ISSN:0975-9646

Comparison of Various Clustering Algorithms

Garima Sehgal[#1] Dr. Kanwal Garg[#2]

Research Scolar[#1], Assistant Professor[#2]
Department of Computer Science and Applications, Kurukshetra University
Kurukshetra, India

Abstract – A comparative study of clustering algorithms across three different datasets is performed. The algorithms under investigation are partitioning based i.e K-means, Farthest First, Expectation maximization and Non Partitioning based i.e Density based, Hierarchical based and Cobweb. All these algorithms are compared according to the factors size of the dataset, number of clusters and time taken to form clusters. Performance of clustering algorithms are compared using clustering tool WEKA(version 3.7.10).

Keywords – ***K-means algorithm, Farthest First algorithm, Expectation Maximization algorithm, Density based algorithm, Hierarchical based algorithm, Cobweb algorithm, WEKA tool.***

I INTRODUCTION

Clustering is division of data into groups of similar objects. Each group called cluster, consists of objects that are similar amongst themselves and dissimilar compared to objects of other groups[3]. Different clustering algorithms are present to form clusters. WEKA(3.7.10) tool is used to compare different clustering algorithms. It is used because it provides better interface to the user than compare to other data mining tools. Clustering algorithms which are compared are partitioning based i.e K-means, Farthest First, Expectation maximization and Non Partitioning based i.e Density based, Hierarchical based and Cobweb. Next section discusses about the various clustering algorithms.

A. K-means Clustering Algorithm

K-means clustering algorithm is first proposed by Macqueen in 1967 which was uncomplicated, non-supervised learning clustering algorithm. K-means is a partitioning clustering algorithm, this technique is used to classify the given data objects into k different clusters through the iterative method, which tends to converge to a local minimum. So the outcomes of generated clusters are dense and independent of each other. The algorithm consists of two separate phases. In the first phase user selects k centres randomly, where the value k is fixed in advance. To take each data object to the nearest centre. Several distance functions are considered to determine the distance between each data object and the cluster centres. When all the data objects are included in some clusters, the first step is completed and an early grouping is done. Then the second phase is to recalculate the average of the early formed clusters. This iterative process continues repeatedly until the criterion function becomes the minimum.[4]

B. Expectation Maximization Clustering Algorithm

Expectation Maximization (EM) algorithm is an iterative method for finding maximum likelihood or maximum a posteriori (MAP) estimates of parameters in statistical models, where the model depends on unobserved latent variables. The EM iteration alternates between performing an expectation (E) step, which creates a function for the expectation of the log-likelihood evaluated using the current estimate for the parameters, and a maximization (M) step, which computes parameters maximizing the expected log-likelihood found on the *E* step. These parameter-estimates are then used to determine the distribution of the latent variables in the next E step. EM assigns a probability distribution to each instance which indicates the probability of it belonging to each of the clusters.[5]

C.Farthest First Clustering Algorithm

Farthest first is a modified of K-Means that places each cluster center in turn at the point further most from the existing cluster center. This point must lies within the data area. This greatly increases the clustering speed in most of the cases since less reassignment and modification is needed.[2].

D. Hierarchical Clustering Algorithm

Hierarchical method creates a hierarchical decomposition of the given set of data objects forming a dendrogram- a tree which splits the database recursively into smaller subsets. The dendrogram can be formed in two ways bottom up or top down. Hierarchical algorithm combine or divide existing groups, creating a hierarchical structure that reflects the order in which groups are merged or divided. The bottom up approach, also called the “agglomerative” approach, starts with each object forming a separate group. It successively merges the objects or groups according to some measures like the distance between two centers of two groups and this is done until all of the groups are merged into one, or until a termination condition holds. The top down also called the “divisive” approach, starts with all the objects in the same cluster. In each successive iteration, a cluster is split into smaller clusters accordingly to some measures until eventually each object is in one cluster, or until a termination condition holds.[1]

E. Density Based Clustering Algorithm

Density based clustering algorithm try to find clusters based on density of data points in a region. The key idea of density based clustering is that for each instance of a cluster the neighbourhood of a given radius has to contain at least a minimum number of instances.

F. Cobweb Clustering Algorithm

The COBWEB algorithm was developed by machine learning researchers in the 1980s for clustering objects in a object-attribute data set. The COBWEB algorithm yields a clustering dendrogram called classification tree that characterizes each cluster with a probabilistic description. Cobweb generates hierarchical clustering, where clusters are described probabilistically. COBWEB uses a heuristic evaluation measure called category utility to guide construction of the tree. It incrementally incorporates objects into a classification tree in order to get the highest category utility.[2].

In this paper there is comparison of partitioning and non partitioning based clustering algorithms. Section 1 gives the introduction about various clustering algorithms. Section 2 defines the dataset used. Section 3 describes the basis for algorithm comparison. Section 4 shows the results and section 5 concludes the paper.

II. DATASET USED

For performing the comparison analysis three dataset has been used. Table 1 shows the description of the three dataset i.e number of attributes and number of instances. These datasets has been collected from web (www.cs.waikato.ac. nz/ml/weka/datasets.html). Dataset 1 is in .csv format and dataset 2 and dataset 3 are in. arff format.

TABLE 1 : DATASET USED

Dataset name	No. Of Attributes	No. Of Instances
Dataset 1	5	150
Dataset 2	9	1253
Dataset 3	9	2924

III. BASIS FOR ALGORITHM COMPARISON

The six clustering algorithms are compared according to the following factors

a) Size of dataset
b) Number of clusters
c) Time taken to form clusters

The clustering algorithms are divided into two categories partitioning based and non partitioning based. Firstly partitioning based clustering algorithms and non partitioning based algorithms are compared separately and the results have been drawn. Then the partitioning and non partitioning based algorithms are compared.

A. Comparison of partitioning based and non partitioning based clustering algorithms

Three datasets are applied to the WEKA (3.7.10) and the results related to time taken to form clusters and number of clusters formed are noted. Table 2 describes the time taken to form clusters by partition and non partitioning based clustering algorithms using different size of datasets. The conclusion drawn is as the size of dataset increases time taken to form clusters increases. Farthest First took least time in forming clusters for all the three datasets whereas Expectation maximization took the longest time.

TABLE 2 : TIME TAKEN TO FORM CLUSTERS

Algorithm	Time taken using DATA SET 1	Time taken using DATA SET2	Time taken using DATA SET 3
K-means	0.02	0.03	0.16
EM	1.98	124.95	966.41
Farthest First	0.01	0.03	0.09
Cobweb	0.06	0.72	1.17
Hierarchical	0.16	5.49	27.61
Density	0.02	0.11	0.17

Three datasets have been applied to weka(3.7.10) and the results related to time taken to form clusters formed have been noted. Figure 2 shows the graphical representation of the results.

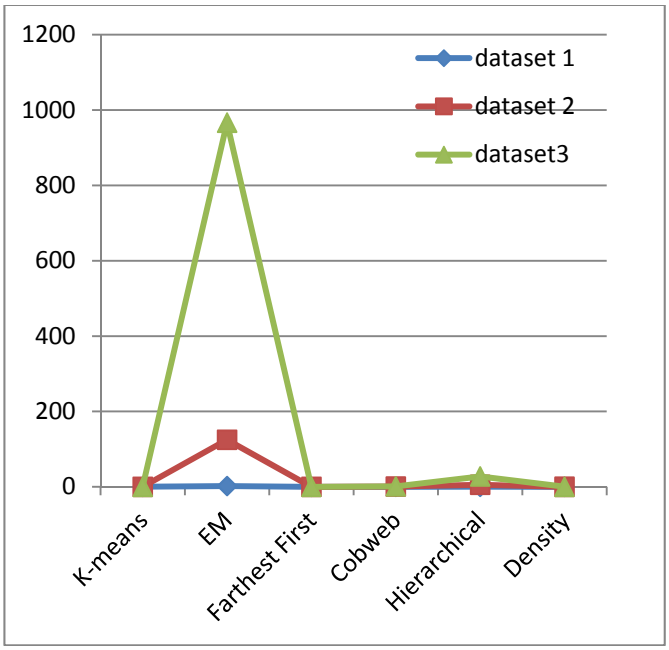

FIGURE 1 – GRAPHICAL REPRESENTAION OF TIME TAKEN TO FORM CLUSTERS

Table 3 describes the number of clusters formed by various algorithms using different size of dataset. In partitioning based clustering algorithms default values for number of clusters have been taken. The value of k is not defined the values are taken by WEKA(3.7.10) itself. Conclusion drawn is number of clusters formed by K-means algorithm and Farthest First algorithm is same for all the datasets, Number of clusters increases as size of dataset increases in expectation maximization. Expectation maximization formed maximum number of clusters.

TABLE 3: NUMBER OF CLUSTERS FORMED USING DIFFERENT SIZE OF DATASET

Algorithm	Number of clusters formed using DATA SET 1	Number of clusters formed using DATA SET 2	Number of clusters formed using DATA SET 3
K-means	2	2	2
Expectation Maximization	7	11	21
Farthest First	2	2	2
Cobweb	4	85	92
Hierarchical	2	2	2
Density	2	2	2

Three datasets have been applied to weka(3.7.10) and the results related to number of clusters formed have been noted. Figure 2 shows the graphical representation of the results.

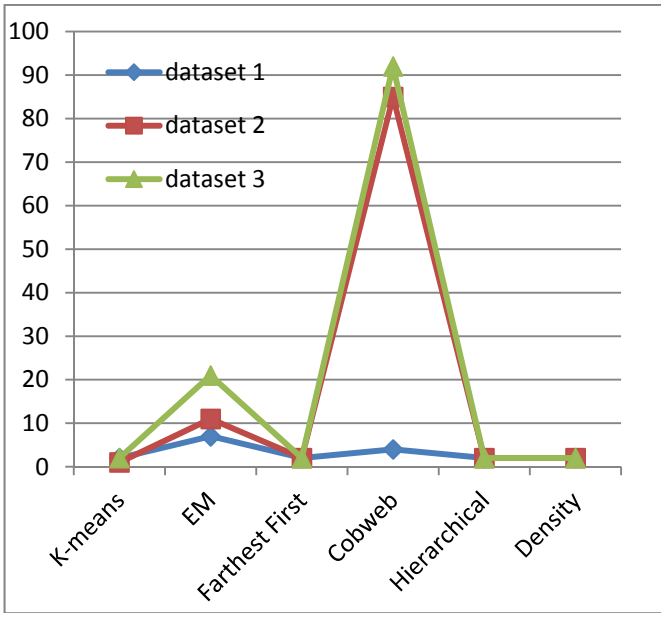

FIGURE 2 : GRAPHICAL REPRESENTATION OF NUMBER OF CLUSTERS FORMED USING DIFFERENT SIZE OF DATASET

IV. RESULT INTERPRETATION

The datasets are applied to the WEKA (3.7.10). As the size of datasets increases time taken to form clusters increases. In partitioning based clustering algorithms Farthest First clustering algorithm took least time in forming clusters whereas Expectation Maximization took maximum time. In non partitioning based clustering algorithm density based clustering algorithm took least time while Hierarchical based clustering algorithm took the maximum time. Farthest First took minimum time in forming clusters in both partitioning and non partitioning based clustering algorithms.

In terms of number of clusters K-means, Farthest First, Hierarchical based and Density based clustering algorithms formed equal number of clusters for all the three datasets. In partitioning based algorithms Expectation Maximization formed maximum number of clusters. In non partitioning based Cobweb clustering algorithm formed maximum clusters.

V. CONCLUSION

A comparative study of clustering algorithms across three different datasets has been performed. The performance of various clustering algorithms is compared based on size of dataset, time taken to form clusters and the number of clusters formed. The experimental results of various clustering algorithms are depicted as graphs. Farthest First algorithm took least time to form clusters and Cobweb algorithm formed maximum number of clusters.

REFERENCES

1. J.Han, M.Kamber and A.K.H Tung “Spatial Clustering Methods in Data Mining : A Survey”
2.. Narendra Sharma , Aman Bajpai and Ratnesh Litoria “Comparison the various Clustering Algorithms of weka tools”, International Journal of Emerging Technology and Advanced Engineering,ISSN:2250-2459,Volume 2, Issue 5, May 2012.
3. Osama Abu Abbas “ Comaprison between Data Clustering Algorithms” The International Arab Journal of Information Technology, Volume 5, July 2008.
4. Richa Loohach and Dr. Kanwal Garg “ Effect of Distance Functions on Simple K-means Clustering Algorithm” International Journal of Computer Applications, Volume 49, July 2012.
5. Sharmila and R.C Mishra “Performance Evaluation of Clustering Algorithms” International Journal of Engineering Trends and Technologies, Volume 4, Issue 7, July 2013