

Comparative Study between K-Means and K-Medoids Clustering Algorithms

Santosh Nirmal¹

¹Santosh Nirmal, Maharashtra, India

ABSTRACT - In many fields clustering algorithms are being used. Clustering is a process of grouping of similar objects into different groups or partitioning of a data set into subsets based on the distance measurement. Clustering techniques are important for statistical data analysis in theory, data points that are in the same group should have similar properties and/or features, while data points in different groups should have highly dissimilar properties and/or features. Clustering is a method of unsupervised learning and is a common technique for statistical data analysis used in many fields. In Data Science, we can use clustering analysis to gain some valuable insights from our data by seeing what groups the data points fall into when we apply a clustering algorithm. A study has been done on comparison between K-means and K-medoids algorithms using the sample datasets with different sizes. The results obtained were in favour of K-medoids algorithm which is better at clustering data when there are outliers than k-means algorithm. K-medoids is better at scalability for the larger dataset and also due to it being more efficient than K-means. It was also found that K-medoids showed its superiority over k means in execution time and to reduce the noise since it employs the method of minimization of the sum of dissimilarities of datasets.

Keywords: Cluster, K-means, K-medoids, Clustering methods, data analysis.

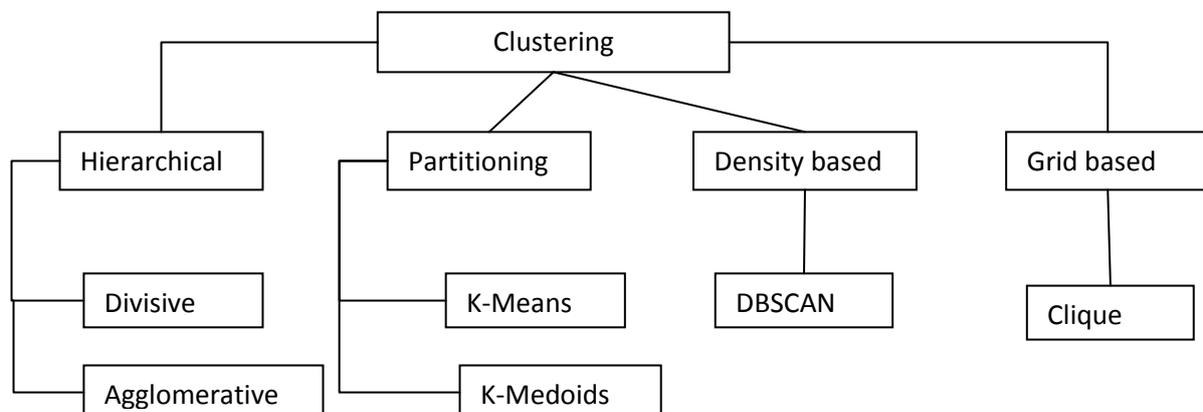
1. INTRODUCTION

Clustering can be considered the most important unsupervised learning problem; so, as every other problem of this kind, it deals with finding a structure in a collection of unlabeled data. A loose definition of clustering could be 'the process of organizing objects into groups whose members are familiar in some way'.

A cluster is therefore a collection of objects which are 'similar' between them and are "dissimilar" to the objects belonging to other clusters.

It's an important method of segregating various objects in a way that homogenous data occupies the same cluster while the heterogeneous are placed in another cluster. Clustering has gained wide usage and its importance has grown proportionally because of the ever-growing amount of data and exponential increase in computer's processing speeds. The importance of clustering can be understood from the fact that it has a wide variety of application like education, industries, agriculture, machine learning, pattern recognition and economics. In the modern technologies of artificial intelligence and pattern recognition too this technique has found its applications.

Clustering techniques are classified as follows.



(Figure – 1)

2. METHODOLOGY

Cluster analysis is a technique that is used to classify the objects into relative groups called clusters. The target is that the objects within the same group be familiar or related to one another and different from other objects in other clusters. So to make a better clustering, homogeneity or similarity should be greater within the group and the difference should be more between the different groups. Various clustering algorithms have been developed for different applications depend on the applications need.

This research work is carried out to compare the performance of k-Means and k-Medoids clustering algorithms based on the clustering result.

2.1 K-Means Method

K-means is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. It performs the division of objects into clusters which are similar between them and dissimilar to the objects belonging to another cluster. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori.

The main idea is to define k centroids, one for each cluster. These centroids should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early groupage is done. At this point we need to re-calculate k new centroids as barycenters of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop we may notice that the k centroids change their location step by step until no more changes are done.

The Euclidean distance between an object and all the nearby centroid is calculated as per the formula given below

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

Where $\|x_i^{(j)} - c_j\|^2$ is the nearest distance measure between a data point x_{ij} and the Centroid C_j , and it indicates the distance between data points from their Centroid. The time complexity of the K-means algorithm is subjected to the formula; $O(n^{dk+1})$.

Algorithm:

You must supply two things to the algorithm:

- The data points themselves
- K - The number of clusters

So, you should know the number of clusters beforehand. This might be a problem in some situations, but there are hacks to use K-means when you don't know the number of clusters.

Input :

$D = \{t_1, t_2, \dots, t_n\}$ //Set of elements

K //Numbers of desired clusters

Output:

K //Set of clusters

K-Means algorithm:

Assign initial values for m_1, m_2, \dots, m_k

Repeat

Assign each item to the cluster which has the closest mean;

Calculate new mean for each cluster

Until convergence criteria is met.

Step-By-Step Example

Datapoints: {2,4,10,12,3,20,30,11,25}

Cluster number : 2

m1 and m2 : mean values

Step 1:

Value of clusters

K1{ 2 }

K2{ 3 6 8 12 15 18 22 25 }

Value of m

m1=2.0 m2=13.625

Step 2:

Value of clusters

K1 {2 3 6}

K2 {8 12 15 18 22 25}

Value of m

m1=3.66 m2=16.66

Step 3:

Value of clusters

K1 {2 3 6 8}

K2 {12 15 18 22 25}

Value of m

m1=4.75 m2=18.4

Step 4:

Value of clusters

K1 {2 3 6 8}

K2 {12 15 18 22 25}

Value of m

m1=4.75 m2=18.4

Step 5: The final clusters by K-means are as follows:

K1 {2 3 6 8 }

K2 {12 15 18 22 25}

K-Medoids Method:

It is one of the important method of partitioning. K-medoid is based on medoids calculating by minimizing the absolute distance between the points and the selected centroid, rather than minimizing the square distance. As a result, it's more robust to noise and outliers than k-means. In k-medoids clustering, each cluster is represented by one of the data point in the cluster. These points are named cluster medoids.

Here, k data objects are selected randomly as medoids to represent k cluster and remaining all data objects are placed in a cluster having medoid nearest (or most similar) to that data object. After processing all data objects, new medoid is determined which can represent cluster in a better way and the entire process is repeated. Again all data objects are bound to the clusters based on the new medoids. In each iteration, medoids change their location step by step. This process is continued until no any medoid move. As a result, k clusters are found representing a set of n data objects

The Distance is calculated as per formula given below (figure):

$$j = \sum_{i=1}^k \sum_{p \in \Omega c_j}^n \|P - O_j\|$$

The time complexity for the K-medoids algorithm is subjected to the formula; $O(k(n-2)^2)$. The efficiency and performance of the results in the cluster are directly dependent on clustering centre chosen. Hence all efforts to improve this algorithm depend on the which k cluster points are chosen as reference.

Algorithm

1 Initialize: randomly select k of the n data points as the medoids

2 Associate each data point to the closest medoid. ("closest" here is defined using any valid distance metric, most commonly Euclidean distance, Manhattan distance or Minkowski distance)

3 For each medoid m :

1. For each non-medoid data point o

1. Swap m and o and compute the total cost of the configuration

4 Select the configuration with the lowest cost.

5 Repeat steps 2 to 4 until there is no change in the medoid.

Step-By-Step Example

Given data points: {2, 4, 10, 12, 3, 20, 30, 11, 25},

Number of clusters $k=2$

Step: 1:

{2}

{3 6 8 12 15 18 22 25}

C2 : S=-18

Step: 2:

{2 3 }

{6 8 12 15 18 22 25}

C2 : S=-10

Step: 3:

{2 3}

{8 6 12 15 18 22 25}

C2 : S=-12

Step: 4:

{2 3 6}

{12 8 15 18 22 25}

C2 : S=-6

Step: 5:

{2 3 6 8}

{15 12 18 22 25}

C2 : S=-3

Step: 6:

{2 3 6 8}

{18 12 15 22 25 }

Step: 7: The final clusters by K-Medoids are as follows:

{2 3 6 8}

{18 12 15 22 25}

EXPERIMENTAL ANALYSIS

Data points

For the purpose of experimental evaluation of superiority of K-Medoids over K-means algorithm, different data points are used with different size of data. The performance of the algorithms have been analyzed for several executions by considering different data points.

K-means Vs K-medoids Algorithms

This paper involves the implantation of the k-means and k-medoids algorithms and testing individual's performance on the basis of quality clustering, number of steps take to create the clusters. Time taken for execution of the algorithm varies from

one run to another run based on the data points and size of the data points. For comparison of both algorithms, java code has implemented with different size of the data as input, the reason being java is a modern, robust but simple, portable and object-oriented programming language which is eventually based on C and C++ programming languages. K-medoids can be used with any similarity measure. K-means however, may fail to converge - it really must only be used with distances that are consistent with the mean. The medoid as used by k-medoids is roughly comparable to the median (in fact, there also is k-medians, which is like K-means but for Manhattan distance). The main drawback of K-Medoids is much more expensive and this is the main drawback.

CONCLUSION AND FUTURE WORK

In this paper I conclude the results of both K-mean and K-Medoids clustering algorithms with respect to the number of clusters formed. K-Medoid also have some disadvantages like it is more costly than k-means method. It does not scale for well for large data sets. For large values of n and k, such computation becomes very costly. Also the result of dataset shows that K-Medoids is better in all aspects such as execution time, non sensitive to outliers and reduction of noise but with the drawback that the complexity is high as compared to K-Means.

We have tried to obtain accurate results of clustering by using two popular clustering algorithms with the number of clusters formed and distance metric. This metric can be extended using three more distance metrics namely euclidian, manhattan and correlation in our future work.

The present work aimed to compared K-medoids algorithm and K-means algorithm to check the improved efficiency and scalability of each of these. The results obtained after performing clustering a number of times prove K-medoids superiority of K-means in the execution time, quality clustered classes and also the number of records.

The data obtained using K-medoids was compared with K-means using real samples obtained from the reliable repository.

REFERENCES

- [1] Kalpit G. Soni and Dr. Atul Patel, "Comparative Analysis of K-means and K-medoids Algorithm on IRIS Data." International Journal of Computational Intelligence Research, ISSN 0973-1873, Volume 13, Number 5, 2017, pp. 899-906.
- [2] Jiawei Han, Han Kamber, "Data Mining Concept and Techniques", 2nd Edition,
- [3] Akshay Agrawal, Sachin Bojewar, "Comparative study of various clustering techniques", International Journal of computer science and mobile computing
- [4] Shalini S Singh and N C Chauhan, "K-means v/s K-medoids: A Comparative Study", National Conference on Recent Trends in Engineering & Technology, 2011.
- [5] Abhishek Patel and Purnima Singh, "New approach for K-mean and K-medoids Algorithm," International Journal of Computer Applications Technology and research Volume 2, {yr}.
- [6] Velmurugan T, "Performance Analysis of K-means and K-medoids clustering algorithm for a randomly generated data set. {yr}.
- [7] C. Kondal raj, "Comparison of K means and K medoids, DBSCAN, algorithms using DNA Microarray dataset, International Journal of Computational and Applied Mathematics, ISSN 1819-4966, Volume 12, Number 1, 2017.
- [8] <https://www.tutorialride.com>
- [9] <https://anuradhasrinivas.files.wordpress.com>
- [10] <https://stackoverflow.com>