S&P 500 Clustering

- **Team:** Assaf Bar-Natan, El Mehdi Ainasse, Nadir Hajouji
- **GitHub Repo link:** SP500-clustering.

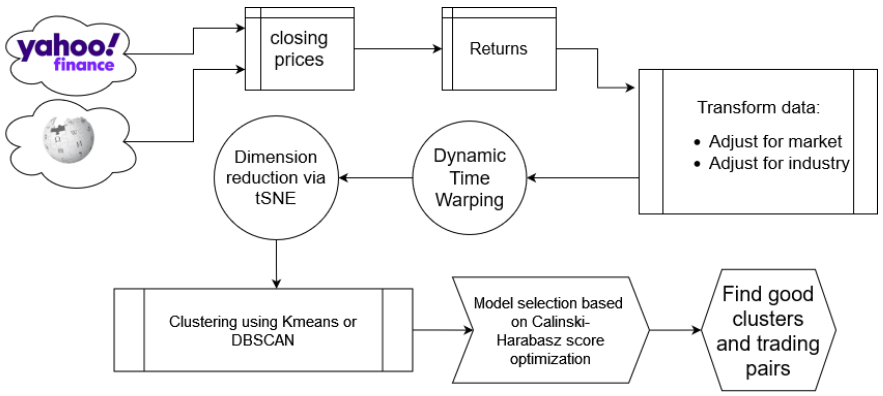
Overview: Portfolio optimization is one of the central problems of modern finance where one aims to maximize the expected return of a portfolio given a prescribed level of risk. This approach to investing relies on a careful and systematic selection of assets. We propose clustering as a systematic method of selecting assets.. Precisely, we seek to cluster the time series corresponding to these assets as our method of portfolio diversification. We will specifically consider the S&P 500 as our universe of assets. The cluster-oriented approach is fairly natural in this context: there are numerous non-quantitative selection methods, such as sector diversification, that effectively group the portfolio assets into clusters. The assets that are selected can subsequently be used as the universe of assets, serving as the basis of a trading strategy or portfolio optimization framework.

Stakeholders: Financial Analysts and Portfolio Managers; FinTech Companies; Regulatory Authorities; Retail Investors.

KPI: Calinski–Harabasz index.

Data Sourcing & Processing: Closing prices for all S&P 500 tickers (except those delisted) over the past two years, from end of November 2021 to end of November 2023, obtained from Yahoo Finance API (using the `yfinance` library). The tickers were scraped from Wikipedia. This allows us to download the data as a pandas dataframe for further use.

Approach:
- **Model Benchmark:** The 11 GICS sectors as the default clusters.
- **Feature Generation:** We first compute the returns of closing prices. We then perform market adjustment[1] and industry adjustment[2] on these returns.
- **Feature Engineering:** We compute the cross-similarity matrix of our adjusted returns using Dynamic Time Warping, which we then feed to t-SNE with 2 components as a means of dimensionality reduction.[3]
- **Modeling:** The reduced t-SNE-based features are fed to either KMeans or DBSCAN for clustering, and the optimal model is the one that maximizes the CH-score.



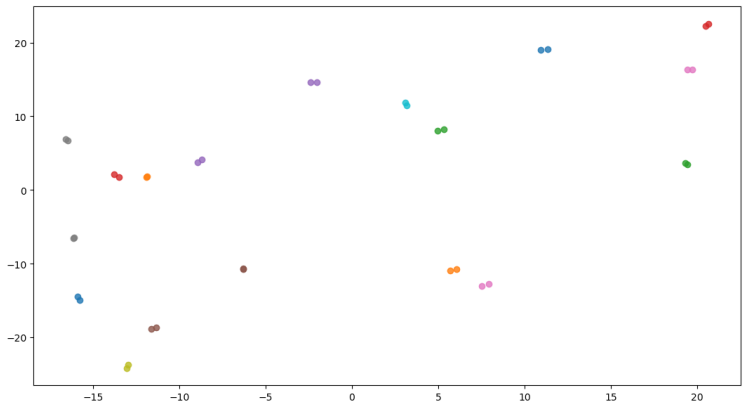
[1] By subtracting the mean of the returns of all of the stocks in the S&P 500.
[2] By subtracting the mean of the returns of the stocks in each industry from the stocks in that industry.
[3] Obviously, this has added visualization benefits.

Results

Our initial exploratory data analysis showed that the industry clusters explained the correlation of the stocks we had, but we were looking for correlations that were independent of industry. In fact, some of the correlations were negative.

With the identification of trading pairs as a particular application of interest, we saw that DBSCAN was the superior model: it produced clusters of assets with highly correlated returns (that were even clearly tracking each other), while the clusters themselves were very far apart. In particular, particular pairs belonged to completely different pairs. Our DBSCAN model easily generalized to clusters consisting of $n > 2$ stocks, was robust against random states in t-SNE, but also generated clusters with more positively correlated adjusted time series than the industry-based clustering/grouping benchmark.

Here's some of the pairs of similar stocks that we found using our strategy, plotted after the t-SNE embedding and the clustering algorithm outlined above.



Next Steps:

By introducing statistical arbitrage into our framework, we can exploit pricing inefficiencies between clustered equities as our data drifts over time. Our model aims to identify groups of stocks within the same cluster that are temporarily mispriced, allowing for short-term trading opportunities via a balanced long-short strategy. Thus, our project aspires not only to be a predictive tool but also an actionable model.