Global Scientific JOURNALS

GSJ: Volume 8, Issue 11, November 2020, Online: ISSN 2320-9186
www.globalscientificjournal.com

Comparison between clustering algorithms: k-means and k-medoids using JAVA Program

Dinesh Bajracharya
Kantipur College of Management and Information Technology
Kathmandu, Nepal

Keywords: k-means, k-medoids, cluster, Euclidean, Manhattan, Sum of squares, nanoTime

Abstract

Clustering is the process of grouping or partitioning data into groups or clusters. Data is grouped into clusters based on the similarity of the objects. Objects of one cluster are similar to each but objects of two different clusters are quite dissimilar. Clustering process has application in several areas like business, medicine, e-commerce etc. Several clustering algorithms have been developed, each has its own pros and cons. K-means and k-medoids are two popular partitioning-based clustering algorithms. K-means algorithm create clusters that minimize the squared-error function and k-medoids create clusters that reduces differences between each object of a cluster and the its corresponding reference point. Effect of size of dataset and different reference points on the execution time of k-means and k-medoids are interesting idea to know.

Introduction

Clustering is process of partitioning or grouping of objects in different groups or clusters based on their similarity with each other. In clustering process objects are compared with each other based on their similarity of their attributes. Clustering process groups objects with similar characteristics in groups (clusters). Clustering process generates one or more clusters, objects of one group are quite similar to each other while objects of two different clusters are quite dissimilar. Clusters or groups of several different kinds of objects like customers, employees, gene, vehicles, books, articles, documents, web-surfers, images etc. can be created. Clustering process have application in several areas like education, medicine, business, e-commerce etc. In department store clusters of customers based on the purchasing behavior of customers can be created, in Internet websites create clusters of web-surfers based on their online click stream behavior can be created. Department stores can develop marketing strategies according to nature of the clusters to increase sales and improve relationships with their customers.

Partitioning data to form clusters is not done by human as it is a very complex process but by clustering algorithms [1]. Clustering if done manually will be a very time consuming and tedious jobs. Several clustering algorithms (methods) have been developed. They are:

- Partitioning based
- Hierarchical based
- Density based etc.

Several partitioning-based, hierarchical-based, density-based algorithms have been developed. In this study comparison between k-means and k-medoids, the partitioning-based algorithms, are done to see performance of each. Java programs were written to implement both algorithms and used to compare performance of both algorithms.

k-means

k-means is a partition-based clustering algorithm. It creates **k** number of clusters by partitioning database containing **n** number of objects. This algorithm works as follows:

Initially k number of objects are randomly selected as cluster centers (centroids) from the given database of size n. Remaining objects of the dataset are distributed to the clusters based on their closeness with the selected k cluster centers and k number of clusters are formed. New cluster centers are calculated for each newly formed cluster. Again, remaining objects are distributed to clusters based on nearness to the cluster centers. The steps for calculating cluster centers and distributing remaining objects continue until some specified condition is not met [1]. Closeness between object and centroid is calculated using Euclidean distance formula. Euclidean distance formula:

$$d(x, y) = \sum_i^n \sqrt{(x_i - y_i)^2}$$

Where x and y are two different objects.

In clusters formed with k-means algorithm, dispersion of objects within the Clusters is low but dispersion of objects between the clusters is high as k-means uses square-error criterion to find distance between objects and centroids.

Goodness of clusters formed by k-means algorithm highly depends upon the correctness of randomly selected initial cluster centroids [5]. Nature of formed clusters can be known by calculating value of Sum of squares error (SSE). The smaller value of SSE means data in clusters are homogeneous and clustering results are good [13]**.**

k-medoids

k-medoids is another partitioning-based clustering algorithm. It also creates k clusters from dataset of size n like k-means algorithm. This algorithm works as follows:

Initially, k objects from the given dataset are selected as cluster centers (medoids). The selected medoids are supposed to be representative of the clusters and are not selected randomly. Remaining objects of the dataset are distributed to the clusters based on their similarity with the selected medoids and clusters are formed. New object which has minimum total cost to the remaining objects of the cluster is selected as new medoids for the cluster. The remaining objects are distributed to the clusters based on their similarity with the medoids. The steps for selecting most representative cluster medoids and distributing remaining objects around medoids continue until some specified condition is not met [1]. Closeness between object and centroid is calculated using Manhattan distance formula (or Euclidean distance formula or some other distance formula can be used).

Manhattan distance formula:

$$d(x,y) = \sum_{i}^{n} |(x_i - y_j)|$$

Sum of Squares within groups, Sum of squares between groups

Sum of squares within groups (SSW) shows how the individual data varies from the mean of the group. Sum of squares between groups (SSB) shows how mean of the individual groups varies from the mean of whole data set (grand mean). If the variance between different clusters is much greater than the variance within a cluster then the cluster means are not equal to grand mean [9].

Sum of Squares within groups is calculated using following formula:

$$SSW = \sum (x_{ij} - \bar{x}_j)^2$$

Sum of Squares between groups is calculated using following formula:

$$SSG = \sum (\overline{x_{ij}} - \bar{x}_j)^2$$

Related works

Velmurugan T. has performed comparison between k-means and k-medoids using JAVA program with arbitrary distributed dataset [2]. Quality of clusters was evaluated based on the distance between two data points and computational time take be each algorithm. The experimental results showed k-Means algorithm performing better than k-Medoids algorithms. Nirmal S. considered dataset with different sizes to evaluate k-Means and k-Medoids algorithms and found execution time of k-Medoids better than that of k-Means [3]. Soni K.G., Patel A. have compared k-means with k-medoids using dataset with one hundred fifty instances of Iris plants with five attributes and found k-medoids to be superior than k-means in terms of execution time, sensitivity towards noise and outliers [4]. Gultom S., et. al. considered data set of 147,679 instances of students of class three with six variables to evaluate k-means and k-medoids algorithms using Eculid Distance, Chanberra Distance and Chebyshev Distance and found k-means method to be more optimal in data clustering compared to k-medoids in all considered distance metrices [7]. Arora P. et al., have evaluated K-means and K-medoids using dataset transaction10K of KEEL. Randomly distributed data points were input to the algorithms. The result of comparison showed that K-medoids was better than K-means in terms of execution time, noise [8].

Methodology

- Two Java programs were implemented, one for k-means, another for k-medoids
- Relevant articles, books were studied
- Three different datasets of different sizes were prepared and stored in MySQL database
- Programs were executed with considered datasets and different cluster centers several times and different metrics were recorded
- On the basis of recorded metrics and reviewed literatures conclusion was derived.

Experimental Setup

Two separate Java programs were developed, one for K-means and another for K-medoids. Each program was provided same set of data with same initial cluster centroids. Dataset were retrieved from

MySQL database. Number of clusters to be formed was fixed to three. Three different datasets were considered for evaluation of the algorithms, first set contained 400 objects, second contained 2292 and third set contained 5949 objects all having two attributes. The first dataset contained randomly generated data. Second dataset contained real data of examination with number of students appeared in exam and number of students passed in the exam as the attributes of objects. Third dataset contained number of male and female students of campuses.

Execution time for the completion of algorithms were recorded using JAVA System.nanoTime() method. Number of iterations taken by each algorithm, time in nanosecond and millisecond, sum of square between groups and within groups were calculated. Clusters formed were drawn in JPanel window.

With the first dataset programs were run three times, with second and third datasets programs were run five times each; number of iterations, execution time, square of sum within formed clusters and square of sum between groups were recorded. Following table shows average of recorded values.

Table 1. Result of programs rum with first dataset containing 400 records

	Nano Second	Iteration	SSB	SSW
k-means	4403633.333	6	15526	9645
k-medoids	65016666.67	8	16063	9226

Figure 1. Clusters formed by k-means and k-medoids from dataset of size 400

Table 2. Result of programs rum with second dataset containing 2282 records

	Nano Second	Iteration	SSB	SSW
k-means	28848420	24	581433	149316
k-medoids	619402820	10	297951	382840

Figure 2. clusters formed by k-means and k-medoids with dataset 2282

Table 3. Result of programs rum with third dataset containing 5949 records

	Nano Second	Iteration	SSB	SSW
k-means	44833580	23	1078605	269004
k-medoids	3082379320	6	627901	712172

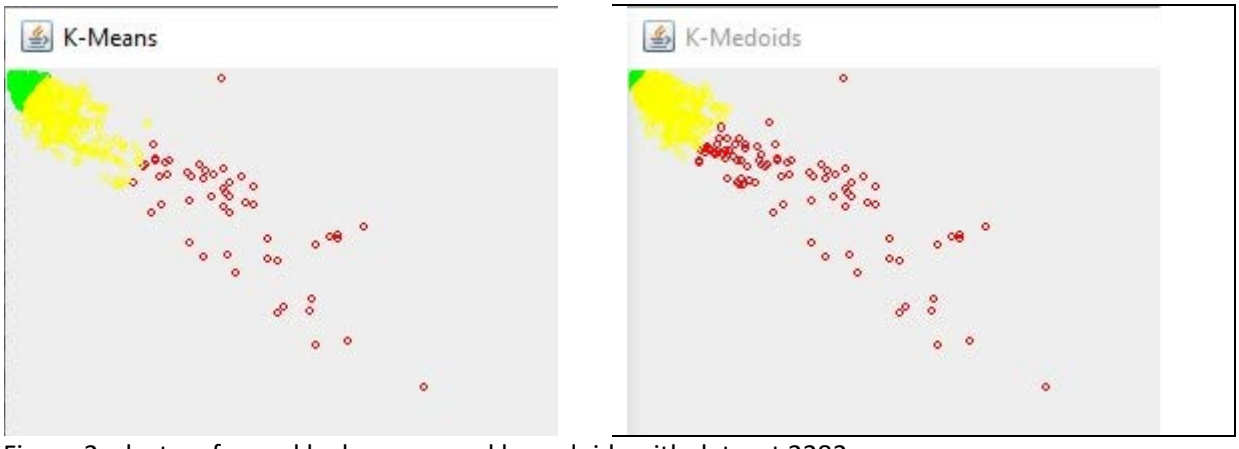

Figure 2. clusters formed by k-means and k-medoids with dataset 2282

From above tables, execution time of k-means is found better than of k-medoids even the number of iterations taken by k-means is more than k-medoids. The objects in clusters formed by k-means are more compact than objects in clusters formed by k-medoids as the sum of squares within groups for k-means are less than for k-medoids for two big sets. But sum of squares between groups are seen more

for clusters formed by k-means than k-medoids. From above observations, execution time of k-means algorithm was found less than k-medoids. Fig. 1, Fig. 2, and Fig. 3 show clusters created by both algorithms. These figures show that size of clusters formed by both algorithms are not same and nature of data has effect on the algorithms.

Discussion

Partitioning algorithms like k-means, k-medoids for clustering require specification of number of clusters and initial cluster centroids. The selection of appropriate cluster centers is a difficult job. Formation of good clusters heavily depends on the initial selection of cluster centers. The experiment was run several times with three different datasets of different sizes and several randomly selected cluster. The result of program executions showed that the execution of k-means algorithm is less than that of k-medoids. This finding is similar with the results is found by Velmurugan T. [1], and Gultom S. et al [7].

From the study it is seen that the nature of clusters formed depends on the nature of data and differs with considered algorithms. As stated by Han and Kamber [1], k-means reduces the sum of squares between the objects, same is found in the experiment. In case of k-medoids, new medoids are formed by selecting the most representative object and proved to a time-consuming process, resulted in high execution time. In k-means, mean of the formed clusters were calculated to find new cluster centers and were not time-consuming steps.

In k-means, centroid calculated for a cluster is mean of all the existing objects within the clusters, so the centroids may not represent any object of the clusters. This complicates explain cluster center and cluster as centroid may not be any of the objects of the considered data set. In k-medoids, medoid considered is one of the existing objects of the cluster which has the minimum total squared sum with the remaining objects of the clusters. So, it is easy to explain clusters and medoids formed by k-medoids algorithm.

Finding similarity between high dimensionality objects may be difficult, objects seem to be highly similar to each other may be seen quite dissimilar or moderately similar when distance-based similarity measures are used and conversely, objects seem to be dissimilar may be seen similar or moderately dissimilar when compared with distance-based similarity measures [10]. Garg S. and Jain R. C. have found degradation in the performance of clustering algorithms with the increase in dimension of clustering objects [11]. Type of data and purpose and application of usage also has effect on the choice of clustering algorithm [12]. The clusters formed by k-means and k-medoids may differ as the distance-based similar measures are affected by high dimensionality of data and also nature of values of considered dimensions.

Conclusion

The comparative study between k-means and k-medoids using JAVA programs showed that execution time of k-means is less than k-medoids, shape and size of clusters formed by both algorithms for the same datasets of same size with same initial centroid differ. The selection of initial centroid is a difficult process and have impact on the shape and size of formed clusters. The size and type of dataset have effect on the performance of both algorithms, and execution time of k-medoids increases with increase with the size of dataset.

Clustering process plays very important role in several disciplines like education, business, medicine etc. Several algorithms have been developed for creating clusters from given datasets. K-means and k-medoids are two popular clustering algorithms. A comparative study between k-means and k-medoids was conducted using Java programs and found k-means better than k-medoids in execution time.

References:

[1] J. Han, M. Kamber, "Data Mining Concepts and Techniques", Morgan Kaufmann Publishers is an imprint of Elsevier, pp. 401-407, 2012. 2006

[2] Velmurugan T., "Efficiency of k-Means and K-Medoids Algorithms for Clustering Arbitrary Data Points", Int. J. Computer Technology & Applications, Vol 3 (5), 1758-1764, 2012.

[3] S. Nirmal, "Comparative Study between K-Means and K-Medoids Clustering Algorithms", International Research Journal of Engineering and Technology (IRJET), Volume: 06 Issue: 03 | Mar 2019

[4] K.G. Soni, A. 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, pp. 899-906, 2017

[5] M. Yedla, S. R. Pathakota, T. M. Srinivasa, "Enhancing k-means clustering algorithm with improved Initial Center", International Journal of Computer Science and Information Technologies, Vol. 1 (2), 121-125, 2010

[6] l B. J. D. Sitompu, O. S. Sitompul, P. Sihombing, "Enhancement Clustering Evaluation Result of Davies-Bouldin Index with Determining Initial Centroid of K-means Algorithm", The 3rd International Conference on Computing and Applied Informatics 2018, IOP Conf. Series: Journal of Physics: Conf. Series 1235 (2019) 012015, 2018

[7] S. Gultom et al., "Clustering analysis of k-means and k-medoids with Ecluidience Distance Algorithm, Chanberra Distance, and Chebyshev Distance for Big Data Clustering", 2nd Nommensen International Conference on Technology and Engineering, IOP Conf. Series: Materials Science and Engineering **420** (2018) 012092, 2018

[8] Arora P., Deepali, Varshny S., "Analysis of K-Means and K-Medoids Algorithm For Big Data", International Conference on Information Security & Privacy (ICISP2015), 11-12 December 2015, Nagpur, INDIA

[9] Newsom, "Notation and Computation of One-Way ANOVA", Psy 521/621 Univariate Quantitative Methods, Fall 2019

[10] M. Steinbach, L. Ertöz, V. Kumar, "The Challenges of Clustering High Dimensional Data. In: Wille L.T." (eds) New Directions in Statistical Physics. Springer, Berlin, Heidelberg, 2004

[11] S. Garg, R. C. Jain, "Variations of k-mean Algorithm: A study for High Dimensional Large Data Sets", Information Technology Journal, 2006

[12] T. Velmurugan, T. Santhanam, "Performance Analysis of K-Means and K-Medoids Clustering Algorithms for a Randomly Generated Data Set, International Conference on Systemics, Cybernetics and Informatics, 2008.

[13] Bernad Jumadi Dehotman Sitompul *et al, "*Enhancement Clustering Evaluation Result of Davies-Bouldin Index with Determining Initial Centroid of K-Means Algorithm*",* 2019 *J. Phys.: Conf. Ser.* **1235** 012015