Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 1, January 2014, pg.334 – 340

RESEARCH ARTICLE

An Analysis on Clustering Algorithms in Data Mining

Mythili S[1], Madhiya E[2]

Assistant Professor, PSGR Krishnammal College for Women, Coimbatore[1]

Assistant Professor, PSGR Krishnammal College for Women, Coimbatore[2]

smythisri@gmail.com, emadhiya@gmail.com

Abstract:

Clustering is the grouping together of similar data items into clusters. Clustering analysis is one of the main analytical methods in data mining; the method of clustering algorithm will influence the clustering results directly. This paper discusses the various types of algorithms like k-means clustering algorithms, etc…. and analyzes the advantages and shortcomings of the various algorithms. In each type we can calculate the distance between each data object and all cluster centers in each iteration, which makes the efficiency of clustering is not high. This paper provides a broad survey of the most basic techniques and identifies .This paper also deals with the issues of clustering algorithm such as time complexity and accuracy to provide the better results based on various environments. The results are discussed on huge datasets.

Keywords: *Clustering; datasets; machine-learning; deterministic*

I INTRODUCTION

A large number of clustering definitions can be found in the literature, from simple to elaborate. The simplest definition is shared among all and includes one fundamental concept: the grouping together of similar data items into clusters. Cluster analysis is the organization of a collection of patterns (usually represented as a vector of measurements,or a point in a multidimensional space) into clusters based on similarity. It is important to understand the difference between clustering (unsupervised classification) and discriminant analysis (supervised classification). In supervised classification, we are provided with a collection of *labeled* (preclassified) patterns [1]; the problem is to label a newly encountered, yet

unlabeled, pattern. Typically, the given labeled (*training*) patterns are used to learn the descriptions of classes which in turn are used to label a new pattern. In the case of clustering, the problem is to group a given collection of unlabeled patterns into meaningful clusters. In a sense, labels are associated with clusters also, but these category labels are *data driven*; that is, they are obtained solely from the data.

Clustering is useful in several exploratory pattern-analysis, grouping, decision-making, and machine-learning situations, including data mining, document retrieval, image segmentation, and pattern classification. However, in many such problems, there is little prior information (e.g., statistical models) available about the data, and the decision-maker must make as few assumptions about the data as possible. It is under these restrictions that clustering methodology is particularly appropriate for the exploration of interrelationships among the data points to make an assessment (perhaps preliminary) of their structure. The term "clustering" is used in several research communities to describe methods for grouping of unlabeled data[2]. These communities have different terminologies and assumptions for the components of the clustering process and the contexts in which clustering is used. Thus, we face a dilemma regarding the scope of this survey. The production of a truly comprehensive survey would be a monumental task given the sheer mass of literature in this area. The accessibility of the survey might also be questionable given the need to reconcile very different vocabularies and assumptions regarding clustering in the various communities [3]. The goal of this paper is to survey the core concepts and techniques in the large subset of cluster analysis with its roots in statistics and decision theory. Where appropriate, references will be made to key concepts and techniques arising from clustering methodology in the machine-learning and other communities.

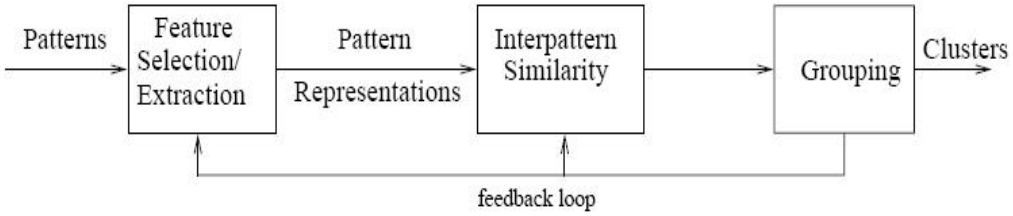

Figure 1: Stages in Clustering

Typical pattern clustering activity involves the following steps [Jain and Dubes 1988]:
(1) pattern representation (optionally including feature extraction and/or selection),
(2) definition of a pattern proximity measure appropriate to the data domain,
(3) clustering or grouping,
(4) data abstraction (if needed), and
(5) assessment of output (if needed).

II LITERATURE SURVEY

Different approaches to clustering data can be described with the help of the hierarchy (other taxonometric representations of clustering methodology are possible; ours is based on the discussion in Jain and Dubes[1988]). At the top level, there is a distinction between hierarchical and partitional approaches (hierarchical methods produce a nested series of partitions, while partitional methods produce only one).

—**Agglomerative *vs.* divisive:** This aspect relates to algorithmic structure and operation. An agglomerative approach[4] begins with each pattern in a distinct (singleton) cluster, and successively merges clusters together until a stopping criterion is satisfied. A divisive method begins with all patterns in a single cluster and performs splitting until a stopping criterion is met.

—**Monothetic *vs.* polythetic**: This aspect relates to the sequential or simultaneous use of features in the clusteringprocess. Most algorithms are polythetic; that is, all features enter into the computation of distances between

patterns [5], and decisions are based on those distances. A simple monothetic algorithm reported in Anderberg [1973] considers features sequentially to divide the given collection of patterns.[14]Here, the collection is divided into two groups using feature *x*1; the vertical broken line V is the separating line. Each of these clusters is further divided independently using feature *x*2, as depicted by the broken lines *H*1 and *H*2. The major problem with this algorithm is that it generates 2*d* clusters where *d* is the dimensionality of the patterns. For large values of *d* (*d* . 100 is typical in information retrieval applications [Salton 1991]),the number of clusters generated by this algorithm is so large that the data set is divided into uninterestingly small and fragmented clusters.

—**Hard *vs.* fuzzy**: A hard clustering algorithm allocates each pattern to a single cluster during its operation and in its output. A fuzzy clustering method assigns degrees of membership in several clusters to each input pattern. A fuzzy clustering can be converted to a hard clustering by assigning each pattern to the cluster with the largest measure of membership.

—**Deterministic *vs.* stochastic**: This issue is most relevant to partitional approaches designed to optimize a squared error function. This optimization can be accomplished using traditional techniques or through a random search[6] of the state space consisting of all possible labelings.

Hierarchical Clustering Algorithms

A representative algorithm of this kind is hierarchical clustering, which is implemented in the popular numerical software MATLAB [15]. This algorithm is an agglomerative algorithm that has several variations depending on the metric used to measure the distances among the clusters. The Euclidean distance is usually used for individual points . There are no known criteria of which clustering distance should be used, and it seems to depend strongly on the dataset. Among the most used variations of the hierarchical clustering based on different distance measures are [16]:

1. Average linkage clustering
The dissimilarity between clusters is calculated using average values.The average distance is calculated from the distance between each point in a cluster and all other points in another cluster. The two clusters with the lowest average distance are joined together to form the newcluster.

2. Centroid linkage clustering
This variation uses the group centroid as the average. The centroid is defined as the center of a cloud of points.

3. Complete linkage clustering (Maximum or Furthest-Neighbor Method)The dissimilarity between 2 groups is equal to the greatest dissimilarity between a member of cluster i and a member of cluster j. This method tends to produce very tight clusters of similar cases.

4. Single linkage clustering (Minimum or Nearest-Neighbor Method): The dissimilarity between 2 clusters is the minimum dissimilarity between members of the two clusters. This method produces long chains whichform loose, straggly clusters.

5. Ward's Method: Cluster membership is assigned by calculating the total sum of squared deviations from the mean of a cluster. The criterion for fusion is that it should produce the smallest possible increase in the error sum of squares.

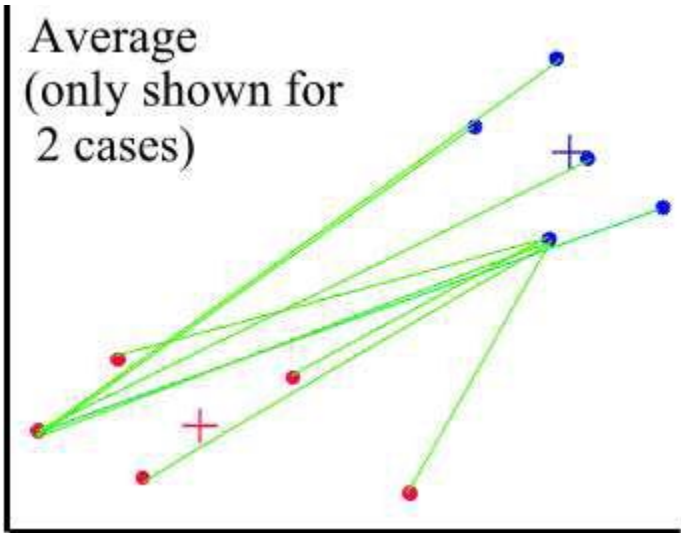

Figure 2: Centroid linkage clustering

Single linkage clustering algorithm Let be D(i; j) the distance between clusters i ,in this case defined as was describe above, and j and N(i) the nearest neighbor of cluster i[7].

1. Initialize as many clusters as data points
2. For each pair of clusters (i; j) compute D(i; j)
3. For each cluster i compute N(i)
4. Repeat until obtain the desired number of clusters

(a) Determine i; j such that D(i; j) is minimized
(b) Agglomerate cluster i and j
(c) Update each D(i; j) and N(i) as necessary

5. End of repeat

Partitional Algorithms

A partitional clustering algorithm obtains a single partition of the data instead of a clustering structure, such as the dendrogram produced by a hierarchical technique. Partitional methods have advantages in applications involving large data sets for which the construction of a dendrogram is computationally prohibitive[8]. A problem accompanying the use of a partitional algorithm is the choice of the number of desired output clusters. A seminal paper [Dubes 1987] provides guidance on this key design decision. The partitional techniques usually produce clusters by optimizing a criterion function defined either locally (on a subset of the patterns) or globally (defined over all of the patterns). Combinatorial search of the set of possible labelings for an optimum value of a criterion is clearly computationally prohibitive. In practice, therefore, the algorithm is typically run multiple times with different starting states, and the best configuration obtained from all of the runs is used as the output clustering.

Nearest Neighbor Clustering

Since proximity plays a key role in our intuitive notion of a cluster, nearest neighbor distances can serve as the basis of clustering procedures. An iterative procedure was proposed in Lu and Fu [1978]; it assigns each unlabeled pattern to the cluster of its nearest labeled neighbor pattern, provided the distance to that labeled neighbor is below a threshold [9]. The process continues until all patterns are labeled or no additional labelings occur. The mutual neighborhood value (described earlier in the context of distance computation) can also be used to grow clusters from near neighbors.

Fuzzy Clustering

Traditional clustering approaches generate partitions; in a partition, each pattern belongs to one and only one cluster. Hence, the clusters in a hard clustering are disjoint. Fuzzy clustering extends this notion to associate each pattern[13] with every cluster using a membership function [Zadeh 1965]. The output of such algorithms is a clustering, but not a partition.

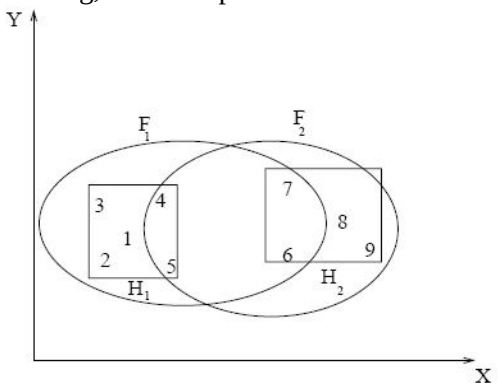

Figure 4: Fuzzy Clustering

III A Comparison of Techniques

In this section we have examined various deterministic and stochastic search techniques to approach the clustering problem as an optimization problem. A majority of these methods use the squared error criterion function. Hence, the partitions generated by these approaches are not as versatile as those generated by hierarchical algorithms. The clusters generated are typically hyperspherical in shape. Evolutionary approaches are globalized search techniques, whereas the rest of the approaches are localized search technique [10]. ANNs and GAs are inherently parallel, so they can be implemented using parallel hardware to improve their speed. Evolutionary approaches are population-based; that is, they search using more than one solution at a time, and the rest are based on using a single solution at a time. ANNs, GAs, SA, and Tabu search (TS) are all sensitive to the selection of various learning/control parameters. In theory, all four of these methods are weak methods [Rich 1983] in that they do not use explicit domain knowledge. An important feature of the evolutionary approaches is that they can find the optimal solution even when the criterion function is discontinuous [11].

K-means algorithm

The K-means algorithm, probably the first one of the clustering algorithms proposed, is based on a very simple idea: Given a set of initial clusters, assign each point to one of them, then each cluster center is replaced by the mean point on the respective cluster [17]. These two simple steps are repeated until convergence. A point is assigned to the cluster which is close in Euclidean distance to the point. Although K-means has the great advantage of being easy to implement, it has two big drawbacks [12]. First, it can be really slow since in each step the distance between each point to each cluster has to be calculated, which can be really expensive in the presence of a large dataset. Second, this method is really sensitive to the provided initial clusters, however, in recent years, this problem has been addressed with some degree of success.

IV Conclusions and Future Directions

Given a data set, the ideal scenario would be to have a given set of criteria choose a proper clustering algorithm to apply. Choosing a clustering algorithm, however, can be a difficult task. Even finding just the most relevant approaches for a given data set is hard. Most of the algorithms generally assume some implicit structure in the data set. The problem, however, is that usually you have little or no information regarding the structure, which is, paradoxically, what you want to uncover. The worst case would be one in which previous information about the data or the clusters is unknown, and a process of trial and error is the best option. However, there are many elements that are usually known, and can be helpful in choosing an algorithm. One of the most important elements is the nature of the data and the nature of the desired cluster. Another issue to keep in mind is the kind of input and tools that the algorithm requires. For example, some algorithms use numerical inputs, some use categorical inputs; some require a definition of a distance or similarity measures for the data. The size of the data set is also important to keep in mind, because most of the clustering algorithms require multiple data scans to achieve convergence, a good discussion of this problems.

An additional issue related to selecting an algorithm is correctly choosing the initial set of clusters. As was shown in the numerical results, an adequate choice of clusters can strongly influence both the quality of and the time required to obtain a solution. Also important is that some clustering methods, such as hierarchical clustering, need a distance matrix which contains all the distances between every pair of elements in the data set. While these methods rely on simplicity, the size of this matrix is of the size m2, which can be prohibitive due to memory constraints, as was shown in the experiments. Recently this issue has been addressed, resulting in new variations of hierarchical and reciprocal nearest neighbor clustering. This paper provides a broad survey of the most basic techniques.

REFERENCES

[1] L. Parsons, E. Haque, and H. Liu, "Subspace clustering forhigh dimensional data: a review," *ACM SIGKDDExplorations Newsletter*, vol. 6, pp. 90-105, 2004.
[2] C. C. Aggarwal, J. Han, J. Wang, and P. S. Yu, "Aframework for projected clustering of high dimensional datastreams," in *Proceedings of the Thirtieth internationalconference on Very large data bases-Volume 30*, 2004, p.863.
[3] R. Agrawal, J. E. Gehrke, D. Gunopulos, and P. Raghavan,"Automatic subspace clustering of high dimensional data fordata mining applications," Google Patents, 1999.
[4] C. Domeniconi, D. Papadopoulos, D. Gunopulos, and S.Ma, "Subspace clustering of high dimensional data," 2004.
[5] X. Z. Fern and C. E. Brodley, "Random projection for highdimensional data clustering: A cluster ensemble approach,"2003, p. 186.
[6] H. P. Kriegel, P. Kröger, and A. Zimek, "Clusteringhigh-dimensional data: A survey on subspace clustering,pattern-based clustering, and correlation clustering," 2009.
[7] R. XU and I. Donald C. Wunsch, *clustering*: A Johnwiley & Sons, INC., Pub, 2008.
[8] Guha, Meyerson, A. Mishra, N. Motwani, and O. C. ."Clustering data streams: Theory and practice ." *IEEETransactions on Knowledge and Data Engineering*, vol. 15,pp. 515-528, 2003.
[9] A. Jain , M. Murty , and p. Flynn " Data clustering: A review.," *ACM Computing Surveys*, vol. 31, pp. 264-323,
1999.
[10] P. C. Biswal, *Discrete Mathematics and Graph Theory*. New Delhi: Prentice Hall of India, 2005.
[11] M. Khalilian, *Discrete Mathematics Structure*. Karaj: Sarafraz, 2004.
[12] K. J. Cios, W. Pedrycz, and R. M. Swiniarsk, "Data miningmethods for knowledge discovery," *IEEE Transactions on Neural Networks*, vol. 9, pp. 1533-1534, 1998.
[13] V. V. Raghavan and K. Birchard, "A clustering strategy based on a formalism of the reproductive process in natural systems," 1979, pp. 10-22.

[14] G. Salton and C. Buckley, "Term weighting approaches in automatic text retrieval," *Inf Process Manage*, vol. 24, pp.513-523, 1987.
[15] G. Salton, "Automatic text processing: the transformation,"*Analysis and Retrieval of Information by Computer*, 1989.
[16] D. Isa, V. P. Kallimani, and L. H. Lee, "Using the self organizing map for clustering of text documents," *Expert Systems With Applications*, vol. 36, pp. 9584-9591, 2009
[17] Shi Na,Liu Xumin, "Research on k-means Clustering Algorithm",IEEE Third International Conference on Intelligent Information Technology and Security Informatics,2010.

