Journal of Computer & Robotics 14 (2), Summer and Autumn 2021, 21-27

Computer & Robotics

Selecting Optimal k in the k-Means Clustering Algorithm

Mojtaba Jahanian[a] , Abbas Karimi [a,*], Faraneh Zarafshan[b]

[a] *Department of Computer Engineering, Faculty of Engineering, Arak Branch, Islamic Azad University, Arak Markazi, IRAN*

[b] *Department of Computer Engineering, Faculty of Engineering, Ashtian Branch, Islamic Azad University, Arak markazi, Iran*

Received 02 January 2022, Accepted 16 January 2022

Abstract

Clustering is one of the essential machine learning algorithms. Data is not labeled in clustering. The most fundamental challenge in clustering algorithms is to choose the correct number of clusters at the beginning of the algorithm. The proper performance of the clustering algorithm depends on selecting the appropriate number of clusters and selecting the optimal right centers. The quality and an optimal number of clusters are essential in algorithm analysis. This article has tried to distinguish our work from other writings by carefully analyzing and comparing existing algorithms and a clear and accurate understanding of all aspects. Also, by comparing other methods using three criteria, the minimum internal distance between points of a cluster and the maximum external distance between clusters and the location of a cluster, we have presented an intelligent method for selecting the optimal number of clusters. In this method, clusters with the lowest error and the lowest internal variance are chosen based on the results obtained from the research.

Keywords: *Clustering Algorithms, K-means, Clustering, the optimal number of clusters.*

1.Introduction

Today, a lot of data is generated daily. Researchers use this vast amount of data to store and manipulate data to extract future ideas. This process of extracting information and patterns from data is called data mining. Data mining consists of three steps: creating, exploring, and classifying patterns. Clustering is one of the data mining methods. Clustering is an unsupervised learning method. Data is not tagged in this way. In terms of data volume, clustering algorithms can be divided into two categories: clustering algorithms for data mining and clustering algorithms for big data. And in general, clustering can be divided into soft and hard clustering. Soft clustering: In this technique, the probability of an observation splitting into a cluster is calculated. A statement is precisely divided into a cluster (no probability is calculated). Some clustering algorithms, such as Connectivity-based Clustering (Hierarchical clustering) [1], Centroids-based Clustering (Partitioning methods) [2] , Distribution-user in advance. The wrong choice can have a direct based Clustering [3], Density-based Clustering (Model-based methods) [4], Fuzzy Clustering [5], Constraint-based (Supervised Clustering) [6], etc. The organization of this paper is as follows. In Section 2, we first construct the learning schema and then propose the Without supervision, k- means clustering. In Section 3, we first review the work done in this area and then explain our proposed algorithm (SONSC). Section 4 describes the results and experiments performed with the proposed methods. Finally, the conclusion is expressed in Section 5.

2.The Global K-Means Clustering Algortithm

The Clustering k-means algorithm is one of the most popular unsupervised learning algorithms. We look at data without a label. Based on the similarity criterion, we try to put similar data in a cluster. In this algorithm, we only have the independent variable. We do not have a dependent/target variable. The algorithm k-means depends on the value of k. and must be specified by the impact on the quality of clusters. Unsuitable selection

* Corresponding Author Email:Akarimi@iau- Arak.ac.ir

of k value can cause the algorithm to get caught in a local minimum [7]. In Section 3, we will talk about K selection methods. In the K-means clustering algorithm, we assume that a data set is given to us in an environment with dimension d.

$$X = \{ x_1, \dots, x_N \} , x_n \in R^d \quad (1)$$

The M-clustering problem aims at partitioning this data set into M disjoint subsets (clusters).Cluster 1 to M

$$C_1, \dots, C_M \quad (2)$$

Now we need to look for a solution that can put the same data in a cluster, the clustering of which uses the distance criterion. First, the cluster centers are determined randomly, and then we calculate the distance of all data points to all cluster centers with the Euclidean distance criterion, and each data point is assigned to the nearest cluster center.

Category	Typical algorithm
Hierarchical Clustering	BIRCH, CURE, ROCK, Chameleon Agglomerative clustering (AGNES) Divisive DIANA (Devise Analysis) Single Linkage, Complete Linkage Average Linkage, Centroid-linkage
Partitioning Methods	K- means , K- medics PAM,CLARA,CLARANS
Density-Based Clustering	DBSCAN, OPTICS ,Mean-shift
Model-based Clustering	COBWEB, GMM, SOM, ART
Fuzzy Clustering	FCM, FCS, MM
Clustering algorithm based on grid	STING, CLIQUE
Clustering algorithm based on distribution	DBCLASD, GMM
Clustering algorithm based on graph theory	CLICK, MST
Clustering algorithm based on fractal theory	FC

$$d(p,q) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2} \quad (3)$$

$p, q = two\ points\ in\ Eucildean\ with\ n\ space.$
$p_i, q_i = Euclidean\ vectors\ , starting\ from$
$the\ origin\ Iinitial\ point).$
$n = n\ space$

This is how the algorithm works:

Step 1: Select the number of clusters based on the algorithm SONSC.

Step 2: Initialize centroids by shuffling the dataset and randomly selecting K. data points for the centroids without replacement.

$$SSE = \sum_{j=1}^{k} \sum_{i=1}^{n} ||x_i^{(j)} - c_j||^2 \quad (4)$$

Where: SSE is the objective function
k is the number of clusters (SONSC)
n is the number of data points
xi is data point ith
CJ is centroid for cluster j

Step 3: Compute the sum of the squared distance between data points and all centroids.

Step 4: Assign each data point to the closest cluster (centroid).

Step 5: Compute the centroids for the clusters by taking the average of all data points that belong to each cluster.

Step 6: Repeat steps 3 to 5 until there is no change to the centroids or the data point belonging to each cluster has not changed. The purpose of the clustering algorithm is to reduce the error function (reducing the variance between data points). Sum of Squared Error (SSE):

3. Select the Optimal Number

Defining the optimal number of clusters in a data set is a central problem in partitioning clustering, such as k-means clustering, which needs the user to determine the number of clusters k to be

generated. Unfortunately, there is no definitive answer to this problem. The optimal number of clusters is subjective and depends on the method used for measuring similarities and the parameters used for partitioning. Among the known methods for selecting the number of clusters, we can mention the Elbow Method. [7], the Average Silhouette Method [8] [9], the Gap Statistics Method [10], and the Hierarchical Clustering [11] [12] [13].

3.1.The Elbow Method Algorithm

In the elbow method, the criterion within-cluster sum of square (WSS) is used to select the number of clusters.

$$minimize(\sum_{k=1}^{k} W(C_k)) \quad (5)$$

Where C_K is K^{th}cluster, and $W(C_k)$ is the within-cluster variation. The total within-cluster sum of squares (WSS) estimates the compactness of the clustering, and we need it to be as small as potential. [14].

The pseudo-code of the algorithm is as follows in Table. 2:

Table 2
Algorithm 1: Elbow Method
Input: $\boldsymbol{X}$= read dataset
1. Define two arrays $\boldsymbol{Wss} = [\,], \boldsymbol{K} = [\,]$
2. for $\boldsymbol{i = 1\ to\ k}$ do
3. $\boldsymbol{Wss}=\sum_{i=1}^{k}\sum \boldsymbol{dis}\,(\boldsymbol{x}, \boldsymbol{c_i})^2$
4. return $\boldsymbol{k}, \boldsymbol{d}$

The value of K in the Elbow method is determined by drawing. But in most cases, this method is not detectable. The elbow method is sometimes vague.

3.2.The Average Silhouette Method

The validity of a cluster is measured statistically. This measurement criterion is speedy. In this way, we determine whether each object is in its cluster or not. If the average ghost width is high, it means good clustering. This algorithm uses two factors, a and b. Factor-a is the distance between a data point and other points in the same cluster, and factor-b is the average distance between a data point and other points in the cluster closest to the data point cluster. [15].

For each data point i, we first define:

$$S_i = \frac{(x_i - y_i)}{\max(x_i - y_i)} \quad S_i \in [-1,1]\ , -1 \le S_i \le +1 \quad (6)$$

x_i *is dissimilarity within cluster* a_i

$$x_i = \frac{1}{x_k - 1}\sum_{\substack{a_i \in c_k \\ a_j \in c_k}} dis(a_i, a_j)_{i \neq j} \quad (7)$$

$$y_i = average\ dis(a_i, a_j)\ where\ \ a_i \in c_k\ but\ a_j \notin c_k\ and\ K = 1,2,\dots,k$$

$$silhouette\ \ coefficients\ \ \ sc = \frac{\sum_{i=1}^{n} s_i}{m} \quad (8)$$

where m is total data point and more s means bettre cluster

The pseudo-code of the algorithm is as follows in Table 3.

Table 3
Algorithm 2: Silhouette Coefficient
1. m=input data set.
2. k=number of clusters.
3. Calculate The mean distance between a sample and all other points in the same class $\boldsymbol{x_i}$
4. Calculate The mean distance between a sample and all other points in the next nearest cluster $\boldsymbol{y_i}$
5. Calculate $\boldsymbol{s_i}$
6. Calculate Global silhouette score is defined as : $\boldsymbol{sc}$
7. The score $\boldsymbol{sc}$ is bounded between -1 for incorrect clustering and +1 for highly dense clustering. Scores around zero indicate overlapping clusters.

Silhouette Coefficient or silhouette score is a metric used to calculate the worth of a clustering method. Its value ranges from -1 to 1.

1: Means clusters are well separated from each other and distinguished.

0: Means clusters are indifferent, or we can tell that the distance between clusters is not meaningful.

-1: Means clusters are assigned incorrectly.

In fact, the difference between the Silhouette method coefficient and the sum of the squares of the SSE error is the sum of the squares of the difference between each observation and the mean of its group. It can be used as a measure of change in a cluster and the Silhouette method coefficient and the distance of a point with all points of the cluster inside. The distance between the moment and the nearest next cluster also shows us that we can consider the best cluster for that point. The silhouette value measures how similar an issue is to its cluster (cohesion) compared to other clusters (separation). [15]

3.3.The Gap Statistic

The gap statistic correlates the total within the intra-cluster variety for various values of k with their demanded values under the null reference distribution of the data. The reference dataset is generated using Monte Carlo simulations of the sampling process. The estimation of the optimal clusters will be the value that maximizes the gap statistic. This means that the clustering structure is far from the random uniform distribution of points. That is, for each variable (x_i) in the data set, we calculate its range between$(\text{minimum}(x_i), maximum(x_j))$ and create values for the n points consistently from the period min to max. As the detected data and the source data, the total intra cluster variation is computed using different values of *k*. The *gap statistic* for a given *k* is defined as follows: [16] [17].

$$gap_n(k) = E_n^* log(W_k) - log(W_k) \quad (9)$$

Where E_n^* means the expectation below a sample size *n* from the source distribution. E_n^* Is determined via bootstrapping (B) by creating B copies of the reference datasets and, by computing the average $\log(W_k^*)$. The gap statistic measures the deviation of the observed $\hat{k}$ value from its supposed value under the null hypothesis. The estimation of the optimal clusters will be the value that maximizes$gap_n(k)$. This means that the clustering structure is far away from the normal distribution of points. The pseudo-code of the algorithm is as follows in Table 4 .

Table 4
Algorithm 3: The Gap Statistic

1- B=read dataset $(\boldsymbol{x1}, \boldsymbol{x2}, \dots, \boldsymbol{xn})$.
2- K=$\boldsymbol{k1}, \boldsymbol{k2}, \dots, \boldsymbol{kn}$
3- Compute$\bar{\boldsymbol{w}} = (\frac{\boldsymbol{1}}{\boldsymbol{B}}) \sum_b \mathbf{log}(\boldsymbol{w}_{\boldsymbol{kb}}^*)$.
4- Compute the standard deviation $\boldsymbol{sd}(\boldsymbol{k}) = \sqrt{(\boldsymbol{1}/\boldsymbol{b}) \sum_b (\mathbf{log}(\boldsymbol{w}_{\boldsymbol{kb}}^*) - \bar{\boldsymbol{w}})^2}$.
5- And compute $\boldsymbol{S_k} = \boldsymbol{sd_k} \times \sqrt{\boldsymbol{1} + \boldsymbol{1}/\boldsymbol{b}}$.
6- Determine the number of clusters as the smallest k such that $\boldsymbol{gap}(\boldsymbol{k}) \geq \boldsymbol{gap}(\boldsymbol{k}+\boldsymbol{1}) - \boldsymbol{s_{k+1}}$

3.4.The Canopy

In this algorithm, we use two threshold limits, T1 and T2. And $T1 > T2$. That Threshold values can vary depending on the needs and type of data [18] [19] [20]

The pseudo-code of the algorithm is as follows in Table 5.

Table 5
Algorithm 4: The Canopy

1. Input x=read data set
2. Definition T1 ,T2 what is T1>T2
3. Definition two array del:[ ] , canopy:[]
4. Delete a point from the set, creating a new 'canopy'. including this point d=p ,p=x-p_i
5. If (d<T2) then
6. del=[d]
7. else
8. canopy=[d]
9. until ($\boldsymbol{x} = \emptyset$)
10. end

This algorithm adds a data point to the canopy each time it is run. If the data point distance is less tha the threshold, it remains in the canopy. Otherwise, it is deleted.

3.5.Proposed Algorithm (SONSC)

In the proposed algorithm, all points in a cluster are at level one (k = 1). We add one unit to K and go to level two; in level two, we have two centers of clusters (the number of clusters in each level is equal to the numerical level). Based on the Euclidean similarity criterion, we calculate the distance from each point to the center of the clusters and then attribute each point to the nearest center of the cluster. In this step, we calculate the new centers of the clusters based on the average of all points in that cluster. We repeat this to keep the cluster centers fixed. After establishing cluster centers, we maintain cluster information and calculate and maintain internal and external evaluation criteria for clusters. Then we go to the next step, select the number of cluster centers, perform the clustering, and calculate the evaluation criteria of the new clusters. At this point, the N (OLD_CLUSTER) and N-1 (NEW_CLUSTER) level evaluation cluster criteria. If the evaluation criteria of the new cluster are better than the old clusters, we delete the old cluster information and save the new cluster information. And we go to the next step; otherwise, we do not consider the clustering of the new level. Like the branch algorithm and

constraint, we return to the previous level, and our clustering criterion is the clusters of the previous level of the equation. (10,11, 12).

n=number of observations, p = number of variables,
q = number of clusters (10)
S_w = sum of the within-cluster distances for her cluster: $\sum_{k=1}^{q} \sum_{i,j \in C_k, i<j} d(x_i, x_j)$

S_b=sum of the between-cluster distances:$\sum_{k=1}^{q-1} \sum_{l=k+1}^{q} \sum_{i \in C_k, j \in C_l} d(x_i, x_j)$ (11)
L=number of her level $(l = 1,2, \dots ,9)$

$SONSC = \frac{\sum_{k=1}^{l}(S_w/S_b)}{l}$ index for her level (12)

The pseudo-code of the proposed algorithm is as follows in Table 6.

Table 6
Algorithm 5: SONSC

1. input<-read_csv("data point")
2. Arbitrarily determine k object from the Data Set as the initial cluster centers.
3. Repeat
4. Find the distances of all data points from the centers of the cluster based on the Euclidean distance. Eq. (3, 4) and assign each point to the nearest center of the cluster.
5. Update mean cluster: calculate the mean value of the objects for each cluster.
6. Until no change.
7. If k=9 go to 11.
8. Compute index SONSC for Using the surface on which it is located . Eq.(10,11,12)
9. If SONSC_NEW> SONSC_OLD then go to 10 else go to 11.
10. K++ and go to 2
11. end

4.Experiments

In this section, we test our proposed algorithm on three valid data and show the results of these experiments. The results obtained from implementing the proposed algorithm show that this algorithm works properly on unknown data. Also, this algorithm works properly on several fixed and unknown clusters. And it has an acceptable speed for big data, in Table 7.

The algorithm SONNC for data IRIS suggests 4 clusters, which provides better clustering than other methods. And for data MPG and MAM, The number of clusters is indicated by 2, which different algorithms have suggested the same number Table. (7) Fig.(2).

In this algorithm, with eight proposed clusters, the error squared with three indices INTERA_CLUSTER(and INTER_CLUSTER and SONSC is compared as shown in the diagram. Criterion has a lower average error than other indicators Fig.(6).

Table 7
Data sets with the optimal number of clusters

Data name	Optimal number of the cluster			
	Elbow	Silhouette	Gap	SONSC
Iris	3	2	2	4
Mpg	3	2	2	2
All.Mamm.is.Milk.1956	3	2	2	2

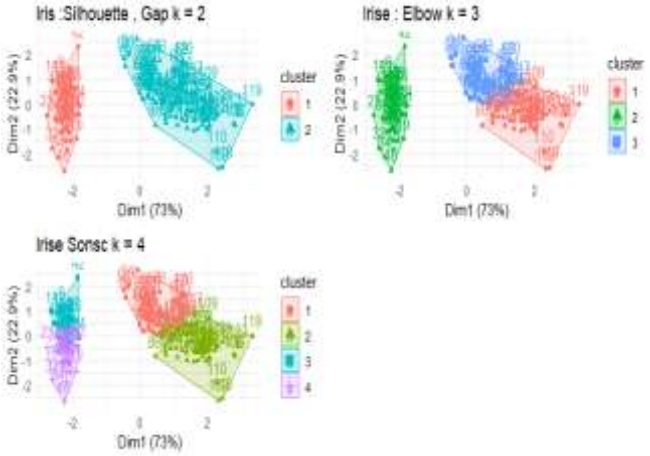

Fig. 1. Select number cluster

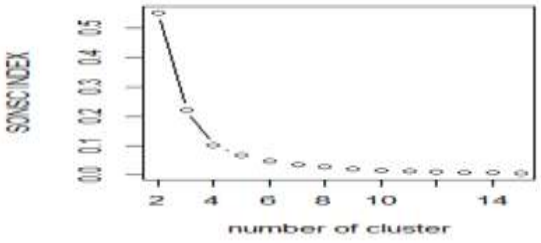

Fig.2. Iris data clustering with 150 samplesusing cluster number prediction with Silhouette Algorithms, Elbow, SONSC.

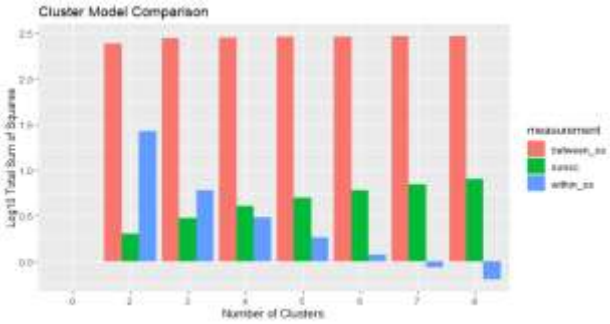

Fig. 3. Gap method k=2

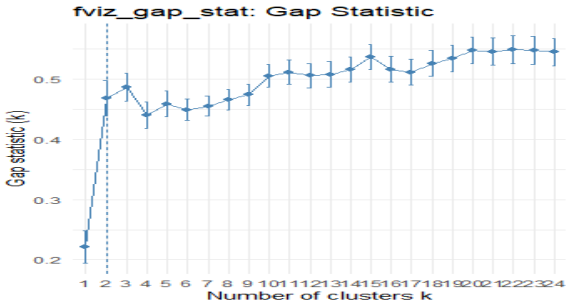

Fig.4.Comparison of SONSC criteria with measurement criteria within Intra–Inter Index for MPG DATA

4.Discussion and Conclusions

The SONSC algorithm provides an index used to predict the number of clusters. This algorithm responds well to small data and shows similar results to extensive data as other algorithms. In addition to the INTER_CLUSTER and INTERA_CLUSTER criteria, the cluster level index helps select the number of clusters more optimally. In this algorithm, we seek to find the minimum value of the SONSC index. This paper performed the research results using R-STDIO software on four valid Iris data, mpg. Mammals and milk. 1959 These results are obtained. Elbow clusters with 3 clusters, silhouette clusters with 2 clusters, and the SONSC method with 4 clusters show better performance and cluster data with a minor error. This proposed method uses the sum of squares of error and the criteria of the proximity of data within the cluster and cluster repetition. The proposed method in 8 selected clusters shows the internal variance of the clusters and the square function with more minor errors than other methods. Our proposal for future work is to use this method for big data, which, based on a combination of hierarchical algorithms, can have good results for big data clustering.

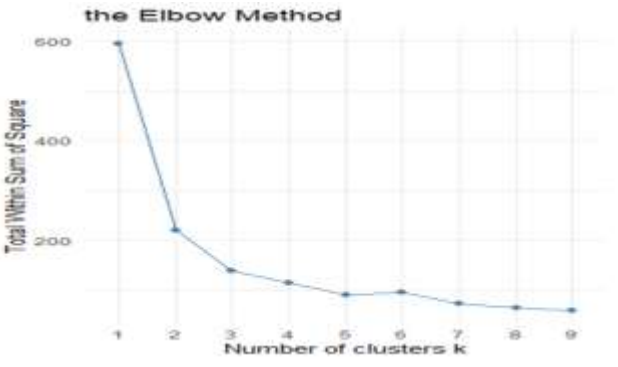

Fig.5. Silhouette method k=2

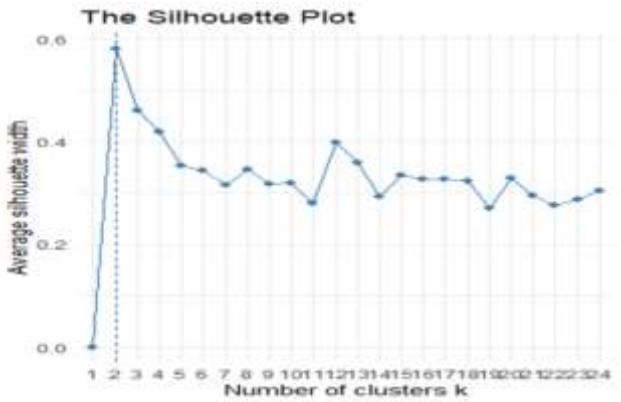

Fig. 6. Gap method k=2

References

[1] Johnson1967hierarchical, "Hierarchical clustering schemes," Psychometrika, pp. 241--254, 1967.

[2] S. K. Uppada, "Centroid based clustering algorithms—A clarion study," International Journal of Computer Science and Information Technologies, vol. 5, pp. 7309--7313, 2014.

[3] Z. a. Z. X. a. W. H.-S. a. Y. J. a. Z. J. a. H. G. Yu, "Distribution-based cluster structure selection," IEEE transactions on cybernetics, pp. 3554--3567, 2016.

[4] R. J. a. M. D. a. S. J. Campello, Density-based clustering based on hierarchical density estimates, Springer, 2013.

[5] D. E. a. K. W. C. Gustafson, Fuzzy clustering with a fuzzy covariance matrix, 1978 IEEE conference on decision and control including the 17th symposium on adaptive processes, 1979, pp. 761--766.

[6] M. a. B. S. a. M. R. J. Bilenko, "Integrating constraints and metric learning in semi-supervised clustering," in Proceedings of the twenty-first international conference on Machine learning, 2004, p. 11.

[7] A. Ng, "Clustering with the k-means algorithm," Machine Learning, 2012.

[8] K. S. a. V. D. L. M. J. Pollard, "A method to identify significant clusters in gene expression data," bepress, 2002.

[9] L. a. R. P. J. Kaufman, "Finding groups in data: an introduction to cluster analysis," John Wiley \& Sons, p. 344, 2009.

[10] Z. Z. X. B. C. e. a. Yin, "Using iterative cluster merging with improved gap statistics to perform online phenotype discovery in the context of high-throughput RNAi screens.," BMC Bioinformatics, p. 264, 2008.

[11] F. a. C. P. Murtagh, "Methods of hierarchical clustering," arXiv preprint arXiv:1105.0121, 2011.

[12] S. C. Johnson, "Hierarchical clustering schemes," Psychometrika, pp. 241--254, 1967.

[13] S. a. P. D. a. D. S. Chakraborty, "Hierarchical clustering with optimal transport," Statistics \& Probability Letters, p. 108781, 2020.

[14] C. a. Y. H. Yuan, "Research on K-value selection method of K-means clustering algorithm," J—Multidisciplinary Scientific Journal, pp. 226--235, 2019.

[15] T. a. K. N. a. D. P. a. K. K. a. K. N. Thinsungnoena, "The clustering validity with silhouette and sum of squared errors," learning, vol. 3, p. 7, 2015.

[16] R. a. W. G. a. H. T. Tibshirani, "Estimating the number of clusters in a data set via the gap statistic," Journal of the Royal Statistical Society: Series B (Statistical Methodology), pp. 411--423, 2001.

[17] Y. a. Y. J. Xiao, "Gap statistic and K-means algorithm," J. Comput. Res. Dev, pp. 176--180, 2007.

[18] N. S. a. Y. S. A. Sagheer, "Canopy with k-means clustering algorithm for big data analytics," AIP Conference Proceedings, p. 070006, 2021.

[19] C. a. Z. R. Yu, "Research of FCM algorithm based on canopy clustering algorithm under cloud environment," Computer Science, pp. 316--319, 2014.

[20] A.a. N. K. a. U. L. H. McCallum, " (Yuan, Research on K-value selection method of K-means clustering algorithm, 2019).," Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 169--178, 2000.

[21] M. a. S. R. a. I. S. M. S. Ahmed, "The k-means algorithm: A comprehensive survey and performance evaluation," Electronics, p. 1295, 2020.

[22] K. a. P. Rousseeuw, "Finding Groups in Data: An Introduction to ClusterAnalysis.," ohn Wiley & Sons,, 1990.

[23] P. J. a. B. J. G. W. Rousseeuw, "The remedian: A robust averaging method for large data sets," Journal of the American Statistical Association, pp. 97--104, 1990.

[24] C. a. Y. H. Yuan, "Research on K-value selection method of K-means clustering algorithm," J—Multidisciplinary Scientific Journal, pp. 226--235, 2019.