**School of Business, Economics and Information Systems**

Chair of Financial Data Analytics

---

Seminar in Machine Learning in Finance

Seminar Paper

# Risk Exposure of S&P 500 Companies before and after COVID-19: Cluster Analysis with Agglomerative Hierarchical Clustering, HDBSCAN and K-means

Supervisors: Prof. Dr. Ralf Kellner, Lukas Reichmann

Edited by: Aleksandra Petrenko

Matriculation number: 108448

Course of studies: M. Sc. Business Administration

Semester: 4th

E-Mail: petren01@ads.uni-passau.de

Passau, submission date: 21.07.2023

# Introduction and literature overview

Fama & French (1993) have developed a three-factor model to describe and explain the stock prices and portfolios. They estimate a following regression model:

$$r - R_f = \beta_{Mkt-R_f}(Mkt - R_f) + \beta_{SMB} \times SMB + \beta_{HML} \times HML + \alpha,$$

where Mkt stands for the market portfolio return, $R_f$ for risk-free return rate, SMB ("Small Minus Big") for the historic excess returns of small caps over big caps and HML ("High Minus Low") for the historic excess returns of value stocks over growth stocks.

Positive beta-loadings imply:

- Mkt-RF: A positive relationship with a market premium, moving with the market
- SMB: A positive relationship with a size premium, small company size
- HML: A positive relationship with a value premium, a company with a high book-to-market value (exposure to value stocks)

Coronavirus pandemic and followed financial crisis has changed many industries. Crisis' impact can be seen in the change of risk exposure. There are several studies where the behavior of beta-coefficients was explored.

For example, Hou & Chen (2021) study the American steel industry before and after COVID-19 using a five-factor Fama-French model. Among other coefficients, the coefficient of market risk decreased, however, SMB and HML were significant, and the change of epidemic situation was not significant. In general, the influence of the COVID-19 outbreak on the US steel industry was dramatic and led to a huge regress in the whole industry. Lim et al. (2014) examines equity price data and the relationship between three factors of systematic risk in a Fama-French model and VIX, an indicator of total risk. Their findings confirm the predicted relationship between the equity risk-premium and risk; moreover, they discover that the size-premium is driven by investors who are flying-to-quality and that investors became increasingly sensitive to changes in the VIX during the global crisis. Sun (2021) discovers the performance of Fama-French five-factor model in US market before and after COVID-19. The research finds that the efficiency of Fama-French five-factor model has increased after the crisis in all industries, while an unexplained factor behavior increased. The pandemic caused change of Fama-French factors

for most of industries respectively and has strong influence on portfolio performance and factor exposure.

# Empirical analysis

## Fama & French (1993) three factor model evaluation

The empirical analysis is conducted using Python. We use stock prices data of S&P 500 companies from 2018-01-02 to 2022-12-30 and calculate discrete returns. S&P 500 contains the stocks of the largest companies listed on stock exchanges in the USA. We also import Fama-French factors for their three-factor model (Fama & French, 1993) for these dates: Mkt-$R_f$, SMB, HML and $R_f$. Then we divide the dataset into two periods: before the COVID-19 financial crisis (from 2018-01-02 till 2020-01-31) and after the crisis (from 2020-05-01 till 2022-12-30).

After that we estimate Fama-French following regression models for each company and for both datasets:

The distribution of beta-coefficients is presented in the Figures 1 and 2:

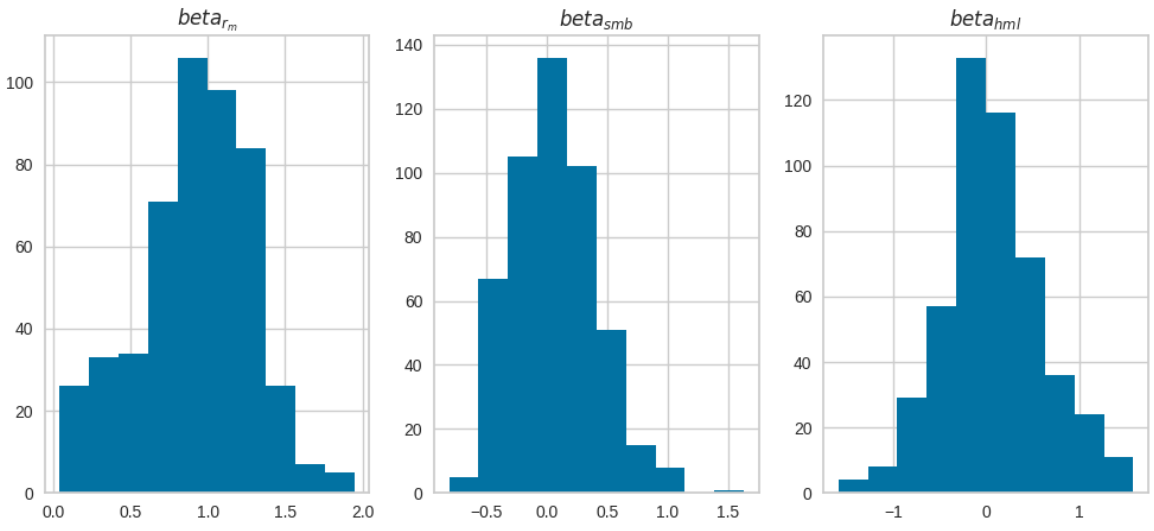


**Figure 1: The distribution of beta-coefficients for the Fama-French model before COVID-19**

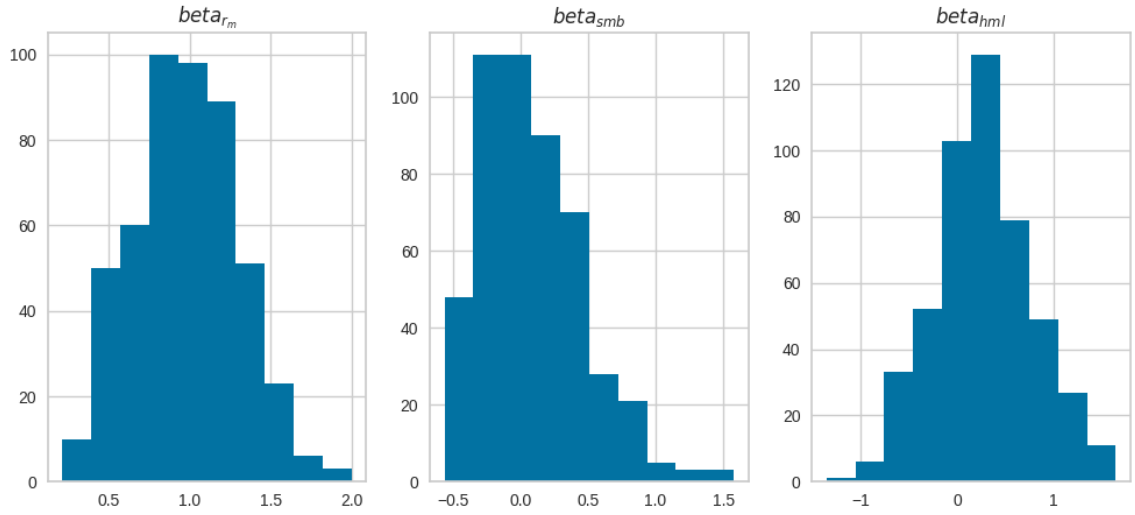


**Figure 2: The distribution of beta-coefficients for the Fama-French model after COVID-19**

As we see, all the distributions are leptokurtic. The distribution of the $\beta_{Mkt-R_f}$ has not changed that much: the mode has decreased from ~0.9 to ~0.75, both distributions are positively skewed. The $\beta_{SMB}$ distribution has become more positively skewed, the mode has changed from ~0.1 to ~ -0.2, the kurtosis has decreased. For the $\beta_{HML}$, the mode has increased from ~-0.2 to ~0.3, but the structure of the distribution has remained similar to the one before COVID-19. To summarize, the distributions of beta-coefficients are slightly different, therefore, further analysis should be conducted to determine how and why the coefficients changed.

For further analysis, beta-coefficients were standardized to remove the effects of different scales.

# Cluster analysis

## Overview of clustering algorithms

For this paper, three following clustering methods were chosen to conduct the analysis. To explain how clustering algorithms work, we introduce a few definitions.

A distance between any pair of vectors or points $i, j, k$ satisfies following properties: Symmetry, $d(i,j) = d(j,i)$ ; positive definiteness, $d(i,j) > 0$; null condition, $d(i,j) = 0 \leftrightarrow i = j$; and

the triangular inequality, $d(i,j) \leq d(i,k) + d(k,j)$. The first three properties embody a dissimilarity. The Euclidean distance is defined as:

$$d(\boldsymbol{x}_a, \boldsymbol{x}_b) = \|\boldsymbol{x}_a - \boldsymbol{x}_b\|_2 = \sqrt{\sum_{i=1}^{n} (\boldsymbol{x}_{i,a} - \boldsymbol{x}_{i,b})^2}.$$

## Agglomerative hierarchical clustering (Ward's linkage)

In general, the Ward's method (Ward Jr., 1963) in agglomerative hierarchical clustering begins with one-point-clusters and merges them step by step by minimizing the within-variance of the clusters being merged. The Ward's method joins two clusters that minimize the increase in the sum of squared errors after merging. Practically, Ward's method says that the distance between two clusters, A and B, is how much the sum of squares will increase, when we merge them, defined as merging cost:

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

Ward's clustering method can be described using the recurring Lance-Williams dissimilarity update formula. If points $i$ and $j$ are agglomerated into cluster $i \cup j$, the new dissimilarity between the cluster and all other points is:

$$d(i \cup j, k) = \alpha_i d(i,k) + \alpha_j d(j,k) + \beta d(i,j) + \gamma |d(i,k) - d(j,k)|,$$

where $\alpha_i, \alpha_j, \beta, \gamma$ define the agglomerative criterion. For the Ward method:

$$\alpha_i = \frac{|i| + |k|}{|i| + |j| + |k|},$$

$$\alpha_j = \frac{|j| + |k|}{|i| + |j| + |k|},$$

$$\beta = -\frac{|k|}{|i| + |j| + |k|},$$

$$\gamma = 0.$$

To summarize, the key steps of the agglomerative clustering (Ward's linkage) are:

1. Make each data point a single-point cluster.
2. Merge two clusters, if they minimize the increase in the sum of squared errors after merging.
3. Repeat step 2 until there the predefined number of clusters is reached.

There are following drawbacks of the Ward's agglomerative clustering:

- Optimal number of clusters: It is hard to define the optimal number of clusters, moreover, the dendrograms are not suitable for large datasets.
- Computational intensiveness: Ward's method is computationally more complex than other hierarchical clustering methods like as single-linkage or complete-linkage clustering.
- Less than optimal clusters: It often converges to local minima, however, the resulting clusters are good enough.
- Difficulties when handling with different sizes of clusters: It does not perform well for clusters of unequal diameters.
- Difficulties with ellipsoidal clusters: Ward's method often leads to misclassifications when the clusters are distinctly ellipsoidal rather than spherical, that is, when the variables are correlated within a cluster.
- Sensitivity to initial conditions: Ward's method is sensitive to the initial order of the observations, and small changes in the order can lead to different results.

Despite of some disadvantages, the Ward's linkage is known as a progressive technique for clustering. (Murtagh & Legendre, 2014; Murtagh & Conteras, 2012)

## K-means clustering (Lloyd's algorithm)

Let $\boldsymbol{X} \in \mathbb{R}^{d \times n}$ denote a data matrix, $\boldsymbol{\Theta} \in \mathbb{R}^{d \times k}$ the centre matrix and $C_j$ the membership set for cluster $j$. Initially the Lloyd's algorithm chooses $k$ starting points, where $k$ is a predefined number of clusters. At every iteration, it assigns each data point $\boldsymbol{x}_i$ to the cluster $C_j$, minimizing the Euclidean distance $\|\boldsymbol{x}_i - \boldsymbol{\theta}_j\|$. After that, the algorithm redefines the centroid $\boldsymbol{\theta}_j$ by averaging the $\boldsymbol{x}_i$ within the cluster $C_j$. The iterations are repeated with new cluster centres until

consistency is achieved and the clusters do not change with every new iteration. The k-means clustering objective is:

$$f_{-\infty}(\boldsymbol{\Theta}) = \sum_{j=1}^{k} \sum_{i \in C_j} \left\| \boldsymbol{x}_i - \boldsymbol{\theta}_j \right\|^2 = \sum_{i=1}^{n} min_{1 \le j \le k} \left\| \boldsymbol{x}_i - \boldsymbol{\theta}_j \right\|^2.$$

To summarize, the key steps of the k-means algorithm are:

1. Randomly choose $k$ points as cluster centres.
2. Allocate the object to the cluster with the nearest centre according to the Euclidean distance function.
3. Determine the "centroid" (mean) in each cluster.
4. Replicate the 2nd and the 3rd steps until the same objects are allocated to each cluster in consecutive rounds.

The main drawbacks of the algorithm are:

- Assumes spherical density: The algorithm does not perform well for clusters with a non-spherical shape.
- Initial centroid: The clustering is extremely sensitive to the initial centroids.
- Noise: Noise or outliers deteriorates the quality of the clustering result.
- Number of clusters: The number of clusters must be determined before the means clustering begins.
- Local minima: It always converges to local minima.
- Inability to cluster non-linearly separable dataset: It fails to split non-linearly separable datasets in the input space.

As soon as the algorithm is notorious to be sensitive to starting points, there are ways to overcome it. One of them is that the whole procedure is repeated $m$ predefined times with various sets of starting points. At the end, the classification with the lowest sum of variances is chosen. Another way is to choose the cluster centres in a notable manner. (Ahmed et al. 2020; Aldahdooh & Ashour, 2013; Ashabi et al., 2020; Xu & Lange, 2019)

## HDBSCAN

The HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise) algorithm (McInnes et al., 2017) is a density-based algorithm. It is an improvement of the existing DBSCAN clustering method with establishing a hierarchical representation of clusters. In contrast to k-means, there is no need to assign every data point to a cluster as it identifies dense clusters. The values which do not belong to any cluster are considered outliers or noise.

There are two input parameters for HDBSCAN: minimum number of points $s$ to be used in the distance calculation and minimum cluster size $p$ that defines a lower bound on the number of points in a cluster.

A first step is to compute the core distances $d_{core}(x_i)$ for each point in the data to obtain an approximate density. The core distance the distance to the $k^{\text{th}}$ nearest neighbour, the value of $k$ includes the point itself. Using the core distances, a new distance metric mutual reachability distance is computed:

$$d_{reach}(x_i, x_j) = \max\{d_{core}(x_i);\ d_{core}(x_j);\ d(x_i, x_j)\}.$$

Next, the mutual reachability distances can be used to construct a weighted graph, where the data points are nodes and an edge between any two points has the weight of the mutual reachability distance of those points. After that, the graph's minimum spanning tree (with the least sum of weights) is computed and then modified by adding a self-edge to each vertex with the point's core distance as the weight – this is called the extended minimum spanning tree.

Thereafter, the HDBSCAN hierarchy is built according to the following algorithm:

1. Assign all points to one cluster label and add it to the list.
2. Sort the graph by edge weight in ascending order.
3. Starting from the bottom of the graph, remove the edges. The weight value of the edge(s) being removed is used to denote the current hierarchical level. Edges with equal weights are removed simultaneously.
4. Cluster split: Explore the cluster containing the removed edge. If the cluster becomes disconnected and is smaller $s$, assign it to noise, otherwise to a new cluster, which is added to the list of clusters. With a new cluster new hierarchy level is created.
5. At the last level of hierarchy assign all the points to noise.

The next step is to identify prominent clusters from the hierarchy. A new measure $\lambda = \frac{1}{edge\ weight}$ as well as $\lambda_{birth}$ (the $\lambda$-value when a new cluster was created from a cluster split),

$\lambda_{death}$ (the $\lambda$-value when that same cluster was itself split) and $\lambda_p$ (for each point, the $\lambda$-value at which that point dropped out of the cluster) are introduced. The stability of a cluster is computed as follows:

$$stability = \sum_{p \epsilon Cluster} (\lambda_p - \lambda_{birth}).$$

The stability needs to be propagated through the clusters. Leaf clusters are clusters with no children, and they can be identified from the list of clusters. Starting with the leaf clusters, go up using the reference to the ancestors. Leaf clusters propagate their stability to their parents and add themselves as a propagated descendent in the parent cluster. For non-leaf clusters, if the cluster being processed has a higher stability than the cumulative stability of its descendants, it alone will be propagated to the parent cluster, otherwise, the cumulative stability of all the current cluster's descendants will be propagated to the parent cluster. No propagation occurs for the root cluster since it has no parent. When this process is finished, the root cluster will contain the references to descendent clusters with the highest stabilities, which are called the most prominent clusters. Using the HDBSCAN hierarchy together with the details of the most prominent clusters, the list of cluster assignments for each point is generated.

Moreover, the outlier scores GLOSH for each point can be computed:

$$GLOSH(x) = 1 - \frac{\epsilon_{max}(x)}{\epsilon(x)},$$

where $\epsilon(x)$ is the weight value of the point right before it was marked as noise and $\epsilon_{max}(x)$ is the lowest propagated child death level from its last labelled cluster.

To summarize, the main steps of the HDBSCAN algorithm are:

1. Compute the core distance for the $k$ nearest neighbours for all points.
2. Compute the extended minimum spanning tree from a weighted graph, where the mutual reachability distances are the edges.
3. Build the HDBSCAN hierarchy from the extended minimum spanning tree.
4. Find the prominent clusters from the hierarchy.
5. Calculate the outlier scores GLOSH.

The advantages of the algorithm are:

- Identification of clusters of varying density.
- Identification of outlier points.
- Discovering non-linear dependences.

The main disadvantage is computational complexity: $O(n^2)$, compared to k-means with $O(n)$. (Stewart & Al-Khassaweneh, 2022; McInnes et al., 2017)

# Optimal number of clusters

First, we must identify the optimal number of clusters for hierarchical and k-means clustering before and after COVID-19. HDBSCAN finds the number of clusters by itself. Three metrics were chosen for identification of optimal number of clusters: Distortion Score (Elbow method), Silhouette Coefficient and Calinski-Harabasz Index.

## Distortion Score (Elbow method)

A distortion score is the average of the squared distances from the cluster centres of the respective clusters to each data point. The scores are calculated iteratively for different numbers of clusters and are plotted against them. The first clusters will minimize the distances but at some point the minimization becomes lower and gives a "kink" in the graph, which is the optimal number of clusters. (Syakur et al., 2018) Below are the results for our datasets.

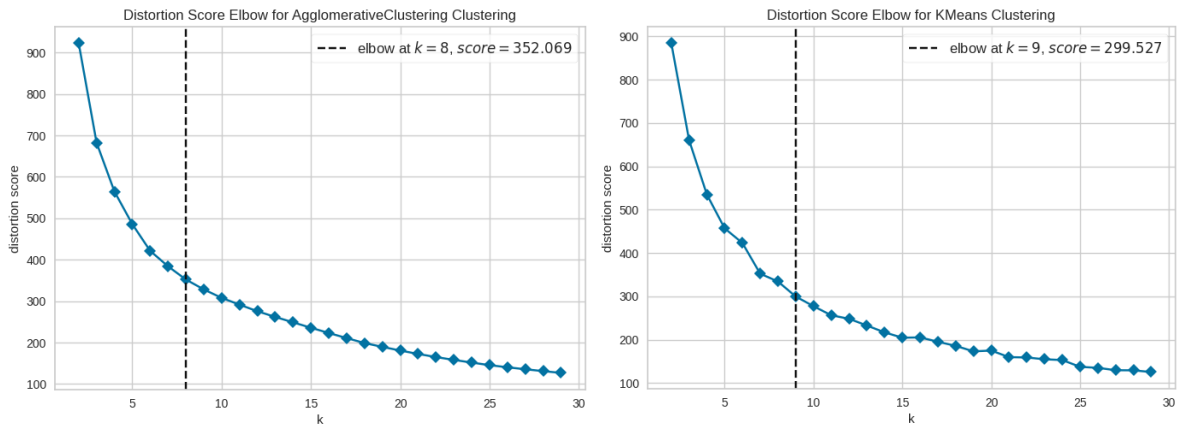


**Figure 3. The Distortion Score for hierarchical clustering and k-means before COVID-19.**

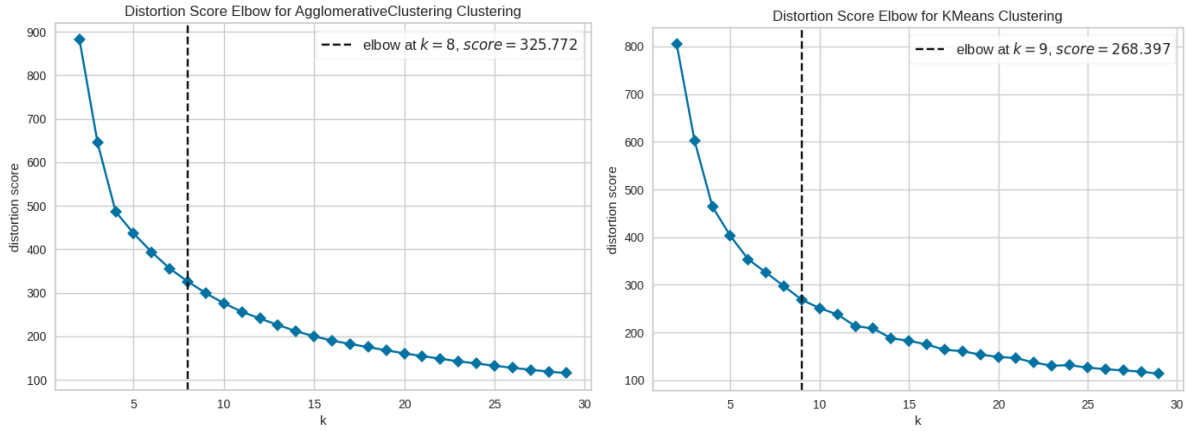


**Figure 4. The Distortion Score for hierarchical clustering and k-means after COVID-19.**

## Silhouette Coefficient

Let us compute a silhouette coefficient (Rousseeuw, 1987). Let there be a cluster $C_l$, $x_i$ is a point in $C_l$ and $a(x_i)$ is the average distance of the $x_i$ to all other members of the same cluster $C_l$. Let $C_{l'}$ be a cluster other than $C_l$, then $d(x_i, C_{l'})$ is the average distance of the $x_i$ to all members of $C_{l'}$. Compute $d(x_i, C_{l'})$ for all clusters except for $C_l$ and let $b(x_i) = min_{C_{l'} \neq C_l} d(x_i, C_{l'})$ . Then the Silhouette Value, which takes values from -1 to 1, is equal to:

$$s(x_i) = \frac{b(x_i) - a(x_i)}{\max\{a(x_i), b(x_i)\}}.$$

A large positive value indicates that the within similarity $a(x_i)$ is much smaller than the smallest between dissimilarity $b(x_i)$, so $x_i$ is well-clustered, and, vice versa, a large negative value stands for poor clustering. A Silhouette Coefficient is simply the average of silhouette values for each point. The number of clusters maximizing the Silhouette Coefficient is the optimal.

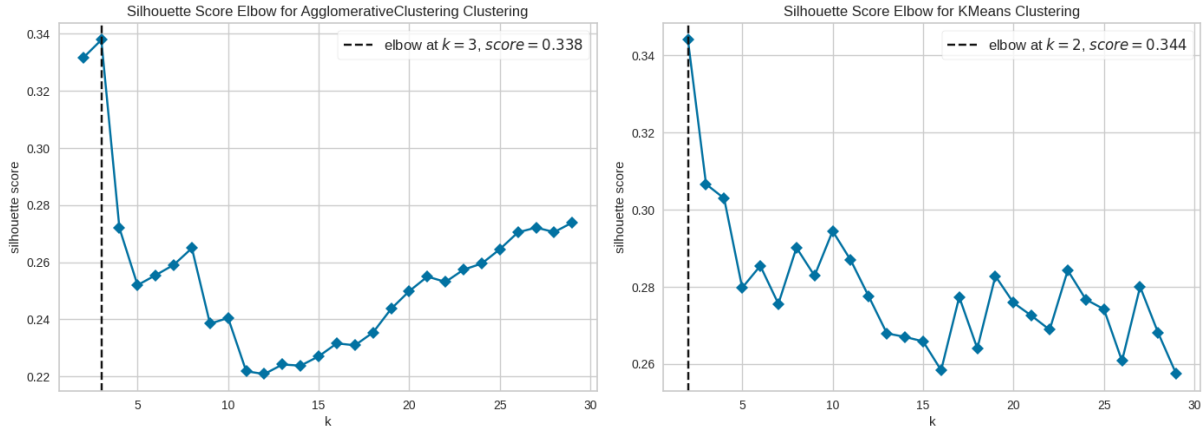


**Figure 5. The Silhouette Coefficient for hierarchical clustering and k-means before COVID-19.**

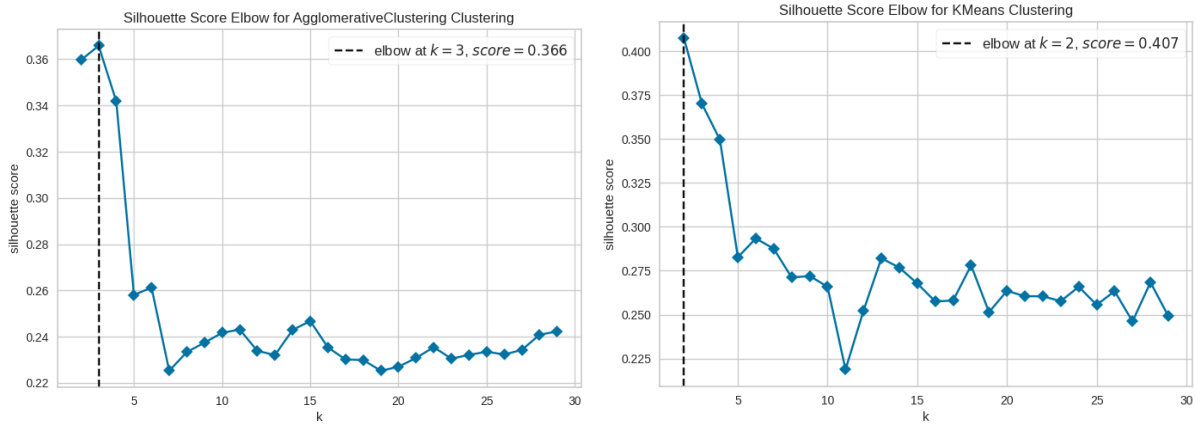


**Figure 6. The Silhouette Coefficient for hierarchical clustering and k-means after COVID-19.**

## Calinski-Harabasz Index

The Calinski-Harabasz index (Caliński & Harabasz, 1974) is expressed as a ratio of between-cluster variance and the overall within-cluster variance:

$$I_{CH} = \frac{N - C}{C - 1} \frac{\sum_{i=1}^{C} d(u_i, U)}{\sum_{i=1}^{C} \sum_{x_j \in CL_i} d(x_j, u_i)},$$

where $N$ is the number of features, $C$ is the number of clusters, $u_i$ is a centroid for each non-empty cluster $CL_i$, $U$ is the centre of gravity of the whole dataset. The larger the numerator, the higher the dispersion between clusters is. The smaller the denominator is, the closer the

relationship is in the cluster. Therefore, the number of clusters maximizing the index is optimal.
(

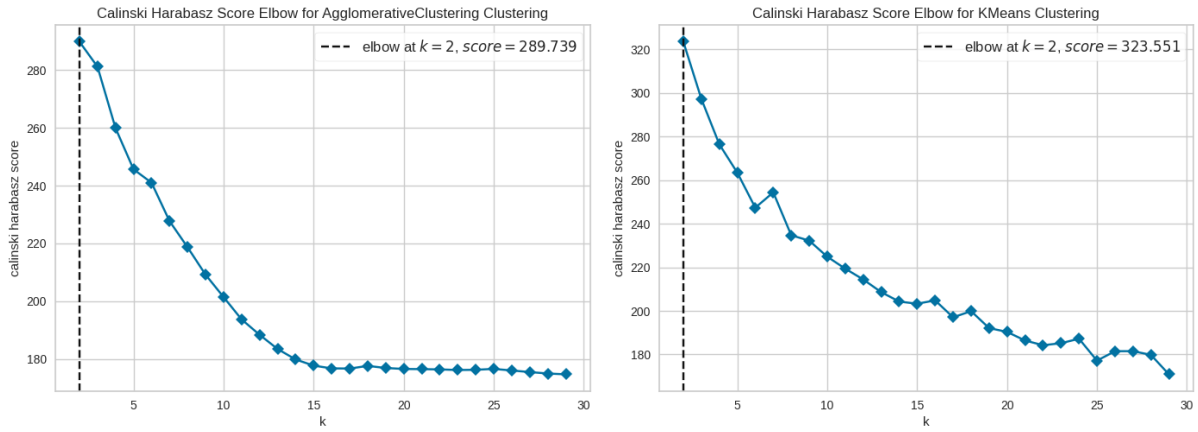


**Figure 7. The Calinski-Harabasz Index for hierarchical clustering and k-means before COVID-19.**

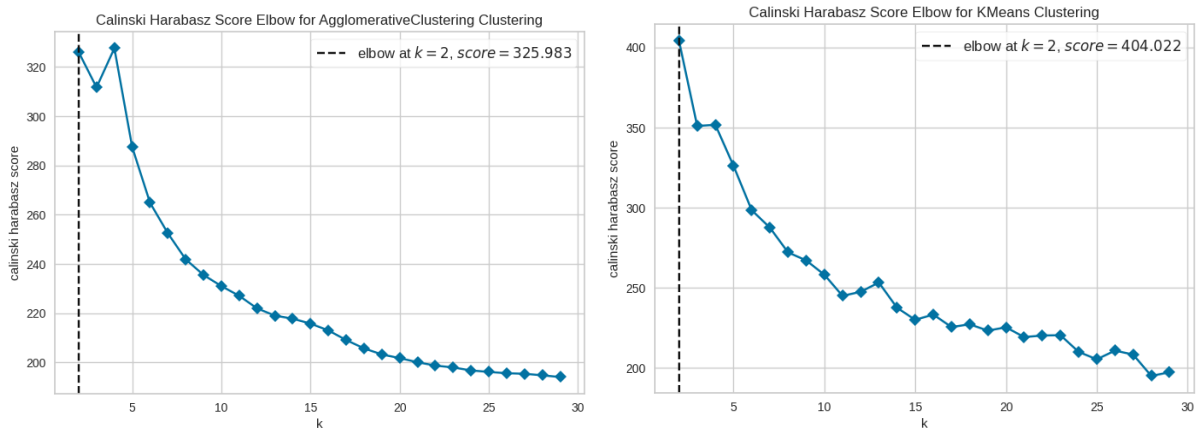


**Figure 8. The Calinski-Harabasz Index for hierarchical clustering and k-means after COVID-19.**

## Analysis of the number of clusters

As we see, there is a discrepancy between different methods. The reason could be that the clusters are not clearly formed in the data. Moreover, the Distortion Score considers only the tightness of the cluster: the closer the points to the cluster centre are, the lower is the distortion. In contrast, the Silhouette Coefficient and the Calinski-Harabasz Index are better when the clusters are tighter and farther from each other. Therefore, we will follow the values of the Silhouette Coefficient and the Calinski-Harabasz Index. The Silhouette Score shows that the

the optimal number of clusters for hierarchical clustering is 3, both before and after COVID, however, the values for 2 clusters are also high. For k-means, the optimal numbers of clusters are equal to 2. The Calinski-Harabasz Index shows that the optimal number of clusters for all methods and periods is 2. To enable better comparison of methods, we finally choose $k = 2$ for all methods and periods.

# Clustering

The results for three clustering methods before and after COVID-19 are presented below: mean coefficients and standard deviation for each cluster as well as the visualization of clustering.

## Before COVID-19

| | mean | | | std | | |
|---|---|---|---|---|---|---|
| | Beta HML | Beta Mkt-RF | Beta SMB | Beta HML | Beta Mkt-RF | Beta SMB |
| Hier | | | | | | |
| 0 | -0.226823 | 0.771323 | -0.102258 | 0.368279 | 0.352117 | 0.292305 |
| 1 | 0.550136 | 1.171352 | 0.315973 | 0.409519 | 0.227364 | 0.271088 |

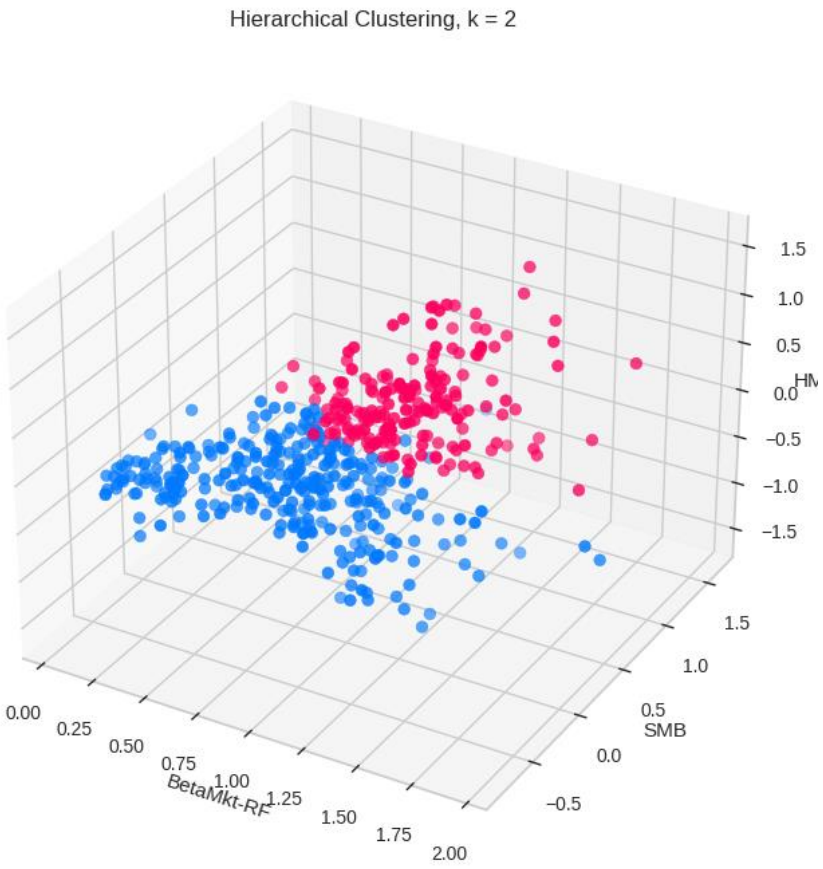


**Figure 9. The results of clustering for hierarchical clustering.**

| | mean | | | std | | |
|---|---|---|---|---|---|---|
| | Beta HML | Beta Mkt-RF | Beta SMB | Beta HML | Beta Mkt-RF | Beta SMB |
| K-means | | | | | | |
| 0 | -0.215188 | 0.706846 | -0.165735 | 0.352920 | 0.316850 | 0.232021 |
| 1 | 0.399189 | 1.172643 | 0.312912 | 0.528532 | 0.238325 | 0.279142 |

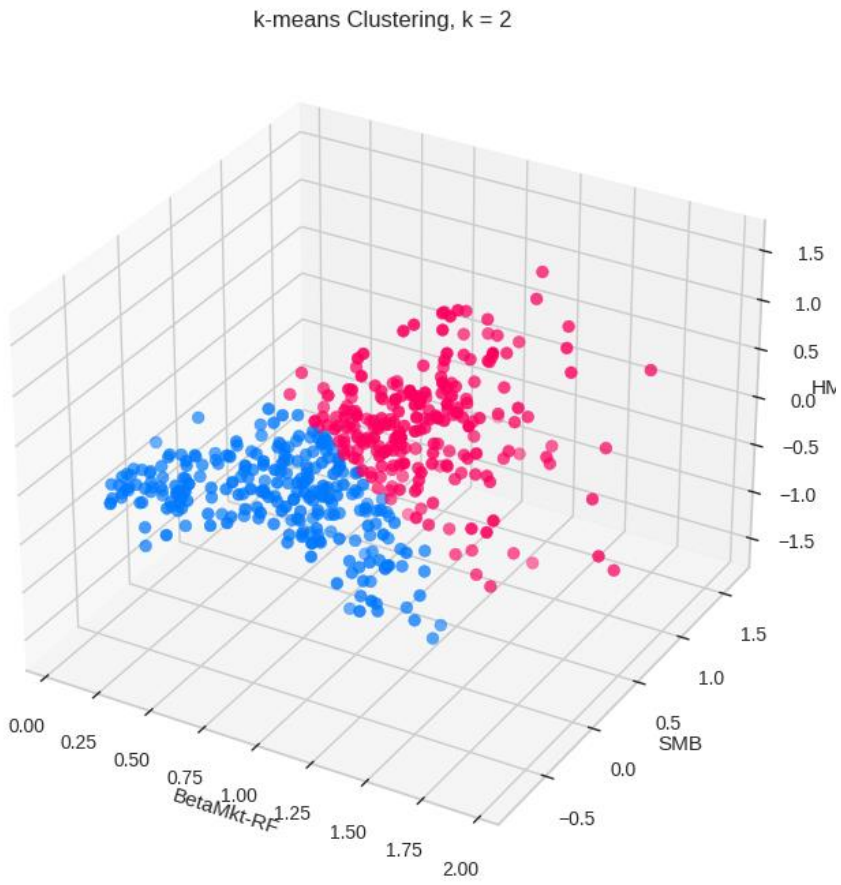


**Figure 10. The results of clustering for k-means clustering.**

| | mean | | | std | | |
|---|---|---|---|---|---|---|
| | Beta HML | Beta Mkt-RF | Beta SMB | Beta HML | Beta Mkt-RF | Beta SMB |
| HDBSCAN | | | | | | |
| -1 | 0.154085 | 1.142366 | 0.214861 | 0.813529 | 0.376473 | 0.436504 |
| 0 | 1.017085 | 1.320570 | -0.022223 | 0.100035 | 0.077982 | 0.066096 |
| 1 | 0.018322 | 0.835848 | 0.005306 | 0.365897 | 0.324311 | 0.298305 |

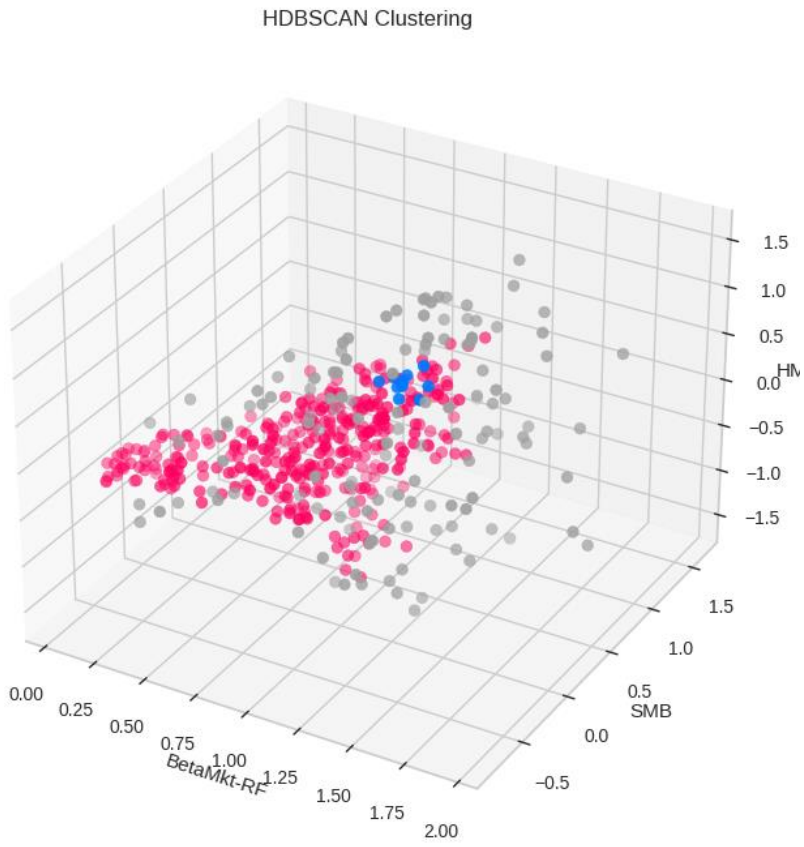


**Figure 11. The results of clustering for HDBSCAN clustering.**

## After COVID-19

| | mean | | | std | | |
|---|---|---|---|---|---|---|
| | Beta HML | Beta Mkt-RF | Beta SMB | Beta HML | Beta Mkt-RF | Beta SMB |
| Hier | | | | | | |
| 0 | -0.039617 | 0.840270 | -0.129949 | 0.340607 | 0.300376 | 0.268636 |
| 1 | 0.694714 | 1.162713 | 0.367665 | 0.370050 | 0.240040 | 0.298422 |

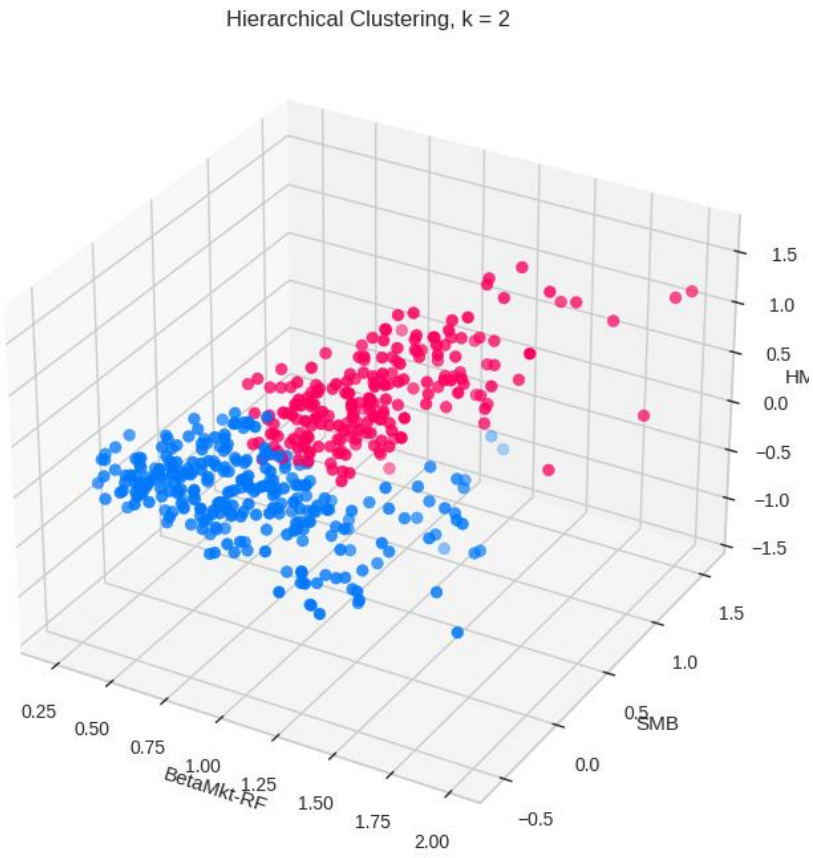


**Figure 12. The results of clustering for hierarchical clustering.**

| | mean | | | std | | |
|---|---|---|---|---|---|---|
| | Beta HML | Beta Mkt-RF | Beta SMB | Beta HML | Beta Mkt-RF | Beta SMB |
| K-means | | | | | | |
| 0 | 0.642948 | 1.240775 | 0.440338 | 0.510623 | 0.222736 | 0.302960 |
| 1 | 0.041000 | 0.810170 | -0.145340 | 0.336111 | 0.249588 | 0.191138 |

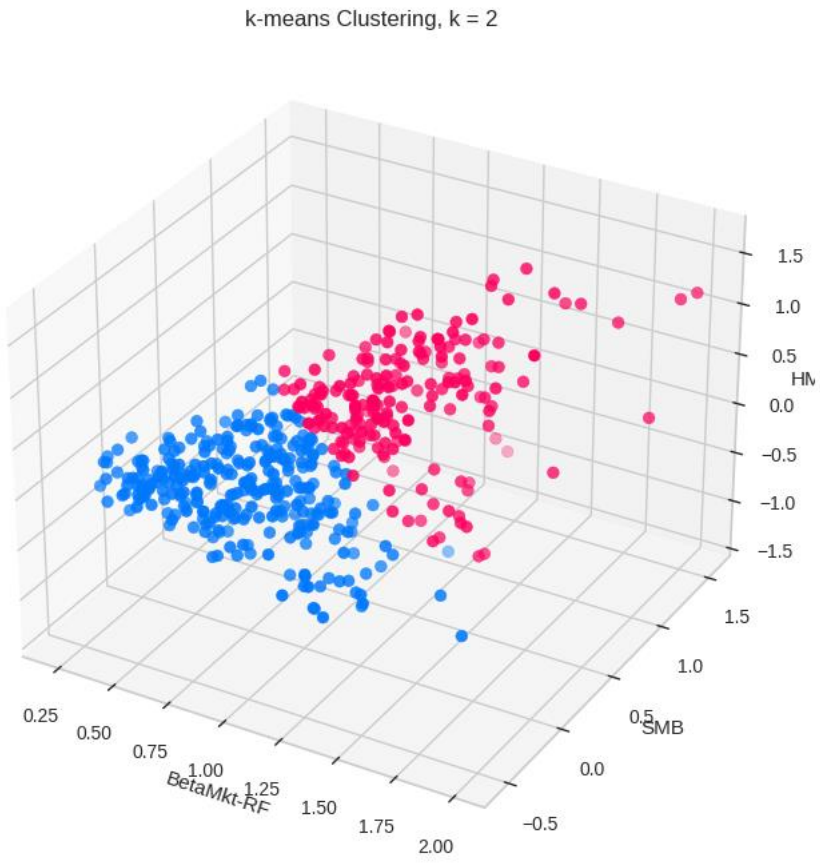


**Figure 13. The results of clustering for k-means clustering.**

| | mean | | | std | | |
|---|---|---|---|---|---|---|
| | Beta HML | Beta Mkt-RF | Beta SMB | Beta HML | Beta Mkt-RF | Beta SMB |
| HDBSCAN | | | | | | |
| -1 | 0.162891 | 1.283918 | 0.448601 | 0.758522 | 0.361862 | 0.477299 |
| 0 | 0.323732 | 0.907221 | 0.011233 | 0.413782 | 0.265722 | 0.296482 |
| 1 | -0.606215 | 1.216026 | -0.181668 | 0.115400 | 0.051216 | 0.132399 |

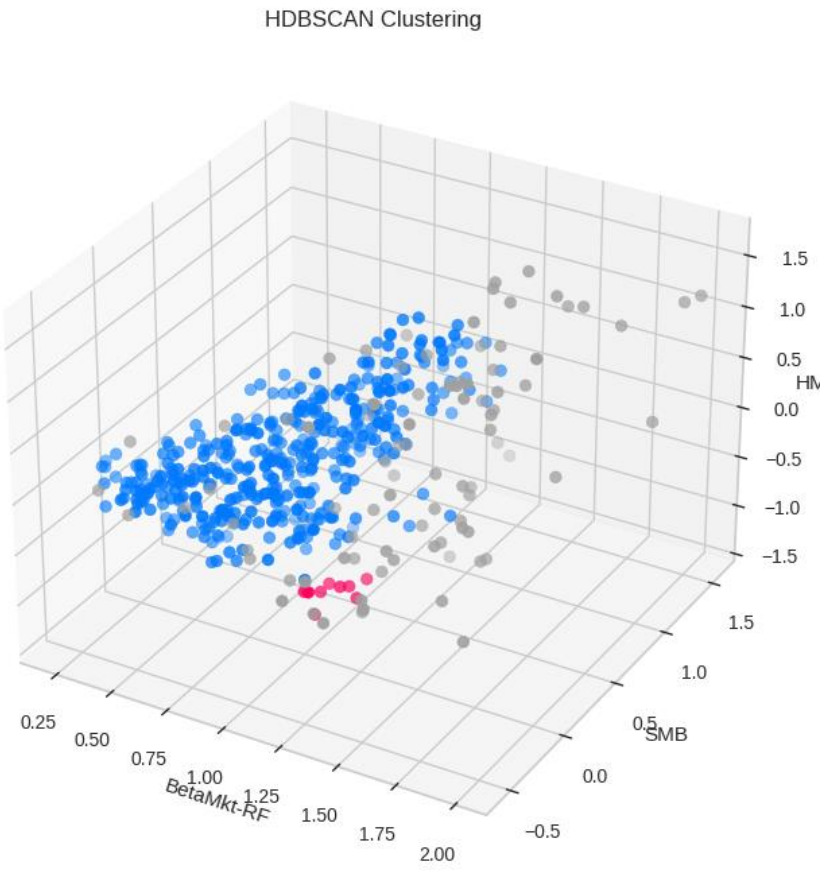


**Figure 14. The results of clustering for HDBSCAN clustering.**

# Comparison of clusters

## Cluster similarities

At the first glance, the distribution of clusters for k-means and hierarchical clustering are similar, whereas the structure of HDBSCAN clusters differs. However, the number of clusters of HDBSCAN is also 2 (excluding the noise which is depicted in grey). The clusters are approximately of the same growth for hierarchical clustering, HDBSCAN produced one large and one very small cluster.

To prove our guesses, we compute the adjusted rand index, which shows similarities between two clustering algorithms. Let us consider each pair of objects and construct a contingency table for comparing partitions $U$ and $V$:

| Partition | $V$ | |
|---|---|---|
| $U$ | Pair in the same group | Pair in different groups |
| Pair in the same group | $a$ | $b$ |
| Pair in different groups | $c$ | $d$ |

**Table 1. Contingency table for clustering $U$ and $V$**

Then the adjusted rand index is:

$$ARI = \frac{\binom{n}{2}(a+d) - ((a+b)(a+c)+(c+d)(b+d))}{\binom{n}{2}^2 - ((a+b)(a+c)+(c+d)(b+d))},$$

where the number of pair in the dataset equals to:

$$\binom{n}{2} = \frac{n!}{(n-k)!\,k!}$$

The adjusted rand index overcomes the problems of the rand index the fact that the expected value of the rand index of two random clusterings does not take a constant value or that the Rand statistic approaches its upper limit of unity as the number of clusters increases. (Santos & Embrechts, 2009) The closer the score to 1, the better is the classification, 0 implies completely different clustering.

As we see in the Table 2, the guesses are proven. K-means and hierarchical clustering are similar within one period, however, not perfectly. HDBSCAN provides clustering different from k-means and hierarchical clustering. Moreover, if we compare clusterings before and after COVID-19, we see that the clusters have changed much, even within one method. The most similar are clusterings within the methods and between k-means and hierarchical clustering.

| | **Before COVID-19** | | | **After COVID-19** | | | **Before vs. after COVID-19** | | |
|---|---|---|---|---|---|---|---|---|---|
| | **K-means** | **HDBSCAN** | **Hier** | **K-means** | **HDBSCAN** | **Hier** | **K-means** | **HDBSCAN** | **Hier** |
| **K-means** | 1.0 | 0.0463 | 0.6393 | 1.0 | 0.0799 | 0.5628 | 0.3022 | 0.0379 | 0.2546 |
| **HDBSCAN** | 0.0463 | 1.0 | 0.0631 | 0.0799 | 1.0 | 0.0009 | 0.0929 | 0.1882 | 0.0203 |
| **Hier** | 0.6393 | 0.0631 | 1.0 | 0.5628 | 0.0009 | 1.0 | 0.3824 | 0.0373 | 0.3582 |

**Table 2. Adjusted rand score for different clusterings.**

What is more, we construct intersection tables, where we can see the number of companies belonging to a certain cluster before and after COVID-19 within one clustering method. It will help to identify the number of changers and find similar clusters.

| **Clusters before COVID-19** | **Clusters after COVID-19** | | | |
|---|---|---|---|---|
| **Hier / K-means** | **Hier** | | **K-means** | |
| **Cluster number** | **0** | **1** | **0** | **1** |
| **0** | 240 | 60 | 222 | 35 |
| **1** | 38 | 152 | 75 | 158 |

**Table 3. Intersection tables with unique tickers before and after COVID-19 within a clustering method for hierarchical and k-means clustering.**

As it can be seen, the clusters 0 and 0, 1 and 1 for both k-means and hierarchical methods are similar as soon as they have many common companies, that is why we can match them. The analysis of changers will follow later.

| **Before COVID-19** | **After COVID-19** | | |
|---|---|---|---|
| | **-1**[1] | **0** | **1** |
| **-1** | 49 | 75 | 5 |
| 0 | 1 | 9 | 0 |
| 1 | 37 | 310 | 4 |

**Table 4. Intersection tables with unique tickers before and after COVID-19 within a clustering method for HDBSCAN clustering.**

For HDBSCAN, the structure is a bit more complicated. The small cluster 0 before COVID-19 has vanished in a large cluster 0 after COVID-19. However, a small new cluster 1 was formed

[1] -1 stands for noise and outliers.

after COVID-19. Clusters 1 before COVID-19 and 0 after COVID-19 are the most similar. A half of the noise cluster has changed after COVID-19.

## Descriptive statistics comparison

Let us compare mean betas (centroids) and standard deviations for the clusters before and after COVID-19. For hierarchical clustering (Figure 15) we have following observations:

- Cluster 0 before COVID-19 consists of growth stocks (negative Beta-HML) of large companies (negative Beta-SMB), moving with the market (large positive Mkt-RF). After COVID-19, beta-HML for the cluster 0 has become less negative, from -0.23 to -0.04: The companies in this cluster still have higher market-to-book ratios and are considered growth stocks, but there is a development towards value stocks. Beta Mkt-RF has grown from 0.77 to 0.84, which depicts a larger movement with the market. Beta-SMB, the growth of companies, has remained the same.
- Cluster 1 before COVID-19 consists of value stocks with low market-to-book ratios of small companies, strongly moving with the market. After COVID-19, beta-HML has grown from 0.55 to 0.69, so the effect of behaving like value stocks has increased. Movement with the market remained the same. Beta-SMB has grown a bit from 0.32 to 0.37, so the stocks still behave like small companies' stocks.
- The standard deviations remained similar.

| | mean | | | std | | |
|---|---|---|---|---|---|---|
| | Beta HML | Beta Mkt-RF | Beta SMB | Beta HML | Beta Mkt-RF | Beta SMB |
| Hier | | | | | | |
| 0 | -0.226823 | 0.771323 | -0.102258 | 0.368279 | 0.352117 | 0.292305 |
| 1 | 0.550136 | 1.171352 | 0.315973 | 0.409519 | 0.227364 | 0.271088 |

| | mean | | | std | | |
|---|---|---|---|---|---|---|
| | Beta HML | Beta Mkt-RF | Beta SMB | Beta HML | Beta Mkt-RF | Beta SMB |
| Hier | | | | | | |
| 0 | -0.039617 | 0.840270 | -0.129949 | 0.340607 | 0.300376 | 0.268636 |
| 1 | 0.694714 | 1.162713 | 0.367665 | 0.370050 | 0.240040 | 0.298422 |

**Figure 15. Descriptive statistics of the hierarchical clustering before (above) and after (below) COVID-19.**

Observations for k-means (Figure 16):

- Clusters before COVID-19 as well as after COVID-19 look very similar to the clusters of hierarchical clustering.
- Standard deviations remained the same within the method. Standard deviations except for cluster 1 beta-HML are comparable with those of k-means.

| | mean | | | std | | |
|---|---|---|---|---|---|---|
| | Beta HML | Beta Mkt-RF | Beta SMB | Beta HML | Beta Mkt-RF | Beta SMB |
| K-means | | | | | | |
| 0 | -0.215188 | 0.706846 | -0.165735 | 0.352920 | 0.316850 | 0.232021 |
| 1 | 0.399189 | 1.172643 | 0.312912 | 0.528532 | 0.238325 | 0.279142 |

| | mean | | | std | | |
|---|---|---|---|---|---|---|
| | Beta HML | Beta Mkt-RF | Beta SMB | Beta HML | Beta Mkt-RF | Beta SMB |
| K-means | | | | | | |
| 0 | 0.041000 | 0.810170 | -0.145340 | 0.336111 | 0.249588 | 0.191138 |
| 1 | 0.642948 | 1.240775 | 0.440338 | 0.510623 | 0.222736 | 0.302960 |

**Figure 16. Descriptive statistics of the k-means clustering before (above) and after (below) COVID-19.**

Observations for HDBSCAN (Figure 17):

- The noise clusters before and after COVID-19 are similar in terms pf descriptive statistics. However, the mean beta Mkt-RF has grown from 1.14 to 1.28 and the mean Beta SMB has increased from 0.21 to 0.45, so there are more outliers/noise in the coefficients of companies moving with the market and behaving like small companies.
- Cluster 1 before COVID-19 and cluster 0 after COVID-19, which are the most similar for HDBSCAN, have similarities in terms of descriptive statistics. Beta HML remained almost the same, nevertheless, beta Mkt-RF has grown from 1.14 to 1.28 (movement with the market) and beta SMB has increased from 0.21 to 0.44 (more behaviour of small companies).

- Small clusters have changed completely. Before COVID-19, cluster 0 consisted of value stocks of large companies moving with the market. After COVID-19, cluster 1 consists of growth stocks of large companies moving with the market.

| | mean | | | std | | |
|---|---|---|---|---|---|---|
| | Beta HML | Beta Mkt-RF | Beta SMB | Beta HML | Beta Mkt-RF | Beta SMB |
| HDBSCAN | | | | | | |
| -1 | 0.154085 | 1.142366 | 0.214861 | 0.813529 | 0.376473 | 0.436504 |
| 0 | 1.017085 | 1.320570 | -0.022223 | 0.100035 | 0.077982 | 0.066096 |
| 1 | 0.018322 | 0.835848 | 0.005306 | 0.365897 | 0.324311 | 0.298305 |

| | mean | | | std | | |
|---|---|---|---|---|---|---|
| | Beta HML | Beta Mkt-RF | Beta SMB | Beta HML | Beta Mkt-RF | Beta SMB |
| HDBSCAN | | | | | | |
| -1 | 0.162891 | 1.283918 | 0.448601 | 0.758522 | 0.361862 | 0.477299 |
| 0 | 0.323732 | 0.907221 | 0.011233 | 0.413782 | 0.265722 | 0.296482 |
| 1 | -0.606215 | 1.216026 | -0.181668 | 0.115400 | 0.051216 | 0.132399 |

**Figure 17. Descriptive statistics of the HDBSCAN clustering before (above) and after (below) COVID-19.**

The reason for increased sensitivity to the market for all clusters could be caused by a similar adaptation of companies after the crisis. A larger exposure to value stocks can be a sign of the growth slowing down after the crisis.

## Industry comparison

### Hierarchical clustering

Cluster 0 before COVID-19 and 0 after COVID-19 mostly consist of the following industries:

- Communication Services: 13 out of 21 companies
- From Consumer Discretionary: Internet and Direct Marketing Retail, Restaurants
- Consumer Staples, especially Consumer Staples Merchandise Retail and FMCG
- Financial Exchanges and Data, Property and Casualty Insurance, Insurance Brokers, Transaction and Payment Processing Services

- Health Care, especially Biotechnology, Health Care Equipment, Health Care Services and Supplies, Life Sciences Tools and Services, Managed Health Care and Pharmaceuticals
- From IT: Application Software and Semiconductor Materials and Equipment, Systems Software
- Real Estate: 14 out of 30 companies
- Utilities

In general, the clusters can be characterized as clusters of cheap goods and services for ordinary consumers, high technologies and utilities.

Cluster 1 before COVID-19 and cluster 1 after COVID-19 consist mostly of the following industries:

- Consumer Discretionary: Luxury Goods, Automotive Industry, Casinos, Hotels
- Energy
- Financials, especially: Consumer Finance, Banks, Life and Health Insurance
- Industrials, especially Construction Machinery and Heavy Transportation Equipment, Industrial Machinery and Supplies and Components, Passenger Airlines
- Technology, Hardware, Storage and Peripherals
- Materials: 15 of 27 companies

Clusters can be characterized as clusters of expensive goods, energy, industrials and materials.

### K-means clustering

Clusters 0 and 1 for k-means before and after COVID-19 respectively are very similar to those in hierarchical clustering. For cluster 0, IT consulting is added, Real Estate is more widely represented (20 of 30 companies). For cluster 1, Semiconductors are added and Materials are less represented.

### HDBSCAN

The small cluster 0 before COVID-19 consists only of Financials with different sub-industries: Asset Management and Custody Banks, Consumer Finance, Diversified Banks, Investment Banking and Brokerage, Life and Health Insurance.

The small cluster 1 after COVID-19 includes mostly Information Technologies. They are seen as growth stocks of large companies, strongly moving with the market. This could be caused by an increased demand of information technologies due to the lockdown (for example, remote working).

All the industries are represented in cluster 1 before COVID-19 and in a similar cluster 0 after COVID-19.

The industries in the noise cluster before and after COVID-19 are not equally spread, the largest industries in the noise cluster before COVID-19 are Financials and Information Technology, after COVID-19 – Information Technology and Consumer Discretionary. Google, Tesla, Amazon, PayPal, Nvidia, Microsoft are among companies referring to noise clusters before and after COVID-19 simultaneously.

## Analysis of changers

To see the differences more clearly, we compute the differences of betas (after COVID-19 minus before COVID-19). The clustering analysis shows that the optimal number of clusters is also 2, and HDBSCAN also shows 2 clusters. However, it is not very informative, that is why we compute Euclidean distances from (0, 0, 0) to corresponding beta differences. The top companies are presented below in the Figure 18. As we see, the most changers are from Consumer Discretionary, especially Hotels, Resorts and Cruise Lines as well as from Real Estate, especially Retail REITs. For all changers, exposure to market, size and value have grown: these companies follow the market, behave like small companies and are value stocks. Some of the changer industries were directly impacted by the lockdown (no travelling, high demand in health care, etc.). Real estate sector has also grown during the pandemic.

| Security | GICS Sector | GICS Sub-Industry | Ticker | Beta Mkt-RF_x | Beta SMB_x | Beta HML_x | Beta Mkt-RF_y | Beta SMB_y | Beta HML_y | Distance |
|---|---|---|---|---|---|---|---|---|---|---|
| Norwegian Cruise Line Holdings | Consumer Discretionary | Hotels, Resorts & Cruise Lines | NCLH | 1,10 | 0,30 | 0,14 | 2,00 | 1,58 | 1,17 | 1,874582925 |
| Carnival | Consumer Discretionary | Hotels, Resorts & Cruise Lines | CCL | 0,99 | 0,25 | 0,31 | 1,98 | 1,45 | 1,20 | 1,794755849 |
| Royal Caribbean Group | Consumer Discretionary | Hotels, Resorts & Cruise Lines | RCL | 1,20 | 0,21 | 0,08 | 1,82 | 1,20 | 1,10 | 1,553745025 |
| Ventas | Real Estate | Health Care REITs | VTR | 0,16 | -0,24 | -0,41 | 1,01 | 0,27 | 0,75 | 1,529779658 |
| United Airlines Holdings | Industrials | Passenger Airlines | UAL | 1,02 | 0,32 | 0,30 | 1,71 | 1,09 | 1,31 | 1,438625579 |
| Welltower | Real Estate | Health Care REITs | WELL | 0,21 | -0,25 | -0,45 | 0,95 | 0,12 | 0,69 | 1,407615554 |
| Simon Property Group | Real Estate | Retail REITs | SPG | 0,51 | 0,11 | 0,17 | 1,33 | 0,71 | 1,06 | 1,356818413 |
| Kimco Realty | Real Estate | Retail REITs | KIM | 0,55 | 0,21 | -0,08 | 1,25 | 0,55 | 1,02 | 1,351207438 |
| Boeing | Industrials | Aerospace & Defense | BA | 1,16 | -0,25 | 0,04 | 1,49 | 0,77 | 0,84 | 1,333656362 |
| Occidental Petroleum | Energy | Oil & Gas Exploration & Production | OXY | 0,97 | 0,33 | 0,61 | 1,52 | 0,97 | 1,64 | 1,330476986 |
| Expedia Group | Consumer Discretionary | Internet & Direct Marketing Retail | EXPE | 0,73 | 0,10 | -0,25 | 1,46 | 0,72 | 0,56 | 1,252186582 |
| Live Nation Entertainment | Communication Services | Movies & Entertainment | LYV | 1,05 | 0,24 | -0,55 | 1,27 | 0,86 | 0,51 | 1,250435872 |
| Regency Centers | Real Estate | Retail REITs | REG | 0,44 | -0,05 | -0,08 | 1,10 | 0,37 | 0,88 | 1,239970362 |
| Federal Realty | Real Estate | Retail REITs | FRT | 0,43 | -0,04 | -0,07 | 1,04 | 0,51 | 0,84 | 1,23040014 |
| Boston Properties | Real Estate | Office REITs | BXP | 0,57 | -0,16 | -0,11 | 1,05 | 0,36 | 0,84 | 1,183877848 |
| Delta Air Lines | Industrials | Passenger Airlines | DAL | 1,01 | 0,20 | 0,14 | 1,47 | 0,83 | 1,01 | 1,165633511 |
| Sysco | Consumer Staples | Food Distributors | SYY | 0,51 | -0,06 | -0,17 | 1,05 | 0,31 | 0,71 | 1,099281366 |
| Healthpeak | Real Estate | Health Care REITs | PEAK | 0,19 | -0,20 | -0,45 | 0,87 | -0,04 | 0,36 | 1,073516535 |
| SolarEdge | Information Technology | Semiconductor Materials & Equipment | SEDG | 1,01 | 0,31 | -0,59 | 1,22 | 1,35 | -0,59 | 1,067992462 |
| Match Group | Communication Services | Interactive Media & Services | MTCH | 0,92 | 0,11 | -1,48 | 1,29 | 0,49 | -0,56 | 1,063128614 |

**Figure 18. Companies with largest differences of beta-coefficients.**

# Conclusion

In this paper, we estimate Fama-French regressions for the companies in S&P 500 and classify the coefficients using hierarchical clustering, k-means and HDBSCAN. The optimal number of clusters as well as the number of clusters found by HDBSCAN is 2. K-means and hierarchical clustering behave in a similar way whereas HDBSCAN find completely different clusters.

The reason for increased sensitivity to market for all clusters could be caused by a similar adaptation of companies after the crisis. A larger exposure to value stocks can be a sign of the growth slowing down after the crisis. In general, clusters of k-means and hierarchical clustering can be characterized as follows: a cluster of cheap goods and services for ordinary consumers, high technologies and utilities and a cluster of expensive goods, energy, industrials and materials. HDBSCAN finds a Financials cluster before COVID-19 and an IT cluster after COVID-19, which could be caused by expanding the IT industry and online technologies after the lockdown.

The most changers in terms of differences in beta coefficients are from Consumer Discretionary, especially Hotels, Resorts and Cruise Lines as well as from Real Estate, especially Retail REITs. Some of the changer industries were directly impacted by the lockdown (no travelling, high demand in health care, etc.). Real estate sector has also grown during the pandemic.

To summarize, there are industry specific changes of risk exposure after the COVID-19 financial crisis.

# References


Ahmed, M., Seraj, R., & Islam, S. M. S. (2020). The k-means algorithm: A comprehensive survey and performance evaluation. *Electronics*, *9*(8), 1295.

Aldahdooh, R. T., & Ashour, W. (2013). DIMK-means "Distance-based Initialization Method for K-means Clustering Algorithm". *International Journal of Intelligent Systems and Applications*, *5*(2), 41.

Ashabi, A., Sahibuddin, S. B., & Salkhordeh Haghighi, M. (2020). The systematic review of K-means clustering algorithm. In *Proceedings of the 2020 9th International Conference on Networks, Communication and Computing* (pp. 13-18).

Caliński, T., & Harabasz, J. (1974). A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, *3*(1), 1-27.

Fama, E. R., & French, K. R. (1993). Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics*, 33(1), 3-56.

Hou, D., & Chen, Z. (2021). Research on the application of Fama-French 5-factor model in the steel industry during COVID-19. In *Journal of Physics: Conference Series* (Vol. 1865, No. 4, p. 042104). IOP Publishing.

Lim, D., Durand, R. B., & Yang, J. W. (2014). The microstructure of fear, the Fama–French factors and the global financial crisis of 2007 and 2008. *Global Finance Journal*, 25(3), 169-180.

McInnes, L., Healy, J., & Astels, S. (n.d.). hdbscan: Hierarchical density-based clustering. *Journal of Open Source Software*, 2(11), 205.

Murtagh, F., & Contreras, P. (2012). Algorithms for hierarchical clustering: an overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(1), 86-97.

Murtagh, F., & Legendre, P. (2014). Ward's Hierarchical Agglomerative Clustering Method: Which Algorithms Implement Ward's Criterion? *Journal of Classification*, 31, 274-295.

Nguyen, T. H. T., Dinh, D. T., Sriboonchitta, S., & Huynh, V. N. (2019). A method for k-means-like clustering of categorical data. *Journal of Ambient Intelligence and Humanized Computing*, 1-11.

Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, *20*, 53-65.

Santos, J. M., & Embrechts, M. (2009, September). On the use of the adjusted rand index as a metric for evaluating supervised classification. In *International conference on artificial neural networks* (pp. 175-184). Berlin, Heidelberg: Springer Berlin Heidelberg.

Stewart, G., & Al-Khassaweneh, M. (2022). An implementation of the HDBSCAN* clustering algorithm. *Applied Sciences*, *12*(5), 2405.

Sun, Y. (2021). Analysis of fore and aft covid-19 impact on industry data based on fama-french five factors. In *2021 international conference on electronic commerce, engineering management and information systems*.

Syakur, M. A., Khotimah, B. K., Rochman, E. M. S., & Satoto, B. D. (2018). Integration k-means clustering method and elbow method for identification of the best customer profile cluster. In *IOP conference series: materials science and engineering* (Vol. 336, p. 012017). IOP Publishing.

Ward Jr., J. H. (1963). Hierarchical Grouping to Optimize an Objective Function. *Journal of the American Statistical Association*, 58(301), 236-244.

Xu, J., & Lange, K. (2019). Power k-means clustering. In *International conference on machine learning* (pp. 6921-6931). PMLR.