Improvement of K-Means Performance Using a Combination of Principal Component Analysis and Rapid Centroid Estimation

Sapriadi1*, Sutarman2, E B Nababan1

1Department of Information Technology, Faculty of Computer Science and Information Technology Universitas Sumatera Utara, Medan, Indonesia
2Department of Mathematics, Faculty of mathematics and science Universitas Sumatera Utara, Medan - Indonesia, Indonesia

E-mail: sapriadi92@gmail.com*

Abstract. K-Means is a simple clustering algorithm, this method starts with randomizing partitions and continuing to reassign samples to clusters based on similarities between data. However, the K-Means method has several disadvantages, including determining the initial cluster center value done randomly, and the distance model used in determining the similarity between data where conventional distance models have the same effect on each data attribute. In this study will try to improve the performance of K-Means by using a combination of Principal Component Analysis (PCA) and Rapid Centroid Estimation (RCE). PCA will determine the weight of each attribute data based on eigen value, and RCE is used to determine the beginning of the cluster center. To see the performance of the proposed method, this research will use 3 datasets obtained from the UCI Repository, including ionosphere, iris, and wine. Analysis of the performance of the proposed method is only measured based on MSE and SSE. The results of this study indicate that the PCA and RCE methods were able to improve the performance of K-Means, the highest performance improvement based on MSE was found in iris data, which amounted to 56.76%, while based on SSE occurred in the ionosphere data which was 86.08%.

1. Introduction
Clustering is a way or method that aims to group data or objects based on information found in data or objects and the relationship between them. [1] Explains that clustering is a process for grouping data based on similarity, where finally data with a high level of similarity are placed in the same group, while the data that is far different is placed in the other groups. [2] Explains clustering is a method of grouping to similar data and grouping data that is not the same in other groups. There are several methods that can be used, among others, K-Medoids, K-Means, Fuzzy C-Means, DBSCAN, etc. [3].

K-Means is an algorithm that uses square-error criteria, this starts with a randomization of partitions and continues to reassign samples to clusters based on similarities between samples and clusters, until the convergence criteria are met. Usually, this criterion is fulfilled when no sample moves from one group to another which will cause a decrease in error square.

However, the K-Means method has several disadvantages, including determining the initial value randomly, where the initial determination of centroid randomly is a problem [4,5]. Because the value of the initial centroid randomly selected has a bad sensitivity and does not guarantee the results of good data grouping [6-7].
The initial determination of the cluster center is very influential on the results of the clustering process [7]. This causes the emergence of various methods in determining the initial cluster center, as was done [9] which uses the neighborhood model algorithm. [4] determines the initial cluster center by means of biogeography based optimization. [10] used Improved Downhill Simplex Search to optimize the value of centroids. Whereas [11,12] uses genetic algorithms. However, these methods make computing more expensive [13]. To overcome the above problem [13] proposed a solution in his research called Rapid Centroid Estimation (RCE).

Another problem that is a weakness of K-Means lies in the distance model used in determining the similarity between data, where traditional distance models such as Euclidean, Manhattan or Minkowsky are used to give the same effect on each attribute of data, this of course can reduce performance in the -cluster.

One of the solutions to the problems above can be overcome by giving weight to each attribute [14], while the methods that can be used include Gain Ratio [15], but the use of Gain Ratio can only be on supervised data. One solution to give weight to the attribute whose data is unsupervised is to use the Principal Component Analysis (PCA) method, PCA can provide good results in determining the interrelationships between attributes [16].

Weaknesses in the clustering method can affect performance in grouping data. Performance can be interpreted as the level of achievement of results or "the degree of accomplishment". Therefore based on previous studies on this time the researchers will try to improve the performance of the K-Means method by using RCE and attribute weighting using PCA, it is hoped that this will be able to overcome the weaknesses in K-Means and produce increased performance in clustering data used.

The rest of this paper is structured as follows. Section 2 will summarize previous studies on the theoretical foundation regarding the topic. In Section 3 we will provide the result and discussions and in Section 4 will provide with conclusions.

2. K-Means

In general, grouping data with the K-Means method can be done with algorithms, including: (1) determining the number of groups, (2) allocating data into groups randomly, (3) calculating the center of the group from the data in each group, (4) allocate each data to the nearest average, (5) back to step (3) if there is still data that moves groups, or if there is a change in the value of the centroid above the specified threshold value, or if the value changes in the function the objective used is still above the specified threshold value. The centroid location of each group taken from the mean of all data values for each feature must be recalculated. If M states the amount of data in a group, i declares the ii feature in a group, and p denotes the data dimension [17].

To calculate the centroid feature i used the formula:

\[ C_i = \frac{1}{M} \sum_{j=1}^{M} X_{ij} \]  

(1)

The formula is done as many as p dimensions so i starts from 1 to p. Measuring distance in Euclidean spacing using a formula:

\[ D(x_2, x_1) = \|x_2 - x_1\|_2 = \sqrt{\sum_{j=1}^{p} |x_{2j} - x_{1j}|^2} \]  

(2)

D is the distance between data x2 and x1, and | . | is absolute value. Measuring the distance in the Manhattan space uses a formula:

\[ D(x_2, x_1) = \|x_2 - x_1\|_1 = \sum_{j=1}^{p} |x_{2j} - x_{1j}| \]  

(3)

Measuring distance in the Minkowsky spacing space using the formula:

\[ D(x_2, x_1) = \|x_2 - x_1\|_\lambda = \lambda^{\frac{1}{\lambda}} \sum_{j=1}^{p} |x_{2j} - x_{1j}|^\lambda \]  

(4)

This method partition the data into clusters / groups so that data that has the same characteristics are grouped into one and the same cluster of data that has different characteristics grouped into other groups. Follow these instructions as carefully as possible so all articles within a conference have the same style to the title page. This paragraph follows a section title so it should not be indented.
3. Rapid Centroid Estimation (RCE)

[13] Proposed a method in his research called RCE. RCE is generally similar to the concept of decision making. [18] RCE is based on the Particle Swarm Clustering (PSC) algorithm, but is reconfigured to reduce computing complexity [19].

RCE is able to achieve performance with higher stability and faster optimization speed than the PSC [16]. The pseudocode of the RCE algorithm is as follows [20]:

**Algorithm**

\[ S = \text{RCE (dataset, total\_cluster)} \]

1. **Initialize** \( n \) particle, random \( x \)
2. **Calculate distance** \( p, g \) for each particle per datum
3. **While** \( t < \text{max\_iteration} \)
   1. **Updated the matrix** distance \( D(t) = d(y_i, x_i) : \forall i,j \) \hspace{1cm} (5)
   2. **Find the closest data point** of each particle \( Dx^{\text{min}}(t)hx^{\text{min}} = \text{min}(D.i) \) \hspace{1cm} (6)
   3. **Find each closest particle** for each data/ \hspace{1cm} (7)
   4. **Determine the winning particle** (Particle closest to the input pattern using the Euclidean distance formula) \hspace{1cm} \[ x_{\text{most\_win}}(t) = x(t) \in \text{min}\left(d(p_i(t) - x_i(t))\right) : \forall i \] \hspace{1cm} (8)
   5. **For each particle**
      1. **Determine the element** of each member particle \( i \) (centroid) \hspace{1cm} \[ y_i^{\text{cluster}} = \forall y \in x_i(t) \]
      2. **Update position** using step (g) if \( N_i > 0 \), if not directly towards the winning particle coordinates.
\[ x_i(t + 1) = \begin{cases} 
    x_i(t) + \Delta x_i(t + 1) & \text{if } N_i > 0 \\
    x_i(t) + \varphi_x\otimes(x_{\text{most\_win}}(t) - x_i(t)) & \text{if not}
\end{cases} \] \hspace{1cm} (11)

**End For**

**End While**

4. Principal Component Analysis (PCA)

PCA is a combination of linear variables which are geometrically combined linearly, this is a new coordinate system that is found from the initial variable rotation process. The PCA method is very useful to use if existing data has a large number of variables, PCA calculations are based on eigenvalues and eigenvectors. The purpose of PCA is to find out the correlated variables without having to lose information from the data.

According to [21], Principal Component Analysis workmanship procedures are as follows: (a) Calculating covariance from observation data using equations:

\[ \text{Cov}(x, y) = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \mu_x)(y_i - \mu_y) \] \hspace{1cm} (13)

With \( \mu_x \) and \( \mu_y \) it is the sample mean of the variables \( x \) and \( y \), where \( x_i \) and \( y_i \) are the values of the \( i \)-observation of the variables \( x \) and \( y \). From the data whose value is used, obtained covariance of size \( n \times n \). (b) Look for the eigenvalue of the covariance matrix with the equation [22]:

\[ \text{Determinant}(A-\lambda I) = 0 \] \hspace{1cm} (14)

\( A \) : matrix \( n \times n \), \( \lambda \) : eigen value, \( I \) : identity matrix. (c) Determine the Principal Component proportion value with the equation:

\[ \text{Principal Component Proportion}\% = \frac{\text{Eigen Value}}{\text{Kovarian Variant}} \times 100\% \] \hspace{1cm} (15)

(d) Calculate attribute weights based on eigenvector with equation [20]:

\[ Ax = \lambda x \] \hspace{1cm} (16)
So that a linear combination is obtained, i.e. (a) \( \lambda_1, \lambda_2, \lambda_3 \ldots \lambda_n \) is eigenvalue matrix A (b) \( x_1, x_2, x_3 \ldots x_n \) is eigenvector according to the eigenvalue (\( \lambda_n \)). The eigenvalue & eigenvector equation is Eigenvalue Decomposition (EVD), with the following equation:

\[
AX = XD
\]  
(17)

Where A: Matrix n x n which has n eigenvalue (\( \lambda_n \)), D: Eigenvalue of the eigenvector, X: Eigenvector of matrix A, X\(^{-1}\): Invers of eigenvector X.

5. The Proposed Method

This study seeks to improve the performance of conventional K-Means by using the Rapid Centroid Estimation (RCE) method to determine the value of the initial cluster center and Principal Component Analysis (PCA) to weight members on each data attribute so that less relevant attributes can be reduced to the clustering process. so as to improve the performance of K-Means [23]. To be more clear in describing the process in this study, we will explain step by step in this sub-chapter. The steps in outline can be seen in Figure 1:

![Figure 1. Detail stages of the proposed method.](image)

From Figure 1 it can be explained that the proposed method has several stages, among others:

Step 1: Determination of attribute weights using PCA based on eigenvalues.

The weight is calculated using the normalization equation min-max [24], where the lowest weight after normalized is 0 and the highest weight after normalized is 1. The equation used is as follows:

\[
W_i = \frac{(E_i - \text{Min}(E))}{\text{Max}(E) - \text{Min}(E)}
\]  
(18)

Where: \( W_i \) = Weight of the \( i \)- Attribute, \( E_i \) = The Eigen Value \( i \), \( \text{Min}(E) \) = The Lowest Value of The Eigen Value, \( \text{Max}(E) \) = The Highest Value of The Eigen Value.

Step 2: Determine the Cluster Center Using RCE.

Step 3: Do the Clustering Process by Using K-Means.

In this study, to see whether the proposed method is able to provide better performance than conventional methods, an analysis of the MSE and SSE results obtained from the two methods will be conducted.

6. Result and Discussion

This research uses data sets obtained from UCI Machine Learning Repository such as ionosphere, iris, wine, glass, haberman, and hayes-roth. The details of the data used can be seen in table 1:

| Data       | Attributes | Type       | Class | Total Data |
|------------|------------|------------|-------|------------|
| Ionosphere | 34         | Real       | 2     | 351        |
| Iris       | 10         | Real       | 3     | 150        |
| Wine       | 13         | Integer, Real | 3     | 178        |
| Glass      | 9          | Real       | 6     | 214        |
| Haberman   | 3          | Real       | 2     | 306        |
This study is to do the clustering process using the K-Means method with attributes that have been weighted using the PCA method, and the initial cluster center values found from the RCE method, while the performance comparison against the conventional method of the three data used can be seen in table 2, 3 and 4.

**Table 2.** Performance for Ionosphere Data.

| Iteration | Conventional K-Means | The Proposed Method |
|-----------|----------------------|---------------------|
|           | MSE      | SSE      | Total Iteration | MSE      | SSE      | Total Iteration |
| Average   | 38.750   | 4.94772  | 6               | 34.685   | 0.68891  | 2               |

**Table 3.** Performance for Iris Data.

| Iteration | Conventional K-Means | The Proposed Method |
|-----------|----------------------|---------------------|
|           | MSE      | SSE      | Total Iteration | MSE      | SSE      | Total Iteration |
| Average   | 1.346    | 2.09074  | 6               | 0.582    | 1.06814  | 3               |

**Table 4.** Performance for Wine Data.

| Iteration | Conventional K-Means | The Proposed Method |
|-----------|----------------------|---------------------|
|           | MSE      | SSE      | Total Iteration | MSE      | SSE      | Total Iteration |
| Average   | 114.902  | 313.13479 | 5               | 74.298   | 293.488  | 2               |

**Table 5.** Performance for Glass Data.

| Iteration | Conventional K-Means | The Proposed Method |
|-----------|----------------------|---------------------|
|           | MSE      | SSE      | Total Iteration | MSE      | SSE      | Total Iteration |
| Average   | 118.06   | 9.32     | 14              | 89.024   | 1.1142   | 14              |

**Table 6.** Performance for Haberman.

| Iteration | Conventional K-Means | The Proposed Method |
|-----------|----------------------|---------------------|
|           | MSE      | SSE      | Total Iteration | MSE      | SSE      | Total Iteration |
| Average   | 91.18    | 18.75    | 14              | 74.51    | 4.478    | 2               |

**Table 7.** Performance for Hayes-roth.

| Iteration | Conventional K-Means | The Proposed Method |
|-----------|----------------------|---------------------|
|           | MSE      | SSE      | Total Iteration | MSE      | SSE      | Total Iteration |
| Average   | 73.80    | 4.45     | 8               | 56.41    | 2.62     | 2               |

To clearly see the performance generated from each method for all data used in this study can be seen in Figure 2.
From figure 2 and table 2, 3, and table 4, it can be seen that the proposed method is able to provide better performance than conventional K-Means. Where the highest increase in performance based on MSE on conventional K-Means was found in the iris data, which amounted to 56.76% and the highest performance increase based on the highest SSE decline occurred in the ionosphere data which was 86.08%. The lowest increase in permeability between the methods proposed before conventional K-Means was 10.5% for MSE found in ionosphere data and 6.27% for SSE found in wine data.

Based on the tests that have been conducted on all data sets used, it can be seen that the proposed method is able to provide better performance than the parameters produced by conventional K-Means.

7. Conclusion

Based on the previous sub-chapter, it is proven that the principal component analysis and rapid centroid estimation can improve performance on conventional K-Means methods, where the highest performance increase towards conventional K-Means is obtained in the iris data set of 56.76% based on MSE, and based on SSE the highest performance increase occurs in the ionosphere data which is equal to 86.08%.

While the lowest increase in performance against conventional K-Means was obtained from the ionosphere dataset, which was 10.49% based on MSE, and based on SSE the lowest performance increase occurred in wine data which was 6.27%.

Based on the tests conducted in the previous chapter, it can be concluded that the principal component analysis and rapid centroid estimation can improve the performance of conventional K-Means on all data used. It is also recommended that as a general principle, for large tables font sizes can be reduced to make the table fit on a page or fit to the width of the text.

References

[1] Han, J., Camber, M., & Pei, J. (2012). Data Mining Concepts and Techniques (3rd ed.). USA: Elsevier.
[2] Bramer, M. 2007. Principles of Data Mining. Springer-Verlag : London.
[3] de Oliveira, J.V & Pedrycz, W (Editor). 2007. Advance in Fuzzy Clustering and It’s Applications. The Atrium, Southern Gate, Chichester. British Library Cataloguing in Publication Data. Jhon Willey and Son, Ltd. England
[4] Kumar, V., Chhabra, J.K. & Kumar, D. 2011. Initializing Cluster Center for K-Means Using Biogeography Based Optimization. Springer-Verlag : Berlin.
[5] A. Aputra., M. Zarlis, & E. B. Nababan. 2017. Performance Analysis of Combined Methods of Genetic Algorithm and K-Means Clustering in Determining The Value of Centroid. International Conference on Information and Communication Technology (IconICT). pp1-6.

[6] Xiong, C., Hua, Z., Lv. Ke. & Li, X. 2016. An Improved K-means text clustering algorithm By Optimizing initial cluster centers. International Conference on Cloud Computing and Big Data : 265 - 268. DOI 10.1109/CCBD.2016.059.

[7] Gorunescu, F. 2011. Data Mining Concept, Model and Techniques. Springer-Verlag: Berlin.

[8] AAhmadyfard. & H Modares. 2008. Combining PSO and K-Means to Enhance Data Clustering. International Symposium on Telecommunication. 688-691.

[9] F Cao, J Liang, & G Jiang. 2009. An Initializing method for the K-Means algorithm using neighborhood model. Computer and Mathematics with Applications58 : 474-483.

[10] E Saboori. S Parsazad. & ASadeghi. 2010. Improving the K-Means Algorithm using Improved Downhill Simplex Search. International Conference on Software Technology and Engineering. V2350-V2354.

[11] W Min. & Y Siqing. 2010. Improved K-Means Clustering Based on Genetic Algorithm. International Conference on Computer Application and System Modeling. V6632-V6639.

[12] B K Khotim., F Irmhami, & T Sundarwati. 2016. Journal of Theoretical and Applied Information Technology 90 (1) : 23-30.

[13] Yuwono, M., W. Su, S. Moulton, B. & Nguyen, H. 2012a. Fast Unsupervised Learning Method For Rapid Estimation Of Cluster Centroids. IEEE Congress on Evolutionary Computation. 1-8.

[14] Kuhkan, M. 2016. A Method to Improve the Accuracy of K - Nearest Neighbor Algorithm. International Journal of Computer Engineering and Information Technology, pp. 90-95.

[15] Nababan A. A., Sitompul O. S, &Tulus. 2018. Attribute Weighting Based K-Nearest Neighbor Using Gain Ratio. MEcNI. 1-6.

[16] Nasution M. Z. N., Sitompul O. S, &Ramli, M. 2018. Pca Based Feature Reduction To Improve The Accuracy Of Decision Tree C4.5 Classification. 2nd International Conference on Computing and Applied Informatics 2017, 1-6.

[17] Prasetyo, E. (2014). Data Mining-Mengolah Data MenjadiInformasiMenggunakanMATLAB . Yogyakarta: ANDI.

[18] Yuwono, M., W. Su, S. Moulton, B. & Nguyen, H. 2012b. Method for increasing the computation speed of an unsupervised learning approach for data clustering. IEEE World Congress on Computational Intelligence. 1-8.

[19] Yuwono, M., W. Su, S. Moulton, B. & Nguyen, H. 2014. Data Clustering Using Variants of Rapid Centroid Estimation. IEEE Transactions on Evolutionary Computation. Pp : 366-377.

[20] Yuwono, M., W. Su, S. Moulton, B. & Nguyen, H. 2012c. Optimization Strategies for Rapid Centroid Estimation. Annual International Conference of the IEEE, Engineering in Medicine and Biology Society (EMBC).pp : 6212-6215.

[21] Jolliffe, I.T. 2002. Principal Component Analysis. 2nd Edition. Springer-Verlag: New York.

[22] Johnson, W.A. &Wichern, D.W. 2007. Applied Multivariate Statistical Analysis. 6th Edition. Pearson Prentice Hall: New Jersey

[23] K U Syaliman. O S Sitompul. & E B Nababan. 2018. Improving the accuracy of k-nearest neighbor using local mean based and distance weight. 2nd International Conference on Computing and Applied Informatics 2017, 1-6.

[24] Saranya, C&Manikandan, G. 2013. A Study on Normalization Techniques for Privacy Preserving Data Mining. International Journal of Engineering and Technology (IJET), pp. 2701-2704.