THE ANALYSIS OF BASKETBALL ATHLETES' POSITIONS BASED ON BODY HEIGHT USING THE DBSCAN ALGORITHM

[1]Yustinus Delvin Permana, [2]Rosita Herawati
[1,2]Program Studi Teknik Informatika Fakultas Ilmu Komputer,
Universitas Katolik Soegijapranata
[2]rosita@unika.ac.id

ABSTRACT

Basketball is one of the most popular sports in the world. In basketball, a proportional height is very important to play optimally. Therefore, the analysis of basketball athletes' positions based on height was made using the DBSCAN algorithm. The DBSCAN parameters in the form of epsilon and minimum points in this study were determined using the elbow method and silhouette which turned out to be unsatisfactory results from the elbow method due to data problems. Comparing the silhouette score with epsilon is an alternative to the elbow method that has been tried and the result is an epsilon of 2.54 while the other parameter, namely the minimum points used is 4 because in processing the data in this study, it is divided into 3 times, each of which has a data dimension of 2. The final result can be obtained well if not using the theory elbow method even though the performance is reduced but the results can be read well.

Keywords: DBSCAN, Silhouette, Basketball

INTRODUCTION

Background

There are many sports in the world today, one of which is basketball. Basketball is one of the most popular sports in the world today as well as in Indonesia there are many basketball competitions ranging from amateur leagues to professional leagues. In basketball, of course, a proportional height is needed in order to compete in matches because of course it is not balanced if short players must face tall players. In basketball, there are 5 classifications of player positions, namely point guard, shooting guard, small forward, power forward center and because basketball is a sport that requires a proportional height, the placement of players must also be important.

From the problems above, that the placement of players' positions based on height is very important in basketball because of course, it will affect the performance of basketball athletes. This study using the concept of clustering and using the DBSCAN algorithm to place the athletes in an optimal position based on the athletes' height.

In this study, the DBSCAN algorithm was used by clustering player positions based on the athletes' height and performance. The total column used is 4 columns, namely the height column and the performance column, which has 3 different columns, namely the points column, rebounds, assists. then the results obtained are to place the athletes in the right position between the 3 main positions in basketball, namely guard, forward, or center.

LITERATURE STUDY

In the first journal from Kanagala, Hari Krishna, and V.V. Jaya Rama Krishnaiah [1] compared the K-Means, DBSCAN, AND Optics algorithms. K-Means algorithm is only applied when the mean of the cluster is defined. in advance, KMeans will not identify Outliers then the DBSCAN algorithm can find clusters of arbitrary shape, determine what information should be classified as noise or outliers. It is very fast when compared to other algorithms. In DBSCAN, the user has the responsibility of selecting the parameter values epsilon and minimum points. Slightly different parameter settings may lead to different clusters. It has some difficulties in distinguishing separated clusters if they are located too close to each other, even though they have different densities. To overcome this difficulty, the OPTICS algorithm was developed. OPTICS ensures good quality clustering by maintaining the order in which the data objects are processed, high-density clusters are given priority over lower density clusters. OPTICS also requires parameters epsilon and minimum points to be specified by the user that will affect the result. The efficiency of clustering algorithms can be improved by removing the limitations of the clustering techniques.

Then in the next research journal Giri, Kinsuk, and Tuhin Kr Biswas [7] made a method to find the epsilon value other than using KNN distances. In this research, a method is developed to find the optimal value of epsilon using empty circles in computational geometry, the results are compared with the calculation of epsilon using KNN distances. The results obtained from the empty circle method are still not as optimal as the KNN distance but can still be an alternative choice if not using the KNN distance.

Then in the next journal Nisa et al [9] created an application that displays clustering of data hotspots using the DBSCAN algorithm. The first step in this research is taking the dataset and then determining the parameters to be used, namely epsilon and minimum points to determine epsilon in this study using the k-dst graph then using the elbow method and then to find the minimum points in this journal stating the formula is D + 1 but because the data which is used is 2-dimensional data, the minimum points that are recommended to be used are 3 or 4 in this study, number 4 was chosen to be the minimum points parameter, then the next step is to display the results of the DBSCAN parameters that have been determined in the form of a website to draw conclusions.

From the explanation of the article above, this research will use the DBSCAN algorithm with NBA player stats data from 1996 to 2021. Then to determine the parameters of DBSCAN, it will use the silhouette score and elbow method, after which the performance will be seen using v-measure for a more complete explanation. seen in research methodology.

RESEARCH METHODOLOGY

Gathering Data

The dataset used for this study was taken from Kaggle's website https://www.kaggle.com/justinas/nba-players-data in CSV format with 22 columns and 4 columns are used namely the height column or player height, points or pts, rebound or reb, assist or ast. Only 4 columns were taken because there are only 3 columns that describe the overall performance in this dataset and the other 1 column is the height column.

Program Implementation

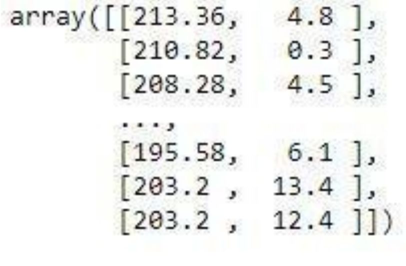
In the clustering process using the DBSCAN algorithm, the first thing to do is take the data to be processed. After taking the data to be processed, check whether the data is clean or not so that it can be used after that determine the parameters, namely epsilon and minimum points. To determine the minimum points, the formula is D+1 but because the data used is 2-dimensional data, the minimum points can be 4. Then to determine the epsilon the first step is to use the k-distance graph and then do the elbow method and then the next step is to see the results of the silhouette to optimize the parameters of the DBSCAN. After determining DBSCAN parameters to optimize the cluster results, do the calculations of the number of clusters to get the silhouette score. After that, the next step is to run the DBSCAN algorithm and visualize to get the results.

TESTING

To get the results of the data needed to be able to advise athletes on the optimal position, the data will be processed. In testing, there are 4 steps, namely Processing Data, define parameters, Calculating Cluster Performance with Homogeneity, Completeness, and V-Measure, Evaluation Result. With these 4 processes carried out to get the best results in carrying out this project, a detailed explanation can be seen below.

Processing Data

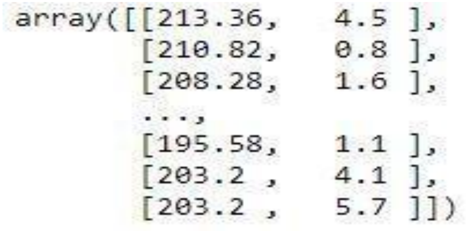
After knowing the data type of each attribute that will be used, it is found that the data type is int then the data type is changed to float because it will be more optimal because the data used is decimal then the data can be processed in DBSCAN the next step is to convert it into an array because DBSCAN it is more optimal if used in data with dimension 2, the author will divide it into 3 different arrays, namely the player_height pts, player_height reb and, player_height ast arrays with the following results:



```
array([[213.36,   4.8 ],
       [210.82,   0.3 ],
       [208.28,   4.5 ],
       ...,
       [195.58,   6.1 ],
       [203.2 ,  13.4 ],
       [203.2 ,  12.4 ]])
```

Figure 1. Array pts

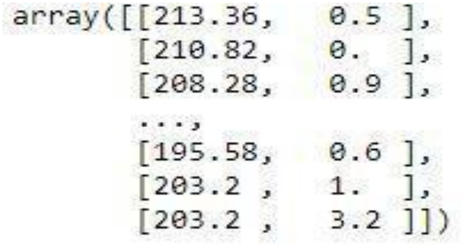
This is an array view with the contents of height and points. Height is identified as player_height or in the image on the left and points are recognized as pts on the right in the image above. So the image above displays an array form with data type float64 from data player_height and points which will be used for visualization of DBSCAN points.



```
array([[213.36,   4.5 ],
       [210.82,   0.8 ],
       [208.28,   1.6 ],
       ...,
       [195.58,   1.1 ],
       [203.2 ,   4.1 ],
       [203.2 ,   5.7 ]])
```

Figure 2. Array reb

Then the picture above is an array of player_height and reb. The player_height pictured above can be seen on the left and the rebound or rebound can be seen on the right. So figure above is an array of data for visualization of DBSCAN rebound.



```
array([[213.36,   0.5 ],
       [210.82,   0.  ],
       [208.28,   0.9 ],
       ...,
       [195.58,   0.6 ],
       [203.2 ,   1.  ],
       [203.2 ,   3.2 ]])
```

Figure 3. Array ast

Then for the ast array, it can be seen above, the same as before, on the left, is the player_height or height data and the right is assist or ast. so the picture above is an array image for the assist data which will later be used for visualization of the DBSCAN assist.

Define Parameters

In doing clustering with DBSCAN, the most important thing is to recognize epsilon and minimum points. Recognizing epsilon and minimum points in DBSCAN, there are various ways in this project, elbow methods and silhouette calculations will be used in determining DBSCAN parameters. Determination of parameters is very important to get optimal results so that the parameters must be precise. The calculation of parameters in this project can be seen with the explanation below.

Elbow Method

Elbow method is used to determine the epsilon value by plot a k-distance and choose the epsilon value at the “elbow” of the graph.

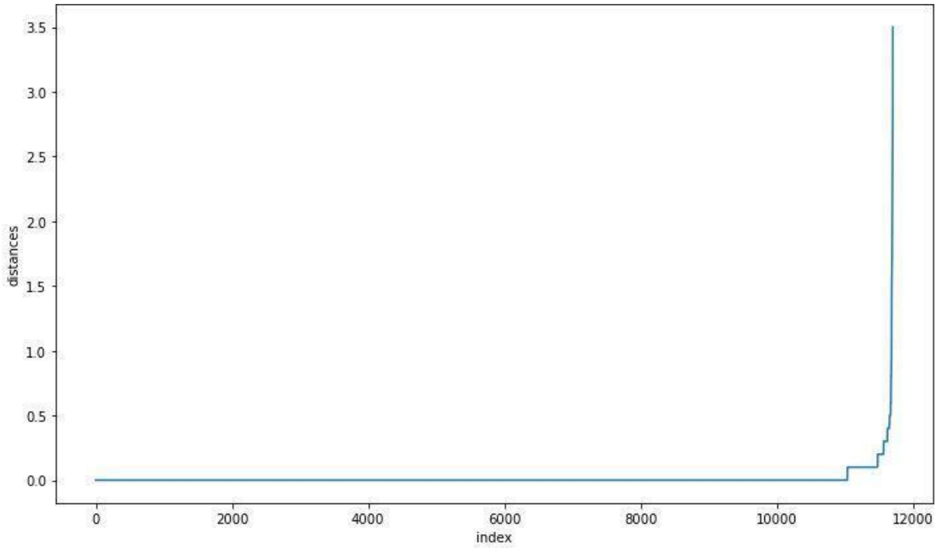


Figure 4. Elbow pts

The picture above is the result of the elbow method from player_height and pts or height and points. From these results, it can be seen that the recommended epsilon value in the elbow graph is between 0.1 to 0.5. So from the elbow results, an epsilon between 0.1 and 0.5 will be used.

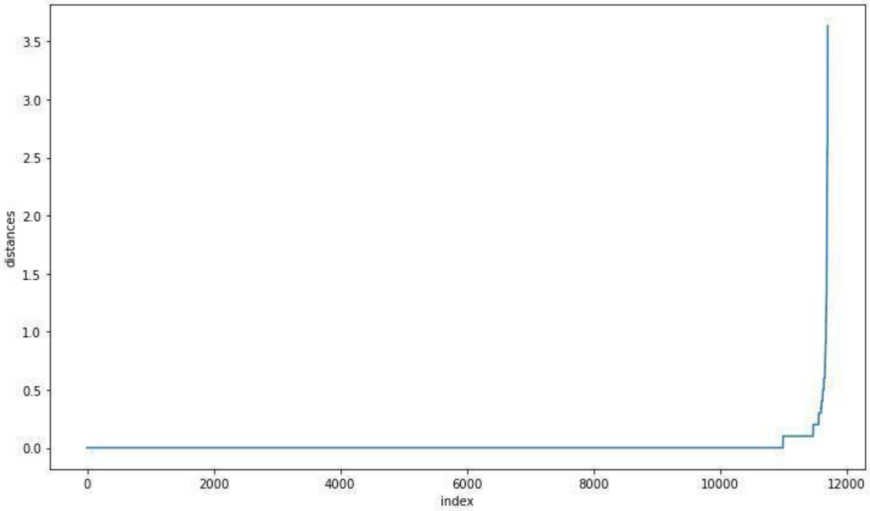


Figure 5. Elbow reb

The image above is the result of the elbow method player_height and reb. Then from player_height and reb or height and rebound the result is 0.1 to 0.5. So from the results of the elbow graph above, an epsilon between 0.1 and 0.5 will be used for rebound.

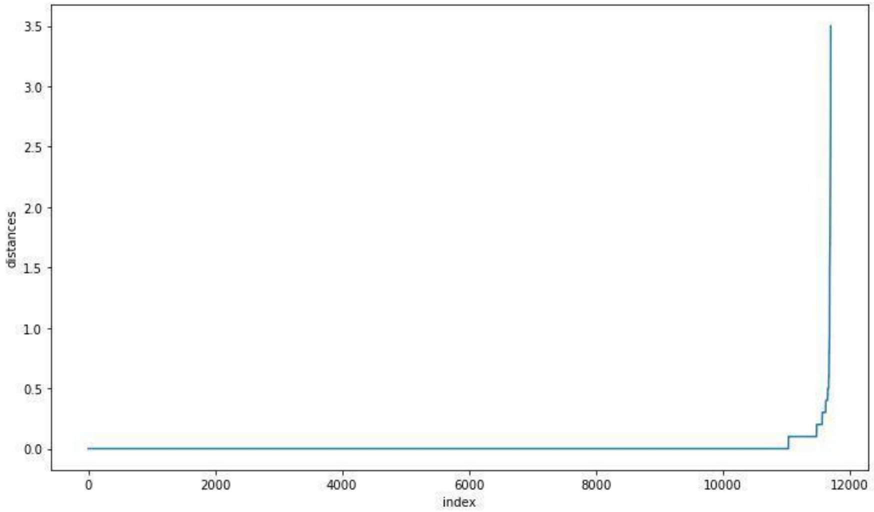


Figure 6. Elbow ast

For the last ones are player_height and ast or height and assists, which is the most optimal value in the numbers 0.1 to 0.5. So it can be concluded that the best epsilon according to the elbow method is 0.1 to 0.5 in every comparison of player height and performance.

Silhouette

In the elbow graph, the epsilon value obtained is between 0.0 to 0.5, so to get the epsilon value the next step is to find the silhouette score because epsilon requires a specific value to be used as a DBSCAN parameter. In this process, what will be done is to calculate silhouettes from 0.1 to 0.5 in each 1 array which includes points height, rebound height, assist height. The formula to get the epsilon value is:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \tag{1}$$

where

a(i) = The average distance of that data point with all other points on the same cluster.

b(i) = The average distance of that data point with all member from closest cluster.

From this formula, the following results are obtained :

Table 1 : Silhouette Epsilon

Attributes	Epsilon	Sillhouete Score	Number of Cluster
pts	0.1	0.4337728516676882	803
pts	0.2	0.022486297484630338	159
pts	0.3	-0.0848043229195178	102
pts	0.4	-0.004100076428562309	71
pts	0.5	0.025565428625584204	49
reb	0.1	0.6119029681990025	544
reb	0.2	0.27348386404451663	71
reb	0.3	0.2603117726498396	39
reb	0.4	0.34975338931087274	34
reb	0.5	0.3538775380391027	31
ast	0.1	0.5789924892829579	444
ast	0.2	0.42925860574902186	62
ast	0.3	0.4289278801612192	38
ast	0.4	0.5243169308165907	37
ast	0.5	0.5323757032535927	30

With these results, it can be seen that the most optimal silhouette value in all attributes is 0.1. But there is a problem with the number of clusters created because the clusters created have a very large value for pts it produces 803 clusters while reb 544 then assists 444 clusters. The number of clusters created later will make it difficult to read the results of this study, therefore a larger epsilon is used to be able to read the results more easily and the selected epsilon is 2.54 for all attributes. Because the value of 2.54 is the most optimal value to reduce the number of clusters with the result :

Table 2 : Silhouette epsilon 2

Attributes	Epsilon	Sillhouete Score	Number of Cluster
pts	2.54	0.20488030533531085	13
reb	2.54	0.3882232972867212	11
ast	2.54	0.4474785739369151	9

After getting the epsilon value, it must know whether the number of clusters is optimal, therefore the number of clusters will be compared with the silhouette to determine the accuracy of the number of clusters with the results :

Table 3 : Silhouette Number of Clusters

Attributes	Number of Clusters	Silhouette
pts	803	0.5694014120442495
pts	13	0.34303459693078636
reb	544	0.6075749842986854
reb	11	0.41197972123526666
ast	444	0.7298066783007097
ast	9	0.4821576782297272

Then to determine the minimum points parameter can use the formula

$$minPts = D + 1 \tag{2}$$

Where,

D = Dimension of the data

According to the existing formula, the minimum points that will be obtained is 3, but because the dimensions of the data are 2, it is also recommended to use 4 minimum points. Because the epsilon value to be used is 2.54 and with MinPts 3 or 4 it will not change the number of clusters created, the MinPts value used is 4. It is different with the epsilon value of 0.1 or the most optimal value for this study, the number of cluster numbers will change with the results :

Table 4 : Minimum Points and Number of Cluster Table

Attributes	MinPts	Number of Clusters
pts	3	957
pts	4	803
reb	3	603
reb	4	544
ast	3	499
ast	4	444

From the silhouette calculation above, the parameters in this study will use epsilon 2.54 and minimum points 4.

Calculating Cluster Performance with Homogeneity, Completeness, and V-Measure

Homogeneity is used to calculate each cluster that has data points with belonging labels. Homogeneity describes the clustering algorithm's closeness to perfection while completeness calculates where all data points belonging to the same class are clustered into the same cluster then the V-Measure is the harmonic mean of the homogeneity and completeness. The homogeneity, completeness, and V-measure values obtained from this study are :

Table 5 : Player Height Performance

Attributes	Epsilon	Minimum Points	Homogeneity Player Height	Completeness Player Height	V-Measure Player Height
player_height (pts)	0.10	3	0.89418	0.37699	0.53037
player_height (pts)	0.10	4	0.84979	0.37510	0.52046
player_height (pts)	2.54	4	0.55592	0.99565	0.71348
player_height (reb)	0.10	3	0.96892	0.42360	0.58948
player_height (reb)	0.10	4	0.95129	0.42228	0.58491
player_height (reb)	2.54	4	0.55671	0.99885	0.71494
player_height (ast)	0.10	3	0.96979	0.48228	0.64420
player_height (ast)	0.10	4	0.95351	0.48167	0.64003
player_height (ast)	2.54	4	0.55568	1.00000	0.71439

From the results of the performance calculation for height, the best homogeneity was found at epsilon 0.1 and a minimum of points 3 then completeness was best at 2.54 and 4 and for V-measure the best at 2.54 and 4.

Table 6 : Performance of player stat

Attributes	Epsilon	Minimum Points	Homogeneity	Completeness	V-Measure
pts	0.10	3	0.75785	0.63371	0.69024
pts	0.10	4	0.71904	0.62950	0.67130
pts	2.54	4	0.01657	0.05887	0.02586
reb	0.10	3	0.83141	0.60485	0.70026
reb	0.10	4	0.81583	0.60264	0.69321
reb	2.54	4	0.03268	0.09756	0.04896
ast	0.10	3	0.75768	0.55033	0.63757
ast	0.10	4	0.74219	0.54760	0.63022
ast	2.54	4	0.04097	0.10769	0.05936

Then for the calculation of player performance with cluster performance, it can be seen that homogeneity, completeness, and V-measure are best at epsilon 0.1 and at minimum points 3. so that according to the calculation of the most optimal performance when using epsilon 0.1 and at least points 3 because of the calculation of height and performance. epsilon 0.1 and minimum points 3 players get the highest total score and if epsilon 0.1 and minimum points 4 are also good but the results are slightly worse than 0.1 and 3 while for epsilon 2.54 and at minimum points 4 the height performance is very good even outperforming 0.1 and 3 but in player performance, the results are not good.

Evaluation Result

For the visualization results of each epsilon and the minimum points aimed are:

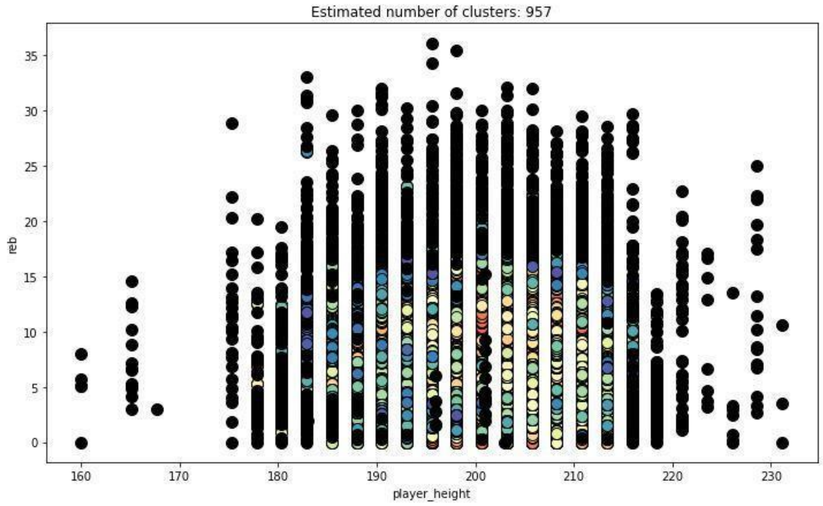


Figure 7. Points

The picture above is a visualization with epsilon 0.1 and a minimum of 3 points. From the visualization results obtained above, it can be seen that the results have a lot of noise and a lot of clusters with different density levels. So from the results above, no conclusions can be drawn for the optimal height in points.

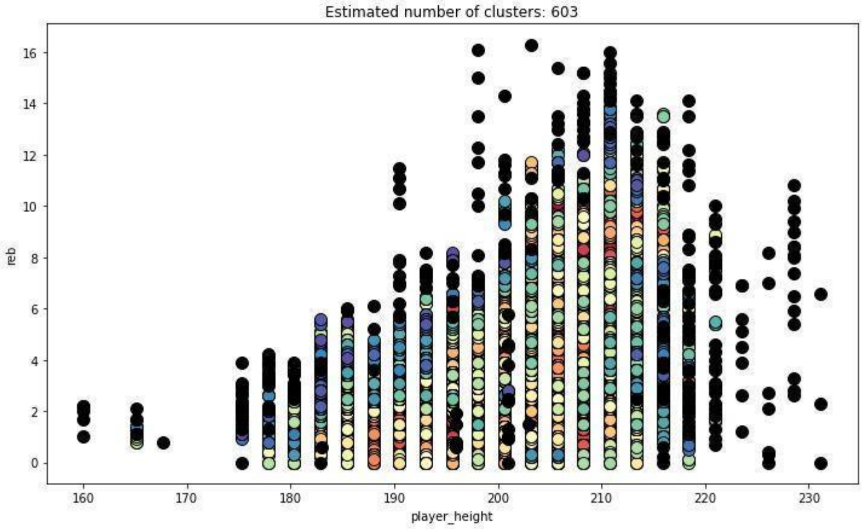


Figure 8. Rebound

Next is the result of the rebound visualization with epsilon 0.1 and minimum points 3. From the results of the visualization, it is still the same as before, namely points that have a lot of noise, lots of clusters, and different densities of each cluster so that it cannot be concluded that the optimal height for rebounding cannot be concluded.

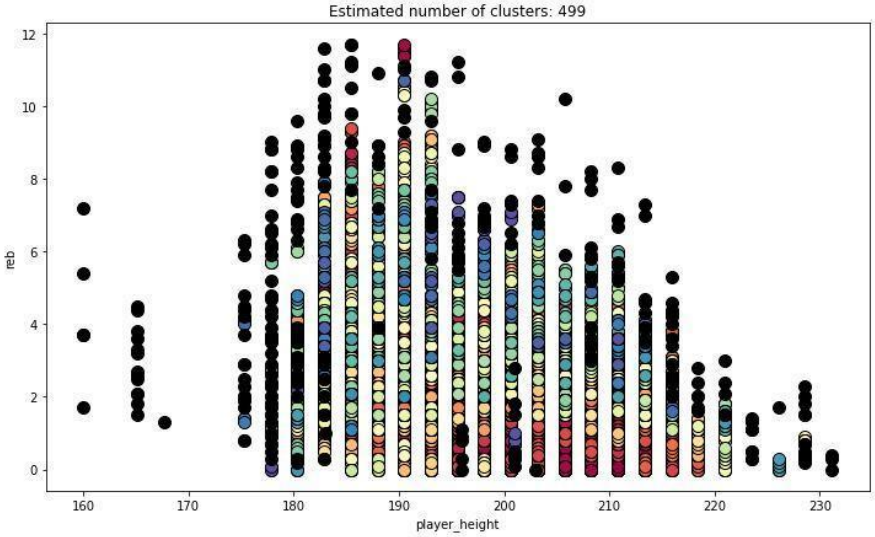


Figure 9. Assist

Then for visualization assists with epsilon 0.1 and minimum points 3 is also the same as a lot of noise, a large number of clusters, and varying densities so that it cannot be concluded that the optimal height for assists cannot be drawn. So for epsilon 0.1 and minimum points 3 will not be used because from the visualization results it can not be concluded that the optimal height for each player's performance.

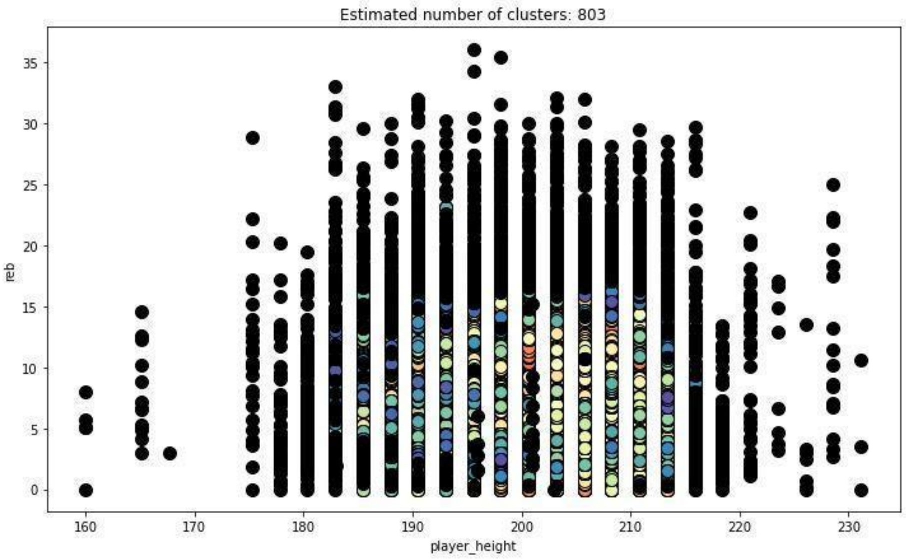


Figure 10. Points

The next image is a visualization of epsilon 0.1 and minimum points 4 of points. From the picture above, it can be seen that there is still a lot of noise even though it has reduced from epsilon 0.1 and minimum points 3 and also the number of clusters is still large and the cluster density is still diverse, so from the results of visualization of points with epsilon 0.1 and minimum points 4, it cannot be concluded that high optimal body.

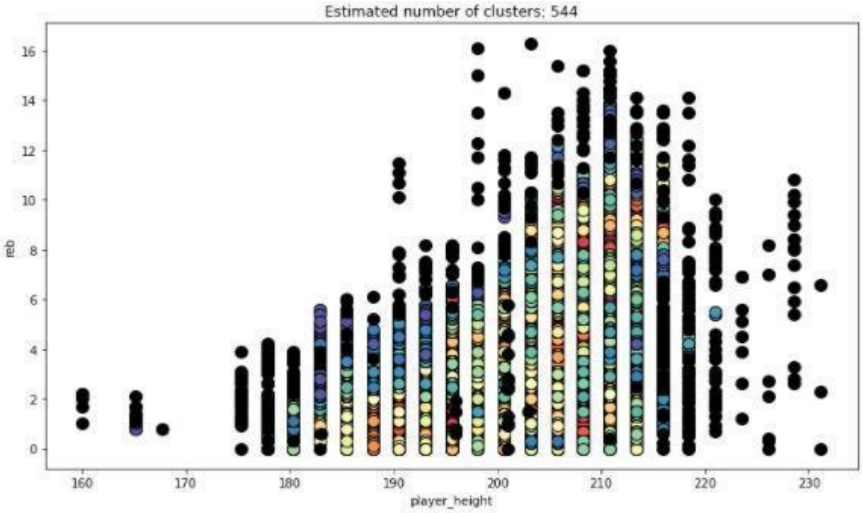


Figure 11. Rebound

Then next is the visualization of the rebound with epsilon 0.1 and minimum points 4 the results can be seen that the noise has decreased from epsilon 0.1 and minimum points 3 but is still the same in a large number of clusters and varying cluster densities so it is still not possible to conclude for optimal height.

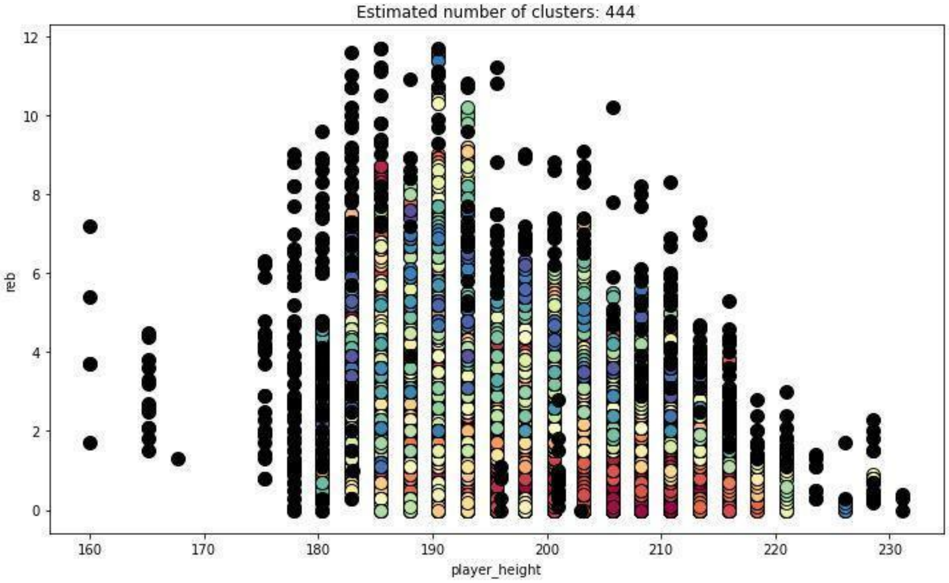
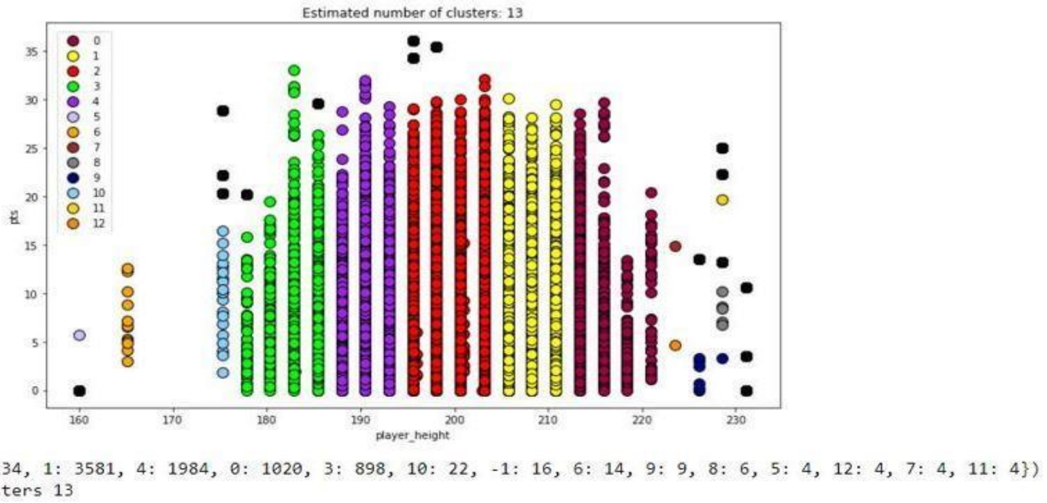


Figure 12. Assist

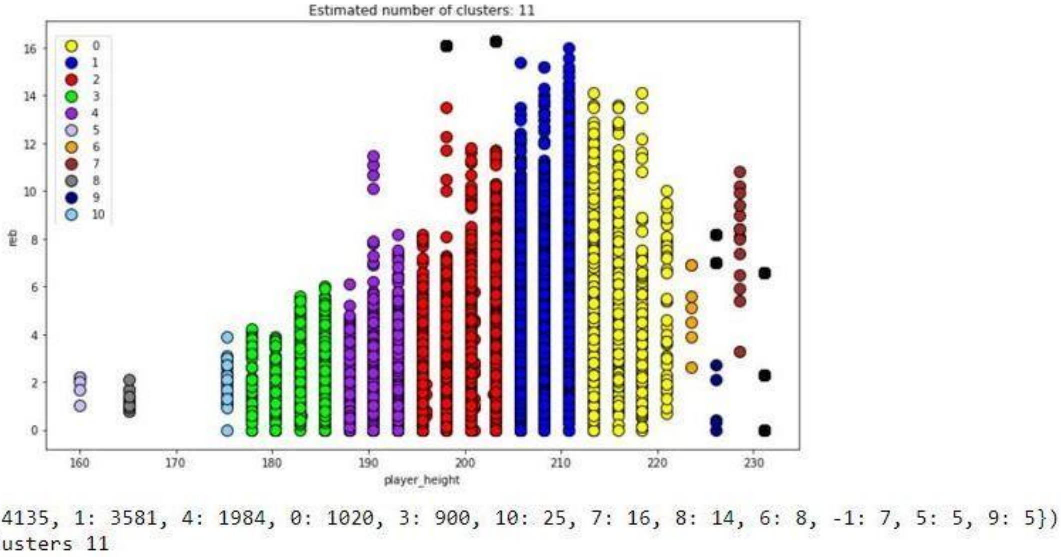
Then on the assist epsilon 0.1 and minimum points 4 it can be seen that the noise has decreased but the number of clusters and the density of clusters is still the same as epsilon 0.1 and minimum points 3. So for epsilon 0.1 and minimum points 4 will not be used because no conclusions can be drawn.



```
Counter({2: 4134, 1: 3581, 4: 1984, 0: 1020, 3: 898, 10: 22, -1: 16, 6: 14, 9: 9, 8: 6, 5: 4, 12: 4, 7: 4, 11: 4})
number of clusters 13
```

Figure 13. Points

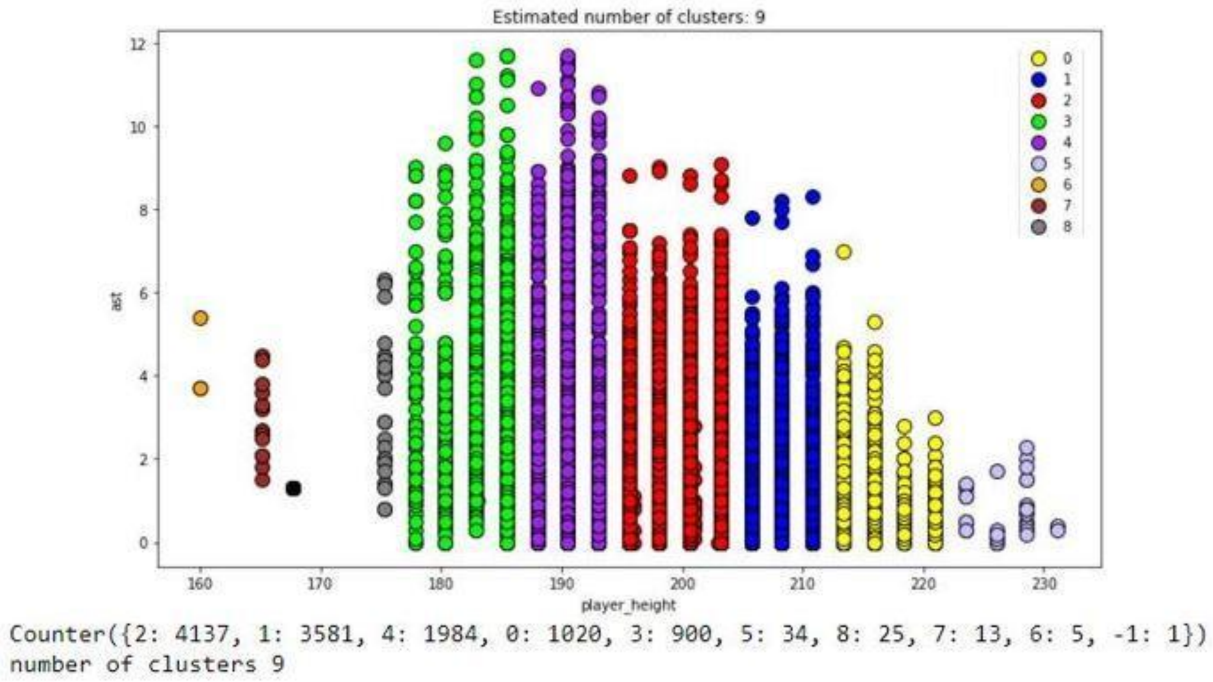
Then with that, the author uses epsilon 2.54 and minimum points 4 to be able to overcome the above problem. It can be seen from the results above that with epsilon 2.54 and minimum points 4 the visualization results of points noise is reduced then cluster density and the number of clusters also improve so it can be concluded that the author chooses to use epsilon 2.54 and minimum points 4 in making optimal height decisions for points.



```
Counter({2: 4135, 1: 3581, 4: 1984, 0: 1020, 3: 900, 10: 25, 7: 16, 8: 14, 6: 8, -1: 7, 5: 5, 9: 5})
number of clusters 11
```

Figure 14. Rebound

Next is the rebound visualization with epsilon 2.54 and the minimum points 4 are the same as the points the author chooses to use epsilon 2.54 and minimum points 4 because the noise produced is less and the number of clusters and cluster density is improving so that conclusions can be drawn for the optimal height in rebounding.



```
Counter({2: 4137, 1: 3581, 4: 1984, 0: 1020, 3: 900, 5: 34, 8: 25, 7: 13, 6: 5, -1: 1})
number of clusters 9
```

Figure 15. Assist

The assists are also the same as points and rebounds, the author uses epsilon 2.54 and minimum points 4 because the number of clusters and the density of clusters is improving from the previous epsilon and minimum points and noise is also reduced. So the visualization of 2.54 and minimum points 4 can be concluded so that the author chooses to use it

Table 7 : Result

Position	**Height**
Point Guard	187-205 cm
Shooting Guard	195-205 cm
Small Forward	195-205 cm
Power Forward	197-210 cm
Center	197-210 cm

Based on the tests carried out on this project in determining the parameters, it is estimated that the usage method is not appropriate. The drawback is in determining the elbow method using the K-dist Graph, in the elbow method the Epsilon results are 0.1 to 0.5 and the performance results achieved are very satisfactory with the largest being at 0.1 then for MinPts using D + 1 or 4 formulas for 2-dimensional data. Then the results obtained are not satisfactory because there is a lot of noise in the middle of the data. then after the elbow method is used, the silhouette comparison is quite satisfactory by getting a value of 2.54 with MinPts 4 can greatly reduce the amount of noise previously obtained and the data is grouped better and more legible. Then for the classification of the position of basketball athletes according to height using DBSCAN taken from the highest cluster and the densest cluster for cluster points and height, it was found that the acquisition of points in many heights was quite average but the cluster results obtained showed that cluster 2 was the most numerous so that for The ratio of points and heights for shooting guards and small forwards is 195-205 cm. For rebounds and height for the power forward and center, it was found that the densest cluster was cluster 2, which means that players with a height of 195-205 are the players who do the most rebounds, but because of the distance that is quite far with the highest cluster, namely, cluster 1 which is also the second-largest cluster. After cluster 2, for the comparison of height and rebound for the power forward and center, 2 clusters were taken, namely clusters 1 and 2 which ranged from 195-210 cm in height, then the last one for the comparison of height and assists for the densest point guard cluster was also in cluster 2. and the same as the rebound, the distance between the densest cluster and the highest value is also quite far, so 2 clusters are also taken for the comparison of height and assists so that they are 187-205 cm tall.

CONCLUSION

From the results, it can be seen that with epsilon 2.54 and minimum points 4 used by the author, it can be concluded that the optimal height for each basketball player position. For the height results produced, it also correlates with an explanation of the tasks of basketball players in each position, With the center being recognized as the tallest player in a team, the height is between 195-210 cm, then the point guard is the shortest player at 187-205 cm, and for the point scorer who is in charge against big and small players between the heights of 195-205 cm. So clustering with DBSCAN can conclude the most optimal height for each basketball player position.

The DBSCAN algorithm is less than optimal in this study because from the use of the most optimal epsilon and minimum points, no conclusions can be drawn so that the authors change the epsilon to be larger to be able to draw conclusions. It cannot be concluded that epsilon and minimum points are the most optimal in this study due to the varying cluster densities and too many clusters.

Suggestions for future research in processing data as in this study is to use the OPTICS algorithm. It is recommended to use the OPTICS algorithm because the data used produces different cluster densities and a large number of clusters. The OPTICS algorithm was created to overcome the weakness of the DBSCAN algorithm which is difficult to handle varying cluster densities.

References

[1] Kanagala, Hari Krishna, and V.V. Jaya Rama Krishnaiah. "A Comparative Study of K-Means, DBSCAN and OPTICS." In *2016 International Conference on Computer Communication and Informatics (ICCCI)*, 1–6. Coimbatore, India: IEEE, 2016. https://doi.org/10.1109/ICCCI.2016.7479923.

[2] Chen, Yewang, Shengyu Tang, Nizar Bouguila, Cheng Wang, Jixiang Du, and HaiLin Li. "A Fast Clustering Algorithm Based on Pruning Unnecessary Distance Computations in DBSCAN for High-Dimensional Data." *Pattern Recognition* 83 (November 2018): 375–87. https://doi.org/10.1016/j.patcog.2018.05.030.

[3] Li, Mingyang, Xinhua Bi, Limin Wang, and Xuming Han. "A Method of Two-Stage Clustering Learning Based on Improved DBSCAN and Density Peak Algorithm." *Computer Communications* 167 (February 2021): 75–84. https://doi.org/10.1016/j.comcom.2020.12.019.

[4] Zhou, Hong Bo, and Jun Tao Gao. "Automatic Method for Determining Cluster Number Based on Silhouette Coefficient." *Advanced Materials Research* 951 (May 2014): 227–30. https://doi.org/10.4028/www.scientific.net/AMR.951.227.

[5] Batool, Fatima, and Christian Hennig. "Clustering with the Average Silhouette Width." *Computational Statistics & Data Analysis* 158 (June 2021): 107190. https://doi.org/10.1016/j.csda.2021.107190.

[6] Rehman, Saif Ur, Kamran Aziz, Simon Fong, and S Sarasvady. "DBSCAN: Past, Present and Future," n.d., 7.

[7] Giri, Kinsuk, and Tuhin Kr Biswas. "Determining Optimal Epsilon (Eps) on DBSCAN Using Empty Circles," n.d., 10.

[8] Hou, Jian, Huijun Gao, and Xuelong Li. "DSets-DBSCAN: A Parameter-Free Clustering Algorithm." *IEEE Transactions on Image Processing* 25, no. 7 (July 2016): 3182–93. https://doi.org/10.1109/TIP.2016.2559803.

[9] Nisa, Karlina Khiyarin, Hari Agung Andrianto, and Rahmah Mardhiyyah. "Hotspot Clustering Using DBSCAN Algorithm and Shiny Web Framework." In *2014 International Conference on Advanced Computer Science and Information System*, 129–32. Jakarta, Indonesia: IEEE, 2014. https://doi.org/10.1109/ICACSIS.2014.7065840.

[10] Aliguliyev, Ramiz M. "Performance Evaluation of Density-Based Clustering Methods." *Information Sciences* 179, no. 20 (September 29, 2009): 3583–3602. https://doi.org/10.1016/j.ins.2009.06.012.

[11] “BASKETBALL POSITIONS” Accessed December 20, 2021.

https://jr.nba.com/basketball-positions/#:~:text=Players%20in%20a%20basketball%20game,point%20guard%2C%20and%20shooting%20guard.&text=The%20center%20is%20the%20tallest,on%20close%20shots%20and%20rebound.