A COMPARATIVE STUDY OF K-MEANS AND K-MEDOIDS CLUSTERING METHODS.

[1]OSEMWENKHAE J.E., [2]EKHATOR O.F., and [3]IDUSERI A.

[1,3] Department of Mathematics, Faculty of Physical Sciences, University of Benin, P.M.B.1154,Benin City 300001, Edo State, Nigeria

[2]Advanced Research Laboratory, Department of Mathematics, University of Benin, P.M.B.1154,Benin City 300001, Edo State, Nigeria

Corresponding Author: Ekhator O.F., Email: sarahtaurus17@yahoo.com, Tel:+2348023394966, +2348079239937

ABSTRACT

The aim of this work is to provide a formal and organized study of the effect of the nature of data and cluster structure on the performance of K-means and K-medoids clustering methods. A cluster validation method called Silhouette analysis is used to assess the quality of cluster partitions created by both methods. An illustration on how Silhouette analysis could be used to determine the optimal number of clusters in a data set is presented. Results obtained reveal that the performance of K-means is at its peak with data in which clusters are of relatively uniform sizes while the K-medoids method tends to perform better than K-means when the input data have varied cluster sizes.

Keywords: Cluster Analysis, Cluster Validation, Distance Functions, K-means, K-medoids, Silhouette Analysis

1.0 Introduction

Cluster analysis encompasses various methods for grouping data objects in a way that the degree of similarity between any two objects in the same group and their degree of dissimilarity between any two objects in different groups is maximal. These groups are referred to as “clusters”.

What cluster analysis does is, discover groups in a data set in a way that reveals underlying patterns (if any) that could either provide immediate insights from which useful conclusions are made or a foundation upon which to perform further analyses. It is an unsupervised process. In some literature, it is called automatic classification, numerical taxonomy, botrology and typological analysis [1]. Clustering techniques have numerous areas of application ranging from Ecology and genetics, to business, spatial data analysis and web mining.

To determine the degree of similarity between two objects and ascertain if they should be in the same cluster, the concept of distance function are employed. The higher the value of the distance function, the less similar the objects are. There are a number of them (Euclidean distance, Squared Euclidean distance, Manhattan distance, Cosine distance, Correlation distance, Canberra distance, e.t.c.) but the ones that will be used in this study are the Euclidean and Manhattan distance functions because of the mathematical tractability. Mathematically, they are expressed as:

EUCLIDEAN DISTANCE FUNCTION:

$$d(X,Y) = \sqrt{|x_{i1} - y_{j1}|^2 + |x_{i2} - y_{j2}|^2 + \cdots + |x_{ip} - y_{jp}|^2} \qquad (1)$$

MANHATTAN DISTANCE FUNCTION:

$$d(X,Y) = |x_{i1} - y_{j1}| + |x_{i2} - y_{j2}| + \cdots + |x_{ip} - y_{jp}| \qquad (2)$$

for two p-dimensional data objects $X= (x_{i1},x_{i2}, ...,x_{ip})$ and $Y= (y_{j1},y_{j2}, ...,y_{jp})$

2.0 Literature Review

Different clustering algorithms often result in entirely different partitions even on the same data. The question is, how do we identify the algorithm that produces the best cluster partition for a given data?. As far as the K-means and K-medoids are concerned, research efforts have been made in the past to compare the performance of both algorithms and some interesting results have been established.

Batra [2] examined the K-means and K-medoids algorithms based on their basic approach. Input data points were generated from the normal distribution and the uniform distribution and the best algorithm in each category was found out based on their performance. It was observed that K means algorithm is efficient for smaller data sets and K medoids seems to perform better than K-means for large datasets.

Velmurugan [3] used arbitrarily distributed input data points to evaluate the clustering quality and performance of K-means and K-medoids. The computational time was calculated for each algorithm in order to measure the performance of the algorithms. The results of both the algorithms were analyzed based on the number of data points and the computational time of each algorithm and the K-means algorithm was found to take less computational time than the K-medoids algorithm and to him, more efficient.

Arora et al. [4] showed that K-medoids is better than K-means in aspects such as, sensitivity to outliers and noise, but with the drawback that the complexity is high as compared to K-means.

In view of the above, the essence of this study is to find out how these algorithms behave, when already known factors that cause anyone of the algorithms to perform better than the other, are kept in check. Examples of these factors are outliers and high data dimension. To achieve this, data without outliers will be used and the dimension of the data used in the analysis will be kept as small as possible.

The major categories of clustering methods are: partitioning methods, model-based methods, hierarchical methods, grid-based methods and density based methods.

These methods each have their plus and minuses. However, since the two clustering algorithms considered in this study are partitional methods, that will be the main area of focus.

3.0 Methodology

The partitioning methods will be the focus. There are a number of such methods (K-means, K-medoids, K-median, K-mode, e.t.c.) but for the purpose of this study, only the K-means and K-medoids methods will be described.

Given the value of k, The K-means and K-medoids methods select k data elements in the data set as cluster representatives. In the K-means method, these representatives are called centroids. The centroid is the mean of the data points in a cluster. The K-medoids method

uses a medoid, which is often the most centrally located data point, as the cluster representative.

3.1 The K-means Algorithm

The basic K-means algorithm was developed by MacQueen [5] and works as follows:

Step 1: Abitrarily choose k data objects as initial cluster centriods and allocate the remaining objects to the cluster with closest centriods (in terms of any of the appropiate distance functions).

Step 2: Compute new cluster cetroids as the mean of the data objects in each cluster and reallocate the remaining objects to the closest clusters

Step 3: Repeat Step 2 until there is no change in the cluster centroids.

Further details and some practical examples can be found in [2].

3.2 The K-medoids Algorithm

The K-medoids algorithm uses the medoids as cluster representatives. These medoids are actual data points in the data, unlike in the K-means algorithm where cluster means (which might not be values contained in the data) are used. As a result of this, the K-medoids algorithm is more tolerant to outliers in the data, than the K-means algorithm. Among many algorithms for K-medoids clustering, partitioning around medoids (PAM) [6] is known to be the most powerful. See Park and Jun [7].

The PAM algorithm works as follows:

Step 1: Select k data points arbitrarily as initail cluster medoids to represent the k clusters and assign each of the remaining data elements to the cluster with the closest medoid based on an appropiate distance function.

Step 2: Swap each medoid with any of the other remaining data elements (non medoids) and calculate the resultant change in total distance due to the swap.

Step 3: Select the set of medoids that result in the lowest distance as the new set of medoids.

Step 4: Repeat steps 2 and 3 until there is no change in medoids.

For further details and some practical examples, see Batra [2].

3.3 Cluster Validity

Here, the aim is to check the quality of the clusters in terms of the level of inter-cluster similarity and intra-cluster dissimilarity. An ideal clustering result is one that has a high degree of similarity among elements in the same cluster and a high degree of dissimilarity among the elements in diferent clusters.

Several cluster validity methods have been developed and introduced but for the purpose of this study, we will focus on Silhouette analysis. Apart from the assessment of cluster quality, cluster validity methods can also used to determine the ideal (optimal) number of clusters k, that the data should be partitioned into.

It can be used to compare the results of two or more clustering results, and also to determine the optimal number of clusters in a data set.

The average silhouette width (ASW), which is displayed alongside the Silhouette plot, is a measure of the quality of cluster structure.

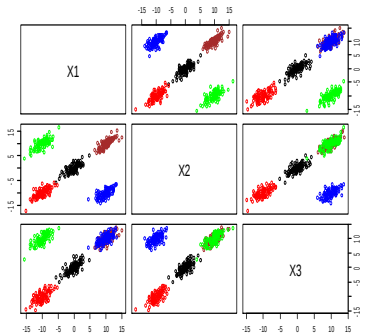
4.0 DATA SIMULATION

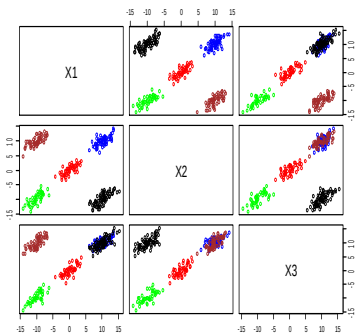
The "clusterSim" package in R was used to generate all the data sets used in this study. Three (3) data sets were generated from the multivariate normal distribution. The first data set is in 3 dimensions and contains 5 clusters, well seperated and of equal size. The second data set was generated using the same model as in the first data, but with unequal cluster sizes while the third data set was generated using a different model from that of the first and second data. This model generates data with complex cluster structure. The data is in 3 dimensions and contains five equal sized, overlapping clusters (not well separated).

Figures 1,2, and 3 show the scatter plot matrices (SPM) for these data.

The scatter plot matrix contains all pairwise scatter plots of the variables in a matrix format. That is, if there are p variables, $X_1, X_2, \ldots, X_p$ in the data, the scatter plot matrix will have p rows and p columns, where the element in the ith row and jth column of this matrix is a plot of X_i against X_j. Often times, the scatter plot matrix is used to check for the following: pairwise relationships between the variables, the nature of relationships (if any), outliers, and clustering by groups in the data

The scatter plot matrix was used in this study as a device to monitor the data generated and make sure that they have the required properties for the study.





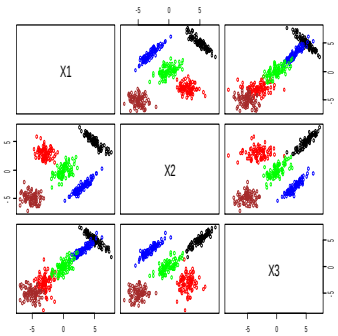


Fig. 1: SPM For Data I **Fig. 2: SPM For Data II** **Fig. 3: SPM For Data III**

Observe that none of the plots in the scatter plot matrices reflects the presence of outliers in the data sets. The matrices also show strong pairwise relationships between variables in the data and the maximum number of clusters in the pairwise scatter plots is five.

However, observe the presence of a complicated cluster structure reflected in the top right corner of the scatter plot for X_1,X_3 in Figure 3, where one cluster completely crosses over the other. This confirms that the 3rd data has a trace of cluster overlap.

5.0 Data Analyses and Interpretations

To analyse the data and compare the results for the K-means and K-medoids algorithm, the bivariate cluster plot and Silhouette plot were used. The bivariate cluster plot is used here to see how well the K-means and K-medoids algorithms are able to detect clusters in different data sets. The Silhouette plot displays a measure of how close a point in one cluster is to points in the neighboring clusters. From the thickness of the silhouette plot, the cluster size can be visualized. The average Silhouette width (ASW), which is displayed along with the Silhouette plot, is a measure of how well placed the elements are in their various clusters and how good the partitions are. The values of ASW lies in the range, $0 < ASW < 1$. The larger the value of ASW, the better the cluster partition. Some values of ASW and their corresponding interpretations can be found in [8].

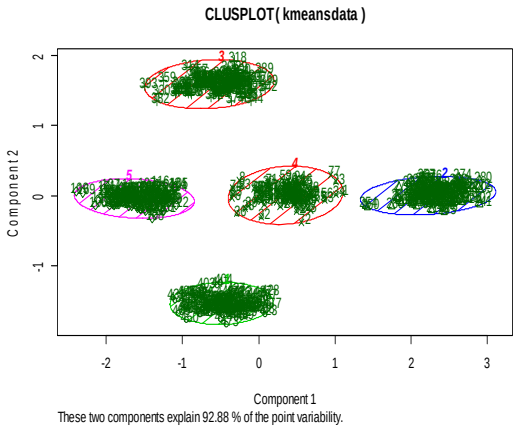


Fig. 4: K-means Cluster Plot For Data I

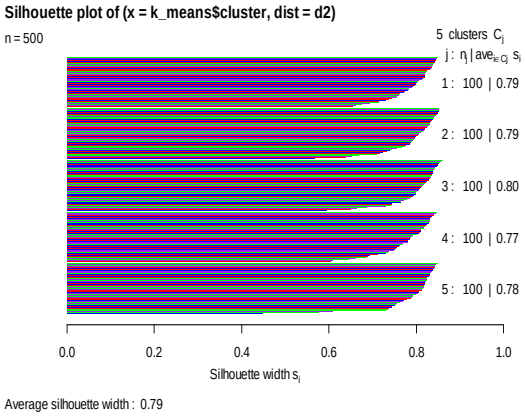


Fig. 5: K-means Silhouette Plot For Data I

The shaded regions in Figure 4 represent the various clusters found by the K-means algorithm for the first data set. By observation, the clusters are well seperated and easy to identify.

The right hand side of Figure 5 displays the cluster number, the respective number of observations in the clusters and Silhouette coefficient for each cluster. In this case, there are 100 observations per cluster and the respective Silhouette coefficients for clusters 1, 2, 3, 4 and 5 are 0.79, 0.79, 0.80, 0.77 and 0.78. These values indicate that the observations in the clusters formed by the K-means algorithm for the first data set are well clustered. The average Silhouette width which is the mean of the Silhouette coefficients for all the clusters, is displayed along with the Silhouette plot as 0.79. This Indicates that the cluster partitions are excellent.

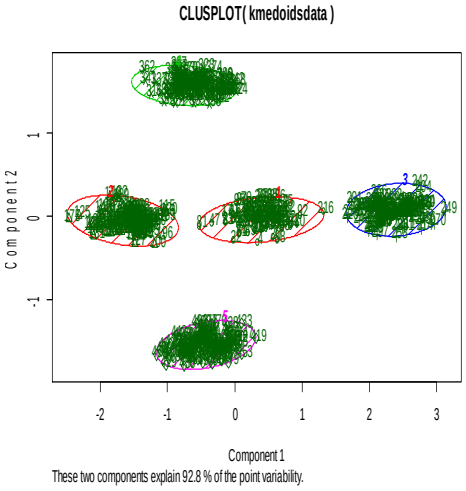


Fig. 6: K-medoids Cluster Plot For Data I

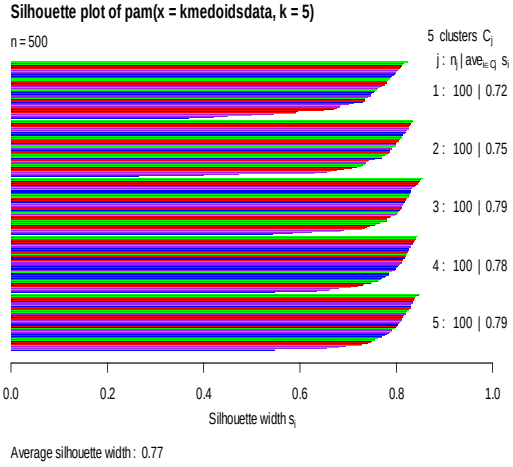
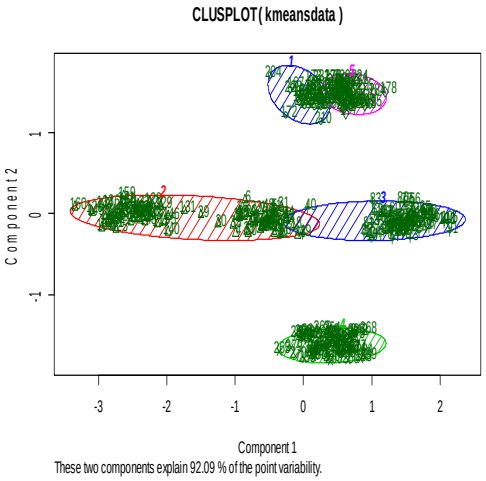


Fig. 7: K-medoids Silhouette Plot For Data I

The K-medoid cluster plot in Figure 6 shows well clustered observations. Although in some of the clusters there are a few data points that are just on the border of the cluster and almost out of the shaded region, it is still easy to tell which of the clusters those data points belong to. So, we can conclude that the clusters are significantly distinct. An ASW of 0.77 as shown in the K-medoids Silhouette plot indicates an excellent cluster partition for the K-medoids algorithm. However, this value is lower than the ASW for the K-means algorithm (i.e. 0.79), so we can conclude that the clustering results for the K-medoids algorithm is only almost as good as that of the K-means algorithm for this data set.



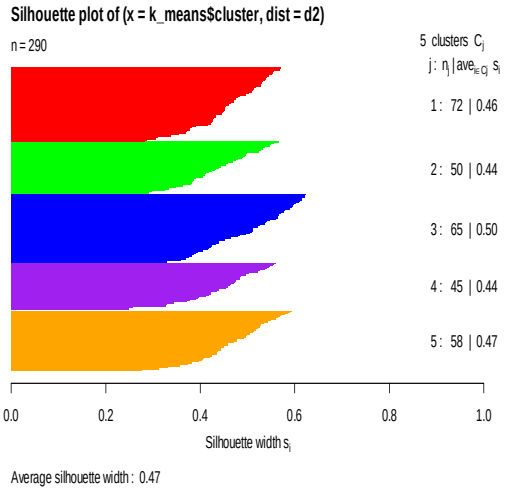
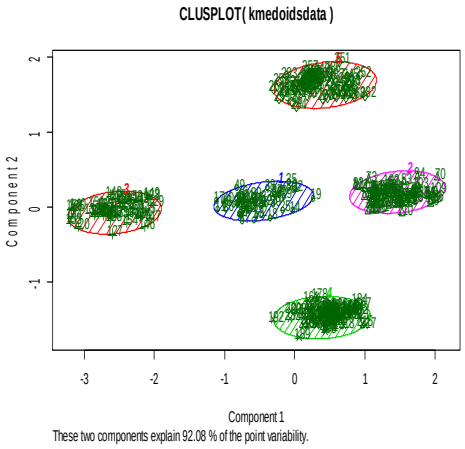


Fig 8: K-means Cluster Plot For Data II

Fig. 9: K-means Silhouette Plot For Data II

In the K-means cluster plot in Figure 8, there are 5 “chunks” of observations that appear to be clusters and some scattered data points which are not properly clustered and this has resulted in an erroneous merge of some of the “chunks” into a single cluster (see the shaded area in the middle of the plot). The clusters (weak) are not as distinct as they should be and it is difficult to tell which observations are in what cluster, especially in the merged clusters. This is also shown (Figure 9) by the ASW of 0.47.



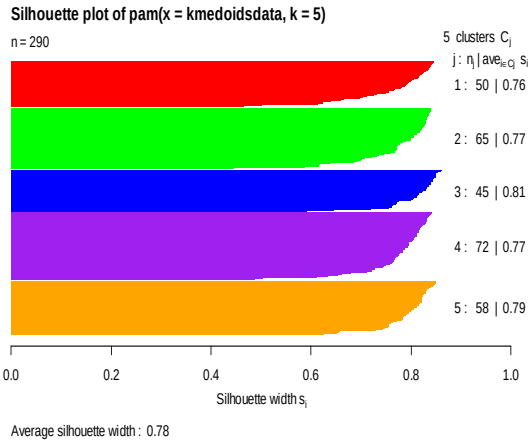
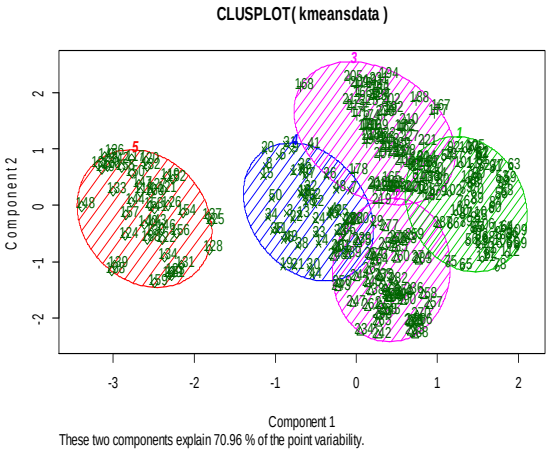


Fig.10: K-medoids Cluster Plot For Data II

Fig. 11: K-medoids Silhouette Plot Data II

The K-medoids cluster plot in Figure 10 reflects well clustered observations and distinct clusters. The K-medoids Silhouette plot shows an ASW of 0.79 which is far better than the ASW obtained for the K-means algorithm (0.47). So, we can conclude that for the second data, the performance of the K-medoids algorithm is better than that of the K-means algorithm.

Figures 12 and 13 show the K-means bivariate cluster plot and Silhouette plot for the 3rd data set with complicated cluster structure.



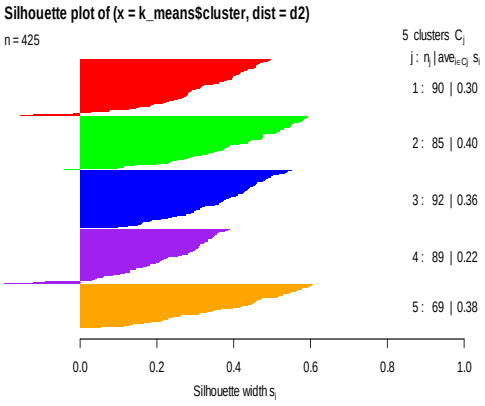


Fig. 12: K-means Cluster Plot For Data III

Fig. 13: K-means Silhouette Plot For Data III

The K-means cluster plot in Figure 12 reveals four overlapping clusters and only one distinct cluster for the 3rd set of data. Also, note from The K-means Silhouette plot in Figure 13, that some tiny fragments of the horizontal bars for the 1st, 2nd and 4th clusters are on the left hand side of the plot. This shows that the Silhouette values for some points in those clusters

are negative which implies that those points are in the wrong cluster. The ASW is 0.33 which indicates that the cluster structure found by the K-means algorithm for this data is very weak.

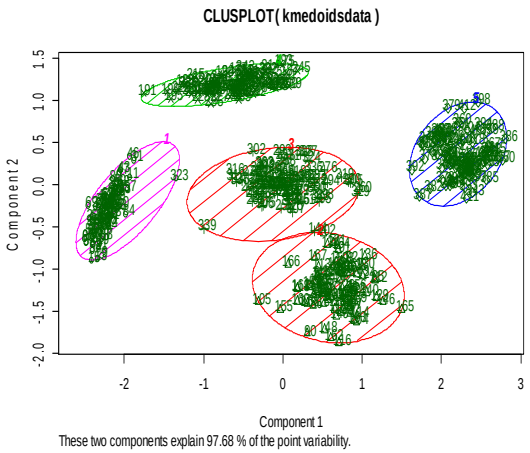


Fig.14: K-medoids Cluster Plot For Data III

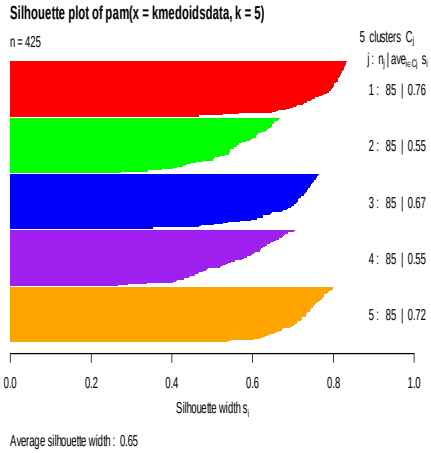


Fig.15: K-medoids Silhouette Plot For Data III

The cluster plot in Figure 14 shows a better cluster solution for the K-medoids algorithm than that of the K-means algorithm. Although some of the clusters are very close, they can still be categorized as distinct. The K-medoids Silhouette plot in Figure 15 shows that the specified number of obeservations per cluster (i.e. 85) was not violated. An ASW of 0.65 shown in the Silhouette plot indicates that a reasonable cluster structure was found by the K-medoids algorithm. Finally ASW of 0.33 for K-means compared with 0.65 obtaianed for K-medoids, shows that the cluster structure found by the K-means algorithm for the 3rd data set is not as good as the cluster structure found by the K-medoids algorithm.

6.0 CONCLUSION AND RECOMENDATION

The results of the analyses show that the K-means algorithm works better with data that has clusters of approximately equal sizes. The K-medoids algorithm tends to accomodate data with unequal cluster sizes more than the K-means algorithm and is less likely to cut incorrect borders between clusters in such data. The K-medoids algorithm also performs better than the K-means algorithm for data with complex cluster structure. For data with no outliers, or complex cluster structure, the k-means algorithm is very efficient and is recommended

The use of at least one cluster validation method to evaluate clustering results is recommended, as this is a reliable way of knowing if the results obtained are good enough.

References

[1] Madhulatha, T. (2012), An Overview on Clustering Methods, IOSR Journal of Engineering, Vol. 2(4) pp: 719-725.

[2] Batra, A. (2011), Analysis and Approach: K-Means and K-Medoids Data Mining Algorithms, 5th IEEE International Conference on Advanced Computing & Communication Technologies, pp 274-279.

[3] Velmurugan, T. (2012), Efficiency of K-means and K-medoids Algorithms for Clustering Arbitrary Data Points. International Journal of Computer Technology & Applications,Vol 3 (5), 1758-1764.

[4] Arora, P. (2015), Varshney, S. Analysis of K-Means and K-Medoids Algorithm For Big Data. International Conference on Information Security & Privacy, Nagpur, INDIA. PP.507-512.

[5] MacQueen, J.(1967), Some methods for classification and analysis of multivariate observations. Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics, 281--297, University of California Press, Berkeley, California.

[6] Kaufman, L. and Rousseeuw, P.J. (1987). Clustering by means of Medoids. In Statistical Data Analysis Based on the L1–Norm and Related Methods, edited by Y. Dodge, North-Holland, 405–416.

[7] Park H.S., Jun C.H. (2009), A simple and fast algorithm for K-medoids clustering. Expert Systems with Applications, 36, (2), 3336–3341.

[8] http://www.stat.berkeley.edu/~spector/s133/Clus