MATH 232: Project Report

Unveiling the Geometry of Equity Markets: Exploring Dimension Reduction Techniques to Enhance Portfolio Diversification

Aryanna Holmes
Department of Mathematics
Yale University
aryanna.holmes@yale.edu

Helen Lyons
Department of Mathematics
Yale University
helen.lyons@yale.edu

Ian Richardson
Department of Electrical Engineering
Yale University
ian.richardson@yale.edu

Elder Veliz
Department of Statistics and Data Science
Yale University
elder.veliz@yale.edu

1 Abstract

Throughout this report, we explore the use of both linear and non-linear dimension reduction techniques and their potential contribution for enhancing portfolio diversification. Previous work has been largely based on PCA, but has not considered the potential of the non-linear methods we employ, such as UMAP, t-SNE, Isomap, Diffusion Maps, PHATE, and Laplacian Eigenmaps. Our data consists of daily stock prices from 30 equities across 6 different sectors: technology, consumer discretionary, industrials, healthcare, financials, and energy. Our results demonstrate that while PCA does provide a reliable method for clustering this kind of data, UMAP and Laplacian Eigenmaps also reveal non-linear relationships that may be useful for further research. There are significant benefits and drawbacks of each method: for example, UMAP excels in maintaining sector clusters — particularly relating to technology and energy — whereas Laplacian Eigenmaps effectively identify low-dimensional manifold structures within the data. Other assessed techniques such as t-SNE struggle with sector clustering due to the high dimensionality of our data relative to the small sample size. This analysis operates under the assumption that a "good" model will cluster sectors well, since there is significant intra-sector co-movement in market data. That being said, prominent idiosyncrasies and inter-sector affinities within market data are responsible for some models' imperfect clustering — which is *exactly* what we hope to one day quantify, by selecting a high-performing model and then understanding the ways in which it deviates from expectation. Overall, this paper contributes to financial analysis by illustrating how dimension reduction can uncover intricate stock correlations and clusters beyond traditional covariance-based methods, suggesting a refined approach to portfolio construction that incorporates both linear and nonlinear analytical strategies.

2 Introduction

2.1 Motivation

Modern portfolio theory, as first proposed by Harry Markowitz in his seminal paper "Portfolio Selection," has long provided investors with a one-size-fits-all golden method to optimize asset holdings. As part of this approach, investors calculate covariances between individual securities in order to determine optimal asset weightings and create a portfolio with the highest return for any given level of risk. This risk is traditionally quantified as the standard deviation of portfolio returns —

which is highly based on the covariances between individual assets.

$$f(x_a, x_b) = \sigma_p^2 = x_a^2\sigma_a^2 + x_b^2\sigma_b^2 + 2x_a x_b \sigma_{a,b}$$
$$x_y = Portfolio\, Weight\, of\ Asset\ y$$
$$\sigma_y^2 = Variance\, of\ Asset\ y$$
$$\sigma_{y,z} = Covariance\ Between\ Asset\ y\ and\ Asset\ z$$

As a result, existing work on portfolio theory has generally made use of covariance-based models to discover linear relationships between equity returns.

2.2 Our Focus

These methods may overlook potential non-linear relationships between stocks, a gap this project aims to address. By employing seven dimension reduction techniques, we seek to identify a series of sound model frameworks with the potential to offer new insights into portfolio diversification and optimization. We employ PCA as a linear dimensional reduction baseline, and 6 nonlinear dimension reduction techniques: Laplacian Eigenmaps, UMAPs, t-SNE, PHATE, Isomap, and Diffusion Maps. We will analyze the soundness of these models by understanding the extent to which they 1) preserve sector relationships, 2) are resistant to random Gaussian noise, and 3) are impacted by the chosen timescale 4) as well as by which stocks are included. We will also use 5 alternative stock metrics (5-day, 21-day, and 63-day moving averages, comparison to the S&P 500 returns, and daily percent change in trading notional) to determine if alternatives to daily returns may be more insightful for clustering purposes.

3 Background: Overview of Nonlinear Dimension Reduction Techniques

Below, we discuss the nonlinear reduction techniques we implemented. Note: we omit discussion of PCA, Laplacian Eigenmaps, and Diffusion Maps as they were discussed in lecture.

3.1 UMAP[1]

The mathematical basis for UMAP, Uniform Manifold Approximation and Projection, is mostly topological and beyond the scope of this course material, but will be described in simpler terms as follows. First, we assume that the given data is uniformly distributed on the data manifold. Typically, real data is not inherently distributed so uniformly, and thus a Riemannian metric must be found that makes it so that the data is approximately uniform over the manifold. Then, since we know the data is uniformly distributed, any neighborhood of some fixed size on the manifold should contain approximately the same number of data points regardless of which point the neighborhood is centered around. This is helpful in determining approximate geodesic distance (distance along the manifold as opposed to Euclidean distance; this is important as some points on certain manifolds may have a long geodesic distance despite having a short Euclidean distance, the classic example being the "Swiss roll").

If our neighborhood contains k points, we find the geodesic distance between some given point x and its neighbors by normalizing the distances between x and its neighbors "with respect to the distance to the k^{th} nearest neighbor of x" (McInnes et al. 2020). This process, however, creates problems with visualizing global structure as it is specifically intended to model the local structure of the data; this distance is independently defined for each neighborhood, thus lacking the consistency to properly model distances between different neighborhoods in the overall structure.

In order to make the individually defined distances compatible with one another, they must be converted to "fuzzy topological representations" using simplicial sets. Simplicial complexes "construct topological spaces by gluing together simple building blocks" (McInnes et al. 2020), and simplicial sets are sets consisting of these building blocks (simplices). As these sets contain the building blocks for a topological space, we may associate simplicial sets with specific topological spaces, and this

[1] McInnes et al., 2020.

will allow for the transformation from simplicial sets to fuzzy simplicial sets. In fuzzy simplicial sets, the members of the set are further characterized by their "membership strength" to the set, where membership of the set is continuous from 0 to 1 rather than discretized, and is determined by the point's normalized distance from the center of the neighborhood. This can be done for each of the neighborhoods described above. Then, by taking the union of these fuzzy sets, a singular fuzzy set is formed which better captures the global structure of our data, but without losing the local distance structure that was formed by the normalization above. Finally, the embedding of this data into low dimensional space can be optimized via use of stochastic gradient descent in order to optimize cross entropy between sets and minimize the error.

Computationally, this process can be explained using the language of graphs. The first part of the process is essentially creating a weighted graph on the k neighbors of x as we defined above. An algorithm is used to determine the nearest neighbor for each point, and this ensures that the constructed graph is connected since each point will be connected to at least its closest neighbor. Furthermore, the weights of the graph are the membership strengths mentioned above. The adjacency matrix for the graph is then determined *probabilistically*; for any two nodes in our graph, their corresponding matrix entry value is determined but the probability that there is an edge between them as determined by the "probabilistic t-conorm used in unioning the fuzzy simplicial sets" (McInnes et. al 2020). This graph is embedded in lower dimensions for visualization. Another algorithm is used to iteratively determine forces between edges and vertices until the overall structure of the data is optimized.

3.2 t-SNE[2]

t-SNE, or t-distributed Stochastic Neighbor Embedding, is another method of dimension reduction somewhat similar to UMAP. One of the drawbacks of using t-SNE for visualization is that it focuses primarily on preserving *local* structure, and thus tends to lose some of the global structure of the data, whilst typically taking longer than UMAP to compute.

For *stochastic* neighbor embedding, Euclidean distances between data points are found and converted into 'similarities,' "the conditional probability, $p_{j|i}$, that x_i would pick x_j as its neighbor if neighbors were picked in proportion to their probability density under a Gaussian centered at x_i." Then, an algorithm for stochastic neighbor embedding is used to minimize the 'Kullback-Leibler divergence,' essentially the difference between actual distance and calculated similarities via gradient descent. The difference between this and t-SNE is that, rather than using a Gaussian probability distribution, t-SNE uses a t-distribution with one degree of freedom to calculate these similarities. This is useful for visualization because the t-distribution helps avoid point overcrowding; in low dimensions, there is simply less room in which to put different data point clusters, and by using a tailed distribution to determine similarity, farther points are weighted higher so that the points can be spread out more without penalizing the distances of close and mid-range distance points. In practice, this makes it so that close points are kept close, but farther points that are more likely to belong to other clusters are mapped to a different area so as not to have many high dimensional clusters map to the same place in lower dimensional space. This results in very closely modeled clusters, but with the clusters being spaced out from one another.

3.3 Isomap[3]

A key feature of Isomap, or Isometric Feature Mapping, is that it is intended to show curvature from the data manifold, and thus focuses on preserving *geodesic* distance. Isomap begins by constructing a neighborhood graph on the data with Euclidean distance edge weights. Then, one of two methods is used to determine whether two points should be considered near one another; in the kNN method, you define some k and consider the k closest points to any point to be close points, and in the epsilon ball method, you define some distance epsilon and consider any points contained in a ball of radius epsilon around a given point to be close points. In either case, you can then algorithmically find the shortest path between all pairs of points (there are several algorithms that can be used to do this). These paths represent the geodesic distances between each pair of data points. Finally, a matrix is formed of the Euclidean distances between each pair of points, and multidimensional scaling is

[2]Hinton & van der Maaten, 2008.

[3]Tenenbaum et al., 2000.

performed using the top eigenvectors of the centered distance matrix—in our project, two dimensions for visualization purposes. This thus creates a low dimensional embedding of the high dimensional weighted graph while preserving the geodesic distances of the original data manifold.

3.4 PHATE[4]

PHATE, or Potential of Heat-diffusion for Affinity-based Transition Embedding, is another strategy specifically designed to mitigate issues like global distortion in t-SNE and overly noisy visualizations from Isomap. Like in several other strategies, the data in PHATE is modeled as a manifold. To determine the distribution of data over the manifold, the Euclidean distances between points are passed through an α-decay kernel, a kernel function which has exponentially decaying tails, to determine point similarities. We use this as opposed to the Gaussian kernel because the decay of the tails for the Gaussian kernel is too slow, thus too heavily weighting points that are somewhat far away from one another.

Furthermore, these similarities are converted to probabilities via diffusion, giving us the transition probability of our point to another in a random walk. The diffusion allows us to place lower weight on paths that are just created by noise in the data, thus giving higher weights to shorter, more efficient paths. Then, this information is used to calculate 'potential distance,' the Euclidean distance between the log transformed probabilities of two points. Lastly, the low dimensional embedding is created via MDS using potential distance as opposed to regular distance.

4 Methods

4.1 Data Preparation

We start by carefully selecting a pool of 30 equities across various six sectors (consumer discretionary, energy, financials, healthcare, industrials, and technology) and market capitalization levels. This diversity ensures our project encompasses a broad spectrum of market behaviors. We obtain 3 years (752 trading days from January 1^{st}, 2021 to January 1^{st}, 2024) of closing price data for these equities from Yahoo Finance, converting them into daily returns (the percentage change in the stock price from the previous day to the current day) to serve as our primary dataset. As is common practice for financial data analysis, we forward filled any missing stock prices (i.e., filled missing values in the time series with the last observed value).

[4]Moon et al., 2019.

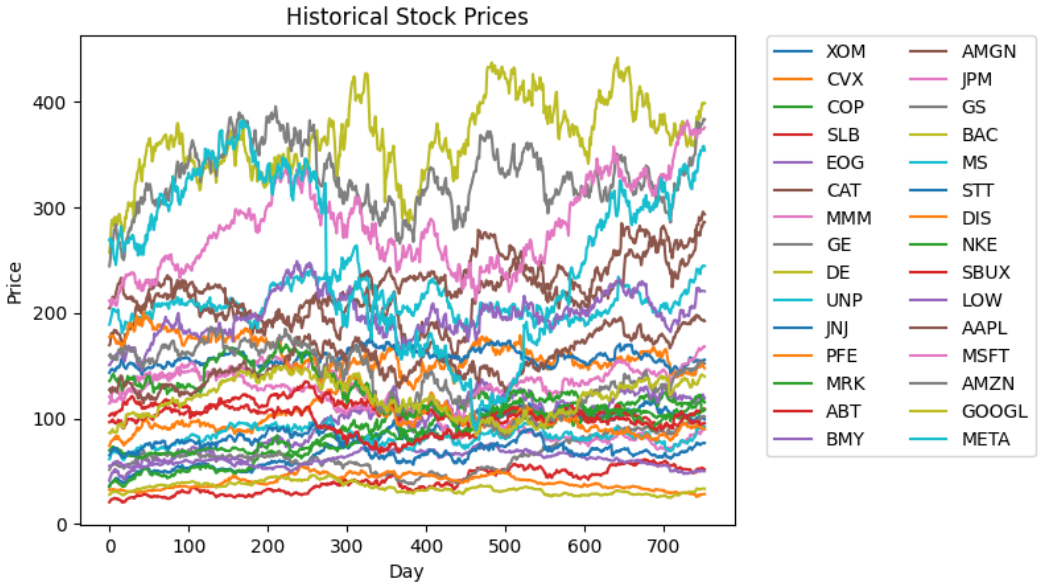

Figure 1: Visualization of Our Stock Prices over Time

We used the **daily returns** dataset to to derive additional metrics potentially useful for our analysis:

- We calculated the **5-day (1-week) moving average of the daily returns** to identify short-term trends and patterns in stock performance.
- We computed the **21-day (1-month) moving average of the daily returns** to analyze medium-term market behavior and volatility, reflecting typical monthly market cycles.
- To assess longer-term market trends and volatility, we calculated the **63-day (3-month) moving average of the daily returns**, corresponding to quarterly financial cycles.
- We measured the **percent change in the trading notional**, defined as the percent change in the price-weighted trading volume, to evaluate shifts in market activity and liquidity for each stock.
- Lastly, we calculated each stock's **daily returns comparison against the S&P 500 daily returns** to gauge performance relative to the broader market. Note: the corresponding embeddings will be inherently similar to daily returns by this construction.

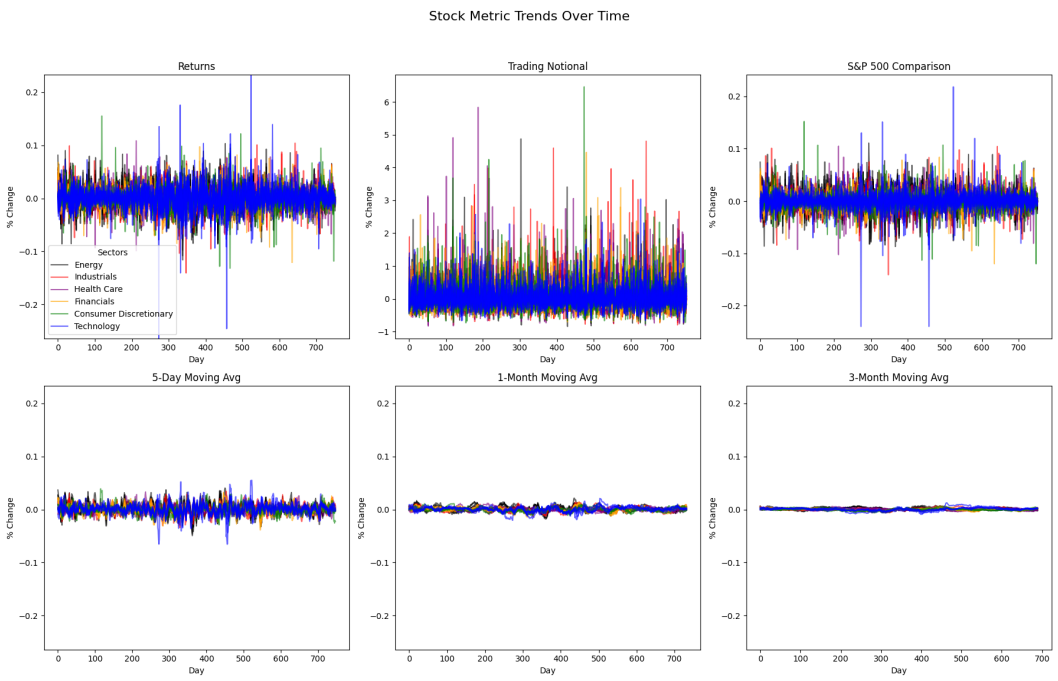

Figure 2: Visualization of Percent Change of Stock Metrics over Time (colored by sector)

In Figure 2, we observe that these stock metrics are roughly zero centered across time, with longer moving averages showing less fluctuation and greater smoothing as noise has been largely averaged out. We observe that the technology sector appears to have the most extreme presence of fluctuations for daily returns but most other sectors occasionally positively spike across the trading notional.

4.2 Nonlinear Dimension Reduction Implementation

We manually implement dimensionality reduction using Laplacian Eigenmaps in Python. In this model, each stock is depicted as a vertex in a graph, connected by edges that reflect the similarity between their returns. This similarity is quantified through a Gaussian kernel function applied to the Euclidean distance between our return vectors $\mathbf{r_1}$ and $\mathbf{r_2}$. The Gaussian kernel calculation is defined below:

$$d(x, y) = exp(-\frac{||\mathbf{r_1} - \mathbf{r_2}||^2}{2\sigma^2}) \tag{1}$$

Finally, we implement the five other nonlinear dimension reduction techniques (UMAP, t-SNE, PHATE, Diffusion Maps, and Isomap) with Python packages.

5 Results

5.1 Principal Components Analysis

As a basis of comparison, we began our analysis by dimension reducing our stock data with PCA, plotting the principal components to observe the explained variance:

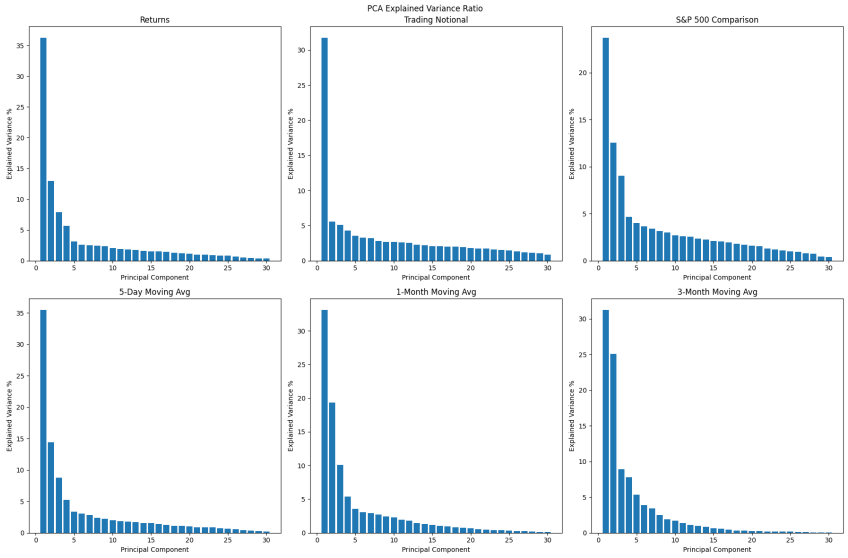

Figure 3: Visualization of Explained Variance Percentage by Principal Component

We observe that the first two principal components of each stock return metric explain a significant portion of the variance; thus, for visualization and consistency, we project our 30 high-dimensional vectors onto their first two principal components.

The projections from PCA served as a solid baseline, capturing the variance and allowing us to discern the distribution and dispersion of the data points.

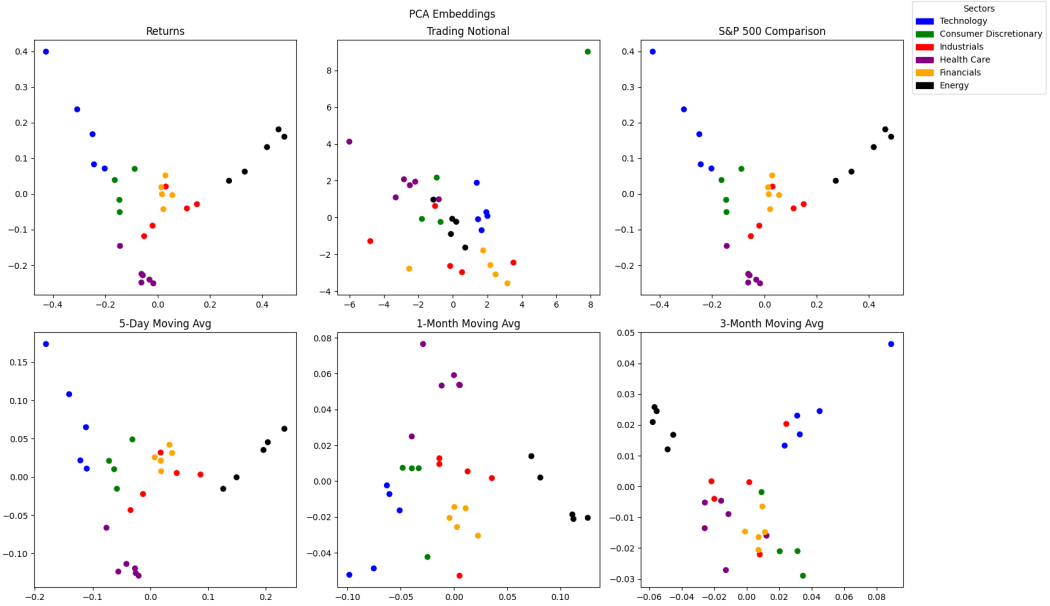

Figure 4: Visualization of 2D PCA Embeddings

We observe that PCA embeddings perform qualitatively well even in just two dimensions, showing very distinct sector-based clustering among our stocks when analyzed using daily, 5-day, and 1-month moving return averages and their relative performance to the S&P 500. Notably, industrial and financial stocks often cluster closely together, indicating a similarity in their market behaviors.

However, our other metrics such as trading notional values and 3-month moving averages seem less effective for clustering, as they show considerable overlap across different sectors. Despite these limitations, the clustering based on the previous metrics suggests PCA is a robust baseline for our analysis.

From here, we use Laplacian Eigenmaps and other nonlinear dimension reduction techniques to uncover any non-linear structure within the data. This will help offer a more nuanced understanding of stock relationships that are sensitive to the manifold's intrinsic geometry.

5.2 Laplacian Eigenmaps Analysis

Our main analysis concerns Laplacian Eigenmaps, since that is the most well-covered dimensionality reduction technique in the course. In projecting our high-dimensional data to a two-dimensional space (Figure 5), we use the two leading *nontrivial* eigenvectors of the RBF-kernelized adjacency matrix. This analysis corroborates the manifold hypothesis: our data neatly arranges itself into a 1-manifold within $\mathbb{R}^2$ — and even when we add a third dimension (Figure 6), the data still tends to lie in a 1-manifold in $\mathbb{R}^3$. The manifold's structure is clear as the second eigenvector (the first nontrivial one) distinguishes idiosyncratic outliers (such as *META*) from the rest of the stocks. The third eigenvector further discerns and roughly clusters the remaining stocks by sector. Note that that this applies to most of the stock metrics, except for trading notional, which seems to cluster all 30 data points into 3 groups. Given that the return-based metrics seem to yield similar-appearing embeddings, we defer to the more interpretable daily returns metric for further analysis.

In our three-dimensional model for daily returns (Figure 7), we observe that the fourth eigenvector performs a dual function: it refines the distinctions among idiosyncratic movements (as observed in stocks like *SLB* and *EOG*) and supports sector-based separation alongside the third eigenvector.

Upon examining the data situated on the one-dimensional manifold (Figure 8), we observe clear sector-based clusterings — energy stocks (denoted in black) often yield high ϕ_4 values and low ϕ_3 values, while technology stocks (indicated in blue) display the inverse pattern. Between these are the stocks from the industrial, healthcare, and consumer discretionary sectors (represented in red, purple, and green, respectively), which, despite a slight overlap, are largely differentiated by sector. It is noteworthy that in our three-dimensional visualization, *META* stands out with a very distinct ϕ_2 coefficient, rendering its position in this three-dimensional plane somewhat less relevant.

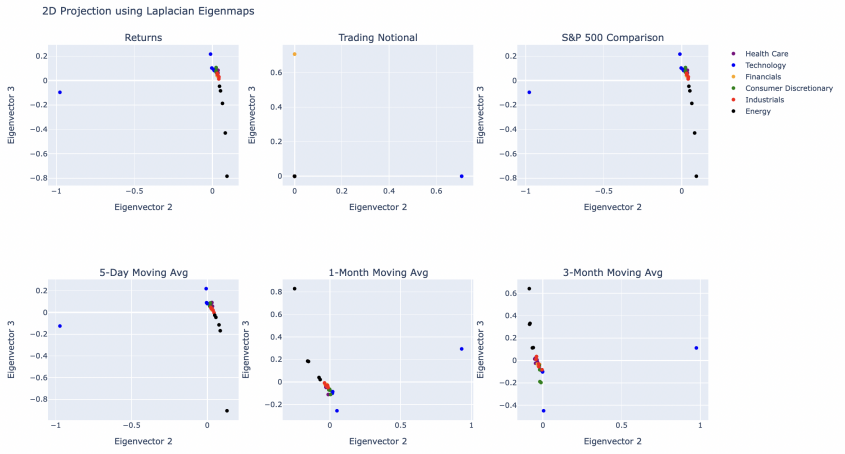

Figure 5: Visualization of the 2D Laplacian Eigenmaps

Figure 6: Visualization of the 3D Laplacian Eigenmaps

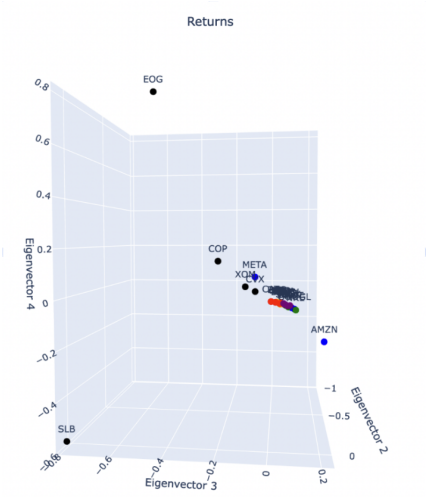

Figure 7: Visualization of the 3D Laplacian Eigenmap for Daily Returns

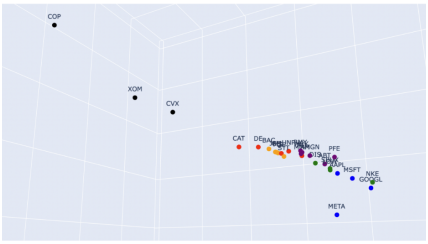

Figure 8: Zoomed-in Visualization of the 3D Laplacian Eigenmap for Daily Returns

5.3 Analysis of UMAP, t-SNE, Isomap, Diffusion Maps, and PHATE

Next, we analyze the five other nonlinear dimension reduction techniques. Upon building all of the different models outlined above, we can see that some of the techniques were much more successful in clustering and identifying trends in our data than others. For one, we can see that PCA and UMAP provide some of the best embeddings in terms of maintaining clusters. Given its popularity for this type of analysis, PCA's success is unsurprising. PCA reduces the data along its principal components (greatest variance); we expect related sectors and stocks to vary in similar respects. UMAP (Figure 9) also does a rather good job of maintaining clusters. These models work particularly well for clustering technology, healthcare, and energy stocks, which suggests these sectors behave similarly to each other and distinctly from other sectors in the market.

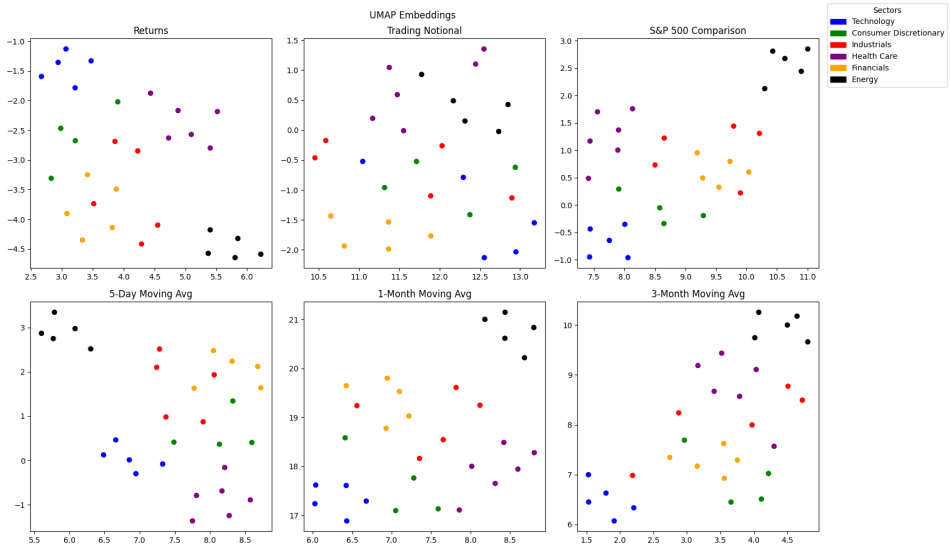

Figure 9: Visualization of the 2D UMAP Embeddings

The t-SNE embeddings (Figure 10) qualitatively yield the weakest sector-based clusterings. For returns, volume, and all moving averages, t-SNE gives us an approximately uniformly distributed embedding with no respect to sector clustering. This not only fails to accurately represent the overall structure of the data but also fails to represent the local structure via proper clustering. This is likely due to the large disparity between the number of stocks we used for analysis and the dimensionality of each one. t-SNE performs poorly when too few observations are used with respect to how many dimensions each observation has. Thus, this issue may be mitigated if further analysis repeated the experiment with a greater number of stocks.

The diffusion map embedding (Figure 11) appears to be using the first basis vector to separate out the most significant outliers—typically, in the technology or energy sectors—depending on the stock metric. As covered in lecture, this is expected: Diffusion Maps are not tuned for a two-dimensional embedding, as the higher eigenvectors may contain significant information that is omitted here.

To investigate this, we removed the first (non-constant) eigenvector and plotted the second eigenvector against the third (Figure 12). This revealed much better sector clustering in an interesting piecewise-linear fashion. Notice that energy stocks are embedded at (0,0)—nearly all their graph distance is captured in the first eigenvector alone. This embedding suggests Diffusion Maps perform better with the trading notional data, since this is less sensitive to fundamental idiosyncrasies and is more correlated with sector trends as a whole. Thus, it's less likely that the diffusion directions would separate so few points without differentiating between other sector-specific behavior.

While the PHATE (Figure 13) and Isomap (Figure 14) embedding techniques generally exhibited decent clustering for clearly distinct sectors—such as energy, technology, and healthcare—across most the stock metrics, these embeddings often failed to disentangle the remaining sectors. This corroborates the hypothesis that many of these sectors' return-derived metrics cannot be appropriately segregated by these high-dimensional non-linear dimension reduction techniques alone. Nevertheless,

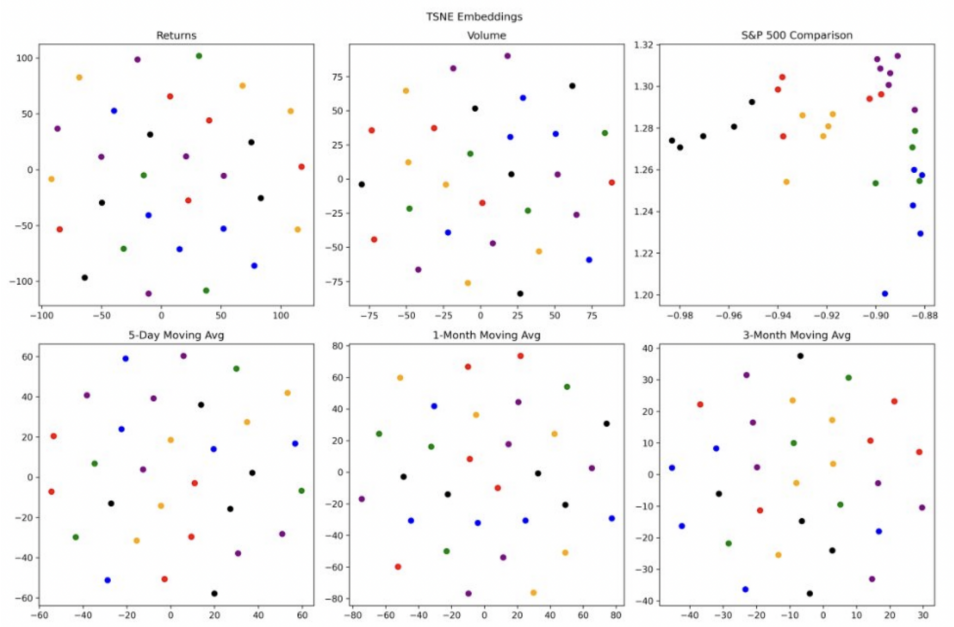

Figure 10: Visualization of the 2D t-SNE Embeddings

PHATE displays the 1-manifold behavior similar to the results of the Laplacian Eigenmaps, especially data that is inherently noisier, i.e. the daily and 5-day moving average returns. The Isomap embedding, on the other hand, embeds the daily returns and S&P 500 comparisons in an interesting piecewise-linear structure. Despite their fundamentally different representations, these embeddings are able to distinguish stocks corresponding to the technology, healthcare, and energy sectors solely using their daily returns.

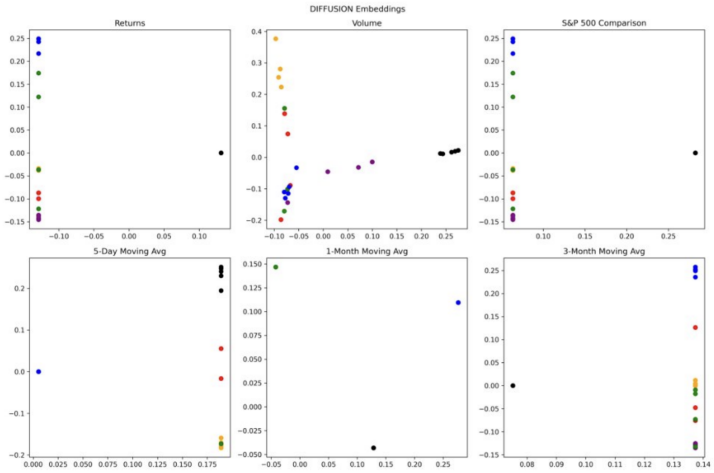

Figure 11: Visualization of the 2D Diffusion Embeddings

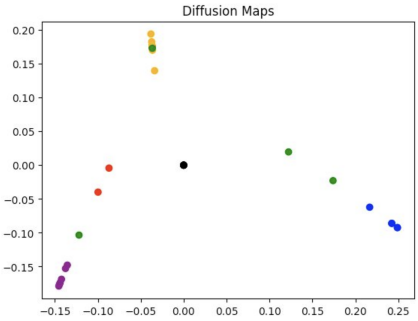

Figure 12: Diffusion Maps Analysis Excluding First Eigenvector

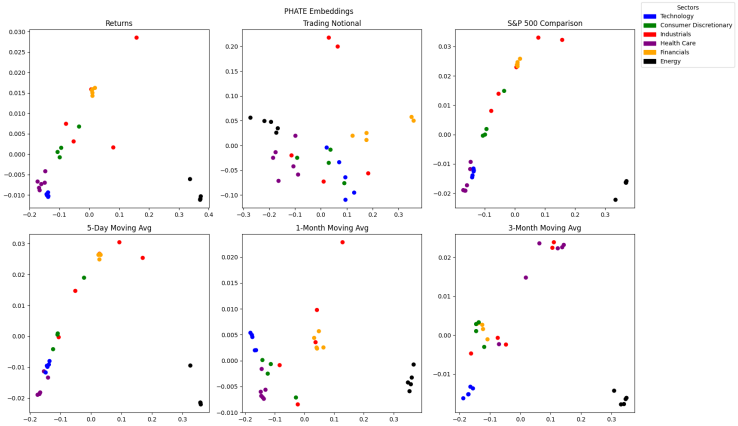

Figure 13: Visualization of the 2D Phate Embeddings

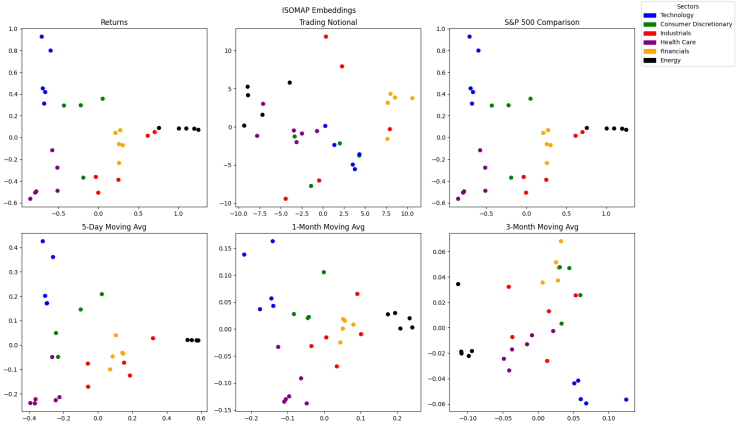

Figure 14: Visualization of the 2D Isomap Embeddings

6 Model Evaluation

We analyze the robustness of these models by understanding the extent to which they 1) are resistant to random Gaussian noise, 2) preserve sector relationships, and 3) are consistent in their clustering with respect to varying timescale.

6.1 Gaussian noise

First, we conduct a sensitivity analysis by injecting Gaussian noise into our returns data to determine our model's sensitivity to low-time-horizon idiosyncratic trends. We modify our representative daily returns matrix as follows:

$$R'_{ij} = R_{ij} \cdot (1 + \varepsilon_{ij}) \tag{2}$$

such that $\varepsilon_{ij} \sim N(0, \sigma)$.

For Laplacian Eigenmaps, we determine that a σ value of 0.8 preserves the local low-dimensional manifold geometry well, and that while higher values of σ distort the local geometry, they still preserve sector clustering (albeit less clearly.)

Note: Graph orientation is irrelevant; the only change is the sign of the eigenvectors.

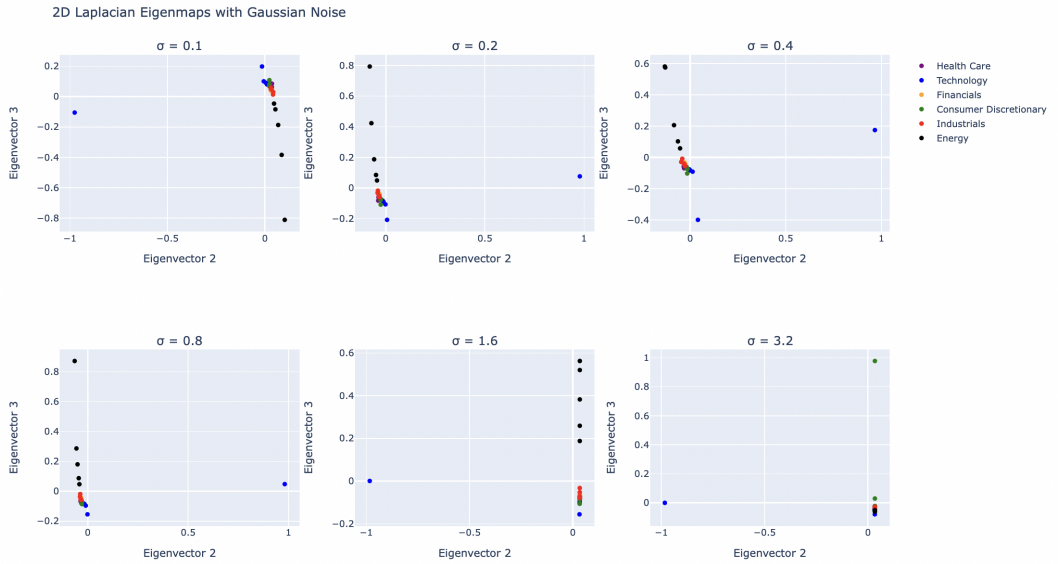

Figure 15: Gaussian Noise applied to 2D Laplacian Eigenmaps

From this analysis, we can conclude that the Laplacian Eigenmaps dimension reduction technique is robust to injected noise in the data, since the overall geometric manifold structure and relative clusters appear to be well-preserved.

6.2 Preservation of Sector Relationships

Second, we analyze the extent to which these models preserve sector relationships. Within all models (especially Laplacian Eigenmaps, UMAP, and PCA), three sectors generally cluster cleanly: technology, energy, and occassionally healthcare. The boundaries of the other sectors often overlap, but some techniques shed light in areas where others don't. For example, the trading notion Diffusion Map embeddings tend to cluster financials abnormally well. As explained previously, while the model as a whole overweights idiosyncrasies just like Laplacian Eigenmaps does, trading notional data is more correlated with sector trends and thus the model is able to pick out sectors rather effectively. Specifically, it makes sense that trading volume in financials is easily separable since bank crises (e.g. the regional bank crisis of March 2023) affect the financials sector as a whole.

6.2.1 Removal of Dominant Sectors

In almost every model analyzed, technology and energy stocks separated with significantly greater magnitude from the other sectors. It was hypothesized that significantly higher covariance in these sectors was adversely impacting the clustering of the other sectors. So, dominant stocks were

removed in groups to reveal the resulting impact on clustering. All analyses were done using Laplacian Eigenmaps on our daily returns dataset.

First, the largest outlier, *META*, was removed (Figure 16). With *META* previously taking up the entire first eigenvector of the Laplacian embedding, the clusters now spanned the first three non-constant eigenvectors—though their structure remained mostly the same.

Figure 16: Laplacian Eigenmaps Model with $META$ removed

Next, all technology stocks were removed (Figure 17). This created an embedding where energy stocks dominated, and all other stocks were clustered together, as seen in the following figure:

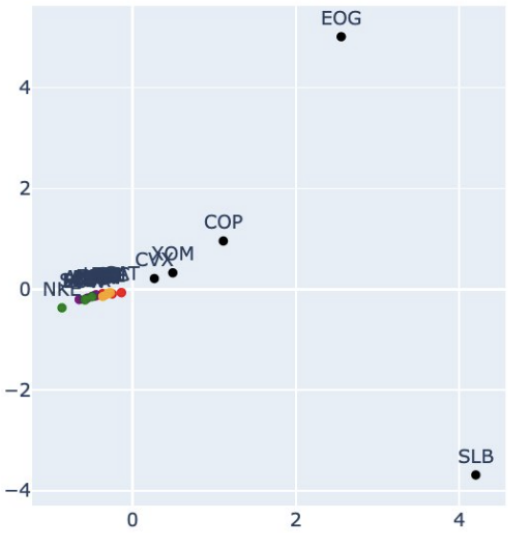

Figure 17: Laplacian Eigenmaps Model with Technology Stocks Removed

Finally, both technology and energy stocks were removed from the analysis (Figure 18). As hypothesized, the resulting embedding much better separated the remaining sectors. In particular, the industrial and financial sectors—previously correlated, display much more complex structure—where it seems that financial stocks form a much tighter cluster (with one outlier inside the larger span of the industrial stocks). Therefore, this supports the hypothesis that the much higher magnitude covariance of the technology and energy stocks had been negatively impacting the clustering of the other sectors.

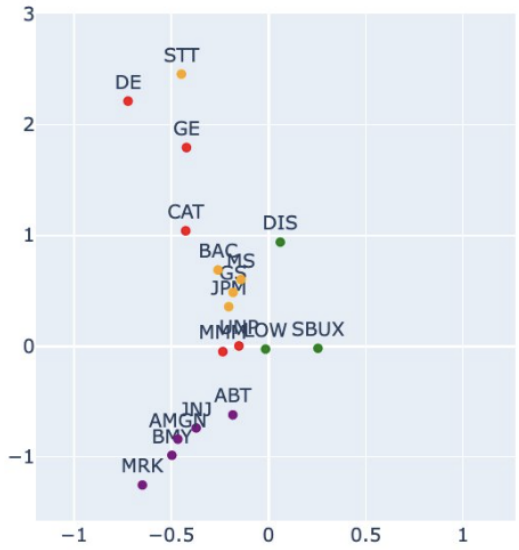

Figure 18: Laplacian Eigenmaps Model with Technology and Energy removed

6.3 Consistency Analysis

A consistency analysis was performed to determine the relationship between data dimensionality and sector clustering. This analysis had two primary goals: First, to determine what timescale is necessary to achieve optimal clustering; and second, to determine whether specific periods of time are biased toward clustering specific sectors better than others.

Addressing the former, the three years of market data were partitioned into six groups: consisting of the first three quarters and last three quarters of each year. Each partition was then independently analyzed using Laplacian Eigenmaps on the daily returns dataset, with the resulting embeddings shown in Figure 19. We observed that the embeddings based off of three quarters of data appeared similar to the 3-year analysis, while the single quarter embeddings were far less consistent and displayed spurious clusters between unexpected sectors. This indicates that our analysis is consistent, in that increasing the dimension of the data converges to a singular scheme of sector clusters.

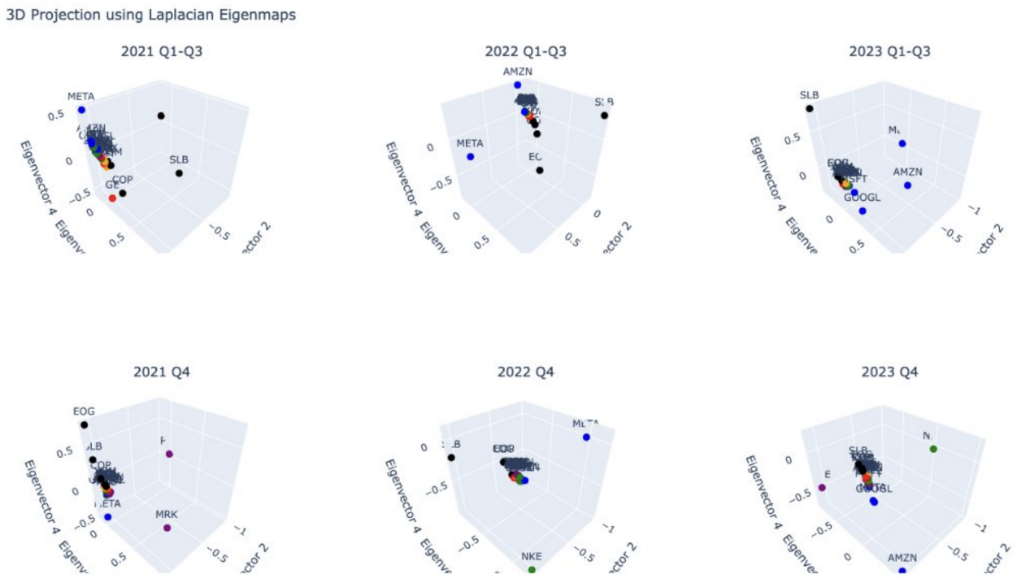

Figure 19: Laplacian Eigenmaps by Partitioned by Various Quarters

Addressing the latter, we next searched for temporal partitions in the data that impacted our sector clustering. The years 2021-2022 were analyzed separately from 2023 using the same method as above. As shown in Figure 20, 2023 separates out technology stocks better, while 2021-2022 separates energy stocks more clearly. From this result we can draw two conclusions. First, we show that our analysis is sensitive to the heuristic market trends that characterize these years (e.g. technology boom in 2023). Second, we find that a timescale longer than one year is necessary to reduce bias toward such short-scale trends in the market.

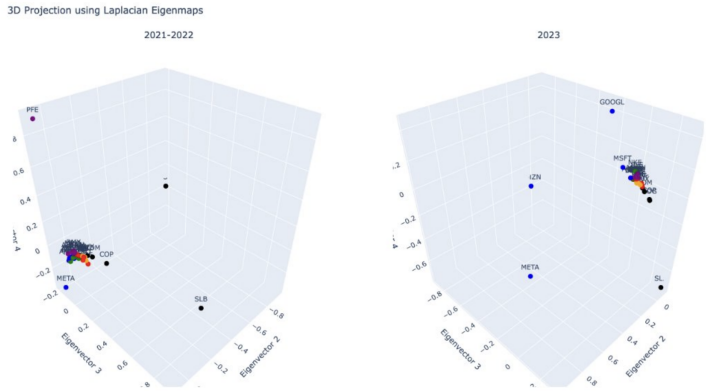

Figure 20: Laplacian Eigenmaps by Year

7 Conclusion

This research involved the creation of 42 models based on seven dimensionality reduction techniques and six stock-related metrics to explore potential diversification strategies beyond covariance-based methods. We find that while PCA works as a baseline for capturing linear covariance, non-linear techniques—such as UMAP and Laplacian Eigenmaps—can complement linear methods in revealing higher-order relationships within market data and reveal stocks exhibiting similar market behaviors.

Since the performance of these methods varied greatly across different datasets, it is important to consider a dataset's characteristics when selecting a dimension reduction technique. Specifically,

our project demonstrated the difficulty in using return-derived metrics to segregate stocks into their respective clusters using non-linear dimension reduction techniques alone.

Given the limited nature of our project, future research could include other features like market volatility, global trends, and economic indicators. The dataset could include more diverse sectors and larger sample sizes to enhance the generalizability of the findings. Since our project was limited to primarily exploring dimension reduction to two-space and comparing results qualitatively, future research with access to greater compute power could also involve larger-scale testing of performance results for techniques that require randomness. Quantitative performance scores were omitted from this project due to the inconsistency across stochastic dimension reduction techniques, however, these could be implemented for alternative comparison metrics.

Even in a simplified scenario with only 30 stocks, 6 sectors, and two-dimensional projections, the diversity in non-linear embeddings shows opportunity for future analyses that may select the best elements of several of these techniques to enhance portfolio diversification and uncover patterns in market behavior that go beyond traditional industry classifications.

References

[1] Chen, Guangliang. Isometric Feature Mapping (Isomap), n.d.

[2] Maaten, Laurens van der, and Geoffrey Hinton. Visualizing data using T-Sne. https://www.jmlr.org/papers/volume9/vandermaaten08a/vandermaaten08a.pdf.

[3] Markowitz, Harry. "Portfolio Selection." *The Journal of Finance* 7, no. 1 (March 1952): 77-91.

[4] McInnes, Leland, John Healy, and Nathaniel Saul. *UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction*, September 2, 2020.

[5] Moon, Kevin R., David van Dijk, Zheng Wang, Scott Gigante, Daniel B. Burkhardt, William S. Chen, Kristina Yim, et al. "Visualizing Structure and Transitions in High-Dimensional Biological Data." *Nature News*, December 3, 2019. https://www.nature.com/articles/s41587-019-0336-3citeas.

[6] Tenenbaum, Joshua B., Vin de Silva, and John C. Langford. "A Global Geometric Framework for Nonlinear Dimensionality Reduction." *Science* 290, no. 5500 (December 22, 2000): 2319–23. https://doi.org/10.1126/science.290.5500.2319.