K-MEANS AND K-MEDOID ALGORITHM APPLICATION IN CLUSTERING STOCK DATA IN INDONESIA

JVC Medellu [1], E S Nugraha [2]
School of Business, Study Program of Actuarial Science, President University, Cikarang, Bekasi
[1]juliano.medellu@student.president.ac.id
[2]*edwin.nugraha@president.ac.id*

ABSTRACT

Stock or share is a sign of a person's or a business party's capital participation in a company. In the context of stock, the meaning of investment itself is an activity of investing funds or capital in a company. The investment is expected to generate profits and not vice versa, therefore a good consideration is needed in the selection of the company's shares to be purchased. Parameters that can be used as the basis for consideration are the level of volatility, liquidity, and also market capitalization. Volatility is an indicator of the rate of change in stock prices that implies the risk of how the stock market price moves. The greater the level of volatility, the greater the risk of the stock market. Liquidity is an indicator of how easy the shares are to be sell and purchased without affecting the asset price. Meanwhile, stocks with high market capitalization will be difficult to play with. Therefore we need to know the characteristics of stocks before investing our fund. The results of data clustering using K-Means and K-Medoid help us to understand the characteristics of stocks. With the effectiveness of the clustering algorithm, we can see the characteristics of stocks, from those with a high risk, high level of liquidity, and also could calculate the average market capitalization in the clustering result for each cluster. The results of this analysis can be the basis for good stock selection, based on the stock historical data.

Keywords: *K-Means; K-Medoid; Volatility; Liquidity; Market Capitalization.*

1. Introduction

Almost all areas of life have used computers as tools. It is expected that in its development, the computer. One of the uses of computers is in data analysis, which we are familiar with machine learning, which can be used to analyze and classify large dataset. Shares are a unit of value or bookkeeping in various financial instruments that refer to the ownership of a company (David, 2020). The number of shares on the Indonesian stock exchange continues to grow over time. Each share listed on the market has different characteristics, both from the company's fundamental side and the stock price movement on the market (technical), therefore it is necessary to group the issuers based on the characteristics of the shares themselves. Clustering is a technique of unsupervised machine learning that aims to classify data based on similar characteristics between data (David, 2020). There are many approaches to clustering such as partitional methods, density methods, hierarchical methods and others. Each clustering method has its own advantages such as the partitional method where the number of clusters can be determined by the user, and each cluster can be searched for its characteristics because it has a cluster center. The K-Means and K-Medoid algorithm that are parts of the unsupervised clustering method could be used on most of the clustering problems starting from the easy to complex problems for example it can be used to cluster the Student’s Grade based on the exams grade, or could also be used to cluster Mall Customers (Customer Segmentation) based on their income and their money spending rate. This article will discuss the clustering of stocks in Indonesia using several algorithms derived from partitional methods, namely K-Means and K-Medoid. The clustering results will be analyzed to identify their characteristics. There is also another method that is used to cluster and classify stocks data, it is the K-Nearest Neighbor (KNN) method. The KNN method tends to be used to classify and predict stocks. As we know KNN is a supervised clustering algorithm while K-means and K-Medoid are the unsupervised clustering algorithms that enable K-Means and K-Medoid to gradually learn and apply themselves to cluster unlabelled points (data). KNN is mainly used for the classification of given data with the known attributes, while K-Means and K-Medoid are more flexible in the context of exploratory data analysis. K-Means and K-Medoid methods are easy to be done. The learning and applying process is relatively fast compared with others, which is also part of the consideration why this project uses K-means and K-Medoid. The clustering in this project will only consider three main technical components that affect stocks which are volatility, liquidity, and market capitalization. The clustering process that will be conducted is based on the daily stock movement data from 2018 to 2020 of the shareholder companies that have been listed on the market before 2017. The methods that will be used in the clustering process are the K-Means and K-Medoid clustering algorithm. The focus of this project consists of two problems, first is to know the application of the K-Means and K-Medoid clustering method in stock data clustering, and the second one is how to classify the stock data in Indonesia based on the predefined parameters which are level of volatility, liquidity, and also market capitalization. The project aims to understand how K-Means and K-Medoid able to do the clustering process using the mentioned parameters, and what can we conclude from the clustering result.

2. Literature Review

2.1. Introduction to K-Means

2.1.1. Based Knowledge of K-Means

Munzir Umran (Umran & Abidin, 2009) states that K Means is a non-hierarchical data clustering method that attempts to partition existing data into one or more clusters/groups. K-means requires as many as k input parameters and divides a set of n-objects into k clusters so that the level of similarity between members in one cluster is high while the level of similarity to members in other clusters is very low. The similarity of members to the cluster is measured by the proximity of the object to the mean value of the cluster members or can be referred to as the centroid or center of mass (Widyawati, 2010). The following is the distance measurement formula :

$$d_{(x,y)} = \|x - y\|^2 = [\sum_{i=1}^{n}(X_i - Y_i)^2]^{1/2} \tag{1}$$

Details :
d = document point
x = data record
y = centroid
This formula basically comes from a Pythagoras formula which sees X and Y as the furthest vertices of a right triangle and the hypotenuse as the distance. The total distance of each cluster is measured by the sum of the total distance between data (document point) to the centroids. As we can see from the formula, calculating Euclidean distance involves a calculation of a square root of the sum of squares (SS) differences in a series between two corresponded points or values. shortest distance between the centroid, and the document determines the position of the document cluster. Recalculate the position of the new centroid for each centroid ($C_{i..j}$) by taking the average of the incoming documents in the initial cluster ($G_{i..j}$). Iteration is carried out continuously until the position of the group does not change. In a mathematical way, the formula to calculate the new centroid is defined as follows:

$$C(i) = \frac{x_1 + x_2 + \dots}{\Sigma x} \tag{2}$$

Details :
x_1 = data *record*-1
x_2 = data *record*-2
∑x = Number of data *records*
C(i) represents the new centroid in cluster *i*, which is calculated by taking the average of all the data inside cluster *i*. The Partitional-clustering strategy that is most often used is based on the square error criterion. In general, the objective of the square error criterion is to obtain a partition (fixed number of clusters) that minimizes the total square error.

2.1.1. K-Means Algorithm Steps

There are steps or stages in processing the K-means algorithm that will be discussed in the following explanation (Larose, 2005). The first step in applying the K-Means algorithm is to determine the number of K clusters that we want. This could be done by choosing it randomly according to the project context, but also could be done by using several methods such as Elbow Method, Silhouette, etc. The next step is to determine the cluster centers (Centroid). For each row, find the closest cluster center. To calculate the distance between the data and the center of the cluster, the Euclidean formula is used (1). After that, group the data based on the shortest distance. For each K cluster, find the centroid (means) of the cluster and update the location of the cluster center into the new centroid values (2). Calculate the data distance to the updated centroid continuously until the iteration value or tolerance value allow the algorithm to stop or when the data position does not change.

2.1.2. Advantages and Disadvantages of K-Means

The K-Means algorithm has its advantages and disadvantages in terms of clustering objects. The common advantage of this method is easy to implement and also easy to adapt. This method does not require a big amount of time to be proceed, the time required to carry out to do and learn this method also is relatively fast. The K-Means method is also known by many people and is commonly used in solving problems. Despite the advantages, there are several disadvantages of this method. This method only works on numeric attributes, so if there any character attributes then they should not be included in the clustering process or they can also be converted as numeric attributes. It is also unclear how many clusters K is the best, but this problem could be solved by using an additional method like the Elbow method in finding the optimum number of K clusters. The K-means itself is very dependent on the initial selection of the centroid.

2.2. Introduction to K-Medoid

2.2.1. Base Knowledge of K-Medoid

K-Medoids or Partitioning Around Method (PAM) is a non-hierarchical cluster method which is a variant of the K-Means method. K-Medoids exist to overcome the K-Means disadvantage of outlier-sensitive because an object with a large value can deviate substantially from the data distribution. It is based on using medoids rather than observing the average each cluster has, to reduce the sensitivity of the partition with respect to the existing extreme values in the dataset. In the K-medoid method, each cluster is represented by an object in the cluster called the medoid. The aim is to find the K-cluster group (number of clusters) among all data objects in the data group. Clusters are built from the results of matching each data object closest to the cluster which is considered a temporary medoid. Distance measurement can be done with the Euclidean Distance (1) formula.

2.2.2. K-Medoid Algorithm Steps

Similar to the K-Means method, the first step of the K-Medoid algorithm is to determine the number of K clusters. This could be done by choosing it randomly based on the project needs, and also by using several additional methods such as Elbow Method, Silhouette Method, and others. The second step is to randomly select the initial center of each cluster (Medoid). After that, allocate each data (object) to the nearest Medoid using the Euclidean Distance formula (1). Calculate the distance of each object or data in each cluster with the new medoid candidate. Compare the closest total distance from the grouping results between the old (previous) and the new medoid. If the total shortest distance from the new grouping result is less than the old one, then swap the position of the old medoid with the new medoid. Calculate the distance between data and the new medoids continuously until the medoids don’t change.

2.2.3. Advantages and Disadvantages of K-Medoid

There are several advantages that K-Medoid method has. The most highlighted advantage is in handling outlier data. The K-Medoid method effectively handle the existing outliers in data compared with K-Mean method, it is because K-Medoid uses a representative object (data) from each cluster as the center of the clusters (medoid), while K-Means uses average calculation in determining the new center of the clusters (centroid) which will possibly force the outlier data to join a cluster even that it has low level of similarity that will affect the quality of the clustering results. The time required to carry out this learning is relatively fast and easy to be done which also has a fast process in clustering objects. The disadvantage is that its able to generate different cluster results for different processes on the same dataset because initially, we randomly select the medoids from the total data objects and assign them to each cluster individually so that it becomes the initial medoid of the clusters. The overall computing time and final distribution of objects in the cluster or group is depend on the initial partition.

3. System Design

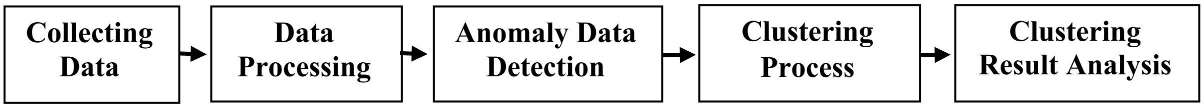


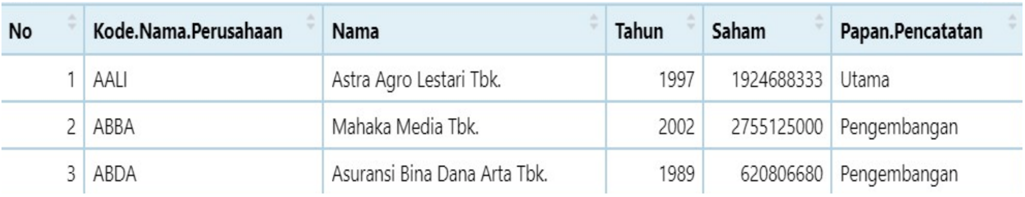
Figure 1. *Clustering process system design*

The starting point of the system design flow is collecting data. The data that will be used is the data on the shareholder company profiles taken from the official website of the Indonesia Stock Exchange (IDX), market capitalization data, and daily share price movement data from 2018 to 2020. In the Data processing section, we will do several calculations and extract the information regarding the volatility and liquidity level (parameter calculation). Knowing that the clustering process is sensitive to the outlier data, the Anomaly Data Detection needed to be done before the clustering process start. The outlier data soon will be removed from the processed dataset (information) before performing the clustering process so as not to affect the quality of the clustering results. The main section is the clustering process. The clustering process will use the K-Means and K-Medoid methods by doing iterations until none of the data inside clusters have changed. Selection of the number of K clusters will be using the elbow method. The last section is the clustering result analysis. This section is carried out to spot the characteristics of the clustering results.

4. Result and Discussion

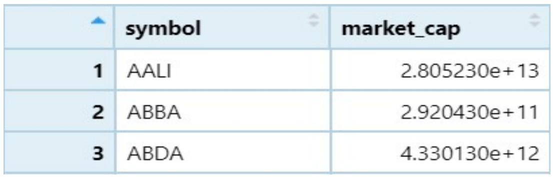
4.1. Initial Data of Clustering Materials

Several datasets will be used in the clustering process, such as company profile data, daily stock price movement data, and market capitalization data. The data that will be clustered are data on the shareholder companies that have been listed on the stock exchange before 2017 with a range of stock price movements from 2018 to 2020. Data processing will be carried out using the assistance of RStudio. Several R packages that need to be activated in data processing are tidyverse, tidyquant, lubridate, ggthemes, scales, factoextra, FactoMineR, dbscan, and cluster. Packages related to the packages that have been previously mentioned will also be activated to support the performance of the main package. So make sure all packages are active so that the data processing can run smoothly. The first thing to do is to input the dataset into RStudio. Company profile data is obtained through an online data collection process on the official website of the Indonesia Stock Exchange (IDX). The dataset is input using read.csv into a variable. For profile data, we filter shareholder company data and retrieve company data recorded before 2017. Then with the new variable, we will pull the symbol data from the profile and add the suffix ".JK" to each symbol. This is done so that we can use these symbols to retrieve stock movement data online from Yahoo because the format for Indonesian shares on Yahoo ends with ".JK". Market capitalization data are loaded on different variables. The display of filtered profile data in R can be seen as follows:



No	Kode.Nama.Perusahaan	Nama	Tahun	Saham	Papan.Pencatatan
1	AALI	Astra Agro Lestari Tbk.	1997	1924688333	Utama
2	ABBA	Mahaka Media Tbk.	2002	2755125000	Pengembangan
3	ABDA	Asuransi Bina Dana Arta Tbk.	1989	620806680	Pengembangan

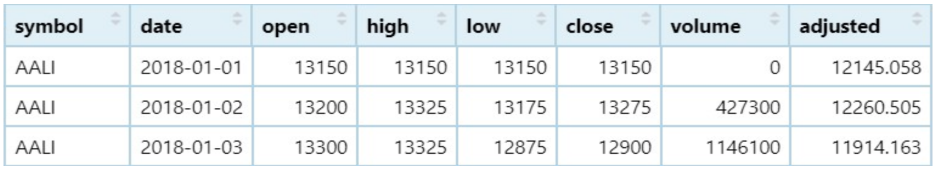
Figure 2. Profile data as a dataframe in R



	symbol	market_cap
1	AALI	2.805230e+13
2	ABBA	2.920430e+11
3	ABDA	4.330130e+12

Figure 3. Market Capitalization as a dataframe in R

Next, we will pull stock movement data online from Yahoo. The "tq_get" function is a function of the tidyquant package which is used to pull stock movement data from Yahoo, in the time span from 2018 (2018-01-01) to 2020 (2020-12-31). After the data inputted into a variable, we will delete the suffix ".JK" in each company symbol using str_remove_all. The following is a display of the data obtained from Yahoo



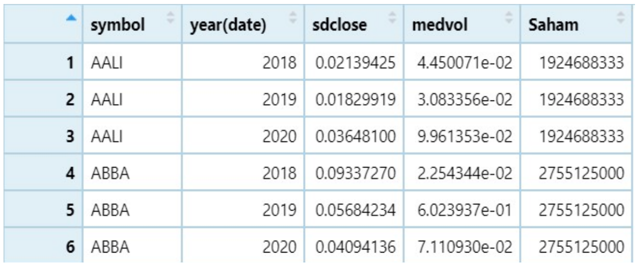
symbol	date	open	high	low	close	volume	adjusted
AALI	2018-01-01	13150	13150	13150	13150	0	12145.058
AALI	2018-01-02	13200	13325	13175	13275	427300	12260.505
AALI	2018-01-03	13300	13325	12875	12900	1146100	11914.163

Figure 4. Stock price movement data from yahoo

The stock movement data consists of 8 variables, the first is "symbol", namely the code of the company issuer on the stock exchange, then "date", which is the date of the stock price, "open" for the opening price, "high" for the highest price, "low" for the lowest price, "close" is for the closing price, "volume" is for the number of shares traded, and "adjusted" represents the closing price that has been adjusted for other corporate actions.

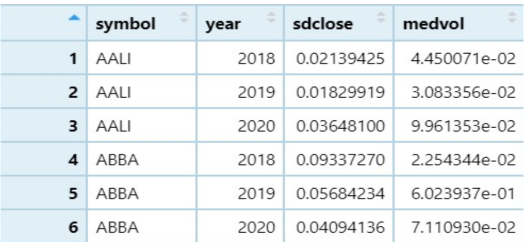
4.2. Data Processing

The parameters for grouping the data in this analysis are volatility, stock liquidity, and market capitalization. To find parameter values, we must first remove empty data by using na.omit () to avoid errors and miscalculations. Volatility can be calculated by finding the standard deviation of the price change ratio/percentage. Meanwhile, the amount of liquidity can be found by taking the median of the share volume. Before performing calculations, we will set the calculation for each data using group_by function, so that each calculation will be carried out based on the symbol and the year. To avoid bias in the data, the previous level of liquidity will be divided by the number of shares based on each company data in the previous profile data. Then we have to combine the calculation data of the parameter magnitude with the profile data based on the code of the company issuer. This can be done by using the left_join function. Temporary data display can be seen in the following table.



	symbol	year(date)	sdclose	medvol	Saham
1	AALI	2018	0.02139425	4.450071e-02	1924688333
2	AALI	2019	0.01829919	3.083356e-02	1924688333
3	AALI	2020	0.03648100	9.961353e-02	1924688333
4	ABBA	2018	0.09337270	2.254344e-02	2755125000
5	ABBA	2019	0.05684234	6.023937e-01	2755125000
6	ABBA	2020	0.04094136	7.110930e-02	2755125000

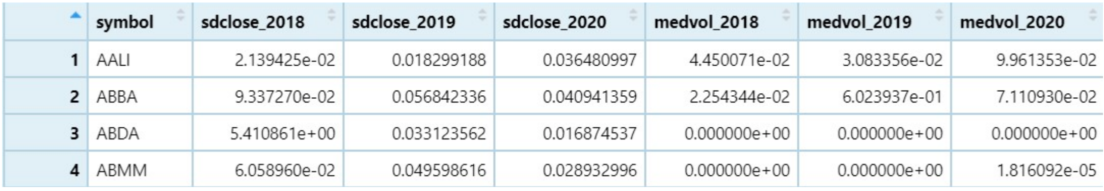
Figure 5. Parameter value calculation result



	symbol	year	sdclose	medvol
1	AALI	2018	0.02139425	4.450071e-02
2	AALI	2019	0.01829919	3.083356e-02
3	AALI	2020	0.03648100	9.961353e-02
4	ABBA	2018	0.09337270	2.254344e-02
5	ABBA	2019	0.05684234	6.023937e-01
6	ABBA	2020	0.04094136	7.110930e-02

Figure 6. Modified parameter result and median volume (liquidity)

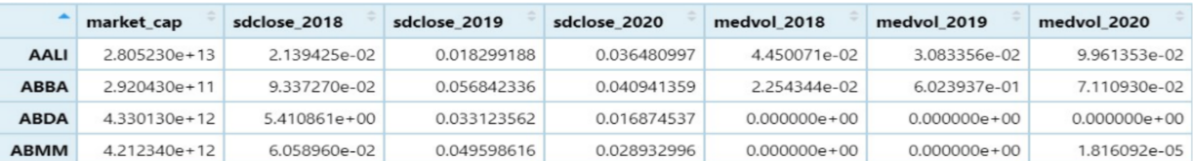
However, we still cannot use the results of this data processing in the clustering process, because each symbol has not been represented by one line due to time differences (years). Therefore we will reformat the data using the pivot_wider function. Data display can be seen in the following figure.



	symbol	sdclose_2018	sdclose_2019	sdclose_2020	medvol_2018	medvol_2019	medvol_2020
1	AALI	2.139425e-02	0.018299188	0.036480997	4.450071e-02	3.083356e-02	9.961353e-02
2	ABBA	9.337270e-02	0.056842336	0.040941359	2.254344e-02	6.023937e-01	7.110930e-02
3	ABDA	5.410861e+00	0.033123562	0.016874537	0.000000e+00	0.000000e+00	0.000000e+00
4	ABMM	6.058960e-02	0.049598616	0.028932996	0.000000e+00	0.000000e+00	1.816092e-05

Figure 7. pivot_wider data processing result

Next, we will add a new column containing market capitalization data. We will also replace each blank data (NA) with 0. There is 1 character column, namely the "symbol" column inside the data. The clustering process only uses numeric data. Therefore we will set the column “symbol” as row names. Data display can be seen in the following figure.



	market_cap	sdclose_2018	sdclose_2019	sdclose_2020	medvol_2018	medvol_2019	medvol_2020
AALI	2.805230e+13	2.139425e-02	0.018299188	0.036480997	4.450071e-02	3.083356e-02	9.961353e-02
ABBA	2.920430e+11	9.337270e-02	0.056842336	0.040941359	2.254344e-02	6.023937e-01	7.110930e-02
ABDA	4.330130e+12	5.410861e+00	0.033123562	0.016874537	0.000000e+00	0.000000e+00	0.000000e+00
ABMM	4.212340e+12	6.058960e-02	0.049598616	0.028932996	0.000000e+00	0.000000e+00	1.816092e-05

Figure 8. Clustering preparation data

4.3. Anomaly Data Detection

The clustering process is sensitive to outlier data because the cluster will force outliers to enter the clusters, thus affecting the quality of the cluster results. At this stage, we will apply the DBSCAN algorithm to spot the outlier data. But to run DBSCAN we must determine the minimum amount of neighbor data in a data environment (minPts) and the maximum distance between 2 data (eps). To determine the eps we can use the knee plot with the help of the KNNdistplot function. The minPts is set to be 8 data. The knee-plot can be seen in the following figure.

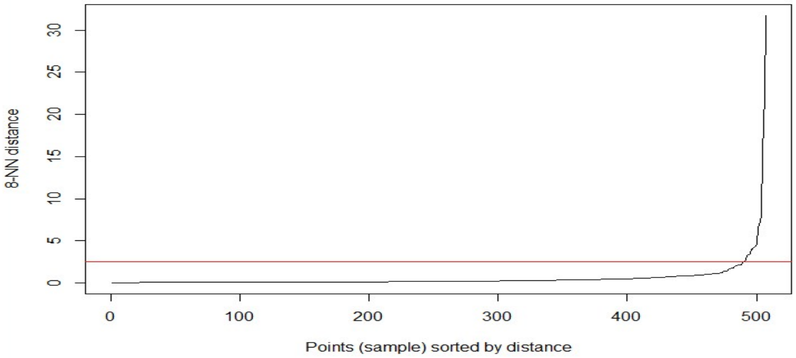


Figure 9. Knee-plot

The knee of the curve shows the optimum eps located in the middle of 0 and 5. Therefore 2.5 will be assigned as the optimum number of eps. Now we are able to use the DBSCAN to identify the outlier data. The data that will be inputted in the DBSCAN algorithm needs to be scaled using the scale function to equalize the range of data. In the DBSCAN result, it is known that there are "10 noise points" or 10 anomalous data. We can draw the outlier data symbol by combining the clustering preparation data (Figure 8) with the cluster column from the DBSCAN result. Outlier data is inside db_clust 0. After that, we are going to filter the data by pulling symbols that are inside the db_clust 0. That way we can draw the anomaly data symbol. For more detailed information about outlier data, we can use Principal Component Analysis (PCA) and then plot outlier data using PCA plot. The plot display can be seen in the following figure.

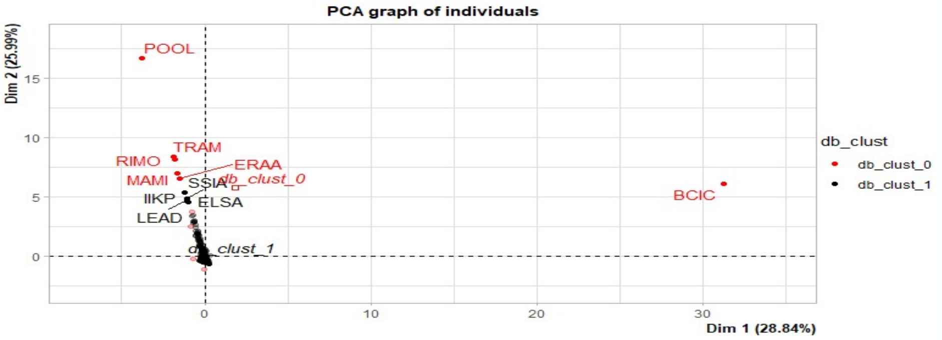


Figure 10. Outlier Data plot

Outlier data can be seen in plots represented by red dots. After knowing the outlier data, we will remove it from the clustering preparation data. We can exclude the outlier data by filtering data or retrieving data that is not in db_clust 0. After that, we can remove the db_clust column and then scale the data.

4.4. Clustering Process

4.4.1. K-Means Clustering

The first step in K-Mean is to determine the number of clusters. This can be done with the help of the Elbow method. The elbow plot view can be seen in the following figure.

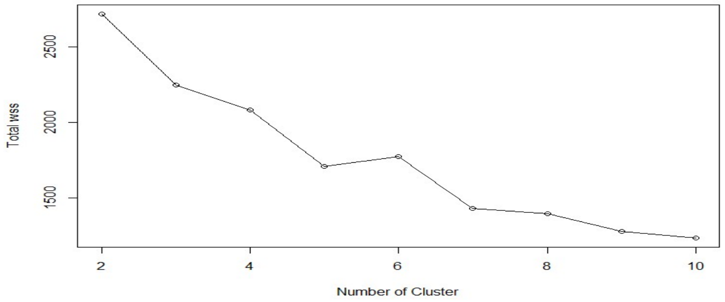


Figure 11. K-Means Elbow method plot

In the plot above there are several elbow points. The optimum value of K is that when the K number is added, the decrease of WSS will no longer be drastic. The suitable number of K in this situation is 9. The WSS at K = 9 is less than 1500. Next, we will do the K-Mean clustering process. With the help of the fviz_cluster function, we will plot (visualize) the results of grouping the data. The plot view can be seen in the following figure.

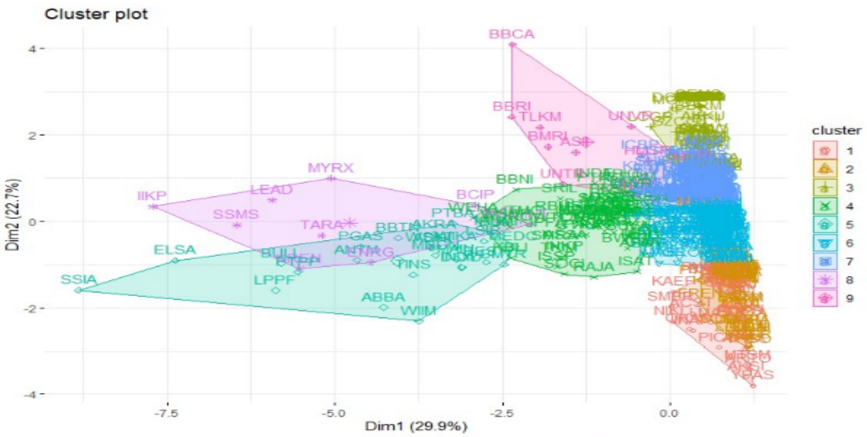
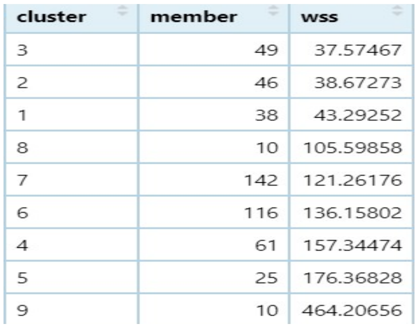


Figure 12. K-Means clustering result plot

After that, we will record the results of clustering into a dataframe in R. The results of grouping data will be made in the form of a dataframe. The display of the results of grouping data can be seen in the following table.



cluster	member	wss
3	49	37.57467
2	46	38.67273
1	38	43.29252
8	10	105.59858
7	142	121.26176
6	116	136.15802
4	61	157.34474
5	25	176.36828
9	10	464.20656

Figure 13. K-Means clustering result as a dataframe

The quality of the clustering results can also be seen from the comparison between the value between_SS and total_SS. The closer to 100%, the better the quality of the clustering result.

4.4.2. K-Medoid Clustering

The first thing to do is determine the number of clusters that will be used. Determination of the number of clusters is assisted by the elbow method. The plot appearance of the elbow method can be seen in the following figure.

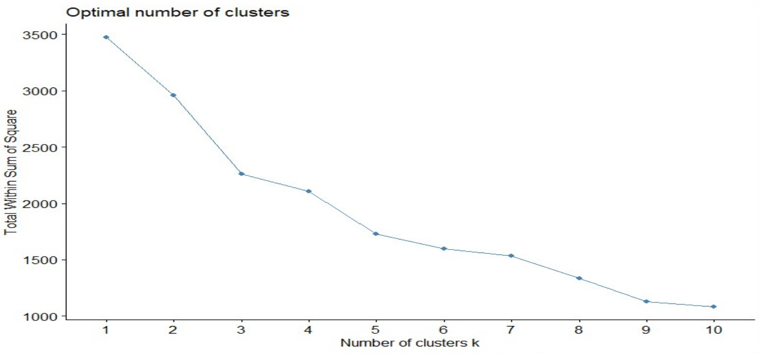


Figure 14. K-Medoid Elbow method plot

From the plot, it can be seen that when K = 6 is chosen as the elbow point of the curve, the decrease in WSS in the next number of K which is K =7 is not drastic (changes in WSS are not drastic). The WSS at K = 6 ranges between 1500 and 2000. Therefore we will assign the number of clusters to 6. Next, we will do the clustering process using the K-Medoid clustering method. With the help of the fviz_cluster function, we will plot the results of the clustering. The plot display can be seen in the following figure.

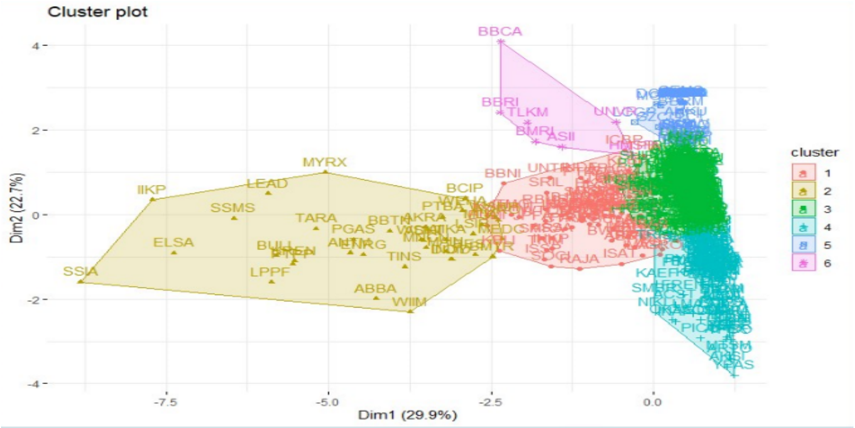
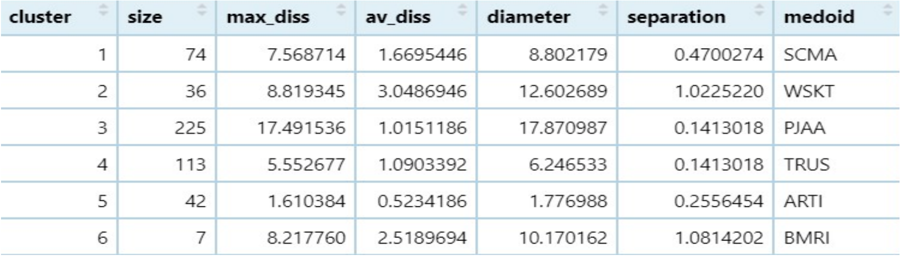


Figure 15. K-Medoid clustering result plot

After that, we will record the clustering results into a dataframe in R. The clustering result will be made in the form of a dataframe, the columns consist of the number of cluster members (size), the largest distance between each data inside a cluster to its medoid (max_diss), the average distance between data to medoid (av_diss), the largest distance between 2 data inside the same cluster (diameter), the smallest distance between data in a cluster to data in other clusters (separation), and Medoid column. The display of the clustering result will be processed into a dataframe which can be seen in the following table.



cluster	size	max_diss	av_diss	diameter	separation	medoid
1	74	7.568714	1.6695446	8.802179	0.4700274	SCMA
2	36	8.819345	3.0486946	12.602689	1.0225220	WSKT
3	225	17.491536	1.0151186	17.870987	0.1413018	PJAA
4	113	5.552677	1.0903392	6.246533	0.1413018	TRUS
5	42	1.610384	0.5234186	1.776988	0.2556454	ARTI
6	7	8.217760	2.5189694	10.170162	1.0814202	BMRI

Figure 16. K-Medoid clustering result

4.4.3. The Comparison of The Clustering Results

After we clustered the data with 2 different methods, namely the K-Means and K-Medoid methods, we will compare the two clustering results. With the help of "ggarrange" in "ggpubr", we are able to compare the 2 plots from the clustering results. The display of the comparison plot (visual) of the clustering results can be seen in the following figure.

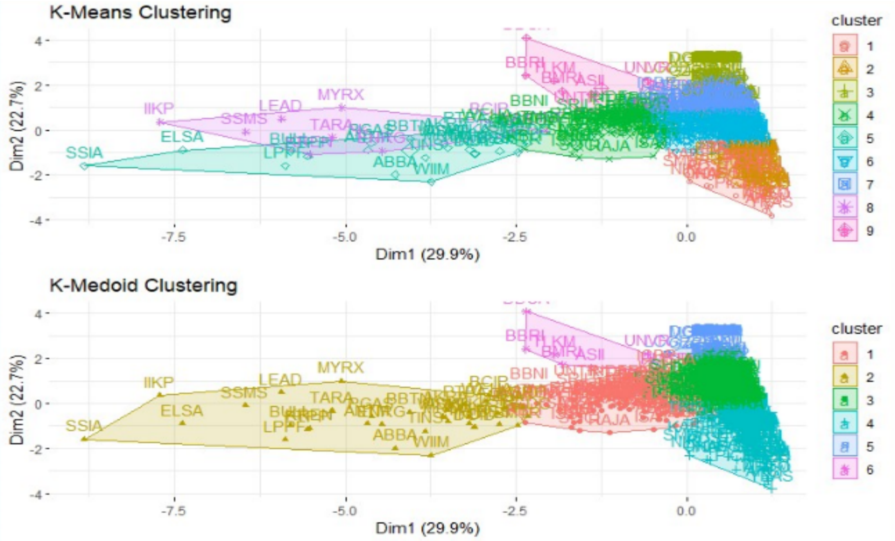
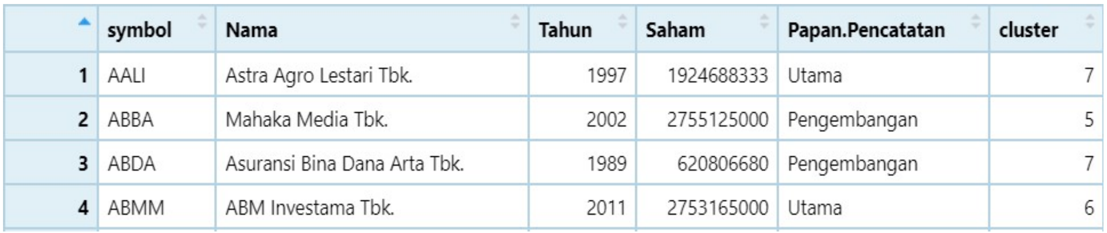


Figure 17. The plot comparison between K-Means and K-Medoid clustering result

From the comparison above, the results obtained with the K-Means method look almost the same as the K-Medoid. The total WSS results obtained by K-Mean are in the range less than 1500, while for K-Medoid is in the range from 1500 to 2000. This implies that the result of clustering using the K-Mean method can be considered more effective because it has less WSS than the K-Medoid. Therefore, we will use the K-Means clustering result to be analyzed.

4.5. Clustering Result Analysis

The clustering result will be analyzed in order to find out the number of members of each cluster and the characteristics of the clusters that we can find based on the clustering parameters. We will combine the clustering result data with the initial company profile data. We will also create a variable containing symbols with its clusters for use in further analyzes. The final profile data display can be seen in the following figure



	symbol	Nama	Tahun	Saham	Papan.Pencatatan	cluster
1	AALI	Astra Agro Lestari Tbk.	1997	1924688333	Utama	7
2	ABBA	Mahaka Media Tbk.	2002	2755125000	Pengembangan	5
3	ABDA	Asuransi Bina Dana Arta Tbk.	1989	620806680	Pengembangan	7
4	ABMM	ABM Investama Tbk.	2011	2753165000	Utama	6

Figure 18. Final profile data

Next, we will plot the number of members calculations that we have done on the K-Mean clustering results. With the help of ggplot, we will visualize the calculation results of each cluster member. The plot display can be seen in the following figure

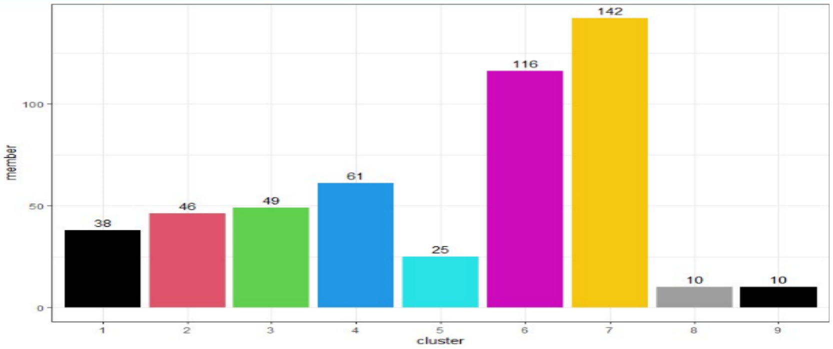


Figure 19. Number of members in each cluster plot

We can easily spot that cluster 7 with 142 companies has the largest number of members, while for clusters with at least 2 clusters, namely clusters 8 and 9, which consist of 10 companies. Next, we will do an analysis of market capitalization. The first step in this analysis is to combine the clustering data with market capitalization data. Replace each blank data (NA) with a value of 0. With these data, we will calculate the average market capitalization based on each cluster. The plot view can be seen in the following figure.

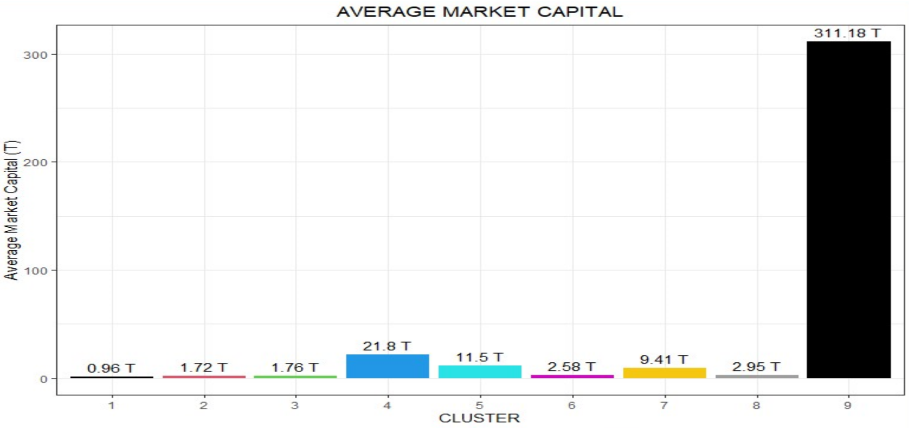


Figure 20. Average Market Capitalization in each cluster

Cluster 9 has the highest average market capitalization, even though it has few members. This implies that the share price of companies in this cluster will be more difficult to be played than companies in other clusters. The next analysis is based on the level of volatility in each cluster. In this analysis, we will calculate the standard deviation of price changes in each cluster in 2018, 2019, and 2020 respectively. The plot display can be seen in the following figure.

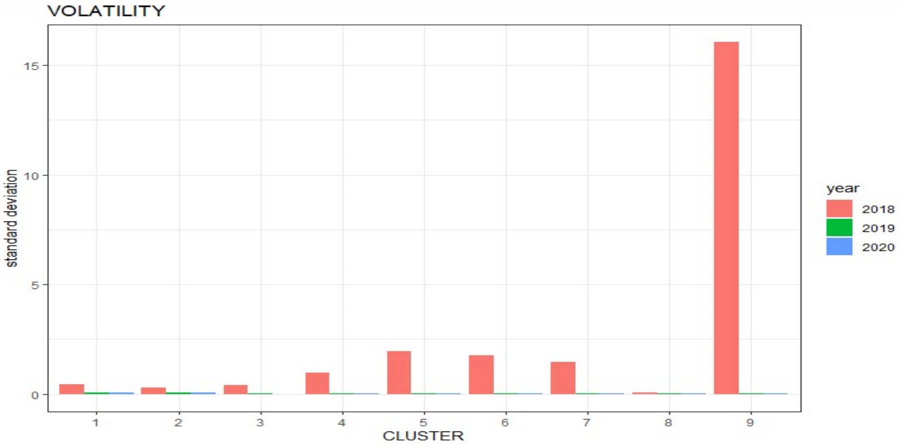


Figure 21. Volatility level

Each cluster has a higher level of volatility in 2018 than in other years. In 2018, cluster 9 has the highest level of volatility compared to other clusters. Further analysis will be carried out based on the liquidity level. In this analysis, we will combine the clustering result data with the aggregate data of the parameter magnitude. Then we will apply a group calculation of average liquidity in each year for each cluster based on clusters and years. We can see the plot display in the following figure.

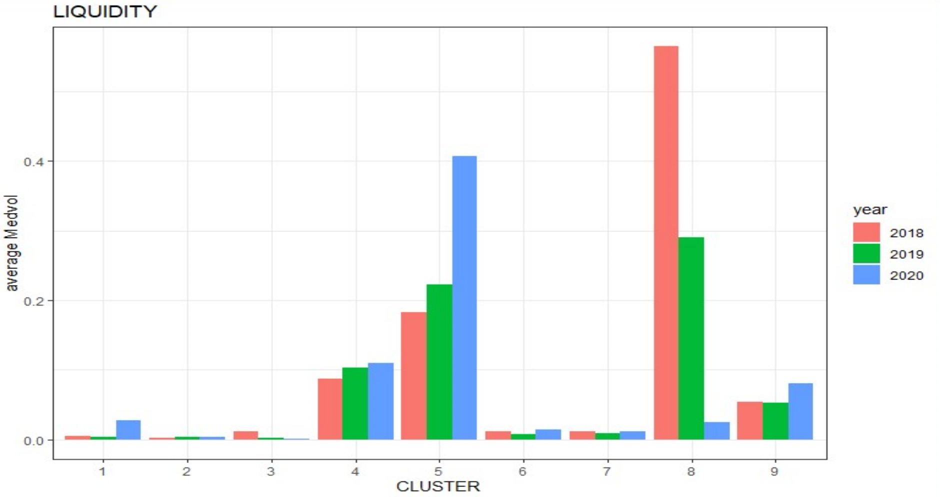


Figure 22. Liquidity level

Based on the plot above, cluster 4 has a relatively stable level of liquidity in the span of 2018 to 2020. Cluster 5 has an increase in liquidity levels during that timeframe, while cluster 8 has the highest level of liquidity in 2018 but has experienced a decrease in liquidity levels in the span of 2018 to 2020.

5. Conclusion

The stock data clustering process only considers the stock's technical features, such as market capitalization, volatility, and liquidity. K-Means and K-Medoid are basic data clustering algorithms that can be applied in clustering Indonesian stock data based on input parameters that have been processed in this article to extract information from stock data. The K-Means method is known to be sensitive to the outlier data. Therefore, outlier data was not included in the clustering process. In this article’s project, the K-Means method is more effective than K-Medoid, since K-Means has less WSS than K-Medoid which is less than 1500. Thus, the analysis of clustering results will be helpful for stock market players to choose stocks based on their characteristics.

References

Belajar, P. (2018, December 16). Algoritma K-Medoid dan Contoh Perhitungan Manual. Accessed April 27, 2021, from http://studyshut.blogspot.com/2018/12/algoritma-k-medoid-dan-contoh.html

Binty, T. (2019, July 8). K-Medoids/Partitioning Around Medoids (PAM) — Non Hierarchical Clustering with R. Accessed April 27, 2021, from https://medium.com/@tribinty/k-medoids-partitioning-around-medoids-pam-non-hierarchical-clustering-with-r-9d0af590bbc0

Cermati.com. (2020, September 11). Investasi Saham: Hal-Hal Dasar yang Mesti Anda Ketahui. Accessed April 28, 2021, from https://www.cermati.com/artikel/investasi-saham-hal-hal-dasar-yang-mesti-anda-ketahui

D. (2020, June 3). RPubs - Unsupervised Learning: Stocks Clustering. Accessed April 24, 2021, from https://rpubs.com/David21/stocksclustering

F, N. H. (2020, April 26). RPubs - Principal Component Analysis (PCA). Accessed April 28, 2021, from https://rpubs.com/nadhifanhf/principal-component-analysis

G. (2019, April 22). Knn distance plot for determining eps of DBSCAN. Accessed April 28, 2021, from http://htydjtk.blogspot.com/2019/04/knn-distance-plot-for-determining-eps.html

Indonesia, P. T. B. E. (n.d.). Daftar Saham. Accessed April 24, 2021, from https://www.idx.co.id/data-pasar/data-saham/daftar-saham/

Larose, D. T. (2005). Discovering knowledge in data: an introduction to data mining. New Jersey: John Wiley & Sons. Accessed April 21, 2021.

Priharto, S. (2021, January 29). Volatility Adalah: Pengertian, Jenis, Penyebab, dan Cara Menghitungnya. Accessed April 27, 2021, from https://accurate.id/ekonomi-keuangan/volatility-adalah/

S. (2018, September 7). Analisis Cluster dengan Menggunakan metode K-Means dan K-Medoids. Accessed April 27, 2021, from https://swanstatistics.com/analisis-cluster-dengan-menggunakan-metode-k-means-dan-k-medoids/

Umran, M., & Abidin, T. F. (2009). Pengelompokkan Dokumen Menggunkan K-Means dan Singular Value Decomposition: Studi Kasus Menggunakan Data Blog. Accessed April 21, 2021, from https://docplayer.info/33226080-Pengelompokkan-dokumen-menggunakan-k-means-dan-singular-value-decomposition-studi-kasus-menggunakan-data-blog.html

Widyawati, N. (2010). Perbandingan Clustering Based On Frequent Word Sequence (FWS) dan K-Means Untuk Pengelompokkan Dokumen Berbahasa Indonesia. Bandung. Accessed April 24, 2021.

Yobero, C. (2018, January 2). RPubs - K-Means Clustering Tutorial. Accessed April 21, 2021, from https://rpubs.com/cyobero/k-means