Using Graph Theory to Improve Portfolio Diversification

Tengiz (Tega) Gabunia

Table of Contents

Introduction

In the midst of taking part in the financial markets, investors construct and build portfolios, which are a collection of assets, like stocks. The whole point of a portfolio is to maximize returns and profits, while minimizing risk of loss. The key towards achieving such a situation is mainly through diversification of assets. Diversification itself, in terms of finance, is a risk-management strategy that involves spreading investments across various assets or stock to reduce exposure to high levels of any single volatility or risk. Diversification is not only about holding many assets and stocks at the same time, but it also depends on how those assets relate and behave relative to each other.

If, let's say, several stocks move in the same way at the same given time, holding all of them does not actually reduce risk. On the other hand, if stocks behave differently from one another, the combination of those stocks in the portfolio can stabilize the risk and reduce the overall volatility which may be fabricated from using highly-correlated stocks, like we mentioned for the first time.

This relationship between assets and stocks can be studied and investigated through Graph Theory. Graph Theory is a branch of mathematics which studies and analyzes networks made up of vertices and edges. In the financial networks, each stock can be represented as a vertex, and the relationship between those stocks can be measured and represented as a weighted edge. The weighted edges will measure how similar the price movements are related to one another.

In this project, Graph Theory will be used to analyze the relationship between several major stocks in the current financial markets, and to determine whether the structure of the financial network, using Graph Theory, can actually help us construct a more diversified portfolio (from our chosen stocks), which in turn will help us maximize returns and minimize risk and volatility.

The project will discuss and will process through the steps of collecting historical stock price data, calculating their daily returns, measuring the correlation between the stocks, then converting those correlations into graph distances, which will help us construct a weighted complete graph. And hence, for the purpose of efficiency and more in-depth research, we will extract a Minimum Spanning Tree from the given graph, which we'll analyze and compare two portfolios, finalized by a short conclusion.

Data Selection

This project will analyze the stocks of 18 one of the most traded stocks from different sectors of the financial market. This includes:

Technology:

- AAPL (Apple)
- MSFT (Microsoft)
- NVDA (Nvidia)
- META (Meta)
- GOOGL (Google)

Consumer Sector:

- AMZN (Amazon)
- WMT (Walmart)
- COST (Costco
- PG (Procter & Gamble Co)
- KO (Coca-Cola)

Finance Sector:

- JPM (JPMorgan)
- BAC (Bank of America)

Energy Sector:

- XOM (Exxon)
- CVX (Chevron)

Defense/Industrial Sector:

- CAT (Caterpillar Inc)
- LMT (Lockheed Martin)

Healthcare Sector:

- UNH (United Healthcare)
- LLY (Eli Lilly And CO)

The goal of selecting these companies and stocks from different sectors is to observe and analyze how stocks from different industries may also relate to one another.

This data consists of daily closing prices over one year. Using this data provides just enough information for us to observe and produce significant statistics, which determines the relationship between stocks. Data was collected from Yahoo Finance (Yahoo Finance, 2024).

Calculating Stock Returns

Before we start constructing the financial network, using the Graph Theory, the raw stock prices we see must be first transformed into some kind of form which allows us to make meaningful comparisons between the chosen assets. The raw prices themselves can’t be directly compared, since all of the companies trade at different price levels. For instance, one stock may be worth $200 (a lot), and the other may be worth $10 (a few dozen), but they both may experience similar relative change in value at any given time. Hence, we must convert these raw prices into daily returns, a comparable measurement, which is used to measure the percentage changes in stock price from one day to the next. This transformation, in short, standardizes the price and data, which will in return help us analyze the behavior of different stocks at the same time, even if they have different initial, raw prices.

After all of this, to apply the Graph Theory to the financial markets - from stock’s daily returns data, correlation between those same assets must be computed, in order to measure how similarly they move at any given time period. Correlation represents a measurement which can be described in three words: similarity of movement. However, since Graph Theory uses distance, mostly, as its measurement, correlations have to be then converted into distance measurements, using a specific set of formulas, in which we’ll dive deeper. Then, these distances will be used to represent edge weights in the network, where each stock is represented as a vertex, and the relationship is represented as an edge between any stock pair. This transformation will allow us to model the financial system as a graph, which can be thoroughly analyzed using the Graph Theory techniques and strategies.

To begin with daily returns, the formula for the calculation is the following:

$$r_t = \frac{P_t - P_{t-1}}{P_{t-1}}$$

Variable Definitions

- $\mathbf{r_t}$ is return of the stock on day t;
- $\mathbf{P_t}$ is price of the stock on day t;
- $\mathbf{P_{t-1}}$ is the price of the stock on the previous trading day of t.

Explanation

This formula measures how much the price changes relative to the previous given stock price. For example, let's suppose the closing price of Apple on one day was 250, and on the next day, it became 255. Hence the daily return shall be:

$$r_t = \frac{255 - 250}{250}$$

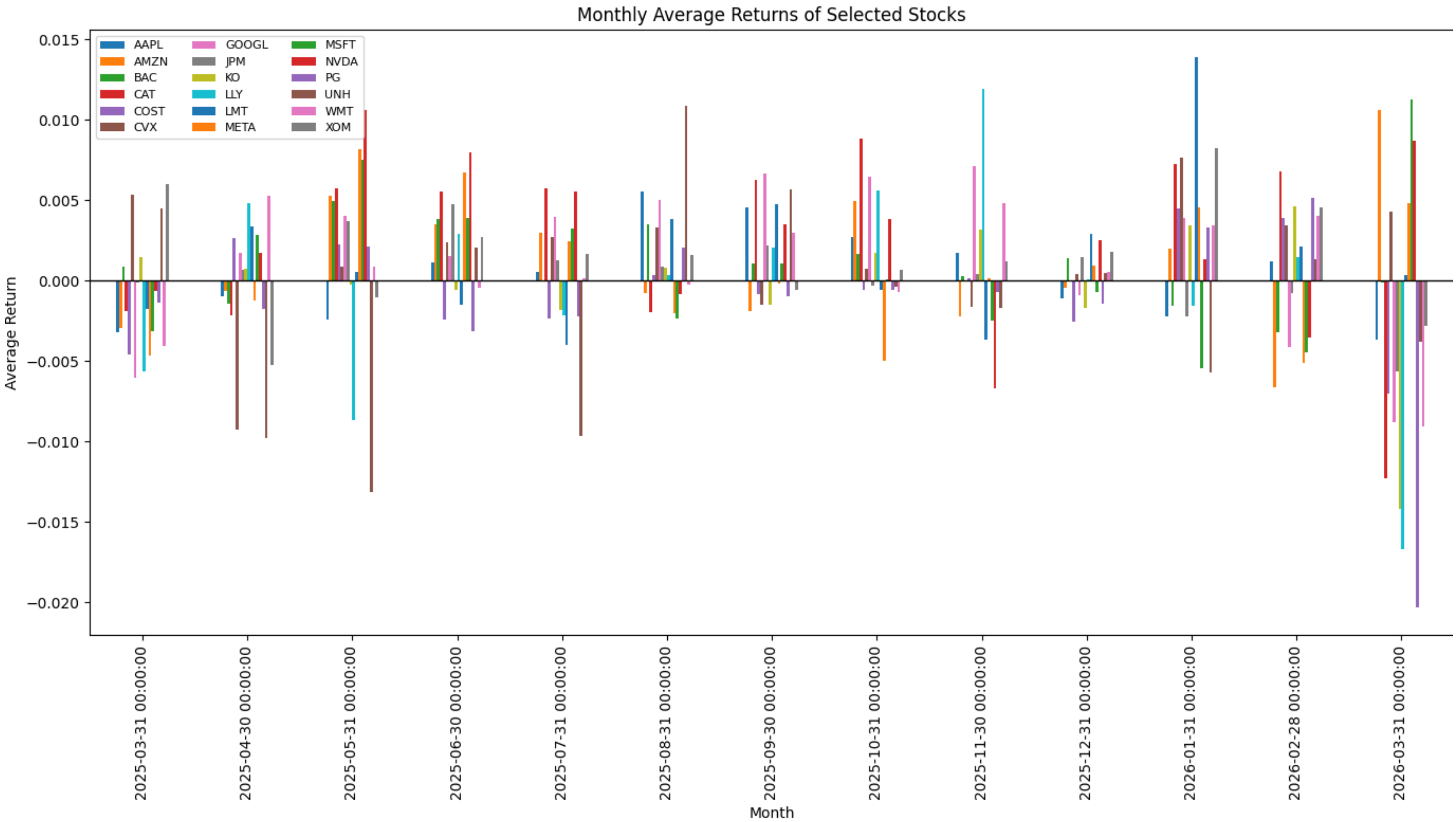
$$r_t = \frac{5}{250}$$

$$r_t \cong 0.02$$

So the return will be approximately 2%. This shows how daily returns are calculated from past and current stock prices, at any given time period.

After demonstrating this calculation, the same formula will be applied in Python, to compute daily returns of all stocks over the entire one year time period. In this graph, monthly returns are used to visually not create a mess, but the daily returns will be used in the future graphs and codes.

Monthly Average Returns Graph



Measuring Relationships Between the Chosen Stocks

After Python calculates returns for every stock, the next step is to measure how similar the movements of those different stocks are relative to each other. So that then, we can transform these correlations into graph distance measurements, which we will need for the Minimum Spanning Tree and the overall Graph. The second step (correlations) can be done using the correlation coefficient, which is showcased through this symbol:

$$\square_{\square\square}$$

Definitions

$\square_{\square\square}$ is the correlation between stock i and stock j, in any given time period.

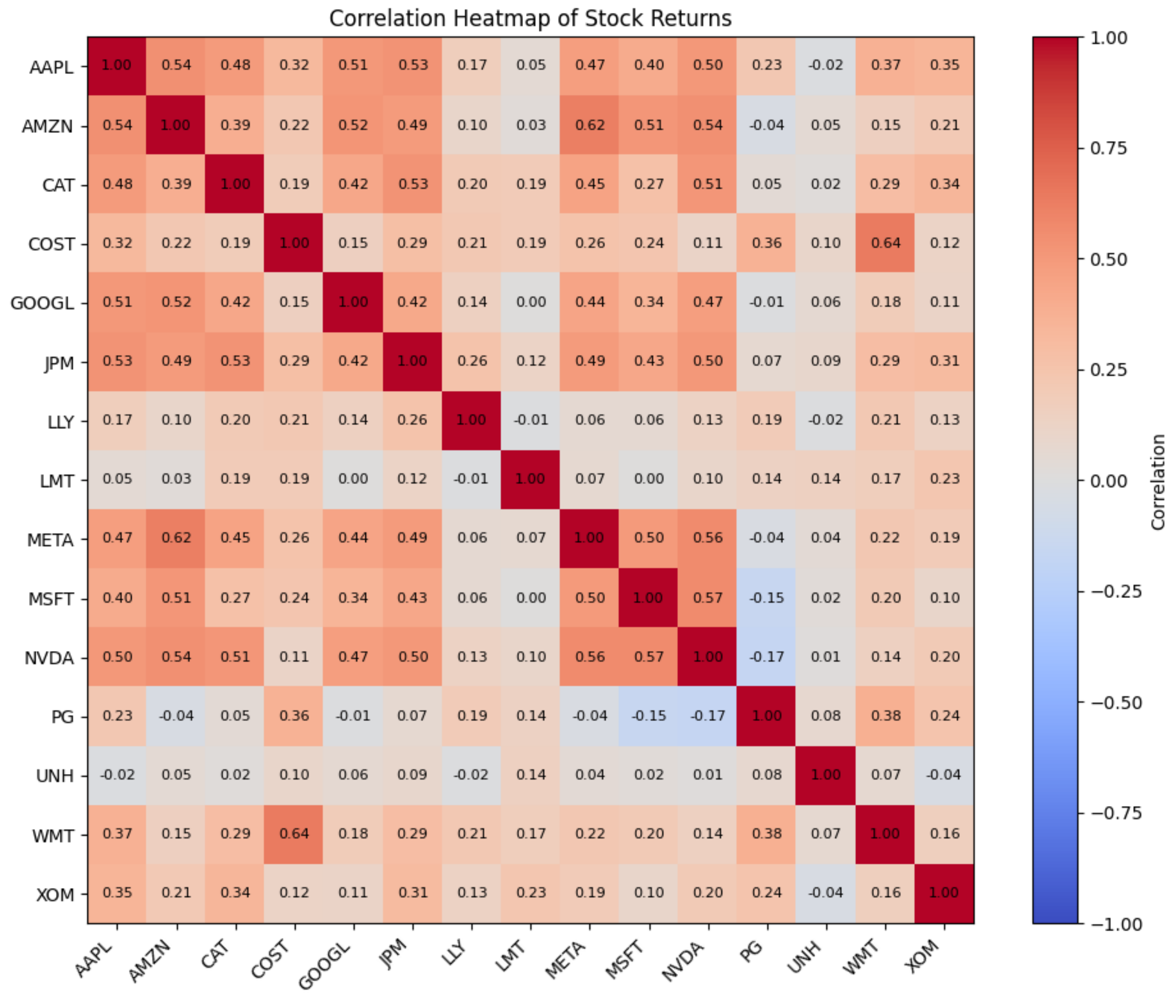
Interpretation of this Symbol

The correlation basically ranges from -1 to 1. 1 - meaning stocks i and j move exactly the same, and -1 - meaning stocks i and j move in opposite directions.

Correlation	Interpretation
-1	Stocks i and j move exactly the opposite way
0	Stocks i and j have no relationship
1	Stocks i and j move almost exactly the same

In portfolios, if successful diversification is the goal, correlations play a huge role, because we have to know which assets and stocks are highly correlated, since if the chosen stocks are highly correlated - it provides less diversification.

Correlations between the 18 stocks were calculated and computed by Python from the daily return series which we've conducted. Our calculations of daily returns and Python made it possible to create a full correlation matrix for each pair of stocks in this portfolio.



From this graph alone, we can obviously see that correlation between these 18 stocks, mostly, is high, hence meaning that these 18 stocks are very much interconnected to each other.

Converting Correlation into Graph Distance

Correlation does measure the similarity, but since graph algorithms and theories usually work with distance, we, therefore, have to convert those correlations into distance measurements, using this formula:

$$\square_{\square\square} = \sqrt{2(1 - \square_{\square\square})}$$

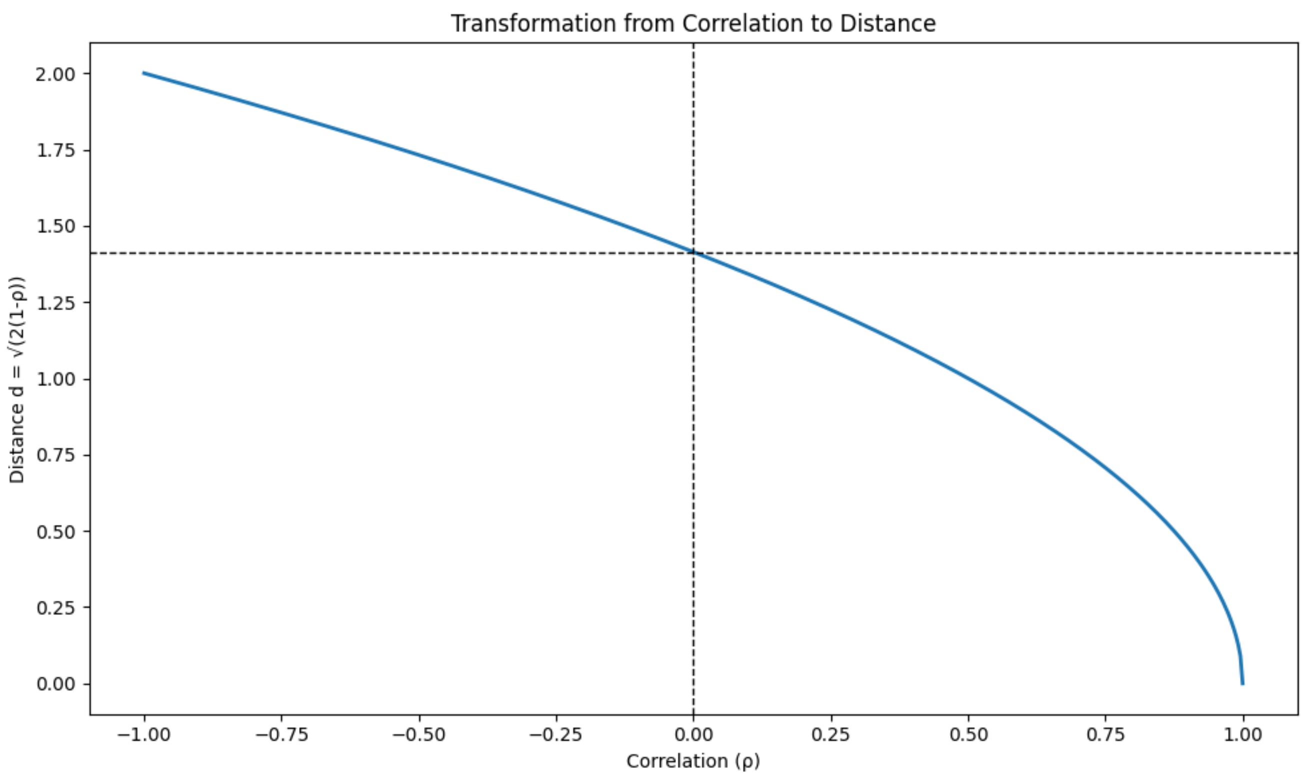
Definitions

d_{ij} is the distance between stock i and j for our graph

This conversion and transformation is necessary, because it converts correlation into a measurement which behaves like distance. For example, if two stocks act very similar and their correlation coefficient is equal to approximately 1, hence their d_{ij} will be equal to approximately 0. However, if stocks are very different in movement, then their distance will become larger and larger, as the correlation gets less and less.

This formula will be implemented in Python to calculate the distances between all stock pairs. This will allow me to graph the complete weighted network and the Minimum Spanning Tree, efficiently.

This graph below shows us why we transform correlation into distance measurement.



Construction of the Graph

Once the distances will be calculated in Python, the financial system of these 18 stocks can be represented as a graph.

Vertices
Each vertex will represent a stock, hence:

V = {"AAPL","MSFT", "NVDA", "GOOGL", "AMZN", "META", "XOM", "LLY", "JPM", "WMT", "CAT", "LMT", "COST", "UNH", "PG", "CVX", "KO", "BAC"}
There will be 18 vertices.

Edges
An edge connects every pair of stocks, hence this means that the initial graph will be a complete graph, since every vertex is connected to every other different vertex.

For n number of vertices, the amount of edges in a complete graph will be:

$$\frac{\square(\square - 1)}{2}$$

And, for 18 stocks:

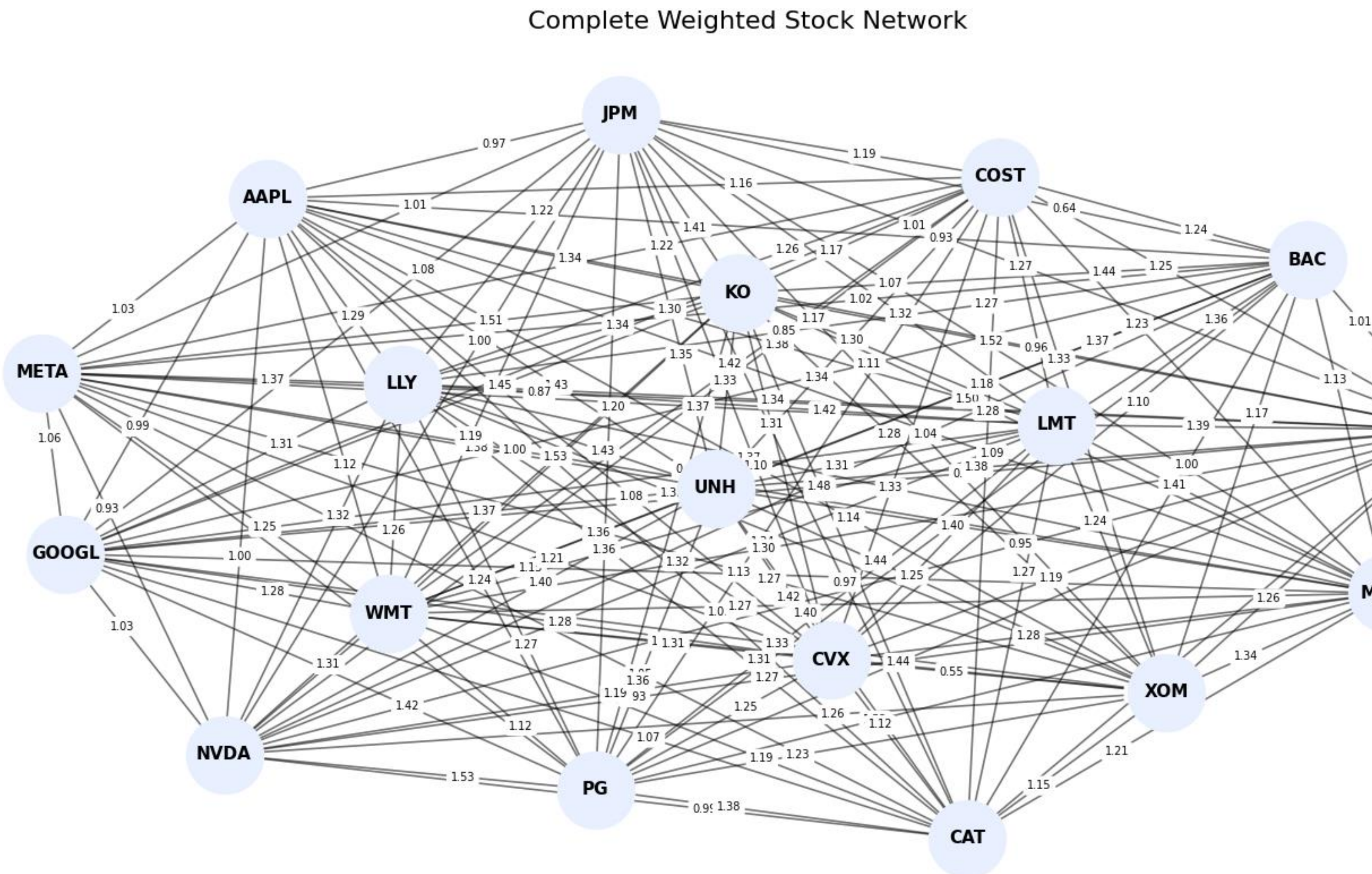
$$\frac{18(18 - 1)}{2}$$

$$\frac{18(17)}{2} = 153$$

With these calculations, we can say that the complete graph will contain 153 edges.

Edge Weights
Each edge will weight the distance derived from correlation, resulting into - $\square_{\square\square}$
Thus each edge's weight will showcase how similar those two stocks are. 2 - very close, 0 - no relationship.

Complete Network Graph



Minimum Spanning Tree

Since this graph is a mess, we apply Graph Theory to compute the Minimum Spanning Tree, to simplify everything.

MST is basically a subgraph that:

- connects all vertices;
- contains no cycles;
- minimizes the total edge weight.

Because the MST reduces the mess and unnecessary connections, it only reveals the essentials and the necessities of the network structure.

For a graph with n amount of vertices, the MST will contain:

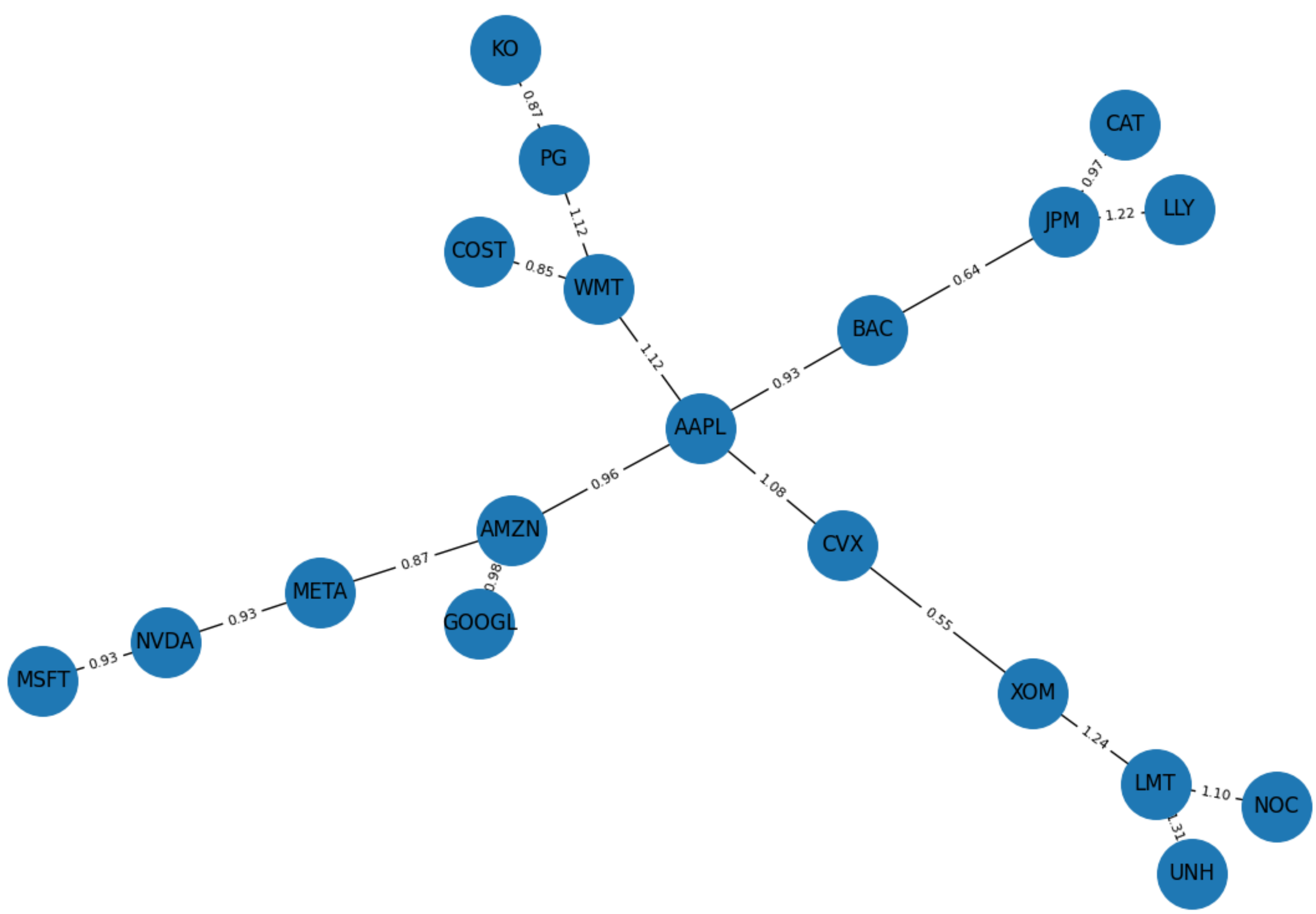
$$n - 1$$

amount of edges.
Since, our graph has 18 vertices, MST will feature:

$$18 - 1 = 17$$

17 edges.

Minimum Spanning Tree with Distance Labels



Network Interpretation

Once we constructed the MST, we can analyze the network.

Central Stocks
Stock(s) with several connections can be considered as a central node(s)

These kinds of stocks tend to move similarly to others, instead of the opposite. For example, Amazon and Apple have the most connections, meaning that they move the most similarly to the connected stocks.

Peripheral Stocks

Stock(s) which have only one connection lie at the edge of the MST. These stocks behave independently from the rest of the herd (our chosen stocks). Such stocks will provide bigger and stronger diversification benefits, instead of the central stocks.

From this graph itself, we can say that Graph Theory will help us out a lot when it comes to identifying which stocks are moving along with the others, and which are moving independently. Through this, I'll create two portfolios, one with central stocks, and another with peripheral stocks. Then we will compare the financial performance metrics, to find if the graph theory helped us determine which stocks are able to give us more options of diversification, and more returns.

Portfolio Construction

To now evaluate the usefulness of Graph Theory, I'll create two portfolios.

Portfolio A - holds `"AAPL", "META", "AMZN", "WMT", "JPM", "XOM", "LMT", "NVDA", "CVX", "BAC"` (stocks which are very close and highly correlated to each other. These stocks (vertices) are typically identifiable from the center of the MST)

Portfolio B - holds `"NVDA", "MSFT", "UNH", "LLY", "NOC", "CAT", "PG", "COST", "GOOGL", "KO"` (stocks which are very far from each other and are correlated very lowly. These stocks (vertices) are typically identifiable from the edge of the MST)

Financial Performance Metrics

To compare those portfolios, we have to use several financial metrics to do so.

Expected Return

One of them is Expected Return, which calculates and measures the return that is expected:

$$\square_{\square} = \sum \quad \square_{\square}\square_{\square}$$

Where $\square_{\square}$ is the portfolio return, $\square_{\square}$ is total portfolio weight of stock i, and $\square_{\square}$ is the total expected return of stock i.

Volatility

Another metric for such comparison is the volatility:

$$\sigma_{\square}$$

Which is the standard deviation of portfolio returns. It basically measures - risk.

Sharpe Ratio
Sharpe ratio measures return relative to risk. It helps investors determine if the returns and profits are caused by smart investment decisions or excessive risk. It is a risk-adjusted performance metric, which shows the investors the amount of excess return a portfolio got, for each unit of risk taken. Helping us evaluate how efficiently risk is converted to returns.

$$\square = \frac{\square_{\square} - \square_{\square}}{\square_{\square}}$$

$\square_{\square}$ - Portfolio’s return
$\square_{\square}$ - Risk-free rate
$\square$ - Sharpe ratio

- If the ratio is less than <1 - investments generally perform worse than the risk-free rate. It is bad.

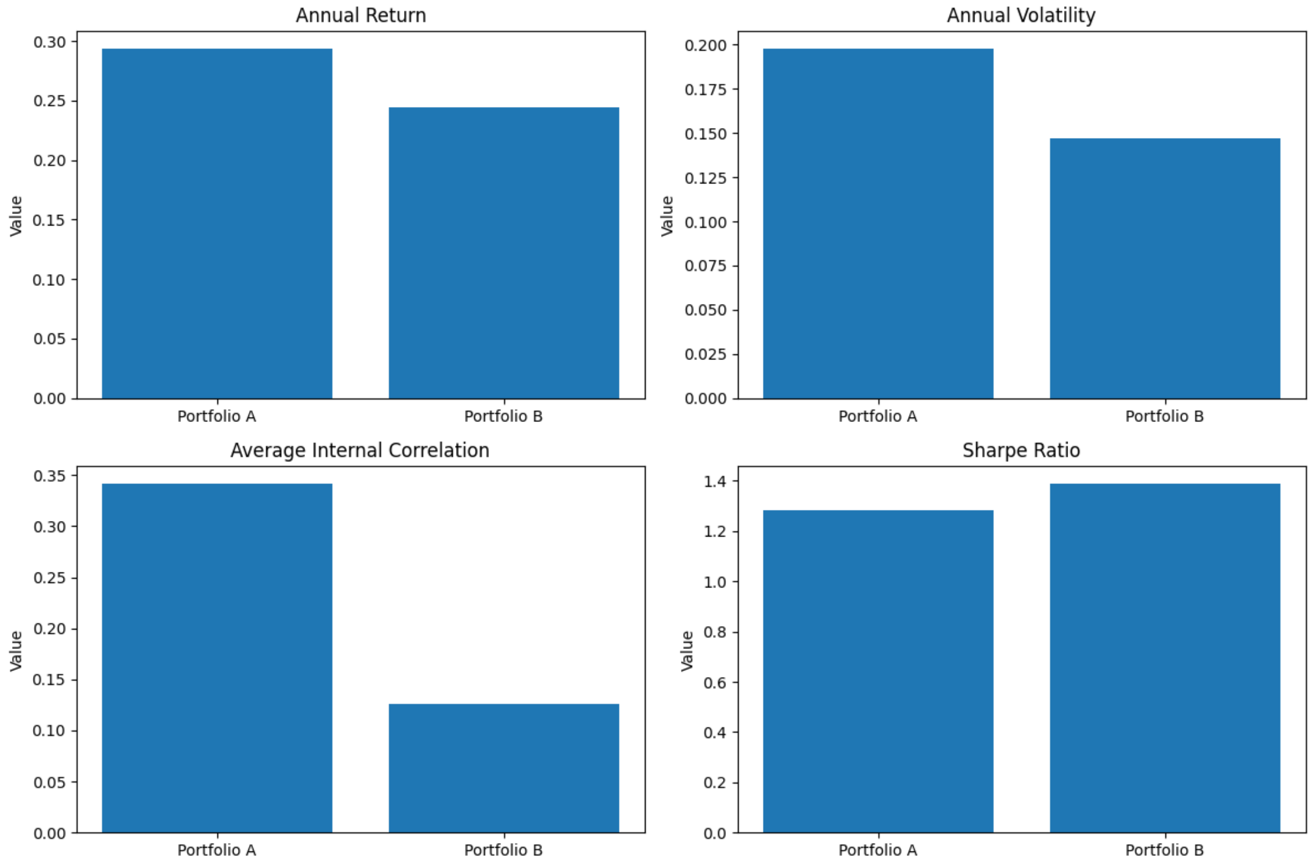
- If the ratio is more than >1 - Low risk-adjusted return. It is acceptable.
- If the ratio is more than >2 - Good risk-adjusted return. It is very good.
- If the ratio is more than >3 - Extremely efficient portfolio. It is excellent.

This ratio is used to compare portfolios' performances.

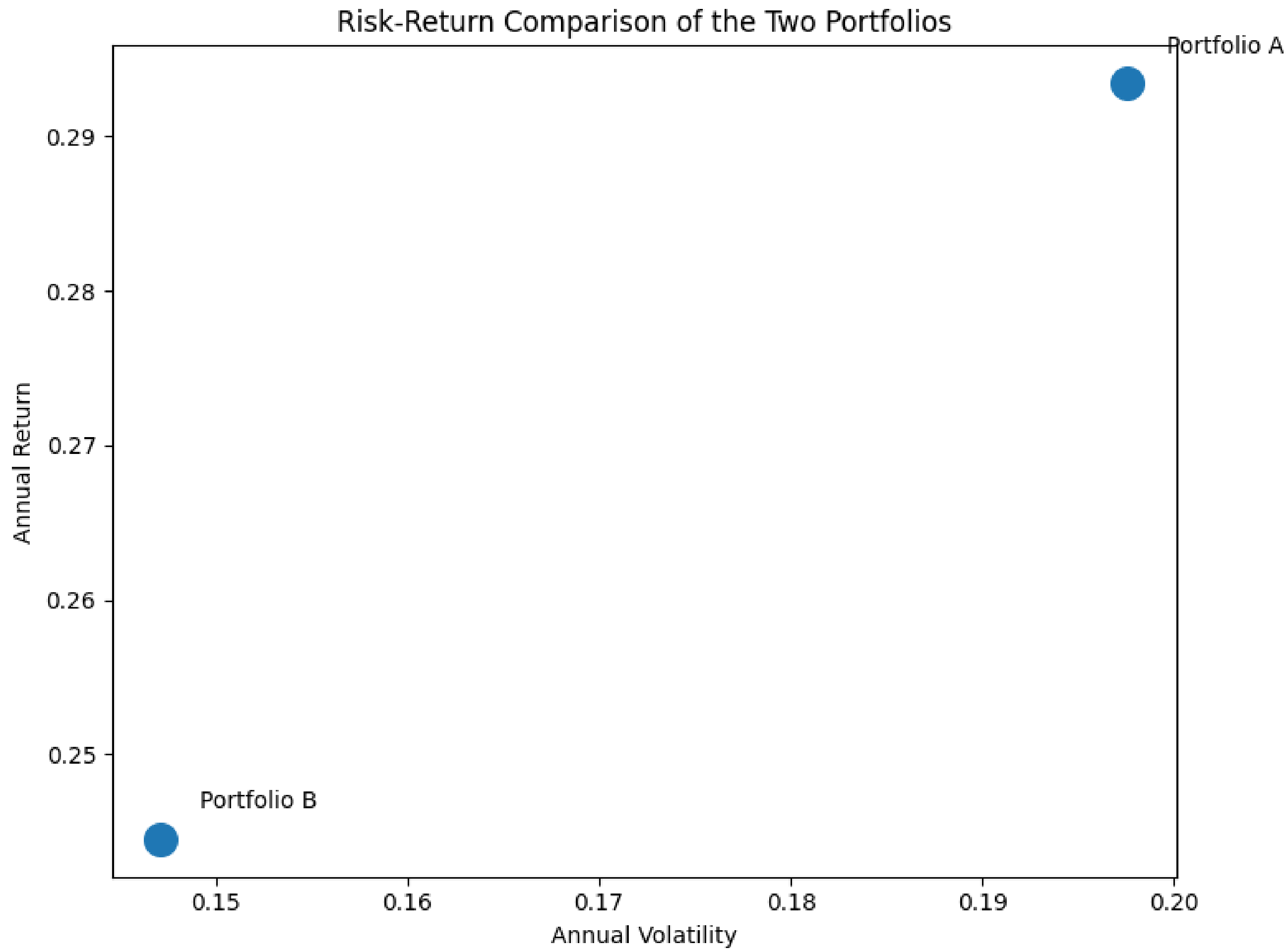
We will put all of these metrics' code into Python, and compare those two portfolios' performance according to those three metrics, over the last year.

Portfolio Risk and Return

Bar Chart Comparison



Risk-Return Scatter Plot



Final Conclusion

This project, at last, applied Graph Theory to analyze the relationship between stocks and their movement, to determine and investigate whether the portfolio construction based on networks and Graph Theory itself, can improve diversification. By first choosing the pool of stock choices we wanted to experiment on, collected from Yahoo Finance, started the report off from. Secondly, collecting stock returns and price data, and then transforming it into correlations between those chosen assets, and then converting those correlations into distance measurements, it became possible for us to construct the financial market as a weighted, complete network, where each stock is represented as vertex and the relationships between them as edges.

Using this structure and skeleton for our project, two portfolios were constructed based on the number of connections they had in the complete and weighted network of Minimum Spanning Tree. Portfolio A consisted of *AAPL, META, AMZN, WMT, JPM, XOM, LMT, NVDA, CVX, BAC,* while Portfolio B consisted of *NVDA, MSFT, UNH, LLY, NOC, CAT, PG, COST, GOOGL, KO.* Then, these portfolios were compared using key financial performance metrics, such as - annual return, correlation between those stocks in each portfolio, annual volatility and the Sharpe Ratio.

The overall comparison results (Yearly) showed that Portfolio A achieved a higher annual return of 29.3% but also exhibited very high amounts of volatility and risk of about ~20%. Plus, Portfolio A had a relatively high internal correlation, which indicated and highlighted that many stocks in that portfolio tended to move together at the same time. This reduces the benefits of diversification and it increases overall risk for the portfolio.

However, Portfolio B showcased that it produced a lower amount of annual return 24.4%, but its volatility was considerably lower at about ~15%. But the more important aspect of these statistics is that Portfolio B had a much lower average correlation, which indicates that stocks in that portfolio moved independently from one another. As a result of these performances from both Portfolios, Portfolio B achieved a higher sharpe ratio of about 1.39, while Portfolio A's Sharpe Ratio was considerably low of about 1.28. Which means that Portfolio B generated a better risk-adjusted performance in the last given year, according to our statistics and graphs.

Now, obviously, these results and analysis show the key core principal of diversification in a portfolio. Choosing stocks and assets which are less correlated can indeed reduce risk and volatility for a portfolio, without necessarily sacrificing performance and returns. By the analysis of relationships between those kinds of stocks, and using Graph Theory and network interpretations, for an investor, it becomes possible to detect which stocks are correlated highly and which are not, hence it becomes possible for us to identify which assets are going to provide stronger and bigger diversification benefits, in the financial market. This project showcases that Graph Theory plays a big role in the process of providing the necessary guidance and structure for understanding financial markets, hence helping us improve portfolio construction. By our construction of graphs using Graph Theory and many more concepts, it became possible to build a network and analyze the relationship between stocks and assets, which would help us and investors make more risk-adjusted decisions, analyze better and be more informed in terms of our portfolio and the market itself, hence helping us improve diversification and the process of minimizing risk.

Bibliography

1. Yahoo Finance. (2024). *Most Active Stocks: US stocks with the highest trading volume today - Yahoo Finance*. Yahoo.com. https://finance.yahoo.com/markets/stocks/most-active/
2. Numpy. (2024). *NumPy*. Numpy.org. https://numpy.org/
3. Matplotlib. (2024, May 30). *Matplotlib: Python Plotting — Matplotlib 3.1.1 Documentation*. Matplotlib.org. https://matplotlib.org/
4. Aroussi, R. (2023). *yfinance: Yahoo! Finance market data downloader*. PyPI. https://pypi.org/project/yfinance/
5. Markowitz, H. (1952). Portfolio Selection. *The Journal of Finance*, *7*(1), 77–91. https://doi.org/10.2307/2975974
6. Mantegna, R. N. (1999). Hierarchical structure in financial markets. *The European Physical Journal B*, *11*(1), 193–197. https://doi.org/10.1007/s100510050929