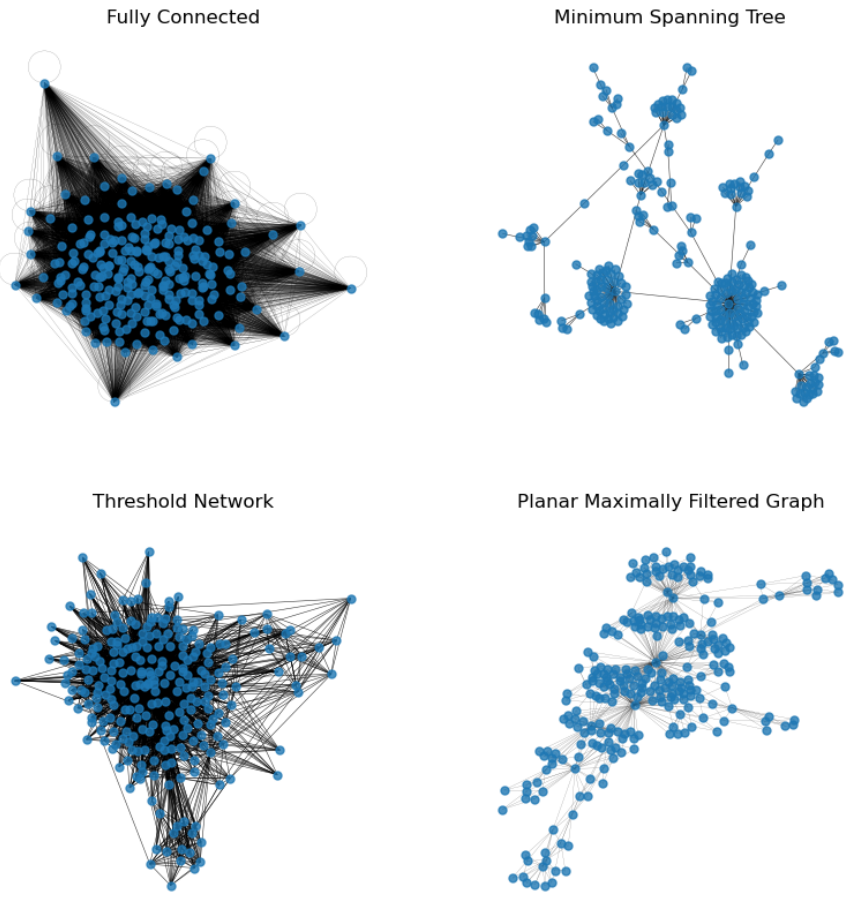

Figure 2: Visualization of the resulting network for each method

correlations are retained in the network. The edge weights are determined by the matrix C. This straightforward approach focuses on excluding the weakest correlations, resulting in the formation of a connected network.

Given its simplicity, the method's effectiveness depends on the careful selection of the percentile p. In this paper, a 5% threshold is applied. This method demonstrates robustness to noise by eliminating the weakest correlations, which are more susceptible to random fluctuations. However, it is essential to recognize that the use of an arbitrary threshold has limitations. This approach may obstruct the identification of communities characterized by internal correlations weaker than the strongest ones but still significantly stronger than external correlations with different communities.

3.2.2 Minimum Spanning Tree (MST)

Another widely accepted methodology for network creation from financial data is the application of the MST method [8]. This method involves the following steps:

1. **Creation of a Distance Matrix**: From the correlation matrix C, The distance matrix D is derived using the formula:

$$D = \sqrt{2(1 - C)}$$

This transforms correlation coefficients into distances.

2. **Graph Construction**: A fully connected graph G is constructed where each node represents an asset, and each edges are weighted based on the distances from matrix D.

3. **Computing the Minimum Spanning Tree**: The MST of G is then computed. The resultant structure G^{MST} includes all nodes (representing the N assets) connected by $N-1$ edges in a way that minimizes the total edge weight.

3.2.3 Planar Maximally Filtered Graph (PMFG)

The PMFG [9], enhances the traditional MST approach by introducing the capacity for a more complex network structure, yet maintaining the graph's planarity. In contrast to the MST, which is restricted to having $N-1$ edges for N nodes, the PMFG is designed to accommodate up to $3N-6$ edges. This is under the stipulation that the graph retains its planar nature, meaning it can be depicted on a plane without overlapping edges. This feature of the PMFG enables it to encapsulate more detailed than MST while containing the MST as a subgraph.

In this study, we first constructed the fully connected graph G from the correlation matrix C. Then, G^{PMFG} was computed from the graph G.

3.3 Communities Detection

Communities in networks are defined as groups of nodes more densely interconnected than one would typically expect under an appropriate null hypothesis. For the detection of these communities within the networks under consideration, the Louvain method [10] is applied.

3.3.1 The Louvain Algorithm

The Louvain algorithm discerns clusters within a graph G using the following process:

1. **Initialization**: Initially, each node in graph G is treated as an individual community. This forms the basis for a bottom-up approach in community detection.

2. **Local Optimization**: Iteratively, the algorithm explores moving nodes to adjacent communities, aiming to maximize the modularity gain—a metric assessing the density of links within communities relative to those between communities.

3. **Aggregation of Nodes**: Following the local optimization, communities are aggregated into single nodes, and the optimization steps are repeated.

4. **Termination**: The iterative process concludes when there is no further improvement in modularity, resulting in a final partition of the network into communities.

This method is particularly suited for large networks, thanks to its hierarchical optimization approach, which boosts computational efficiency and uncovers the layered complexity of the networks.

3.4 Leading Assets Identification

Once the community are detected, Principal Component Analysis (PCA) can be performed on each community in order to identify which asset explains the largest amount of variance. The cryptocurrency which explains the highest amount of variance within the first principal component of the returns data is considered as the *leading coin* of the community [11].

The most recurrent *leading coins* are Ethereum, Bitcoin and Chainlink. These cryptocurrencies hold prominent positions in the market capitalization rankings, reflecting their significance within the crypto space.

Bitcoin, recognized as the inaugural cryptocurrency, retains a central position, often leading the market. Ethereum distinguishes itself by its smart contract capabilities, making it a cornerstone in decentralized finance (DeFi) and diverse decentralized applications (DApps). Chainlink, while not a direct competitor to Bitcoin and Ethereum, is frequently clustered with the leading coins of 2017, including Litecoin, IOTA, Monero, and Stellar, all of which have experienced a decline in market capitalization rankings over time.

The fact that such important coins appear in the *leading coins* indicates that the method of using community detection and PCA could be useful for creating stock portfolios. Limiting the selection to the *leading coins* when optimizing a portfolio has the advantage of having a correlation matrix that is often less affected by statistical uncertainty and therefore more stable in time. The portfolio would still be diversified as we are investing in each cluster. Hence, portfolio optimization based on *leading coins* can potentially achieve a more balanced risk exposure.

3.5 Markowitz Portfolio Optimization

Markowitz portfolio optimization, introduced by Harry Markowitz in 1959 [2], addresses the challenge of constructing an optimal investment portfolio with N risky assets. The portfolio is determined by the weigths w_i ($i = 1, \ldots, N$) representing the fraction of wealth invested in each asset. The portfolio formed by those weights has the following average return and variance:

$$r_p = \sum_{i=0}^{N} w_i \mu_i,$$

$$\sigma_p^2 = \sum_{i=0}^{N} \sum_{j=0}^{N} w_i w_j \sigma_{ij}$$

where μ_i is the average return of asset i and σ_{ij} is the covariance between assets i and j. The goal is to minimize the portfolio volatility σ_p for a given level of return r_p. The optimization problem can be succinctly expressed in matrix notation as minimizing $\sigma_p^2 = w^T \Sigma w$, subject to the constraint $w^T 1 = 1$, where Σ is the covariance matrix. In the following, we focus our attention on the minimum variance portfolio. This allows us to limit the number of potential control parameters. The optimal weights for the minimum variance portfolio are derived using Lagrangian multipliers, resulting in $w_{opt} = \Sigma^{-1} 1 / (1^T \Sigma^{-1} 1)$. This methodology provides a quantitative framework for constructing portfolios that balance risk and return.

3.6 Rolling Window Method

To evaluate the stability and practicality of this community detection system, a rolling window approach was adopted. This method capitalizes on the temporal dimensionality of the data, aiming to uncover community structures across various time intervals. The essence of this approach is to iteratively analyze segments of the data over a defined period L, detecting communities within these segments.

The process begins by selecting a window of time L and conducting community detection for this initial segment. Following this, the window is shifted forward by an interval I, and the community detection process is reapplied to the new segment. This shift-and-cluster cycle continues, progressively moving through the dataset.

By employing this rolling window technique, the study benefits from a dynamic analysis of how communities evolve over time. It allows for the observation of patterns in community formation and dissolution, as well as shifts in asset relationships.

4 Results

In our investigation of the various correlation-based networks, we employed a rolling window method with a period $L = 720$ hours (equivalent to one month) and an interval of $S = 24$ hours. Within each time window, a systematic approach was applied:

- **Correlation Matrix Filtering with RMT:** The robustness of the correlation matrix is improved through the application of RMT.
- **Graph Creation Methods:** Following the RMT filtering, we explored different graph creation methods on the correlation matrix.
- **Graph Clustering with Louvain Algorithm:** The Louvain algorithm was then employed for graph clustering. To ensure a fair comparison across methods, the resolution parameter of the Louvain algorithm was carefully selected to achieve a similar number of clusters for each method. The distributions of the number of clusters for all the graph models are shown in Figure 3.
- **Identification of *Leading Coins* and Portfolio Optimization:** Within each cluster, we identified *leading coins* and performed portfolio optimization exclusively using these identified assets. The optimization process involved utilizing the unfiltered correlation matrix of these *leading coins*.
- **Out-Sample Risk Evaluation:** To assess the effectiveness of each strategy, out-sample risks are evaluated in the subsequent non-overlapping time window.

In the subsequent plots, the Threshold method is denoted by THRESH and the Fully Connected method is denoted by FULL.

4.1 Stability of the Methods

To evaluate the stability of identified clusters over different time periods, we employ the Adjusted Rand Index (ARI). The ARI is a widely used metric in cluster analysis that quantifies the similarity between two sets of clustering results, considering both the agreement and chance.

The Adjusted Rand Index is defined as:

$$\text{ARI} = \frac{\text{RI} - \text{Expected}_{\text{RI}}}{\max(\text{RI}) - \text{Expected}_{\text{RI}}}$$

where RI (Rand Index) measures the similarity between two clustering results, considering both correct and incorrect assignments. If a is the number of pairs of samples that are in the same cluster in both clustering 1 and clustering 2 and b is the number of pairs of samples that are in different clusters, then $RI = \frac{a+b}{\binom{n}{2}}$. $\text{Expected}_{\text{RI}}$ represents the expected similarity under a random clustering and max(RI) is the maximum value for RI.

The Adjusted Rand Index ranges from -0.5 to 1, where a higher ARI indicates a higher similarity between the compared clustering. If ARI=1, then we have the same clusters. If ARI=0, the clustering is as good as a random clustering.

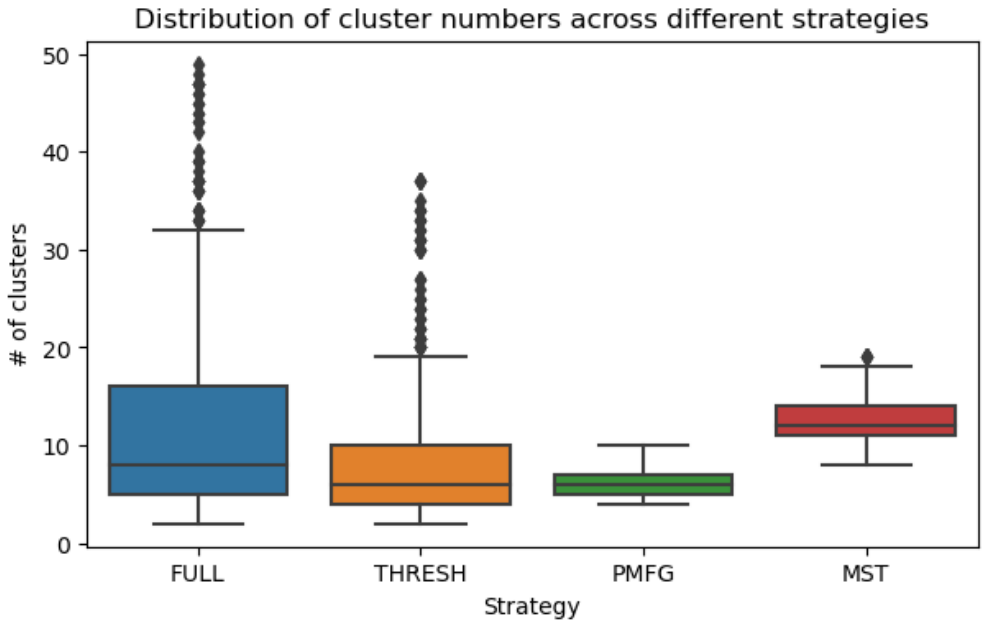

Figure 3: Distribution of the number of clusters obtained for each time window.

To assess the stability of our various methods, we calculate the Adjusted Rand Index (ARI) between consecutive non-overlapping time windows. This metric quantitatively measures the similarity or consistency in clustering results as we transition from one temporal segment to the next. The results are displayed in Figure 4.

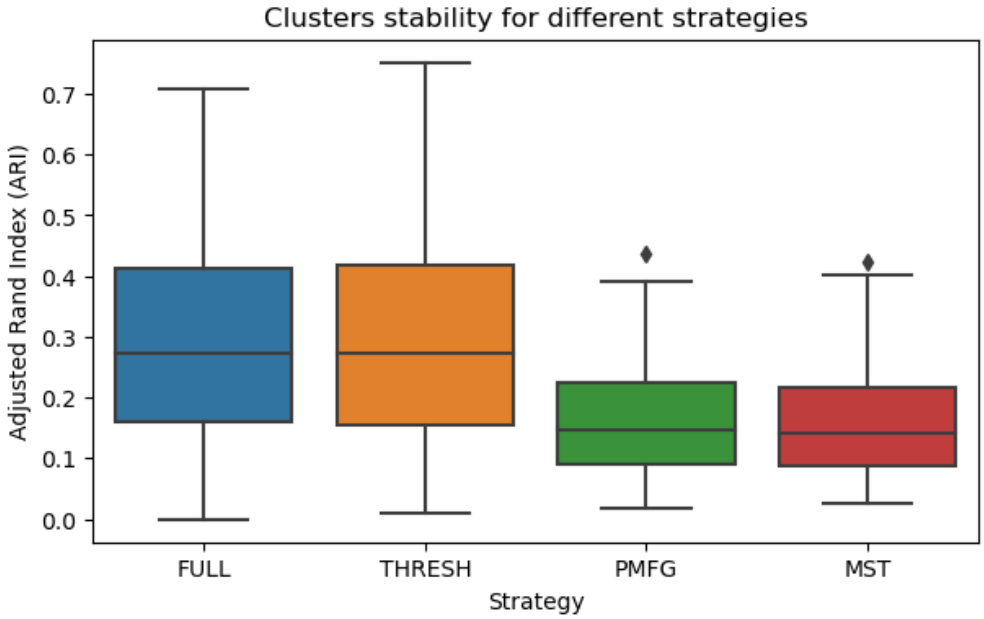

Figure 4: ARI scores for each strategy as a measure of similarity between clusterings, indicating the quality of clustering results across different strategies.

4.2 Risk Minimization

To build an optimal portfolio at a given time t based on the empirical $T \times N$ matrix (R) of past returns, one must choose the fraction of investment to invest in each coin. The component i of the vector w represents the fraction invested in the i^{th} coin. The risk of the portfolio can be measured by the standard deviation of the portfolio returns σ_p. If Σ represents the correlation of the past returns R, then $\sigma_p = \sqrt{w^T \Sigma w}$. To measure the performance of the clustering methods, we use the realized risk (out-of-sample)

$$\sigma_p^{\text{realized}} = \sqrt{w_*^T \Sigma^{\text{realized}} w_*}$$

where w_* represents the minimum variance weights derived from the Markowitz model utilizing the empirical in-sample covariance matrix and Σ^{realized} represents the covariance matrix of the next non-overlapping time-window. Our evaluation of performance, based on realized risk, relies on two metrics:

- **Relative Difference in Risks:** This metric compares the in-sample risk σ_p with the realized risk $\sigma_p^{\text{realized}}$. The relative absolute difference provides a measure of how closely the anticipated risk aligns with the actual realized risk [12].

$$\Re = \frac{|\sigma_p^{\text{realized}} - \sigma_p|}{\sigma_p}$$

  Figure 5 displays the distributions of $\Re$ for the different strategies.

- **Realized Risk:** The realized risk ($\sigma_p^{\text{realized}}$) independently serves as a metric, offering a direct assessment of the actual risk encountered during the investment period [13]. The distribution of the realized risks are presented in Figure 6.

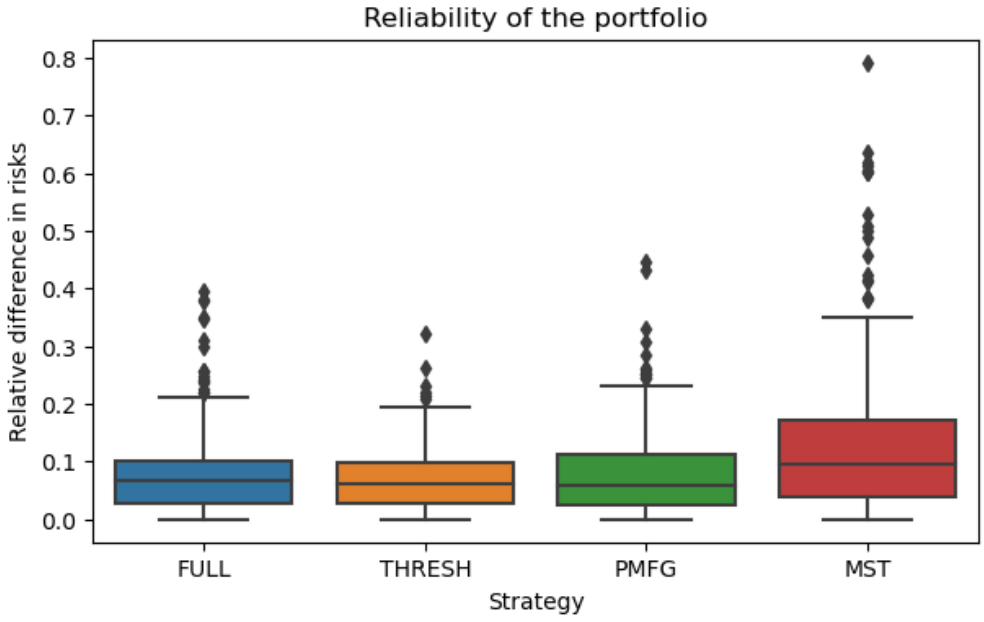

Figure 5: Analysis of relative differences in portfolio risks: The plot showcases the distribution of absolute relative differences between in-sample and out-of-sample portfolio risks.

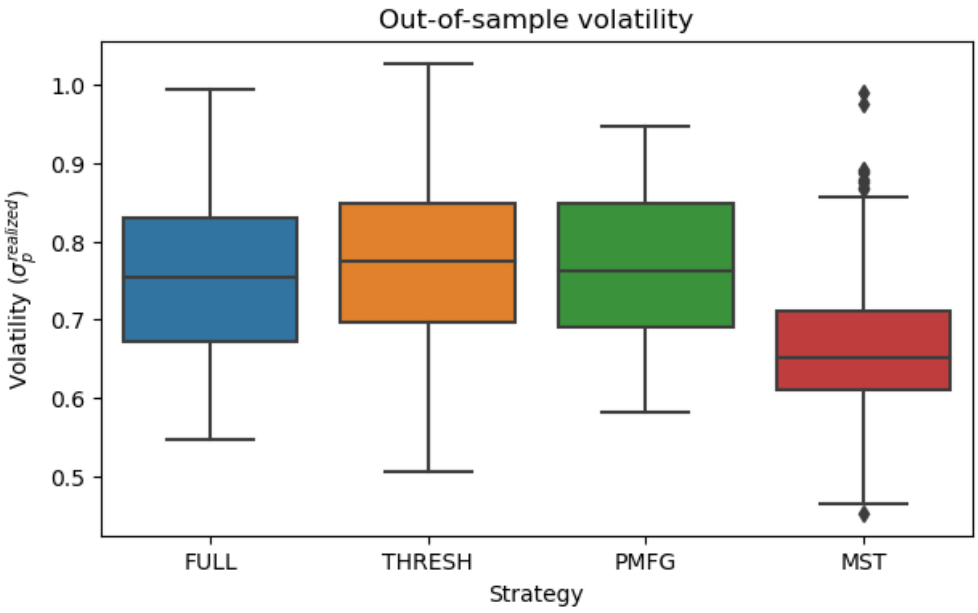

Figure 6: Comparison of realized portfolio risk across methods: The plot illustrates the distribution of the standard deviation of the optimal portfolio formed under each strategy.

These metrics collectively highlight the effectiveness of the selected portfolio weights, providing a comparative measure of how well the model's predictions align with the observed risks in real-world conditions.

5 Discussion & Conclusion

5.1 Results interpretation

The analysis reveals that both fully connected and threshold graphs demonstrate remarkable stability over time, indicated by their high Adjusted Rand Index. This initial sign of reliability is precious, as it suggests a reduced need for heavy rebalancing at the close of each month. Another indicator of stability is the smaller relative difference in risks, notably observed in the fully connected and threshold graphs. When examining realized volatility, MST graphs appear to outperform other strategies. Finally, despite their conceptual appeal, Planar Maximally Filtered graphs do not exhibit any distinct advantages over other strategies when considering the metrics employed in this study.

5.2 Challenges encountered

In this section, we discuss the various challenges encountered during this project.

5.2.1 Challenges with Computational Time

One of our primary challenges centered around data collection, a process detailed in Section 2.2. We had to deal with constraints while using the API, which allowed access to only one asset per request. Furthermore, the API imposed a maximum limit of 90 days of data per request for hourly granularity, and a time restriction of approximately 9 requests per minute. To avoid potential blocking, we maintained

a small margin on the number of requests per minute. Consequently, the entire data retrieval process required approximately 13 hours to complete.

Additionally, creating planar maximally filtered graphs posed another computational challenge. The task of computing all the clustering within the rolling time window required a substantial 15 hours of computation.

5.2.2 Cryptocurrency market dynamic

Another challenge encountered was the inconsistency of the data collected, primarily due to the dynamic nature of the cryptocurrency market. We observed that a considerable number of assets had a limited lifespan. This transient nature of assets posed a problem. Indeed, as new cryptocurrencies emerged and others faded, ensuring the completeness and accuracy of our dataset across time was difficult. This issue required us to exclude many assets as described in Section 2.3. This exclusion was necessary to ensure that our analysis remained relevant and reliable despite the market's frequent fluctuations.

5.3 Methodology discussion

In the analysis, several methods were employed: Random Matrix Theory (RMT), Graph creation, Louvain community detection, PCA and Markowitz's theory. It can be argued that only the different approaches to graph creation were examined for their impact on performance. One can rightly ask if the other methods employed are the optimal choices and if they fit well together. This decision is well supported by the established literature. Random Matrix Theory as a filtering procedure is a commonly used technique [7] [14] [6] [5]. RMT is valuable in identifying real correlations in financial data instead of mere noise, which is crucial in large and noisy datasets. The Louvain Community Detection algorithm is a significant tool in network analysis, particularly for identifying clusters or communities within large networks. It is also widely used for financial data [15] [11]. The PCA method to find *leading coins* is conceptually attractive and is efficient for cryptocurrencies as shown in [11]. Finally, Markowitz's portfolio theory is a classic of modern portfolio management, emphasizing diversification to balance risk and return [16] and the choice of using the global minimum variance portfolio is not a severe limitation [12].

In this project, adopting these methodologies likely provided a robust framework, enabling us to focus primarily on the methods of graph creation.

The application of different graph creation methodologies can lead to diverse outcomes in terms of cluster numbers. To mitigate the impact of this variability which directly affects the number of cryptocurrencies optimized through Markowitz, the Louvain parameters are fined-tuned to ensure similar distributions of cluster numbers across strategies. Figure 3 illustrates that, while the distributions exhibit similarities, they are not identical. A more in-depth investigation is required to assess whether these nuances influence our risk results.

5.4 Future work

In conclusion to our study, we express certain aspects of this project that would profit from additional analysis. Firstly, it would be beneficial to explore how varying the number of clusters impacts the results. This investigation could provide deeper insights into the data segmentation and its effects on the overall outcomes. Secondly, examining the effects of various clustering algorithms on our results could provide valuable insights. One could explore methodologies like hierarchical clustering [17], K-means++ clustering [18], or a density-based clustering algorithm like DBSCAN [19]. Thirdly, conducting thorough back-testing of the strategy would be helpful in further validating its effectiveness. Such analysis would offer a clearer understanding of the strategy's potential performance under different market conditions.

References

1. https://www.statista.com/statistics/1202503/global-cryptocurrency-user-base/.
2. Markowitz, H. Portfolio Selection. *The Journal of Finance* **7,** 77–91. ISSN: 00221082, 15406261. http://www.jstor.org/stable/2975974 (2024) (1952).
3. https://www.coingecko.com/en/api.
4. Wigner, E. P. in *The Collected Works of Eugene Paul Wigner: Part A: The Scientific Papers* (ed Wightman, A. S.) 409–440 (Springer Berlin Heidelberg, Berlin, Heidelberg, 1993). ISBN: 978-3-662-02781-3. https://doi.org/10.1007/978-3-662-02781-3_26.
5. Laloux, L., Cizeau, P., Bouchaud, J.-P. & Potters, M. Noise Dressing of Financial Correlation Matrices. *Phys. Rev. Lett.* **83,** 1467–1470. https://link.aps.org/doi/10.1103/PhysRevLett.83.1467 (7 Aug. 1999).
6. Plerou, V., Gopikrishnan, P., Rosenow, B., Nunes Amaral, L. A. & Stanley, H. E. Universal and Nonuniversal Properties of Cross Correlations in Financial Time Series. *Phys. Rev. Lett.* **83,** 1471–1474. https://link.aps.org/doi/10.1103/PhysRevLett.83.1471 (7 Aug. 1999).
7. MacMahon, M. & Garlaschelli, D. Community Detection for Correlation Matrices. *Physical Review X* **5.** ISSN: 2160-3308. http://dx.doi.org/10.1103/PhysRevX.5.021006 (Apr. 2015).
8. *Progress in Information Geometry: Theory and Applications* ISBN: 9783030654597. http://dx.doi.org/10.1007/978-3-030-65459-7 (Springer International Publishing, 2021).
9. Tumminello, M., Aste, T., Di Matteo, T. & Mantegna, R. N. A tool for filtering information in complex systems. *Proceedings of the National Academy of Sciences* **102,** 10421–10426. ISSN: 1091-6490. http://dx.doi.org/10.1073/pnas.0500298102 (July 2005).
10. Blondel, V. D., Guillaume, J.-L., Lambiotte, R. & Lefebvre, E. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* **2008,** P10008. ISSN: 1742-5468. http://dx.doi.org/10.1088/1742-5468/2008/10/P10008 (Oct. 2008).
11. Gavin, J. & Crane, M. *Community Detection in Cryptocurrencies with Potential Applications to Portfolio Diversification* Papers 2108.09763 (arXiv.org, Aug. 2021). https://ideas.repec.org/p/arx/papers/2108.09763.html.
12. Tola, V., Lillo, F., Gallegati, M. & Mantegna, R. N. Cluster analysis for portfolio optimization. *Journal of Economic Dynamics and Control* **32.** Applications of statistical physics in economics and finance, 235–258. ISSN: 0165-1889. https://www.sciencedirect.com/science/article/pii/S0165188907000462 (2008).
13. Bongiorno, C. & Challet, D. Covariance matrix filtering with bootstrapped hierarchies. *PLOS ONE* **16** (ed Sinatra, R.) e0245092. ISSN: 1932-6203. http://dx.doi.org/10.1371/journal.pone.0245092 (Jan. 2021).
14. Potters, M., Bouchaud, J. P. & Laloux, L. *Financial Applications of Random Matrix Theory: Old Laces and New Pieces* 2005. arXiv: physics/0507111 [physics.data-an].
15. Orman, G. K., Labatut, V. & Cherifi, H. Comparative Evaluation of Community Detection Algorithms: A Topological Approach. *CoRR* **abs/1206.4987.** arXiv: 1206.4987. http://arxiv.org/abs/1206.4987 (2012).

16. Bu, G. & Liu, Y. *Optimizing Returns of Diversified Investment Portfolio with Markowitz Model* in *Proceedings of the 2023 2nd International Conference on Economics, Smart Finance and Contemporary Trade (ESFCT 2023)* (Atlantis Press, 2023), 123–135. ISBN: 978-94-6463-268-2. https://doi.org/10.2991/978-94-6463-268-2_16.

17. Cecil C. Bridges, J. Hierarchical Cluster Analysis. *Psychological Reports* **18,** 851–854. eprint: https://doi.org/10.2466/pr0.1966.18.3.851.https://doi.org/10.2466/pr0.1966.18.3.851 (1966).

18. Arthur, D. & Vassilvitskii, S. *K-Means++: The Advantages of Careful Seeding* in *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms* (Society for Industrial and Applied Mathematics, New Orleans, Louisiana, 2007), 1027–1035. ISBN: 9780898716245.

19. Ester, M., Kriegel, H.-P., Sander, J. & Xu, X. *A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise* in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining* (AAAI Press, Portland, Oregon, 1996), 226–231.