Portfolio Construction Through Clustering

Melique Daley

April 22$^{\text{th}}$, 2021

1 Introduction

It is common in the financial world to work with time-series data making it difficult to use classical statistical learning techniques such as regressions to make predictions since they can be naive. In addition, financial data is highly variable and complex since many factors are interacting with each other. For instance, macro and microeconomics, government policies, global capital markets, etc. Clustering is an unsupervised learning technique, where one can discover patterns directly from data that is unlabeled. Clustering offers a great opportunity to discover complex patterns and relationships within financial data.

This report will focus on portfolio construction through clustering. That is, develop a clustering model that will construct a portfolio from a set of stocks. I am doing this because it is difficult to develop portfolios in the classical sense and clustering offers a systematic and flexible way to group and weigh assets that a Portfolio Manager could miss. I don't believe clustering could replace Portfolio Managers and Analysts, but we do believe the technique will help them greatly in a world that gets more complex and unpredictable every day.

2 Background

2.1 K-Means

K-Means is a clustering algorithm that partitions the data into disjoint sets such that each point belongs to one single cluster. That is, for K clusters randomly assign K vectors that represents the geometric center of the clusters, known as the centroids. We can form this as an optimization problem and minimize the objective function,

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} ||x_n - \vec{\mu_k}||^2$$

where x_n is a data point, N is the total number of observations and K is the number of clusters. $r_{nk} \in \{0, 1\}$ indicates which cluster x_n is in. Meaning, if x_n is assigned to cluster k then $r_{nk} = 1$ and 0 otherwise. $\vec{\mu_k}$ us the vector associated with cluster k, the centroid [1]. A general K-Means algorithm is given below,

```
while $\Delta J < \theta$ do
    minimize $J$ with respect to $r_{nk}$ and keeping $\mu_k$ fixed;
    minimize $J$ with respect to $\mu_k$ and keeping $r_{nk}$ fixed;
end
```

2.2 Hierarchical Cluster

In hierarchical clustering nested groups are formed iteratively using a bottom up approach or a top down approach [1]. Hierarchical Cluster works well where the structure of the underlying data is a tree [2] and we will see later hierarchical clustering struggles with our data.

Since, there are many Hierarchical Clustering methods this report will focus only show Ward's method as it had the best performance on the data. Ward's methods define the distance between two clusters A and B as how much the sum of squares will increase when they are merged,

$$\begin{aligned}\Delta(A,B) &= \sum_{i\in A\cup B} ||\vec{x}_i - \vec{\mu}_{A\cup B}||^2 - \sum_{i\in A} ||\vec{x}_i - \vec{\mu}_A||^2 - \sum_{i\in B} ||\vec{x}_i - \vec{\mu}_B||^2 \\ &= \frac{n_A n_B}{n_A + n_B} ||\vec{\mu}_A - \vec{\mu}_B||^2\end{aligned}$$

where $\vec{\mu}_j$ is the center of the cluster j, and n_j is the number of points in it. Δ is the merging cost of combining the clusters A and B.

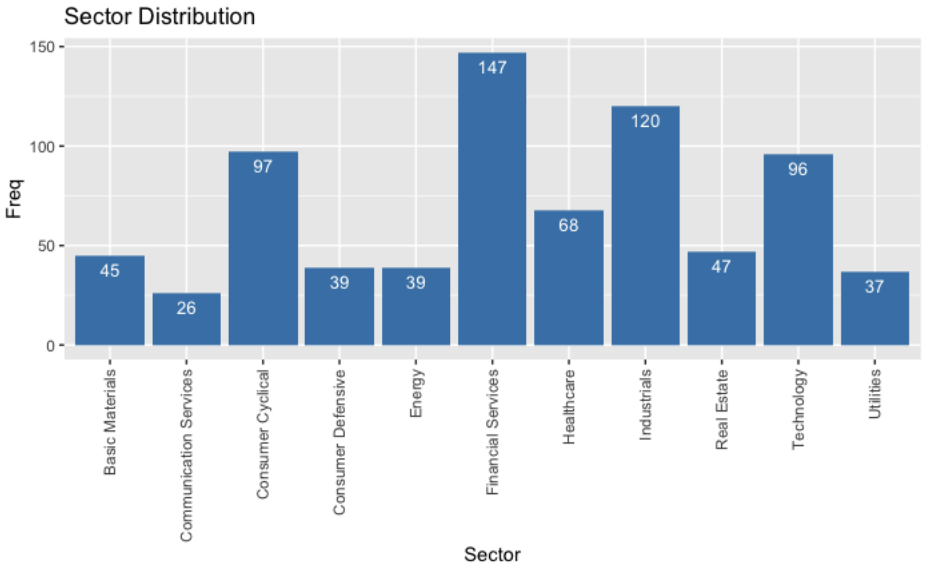
With all hierarchical clustering methods, the sum of squares starts out at zero since every point is its own cluster and then grows as cluster merge. Ward's method wants to keep this as small as possible. Given two pairs of clusters whose centers are equally far apart Ward's method would like to merge the smaller ones [3].

3 Data Description

We found the log monthly returns of 773 stocks from 2010-01-01 to 2020-12-31 and same for the S&P 500 ($\hat{\text{G}}$SPC) that will be used as a benchmark.

The following figure shows the sector distribution of the 773 stocks.

Figure 1



4 Methodology

To find the optimal number of clusters we used the Silhouette metric given by,

$$s(i) = \begin{cases} 1 - a(i)/b(i) & a(i) < b(i) \\ 0 & a(i) = b(i) \\ b(i)/a(i) - 1 & a(i) > b(i) \end{cases}$$

$$a(i) = \frac{1}{|C_i| - 1} \sum_{j\in C_i, i\neq j} d(i,j) \qquad\qquad b(i) = \min_{k\neq i} \frac{1}{C_k} \sum_{j\in C_k} d(i,j)$$

where $d(i, j)$ is some distance function.

The Silhouette metric is a measure how similar a data point is to its own clusters compared to other clusters and ranges from -1 to 1. Thus, a high Silhouette value indicates dense and well separated [4]. Once the clusters are formed, we need to find n stocks that will be in the portfolio. We rank each cluster by finding the average Sharpe ratio of the cluster and sort them in descending order. That is,

$$\frac{1}{|C_k|} \sum_{i \in C_k} \frac{R_{si} - R_f}{\sigma_{si}} \quad \text{for each } 1 \leq k \leq K$$

The rationale behind this is better performing stocks will be in the same cluster and Sharpe ratio helps us quantify this. For this 10-year period we assume the risk free rate, $R_f = 0.01$.

Since better performing stocks will be in the same cluster, we want to pool more stocks from clusters with higher Sharpe ratio. To do this, we construct a geometric sequence that starts from n and has length K and take the ceiling of it to get integers. We ignore the first number in the sequence, decrease the second number in the sequence by 1, then for the remaining numbers in the sequence we randomly add 1 to one of them. We decrease by 1 to ensure the best performing cluster doesn't have a majority in the portfolio. We have to add back 1 and we do it randomly to increase the diversity of the portfolio backup. Note, that this process doesn't always produce n stocks but does the majority of time or very close such as $n - 1$ or $n + 1$.

An example of this, we want 7 stocks in a portfolio and have 5 clusters. The clusters ranks are $r = (4, 2, 5, 3, 1)$ where the index of r is the cluster it represents. The geometric sequence after taking the ceiling is $(7, 4, 2, 1, 1)$. We ignore the 7, decrease 4 by 1 and randomly add 1 to cluster 5. Thus, 3 stocks from cluster 4, 2 stocks from cluster 2, 2 stocks from cluster 5, and 1 stock form cluster 1.

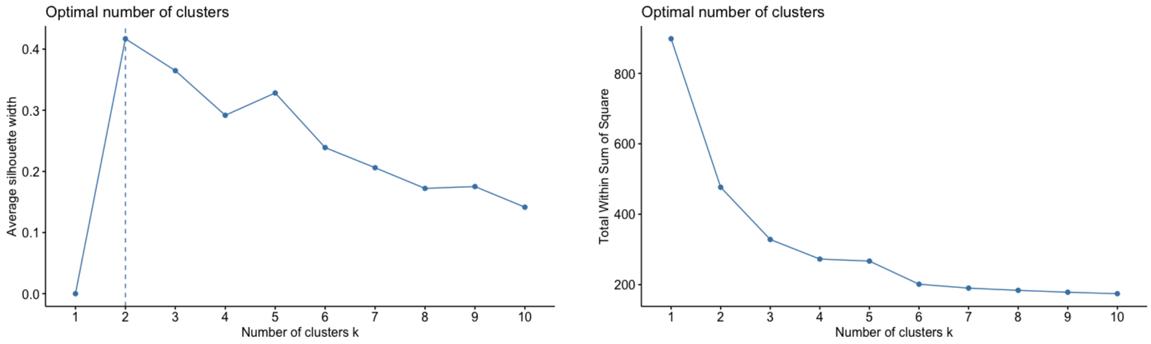
To find the optimal weights for the portfolio, we generate a large number of portfolios such as 5000, calculate the returns, risk and Sharpe ratio of each portfolio. Doing this constructs an efficient frontier and from Modern Portfolio theory we can find minimum variance portfolio and the tangency portfolio.

5 Experiment and Performance

For each method we construct 5000 portfolios composed of 7 stocks.

5.1 K-Means

The following plots tells us the optional number of clusters to use using 2 different metrics.



(a) Average Silhouette

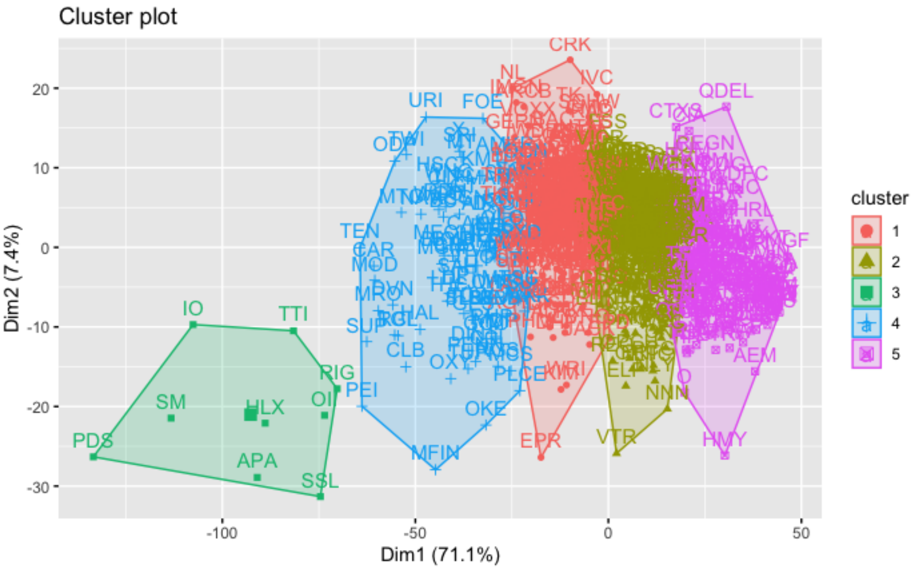
(b) Within-Cluster Sum of Square (WSS)

Figure 2

We focus on Average Silhouette since it is more suitable in this case and pick 5 clusters.

Then we construct clusters on the covariance matrix of the log monthly returns and multiple it by 21 to annualized the returns. We can visualize the clusters made by K-Means below,

Figure 3: K-Means' Cluster Plot



We see for the most part there is good separation between the clusters. The process than produces 7 stocks and their characteristics are below,

	Market.Cap	Sector	Industry	Country
COST	1.45000e+11	Consumer Defensive	Discount Stores	United States
ROL	1.62060e+10	Consumer Cyclical	Personal Services	United States
WCN	2.72700e+10	Industrials	Waste Management	Canada
SNPS	3.56630e+10	Technology	Software—Infrastructure	United States
CDNS	3.63110e+10	Technology	Software—Application	United States
CGNX	1.47580e+10	Technology	Scientific & Technical Instruments	United States
LRCX	7.81960e+10	Technology	Semiconductor Equipment & Materials	United States

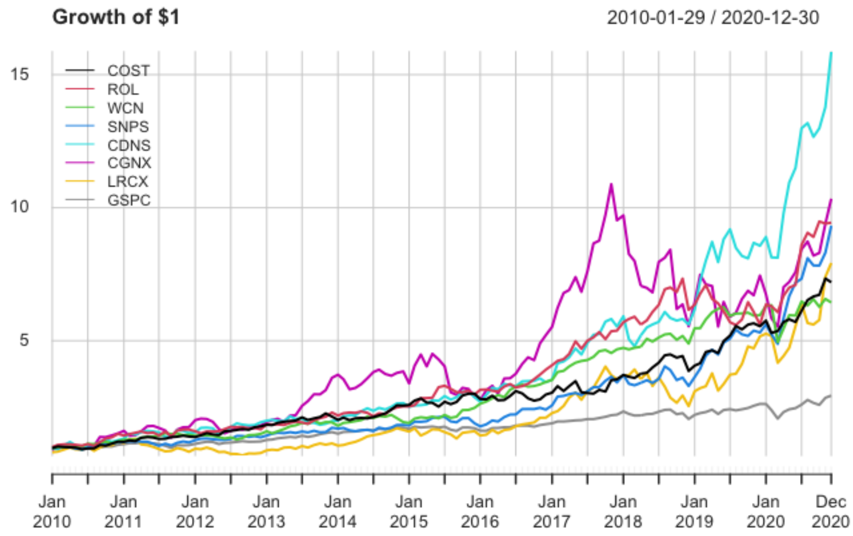
We see that K-Means picked half the stocks from the Technology which isn't surprising as that sector has been performing very well. Overall the profile is diverse and in theory if any of these sectors are not doing well the portfolio should be resilient.

Figure 4 on the next page shows the Compound Annual Growth Rate (CAGR) for each stock and the S&P 500. We can clearly see that each stock picked outperformed the S&P 500 on the given time period.

Next, we generate 5000 portfolios with those K-Means' stocks and find the minimum variance and tangency portfolios. We can look at the head of the generated portfolios and see the different weight combinations and their associated returns, risk and, Sharpe ratio,

	COST	ROL	WCN	SNPS	CDNS	CGNX	LRCX	Return	Risk	Sharpe ratio
1	0.06	0.18	0.07	0.12	0.26	0.15	0.17	0.52	0.23	2.23
2	0.08	0.09	0.13	0.15	0.10	0.24	0.21	0.51	0.24	2.04
3	0.12	0.43	0.02	0.03	0.01	0.37	0.03	0.51	0.25	1.99
4	0.26	0.21	0.08	0.04	0.26	0.01	0.14	0.49	0.19	2.49
5	0.12	0.09	0.03	0.14	0.27	0.09	0.26	0.52	0.24	2.15
6	0.20	0.04	0.24	0.04	0.18	0.16	0.15	0.48	0.21	2.25

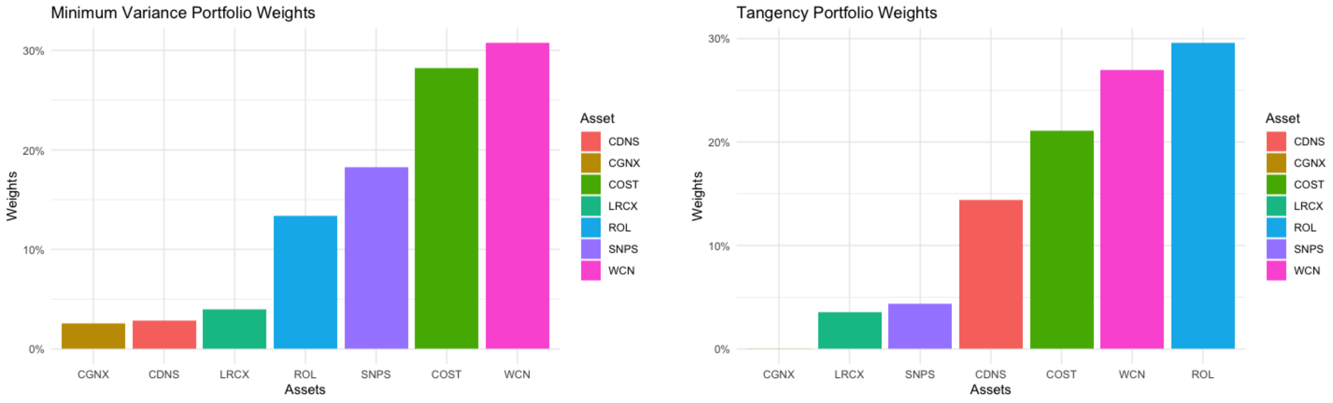
Figure 4: K-Mean's CAGR



The following table shows the minimum variance portfolio and the tangency portfolio weights and their associated returns, risk and, Sharpe ratio. Observe the portfolios are very close in terms of returns, risk, and Sharpe ratio.

	COST	ROL	WCN	SNPS	CDNS	CGNX	LRCX	Return	Risk	Sharpe ratio
minimum variance	0.28	0.13	0.31	0.18	0.03	0.03	0.04	0.43	0.17	2.52
highest sharpe	0.21	0.30	0.27	0.04	0.14	0.00	0.04	0.45	0.17	2.62

The following plots show the weight distribution of the minimum variance portfolio and the tangency portfolio,



(a)

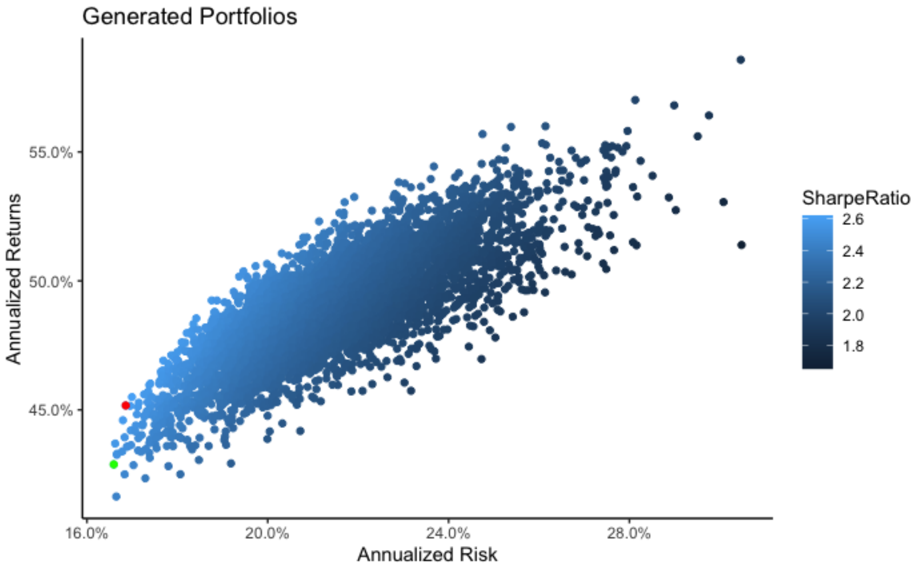
(b)

Figure 5

The weight distribution is expected since in the minimum variance portfolio will put more weight on stocks that are less volatile and the tangency portfolio does the opposite.

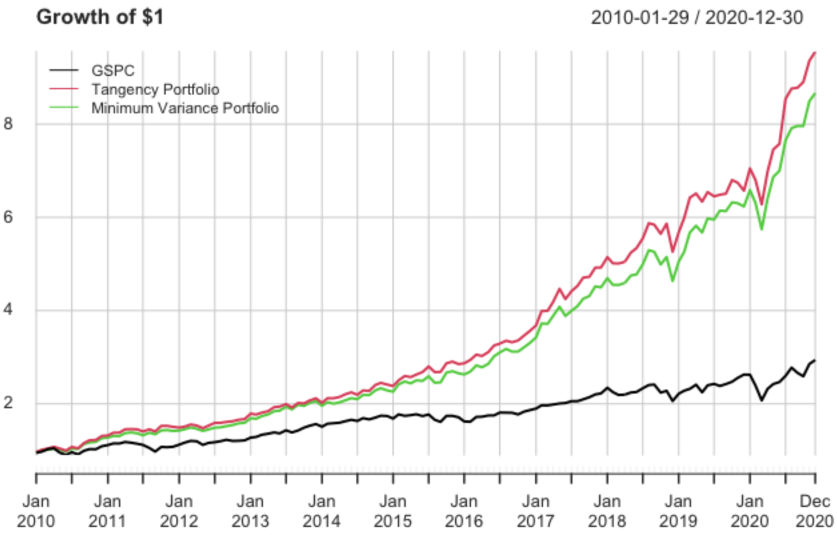
Figure 6 on the next page shows a visualization all 5000 K-Means' generated portfolios. The green point is the minimum variance portfolio and the red point is the tangency portfolio. Observe all generated portfolios have high returns but are risky.

Figure 6: Generated Portfolios through K-Means



Comparing the minimum variance portfolio and tangency portfolio against the S&P 500,

Figure 7: K-Means' CAGR



We see that the tangency portfolio outperforms the minimum variance portfolio which is expected but not by a lot. We clearly see both portfolios outperform the S&P 500.

We can conclude that the clustering algorithm K-Means did an excellent job to at picking stocks that can easily outperform the S&P 500.

5.2 Ward's Method

We take the general approach as K-Means. We do expect the performance to be worse since as mentioned earlier, hierarchical clustering works well on tree structure data and the log monthly returns are not that [4].

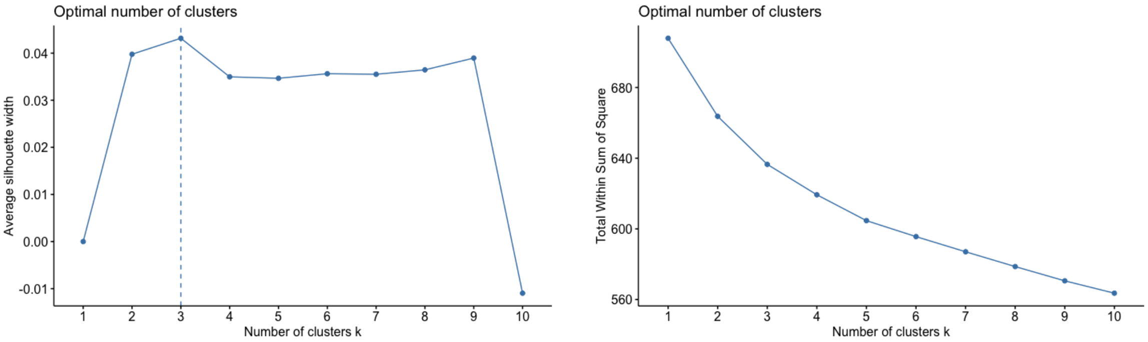
The distance function we use for stocks i and j is,

$$D_{i,j} = \sqrt{2(1-\rho_{i,j})}$$

where $\rho_{i,j}$ is the Pearson correlation coefficient,

$$\rho_{i,j} = \frac{\mathrm{Cov}(i,j)}{\sigma_i \sigma_j}$$

Looking at the optimal cluster metrics we see a big difference between it and K-Means,



(a) Average Silhouette

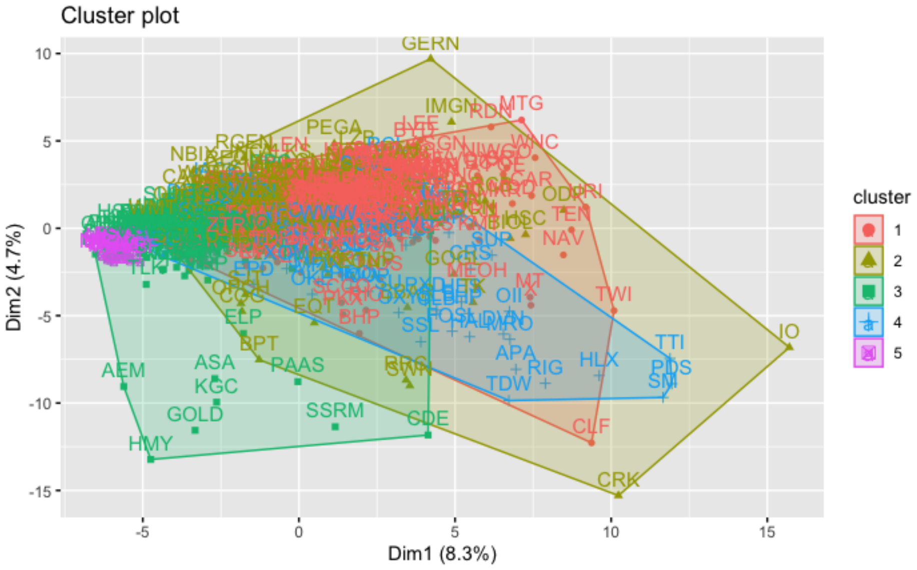
(b) Within-Cluster Sum of Square (WSS)

Figure 8

Average Silhouette is very small for all values and WSS is larger for all values.

Looking at the cluster plot we see a lot of overlap with the clusters,

Figure 9: Ward's Cluster Plot



Ward is unable to create well dense and separated clusters like K-Means.

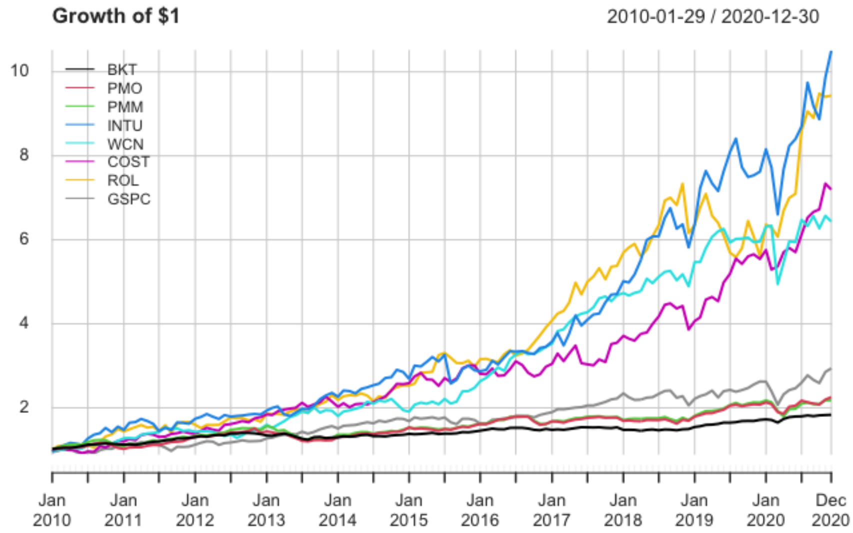
Again, doing the same process we get 7 stocks from Ward's clusters and can look at the characteristics of the stocks,

	Market.Cap	Sector	Industry	Country
BKT	3.85972e+08	Financial Services	Asset Management	United States
PMO	4.64427e+08	Financial Services	Asset Management	United States
PMM	3.91040e+08	Financial Services	Asset Management	United States
INTU	1.08000e+11	Technology	Software—Application	United States
WCN	2.72700e+10	Industrials	Waste Management	Canada
COST	1.45000e+11	Consumer Defensive	Discount Stores	United States
ROL	1.62060e+10	Consumer Cyclical	Personal Services	United States

Ward's stocks are very different form K-Means. It has preferred Financial services companies rather than technology companies.

Looking at Ward's CAGR,

Figure 10: Ward's CAGR



Observe, 3 companies that Ward picked are underperforming against the S&P 500. A big contrast, to K-Means were all its stocks outperformed the index.

We now generate 5000 portfolio from Ward's stocks and look at the head of the data frame,

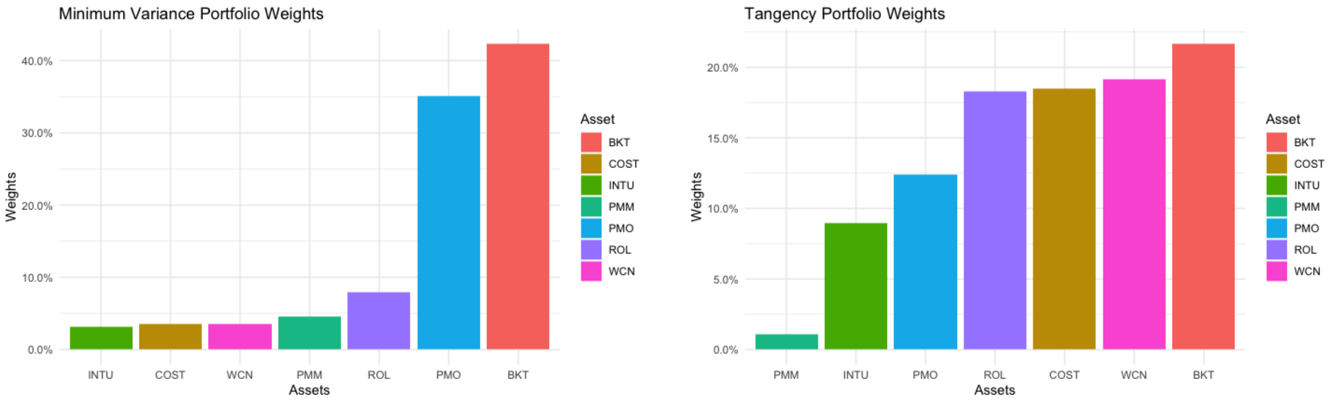
	BKT	PMO	PMM	INTU	WCN	COST	ROL	Return	Risk	Sharpe ratio
1	0.06	0.18	0.07	0.12	0.26	0.15	0.17	0.33	0.13	2.51
2	0.08	0.09	0.13	0.15	0.10	0.24	0.21	0.34	0.13	2.50
3	0.12	0.43	0.02	0.03	0.01	0.37	0.03	0.25	0.11	2.17
4	0.26	0.21	0.08	0.04	0.26	0.01	0.14	0.25	0.10	2.28
5	0.12	0.09	0.03	0.14	0.27	0.09	0.26	0.35	0.14	2.49
6	0.20	0.04	0.24	0.04	0.18	0.16	0.15	0.27	0.11	2.45

Another big difference between Ward's stocks and K-Means' stocks is that Ward's stocks have lower returns but also lower risk and their Sharpe ratio is good. This indicates that Ward's method can still construct good portfolios with less risk and returns.

The following table shows Ward's minimum variance portfolio and the tangency portfolio weights and their associated returns, risk and, Sharpe ratio,

	BKT	PMO	PMM	INTU	WCN	COST	ROL	Return	Risk	Sharpe ratio
minimum variance	0.42	0.35	0.05	0.03	0.03	0.03	0.08	0.18	0.08	2.01
highest sharpe	0.22	0.12	0.01	0.09	0.19	0.18	0.18	0.31	0.12	2.56

The following plots show the weight distribution of the minimum variance portfolio and the tangency portfolio.



(a)

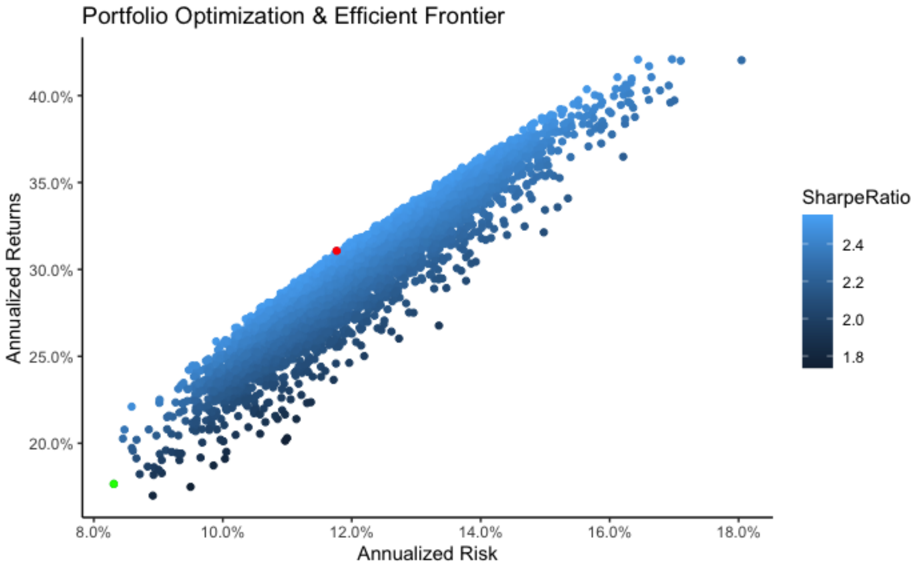
(b)

Figure 11

We see for the minimum variance portfolio it is composed mostly from only 2 stocks while the tangency portfolio is more uniform.

We can visualize all 5000 Ward's generated portfolios. Again, the green point is the minimum variance portfolio and the red point is the tangency portfolio.

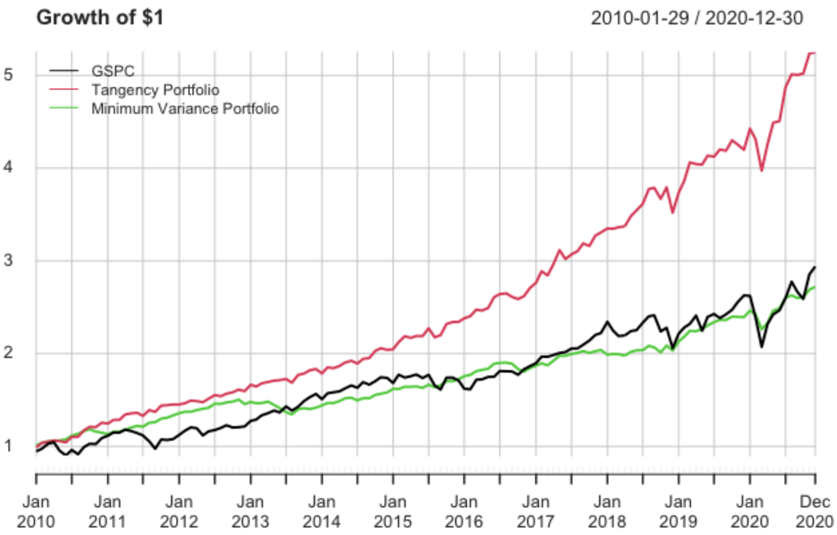
Figure 12: Ward's Generated Portfolio



Another contrast from K-Means, we see the efficient frontier of Ward is a lot linear and its minimum variance portfolio and tangency portfolio are further apart.

Finally, comparing the minimum variance portfolio and tangency portfolio against the S&P 500 which can be found in figure 13 in the next page. We see that the minimum variance portfolio essentially tracks the S&P 500 and there are multiple inflection points with it and the index. Moreover, take note that the minimum variance portfolio has more stable returns than the S&P 500. Moreover, the tangency portfolio outperforms the index which is expected.

Figure 13: Ward's CAGR



Comparing the performance to K-Means and Ward's method it's clear that K-Means outperformed Ward's method and it's fair to say any partitioning clustering algorithms would do better than a hierarchical one. Observe, the tradeoff of K-Means performance is that K-Means constructed more risky investments, while Ward did not. Therefore, Ward can still be used for investors with lower risk tolerance.

6 Conclusion

This report has demonstrated that the unsupervised machine learning algorithm clustering is a suitable technique to construct well performing portfolios. Partitioning algorithms such as K-Means are able to form dense, well separated clusters and find high preforming stocks but at the cost of increase risk. Hierarchical clustering algorithms such as Ward's method performance suffer compared to K-Means since the structure of the underlying data is not a tree. Despite this, Ward's method and the process was still able to produce stocks that can outperform the S&P 500 and has less risk. Both methods can be attractive to a large range of investors since every investor is different.

The methods and process of constructing portfolios shown in the report can easily be expanded to take into account investor's preferences such as filtering stocks based on their sector, industry, region, and many other metrics. In addition, the technique provided in the report is the static and can me made dynamic by implementing a system that use our technique with the addition of some maintenance feature that can enter and exit positions.

Bibliography

[1] Ruxandra Vulpoiu. *Diversified stock allocation through clustering*, pages 35–61. 2018.

[2] Karina Marvin Adviser and S. Bhatt. Creating diversified portfolios using cluster analysis. 2015.

[3] Cosma Shalizi. Distances between clustering, hierarchical clustering, September 2009.

[4] Diego León, Arbey Aragón, Javier Sandoval, Germán Hernández, Andrés Arévalo, and Jaime Niño. Clustering algorithms for risk-adjusted portfolio construction. *Procedia Computer Science*, 108:1334–1343, 2017. International Conference on Computational Science, ICCS 2017, 12-14 June 2017, Zurich, Switzerland.

[5] S.R. Nanda, B. Mahanty, and M.K. Tiwari. Clustering indian stock market data for portfolio management. *Expert Systems with Applications*, 37(12):8793–8798, 2010.

[6] Jinwoo Park. Clustering approaches for global minimum variance portfolio, 2020.

[7] Bettina Grün. Model-based clustering, 2018.