K-Means Algorithm Performance Analysis With Determining The Value Of Starting Centroid With Random And KD-Tree Method

Kamson Sirait¹, Tulus², Erna Budhiarti Nababan³
¹,²,³Department of Computer Science, Faculty of Computer Science and Information Technology, Universitas Sumatera Utara, Medan, Indonesia, 20155
Email: kamsonsirait@gmail.com¹, tulus_jp@yahoo.com², ernastp@yahoo.com³

Abstract. Clustering methods that have high accuracy and time efficiency are necessary for the filtering process. One method that has been known and applied in clustering is K-Means Clustering. In its application, the determination of the beginning value of the cluster center greatly affects the results of the K-Means algorithm. This research discusses the results of K-Means Clustering with starting centroid determination with a random and KD-Tree method. The initial determination of random centroid on the data set of 1000 student academic data to classify the potentially dropout has a sse value of 952972 for the quality variable and 232.48 for the GPA, whereas the initial centroid determination by KD-Tree has a sse value of 504302 for the quality variable and 214.37 for the GPA variable. The smaller sse values indicate that the result of K-Means Clustering with initial KD-Tree centroid selection have better accuracy than K-Means Clustering method with random initial centroid selection.

Keyword: Clustering, K-Means, KD-Tree.

1. Introduction
One of the known and applied clustering methods is K-Means Clustering. Clustering is one of the data mining techniques used to obtain groups of objects that have common characteristics in large enough data. K-Means Clustering algorithm can be used to predict student academic performance [1]. In terms of time and accuracy, K-Means produces better results [2]. K-Means grouping is a partitioning grouping technique in which groups are formed with the base of the centroid [3]. K-Means is a known grouping technique based on partitions that try to find centroid that represent the number of clusters. But the output is enough sensitive to the beginning of the cluster center position. Finding the starting point of centroid can make the algorithm more effective and efficient so that the results obtained are more accurate and the time required to process the data is substantially reduced [4]. The K-Means algorithm always selects a different starting center for each run of algorithm for the same dataset and k value, then the algorithm is optimized by modifying it with the initial center selected, resulting in fewer execution times for large datasets [5].

Selection of the starting position from the middle point of a bad cluster will result in the algorithm of K-Means stuck in an optimal local solution [6]. Therefore, the determination of the starting point value of the clustering center will greatly determine the results obtained by the K-Means Clustering algorithm [7]. To determine the correlation between two objects by using the following Euclidean Distance formula:

\[ d_{Euclidean}(x, y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \] .................................(1)

Dimana :
\( d(x,y) \) = The distance of data to x to the center of the cluster y.
\( x_i \) = data i in n data.
\[ y_j \quad = \quad \text{data j in n data.} \]

Clustering using K-Means with initial centroids is determined randomly.

a. Determine the value of K desired, where the value of K is the number of clusters to be formed.

b. Determine the initial centroids. The initial centroid is randomly assigned from the existing data and the number of cluster is equal to the number of initial centroid.

c. Find the nearest centroid of each data by calculating the distance to each centroid using the Euclidean Distance formula.

d. Grouping data by the minimum distance. A data will be a part of a cluster that has the closest (smallest) distance from its cluster center.

e. Based on these groupings, then look for new centroids based on the average of the data of each cluster.

f. Go back to stage 3.

g. The recurrence stops if no data is migrated.

KD-Tree is a data tree structure that which each node can accommodate many data sets and is used as a tool to accommodate the evaluation data elements, by devide the the data set into two subsets in which each evaluation stage. Each subset is stored in a node tree to form a left and right subtree. The same evaluation on both subtrees is done as long as the node is found with the number of members more than the specified maximum limit. If each node is assumed as "h" and many desired maximum data elements for each node are "w", then each leaf node must meet the following conditions:

\[ 1 \leq h \leq w \quad \text{........................................}(2) \]

Simply the properties of the k-d tree data structure can be described as follows [8]:

a. KD-tree is a binary tree.

b. The root node holds all elements of the data set.

c. The data elements at each node are divided into two parts based on a particular variant, such that one part becomes the left subtree elements and one part becomes the right subtree elements.

d. The subtrees on the left and right will be recursively reconstructed, as long as each node "h" does not meet \( 1 \leq h \leq w \).

KD-Tree K-Means Clustering is clustering method that the initial centroid determined by KD-Tree.

a. The establishment of KD-Tree from the data set. The KD-Tree created has a leaf bucket.

b. For each leaf bucket (L1, L2, ... Lj), calculate the density value (Pj) of each leaf bucket Lj, and calculate the middle point value of leaf bucket (Mj) by finding the average value of all points on the leaf bucket Lj.

c. Select the middle point of the first cluster C1 = Mz, where \( z = \arg \max (Pj) \).

d. For \( t = 2, ..., K \):
   For \( j = 1, ..., q \) calculate the leaf bucket ranking (Gj).

e. \( C_t = Mz, \text{where} \ z = \arg \max (Gj) \)

f. Remove 20% leaf bucket with the lowest density value. Go back to step 3 and calculate the position of the new middle point of the cluster K (c1, c2, ... ck)

g. Run the K-Means Clustering algorithm with the initialization of the middle point (C1, ... Ck).

2. Computational Analysis

2.1 Clustering Using K-Means With Starting point of Centroid Specified by Random.

In this section will be discussed about the results obtained in this study, to the analysis that has been done is whether by determining the initial centroid using KD-Tree on the grouping of data with K-Means can give better results with a smaller sum square error Compared to the initial determination of
centroid randomly in determining the potential dropout of students' academic data. GPA data is a cumulative prastasi index of students from the semester that has been lived, while the quality data is the achievement of SKS that has been lived times with the GPA obtained by students. The first stage is initialization of the beginning of cluster where the value of K to be formed is 3. Conducted selection K = 3 data as the initial centroid randomly, that is selected 100th, 500th, and 900th data. Table 1 presents the centroid values named C1, C2 and C3. Data to 100 (N100) becomes first Centroid (C1), data to 500 (N500) becomes second Centroid (C2) and data to 100 (N100) becomes Centroid third (C3). Next each data is calculated the distance to the nearest centroid. The nearest Centroid will be the cluster followed by the data. Next is calculated the new centroid for each cluster ie the average of all the data that joins in each cluster as shown in table 2. The average value on each cluster will be the new centroid value for the next iteration.

| Centroid | Data   | Quality | GPA  |
|----------|--------|---------|------|
| 1        | N100   | 424     | 3.59 |
| 2        | N500   | 362     | 3.07 |
| 3        | N900   | 399     | 3.33 |

Table 2. Result Clustering New Cluster Iteration 1

| Cluster | Member | Quality Average | GPA Average |
|---------|--------|-----------------|-------------|
| 1       | 253 point | 430             | 3.64        |
| 2       | 356 point | 240             | 2.51        |
| 3       | 391 point | 396             | 3.34        |

In this 3rd iteration, the average value of each cluster is equal to the initial cluster value, so that on the 2nd and 3rd iterations this cluster position does not change anymore and no more data moves from one cluster to another. Then the iteration is stopped and the final result obtained by 3 clusters with 10 iterations as seen in table 3. It is known that the more potential drop out students are in cluster 2 due to the highest Quality and GPA of the lowest value compared to cluster 1 and cluster 3. On Table 3 can be seen that there are 162 students who joined in cluster 2 which means most potentially dropout. To see the accuracy of clustering process by using K-Means clustering method then calculated the square error value (SE) of each data in cluster 2. The value of square error is calculated by squaring the difference of the quality score or GPA of each student with the value of centroid cluster 2. The value of sum square error (SSE) obtained is 952972 for the quality variable and 232.48 for the GPA which is the sum of all square error values. This SSE value will be compared with the SSE value of the groupings obtained by using K-Means Clustering K-Means Clustering method which determines the initial centroid in KD-Tree.

| Centroid | Quality | GPA  | Member |
|----------|---------|------|--------|
| C1       | 432     | 3.65 | 237 point |
| C2       | 83      | 1.76 | 162 point |
| C3       | 388     | 3.28 | 601 point |

2.2 K-Means Clustering with Early Centroid Determination by KD-Tree

Student's GPA and Quality Data will be clustered using K-Means KD-Tree Clustering to predict students with the potential to drop out. The initial determination of the KD-Tree cluster center is done
by dividing the data into 32 leaf bucket (Lj) by limiting each leaf bucket to a maximum of 35 data. Further calculated the middle point value of each leaf bucket (Mj) and its density (Pj). The values of M1 and M2 are respectively the average values of data incorporated in L1 and L2. Next is calculated the range of each leaf bucket (Lj) that is by taking the value of data range (Vj) is significant (in this case is taken the value of Quality range, the value of the GPA range can be ignored because it is small enough than the value of Quality) is the highest data difference with the data Lowest on the leaf bucket. For data at L1 has the highest data (Quality value) 8 and the lowest data 0, so that obtained value V1 = 8 - 0 = 8. For data at L2 has the highest data (Quality value) 55 and the lowest data 10, so obtained value V2 = 55 - 10 = 45. Next we calculate the density value (Pj) of each leaf bucket with equation (3):

\[ P_j = \frac{N_j}{V_j} \] .................................(3)

The highest density (Pj) leaf bucket (Lj) density obtained is L20 with value P20 = 15,00 and is directly selected to be the first initial C1 centroids. Furthermore, throw 20% leaf bucket which has the lowest density value to avoid the election of outlier data becomes the middle point candidate. 20% of 32 Leaf bucket is 6. The next cluster is determined by calculating the rank of leaf bucket (Gj) which is the multiplication of distance (dj) between the middle point candidate with centroid C2 with the density value (Pj) as equation (4):

\[ G_j = \left\{ \min_{k = 0 \ldots r} \left[ d(C_k, M_j) \right] \right\} \times P_j \] ..............................................(4)

Calculation of Gj value for leaf bucket L1, L3 and L7 can be shown as follows:

\[ G_1 = d(C_1, M_1) \times P_1 = \sqrt{(1.4 - 401)^2 + (0.01 - 3.40)^2} \times 3.75 = 400 \times 3.75 = 1499.30 \]
\[ G_3 = d(C_1, M_3) \times P_3 = \sqrt{(55 - 401)^2 + (1.02 - 3.40)^2} \times 0.70 = 346 \times 0.70 = 241.40 \]
\[ G_7 = d(C_1, M_7) \times P_7 = \sqrt{(367 - 401)^2 + (3.11 - 3.40)^2} \times 3.33 = 34 \times 3.33 = 114.45 \]

Leaf bucket (Lj) which has the highest Gj is L1 with the value G1 = 1499.30 and immediately become the second initial centroid C2 with the value of Quality 1.4 and the value of GPA 0.01. Furthermore, throw 20% leaf bucket which has the lowest density value to avoid the election of outlier data becomes the middle point candidate. 20% of 25 leaf bucket is 5. The next cluster is determined by calculating the rank of leaf bucket (Gj) which is the multiplication of distance between candidate middle point with centroid C2 (dj) with the density value (Pj) as equation (3). Obtained leaf bucket which has the highest Gj leaf bucket L26 and directly elected to the third initial centroid C3 with the value of Quality 425 and the value of GPA 3.59. Then obtained the initial 3 centroid C1, C2, and C3 as table 4. Next, the initial centroid processed with K-Means Clustering. In the 8th and 9th iterations the cluster position does not change anymore and no more data moves from one cluster to another, the iteration is stopped and the end result is 3 clusters with 9 iterations.

| Centroid | Quality | GPA  |
|----------|---------|------|
| C1       | 401     | 3.40 |
| C2       | 1.40    | 0.01 |
| C3       | 425     | 3.59 |
Table 5. End Cluster Center KD-Tree K-Means Clustering

| Centroid | Quality | GPA | Member |
|----------|---------|-----|--------|
| C1       | 377     | 3,21 | 444    |
| C2       | 68      | 1,67 | 149    |
| C3       | 421     | 3,56 | 407    |

Table 6. SSE Value Method K-Means Clustering and KD-Tree K-Means Clustering

| Method            | SSE  | Member |
|-------------------|------|--------|
|                   | Quality | GPA     | Member |
| K Means           | 952972 | 232,48  | 162    |
| KD Tree K Means   | 504302 | 214,37  | 149    |

It can be seen that more potential drop out students are in cluster 2 due to the highest Quality and GPA of the lowest score compared to cluster 1 and cluster 3. Table 5 shows that there are 149 members in cluster 2. The value of square error obtained by calculating the difference in the value of the data with the average of all data in the cluster group 2. The sum of square error (SSE) obtained is 504302,2 for the Quality variable and 214,37 for the GPA which is the sum of all square error values. Students with the most potential to drop out are in clusters that have the highest Quality and GPA of the lowest value compared to other clusters. In table 6 can see the results of grouping with two methods that have been tested and then compared the value of sum square error (sse) between K-Means Clustering results with KD-Tree K-Means Clustering. The method with the smallest sse value is the best.

3 Conclusion

Value of sum square error (SSE) obtained from K-Means Clustering method with initial centroid determination at random is 952972 for Quality variable and 232,48 for GPA. Value of sum square error (SSE) obtained from KD-Tree K-Means Clustering Method with initial centroid determination by KD-Tree is 504302 for Quality and 214,37 for IPK. It can be seen that the SSE value of the K-Means Clustering KD-Tree Method is smaller than the K-Means Method which selects the centroid initially at random. This suggests that early KD-Tree centroid selection in the K-Means method provides better accuracy than the K-Means Clustering method that selects the initially random centroid.

4 References

[1] Oyelade, O. J., Oladipupo, O. O. & Obagbuwa, I. C. 2010. Application Of K-Means Clustering Algorithm For Prediction Of Students Academic Performance. *International Journal of Computer Science and Information Security*. Vol. 7, No. 1.

[2] Godara, S. & Yadav, R. 2013. Performance Analysis Of Clustering Algorithms For Character Recognition Using Weka Tool. *International Journal of Advanced Computer and Mathematical Sciences*. Vol 4, pp 119-123.

[3] Chaturvedi, S. K., Richariya, V. & Tiwari, N. 2012. Anomaly Detection in Network using Data mining Techniques. *International Journal of Emerging Technology and Advanced Engineering*. Vol. 2, No. 5.

[4] Dash, R., Mishra, D., Rath, A. K. & Acharya, M. 2010. A Hybridized K-Means Clustering Approach For High Dimensional Dataset. *International Journal of Engineering, Science and Technology*. Vol. 2, No. 2, pp. 59-66.

[5] Bhatia, M. P. S. & Khurana, D. 2013. Experimental study of Data clustering using k Means and modified algorithms.*International Journal of Data Mining & Knowledge Management Process*. Vol.3, No. 3.
[6] Jain, A. K. 2009. Data Clustering: 50 Years Beyond K-Means. *Pattern Recognition Letters*. Vol 31, pp 651–666.

[7] Sahu, S. K. & Jena, S. K. 2014. A Study of K-Means and C-Means Clustering Algorithms for Intrusion Detection Product Development. *International Journal of Innovation, Management and Technology*. Vol. 5, No. 3.

[8] Likas, A., Vlassis, N. & Verbeek J. J. 2003. The Global K-Means Clustering Algorithm. *The Journal of the Pattern Recognition Society*. Vol 36, pp 451 – 461.