A Comparison Framework for Clustering Algorithms

Ferenc Kovács, Zoltán Dávid and Renáta Iváncsy

Department of Automation and Applied Informatics
Budapest University of Technology and Economics
Goldmann Gy. tér 3, 1117 Budapest, Hungary
{ferenc.kovacs, zoltan.david, renata.ivancsy}@aut.bme.hu

Abstract: Clustering means grouping target objects into different clusters in such a way, that each cluster contains similar objects and the objects in different clusters are dissimilar in a ceratin way. The main question of clustering is how to define the similarity and dissimilarity of the objects, and how to verify that the resulting clusters are good enough for a given purpose. There exist several methods for clustering that have different advantages and disadvantages regarding noise filtering, the shape of the resulting clusters or the number of clusters. Furthermore, validity indices are defined for measuring the goodness of the results. Comparing the clustering algorithms is a difficult task as there are many different approaches and they can have different cluster definitions as well. For this reason a novel comaprison approach is suggested in this paper that introduces a general clustering algorithm model. The main advantage of this model is that it gives a better overview of the clustering problem; furthermore, it divides the whole process into smaller and well separated substeps, that makes easier the investigation of the clustering algorithms.

Keywords: clustering, cluster validity, cluster model

1 Introduction

Clustering is a well known and widely used process for grouping items into different classes while the features of the target classes are not known in advance. In this manner clustering – in contrary to classification – is an unsupervised learning, where the identification of target classes is based on different features of the investigated objects.

There are huge amount of algorithms that have been developed for clustering large number of objects. These algorithms have different basic approaches to the problem, that can be categorizes as follows: (i) partition-based algorithms, (ii) hierarchical algorithms, (iii) density-based algorithms, (iv) grid-based algorithms,

(v) model-based algorithms and (vi) fuzzy algorithms. These algorithms are very different from each other in many aspects, for example some of them can handle noise while other do not care about it. In some cases the distance of the objects is important, in other cases the density or the distribution of the objects is an essential aspect. For this reason, comparing the different methods is a difficult and challenging task.

In this paper we give a novel approach for describing the general behavior of the clustering algorithms. The main contribution of the paper is to set up a framework that defines the different steps, input parameters and constrains for the whole process that can be applied to the current clustering algorithms.

The organization of the paper is as follows. Section 2 describes the basic idea behind the different clustering approaches mentioned before, and states the difficulties of comparison. Section 3 introduces the fundamental elements of the novel model that can be generally used for describing various types of clustering algorithms. In Section 4 the model is validated on some well-known algorithms. Conclusion can be found in Section 5.

2 Classification of Clustering Algorithms

This section describes the basic idea behind the different approaches for clustering huge amount of data.

The aim of the **partition-based** algorithms is to decompose the set of objects into a set of disjoint clusters where the number of the resulting clusters is predefined by the user. The algorithm uses an iterative method, and based on a distance measure of the objects it updates the clusters in every iteration. It is done until any changes can be made. The most representative partition-based clustering algorithms are the k-means and the k-mediod, and in the data mining field the CLARANS [1]. The advantage of the partition-based algorithms that they use an iterative way to create the clusters, but the drawback is, that the number of clusters has to be determined in advance and only hyper sphere like clusters can be discovered by these algorithms.

Hierarchical algorithms provide a hierarchical grouping of the objects. There are two main approaches, the bottom-up and the top-down approach. In case of bottom-up approach, at the beginning of the algorithm each object represents a different cluster and at the end all objects belong to the same cluster. In case of top-down method at the start of the algorithm all objects belong to the same cluster which is split, until each object constitutes different clusters. The steps of the algorithms can be represented by dendrogram. The resulting clusters are determined by cutting the dendrogram in a certain level. One of the key aspects in these kinds of algorithms is the definition of the distance measurements between

the objects and between the clusters. Many definitions can be used to measure distance between the objects, for example Eucledian, City-Block, Minkowski and so on. The clusters distance can be defined as the distance of the two nearest objects in the separated cluster, or as the two farthest or as the distance of the clusters mediods. The drawback of the hierarchical algorithm is that after an object is assigned to a cluster it cannot be modified later. Furthermore, like in partition-based case, only hyper spherical clusters can be obtained. The advantage of the hierarchical algorithms is that the number of the clusters and the cut level of the dendrogam can be defined by the validation indices (correlation, inconsistency measure), which can be defined on the clusters. The best known hierarchical clustering methods are CHAMELEON [2], BIRCH [3] and CURE [4].

Density-based algorithms identify the core objects at first then they expand the clusters from these cores in the following way: if an object is closer to a core object than a given radius then they are in same cluster. The advantage of these type of algorithms is that they can detect arbitrary shaped clusters and it can filter out the noise. DBSCAN [5] and OPTICS [6] are well-known density-based algorithms.

The **grid-based** algorithms use a hierarchical grid structure to decompose the object space into finite number of cells. For each cell statistical information is stored about the objects and the clustering is achieved on these cells. The advantage of this approach is the fast processing time that is in general independent of the number of data objects. Grid-based algorithms are STING [7], CLIQUE [8] and WaweCluster [9].

Model-based algorithms use different distribution models for the clusters which should be verified during the clustering algorithm. A model-based clustering method is MCLUST [10].

Fuzzy algorithms suppose that no hard clusters exist on the set of objects, but one object can be assigned to more than one cluster. The best known fuzzy clustering algorithm is FCM (Fuzzy CMEANS) [11].

As it can be seen well, the different approaches differs really in basic aspects, thus comparing them is a hard task.

3 A Novel Model for Clustering

In this section we suggest a new approach for handling the problem of comparing different clustering methods. The main idea behind this is to set up a framework that can be applied to the most well-known and widely used algorithms. If it is possible to identify general well-separated subtasks of clustering problem the

whole problem statement can be better understand. Furthermore, using this aspect, the different algorithms are more comparable to each other.

3.1 Clustering Process

If a certain algorithm has been chosen for solving a data mining problem, several parameters have to be calculated and various considerations have to be made. First of all the investigated objects have to be converted into the input format of the algorithm. In most cases it means, that the different features of the objects is converted into a numerical format, and a vector is created from these numbers. Each vector represents an object in the vector space. Of course, there are other methods, like binary representation or method for supporting categorical features, as well. In our case we draw our attention only on the cases where the objects are represented with vectors.

After the definition of the input vectors the similarity measure of the vectors has to be defined. In most cases this is some kind of distance measures. The noise model also has to be provided. This model defines the properties of outliers. One of the most critical aspects of the planning phase is to identify the cluster definition that fits to the current data mining problem. During this pahes the main question is what he cluster is. For example there is a method that uses a cluster definition as follows: the narrowest points belong to same cluster or in another the uniform density of the objects is the key feature in a cluster.

When a clustering result is created, some validity functions can be used for evaluating the result of the algorithm. . If it indicates that the clustering result is not acceptable the whole process is repeated.

3.2 Model for Clustering Algorithms

Choosing the right clustering algorithm is fundamental task during the clustering process. Therefore a general model is needed for comparing them.

The novel suggested model works as follows. The input of the whole process is the set of the vector representation of the objects. It is called target items and is denoted in the sequel with D_0. First of all the noise model is applied on these items (phase 1 - P_1), which filters out those objects, that are considered as noise. The remaining items are called filtered items (D_1). The clustering algorithm is applied only these set of items. Each clustering process has an initial clustering phase (P_2) that produces the initial clusters C_0. It can be really simple, like in a hierarchical method, where each item belongs to a different cluster (or even opposite, each item belongs to the same cluster), or it can be a more complicated, like in k-means clustering, where random items are selected as cluster centers, and an item belongs to that cluster center, that is the most narrow to it.

After the initial clusters (C_0) are created a loop is applied on them. By calculating an error function on the clusters (P_3) a so called search space pruning (P_4) is applied on the actual clustering (C_i, i = 0..k, where k states for the number of loops). Search space pruning is a heuristic, that makes the clustering faster by pruning the search space for not to try all the possible cases of different clustering results. In this step the heuristic determines what has to be done with each item. There exist different possibilities, like the item has to be moved to a different cluster, or new cluster centers have to be calculated, or even a new cluster has to be generated and so on.

After the pruning step it has to be decided, wheter the clustering has finished, or not. For this reason some constraints are used. If the clustering has not yet been finished, the whole process is repeated, until the decision is made, that the clusters are good enough along the given parameters and constraints. The output of the model is the resulting clusters (C_r). The different steps of the model can be seen in Figure 1.

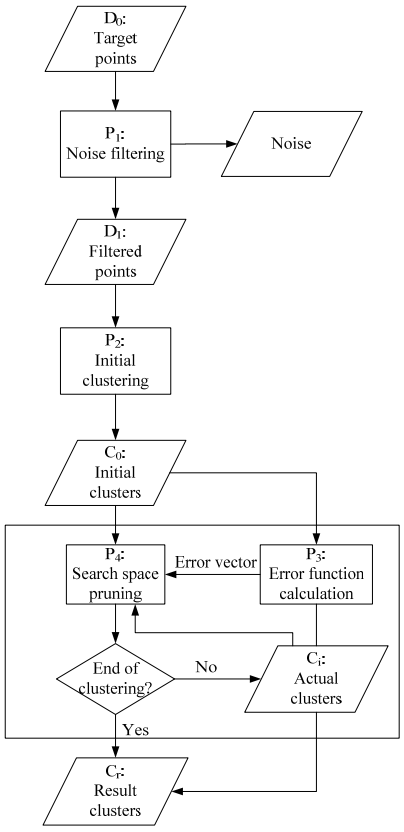

Figure 1
General clustering model

4 Model Verification

In this section we show that the model suggested previously can be applied on the most known and widely used algorithms. The aim of the verification is to identify appropriate sub steps, error function and constraints according to a chosen algorithm. The algorithm, of course, can work in different way but in point of clustering result it is similar to the algorithm specific parameterized framework.

In the following it is presented, how to apply the model to the well-known algorithms of different types of clustering approaches, namely partitioning-based, hierarchical and density-based methods.

4.1 K-means

K-means is a partitioning method. It takes n number of objects as input along with the parameter k, that means the algorithm has to create k number of clusters. Each cluster has to contain at least one object, and each object has to belong to exactly one cluster. In this manner the k-means algorithm does not handle noise, thus in our model applying the noise model is an empty step (P_1 = NULL, $D_0 = D_1$).

K-means algorithm creates an initial partitioning where randomly selected objects are considered as the center of a cluster, and those objects belong to a given cluster, that are the nearest to it considering the distance measure, that is also an input for the algorithm. Thus, in this manner P_2 is applied on the input object set and creates the initial clusters C_0.

Afterwards the K-means algorithm uses an iterative relocation algorithm (P_4), where each object is moved to that cluster that is nearer than the original cluster (P_3). For this reason a new center is calculated (P_3). C_i of the model is the different temporary clusters during the iteration. This process iterates until the criterion function converges. This reallocation means that the k-means algorithm tries to minimize the root mean square of the clusterings.1

4.2 AGNES

AGNES (AGglomerative NESting) [12] is an agglomerative hierarchical clustering method. It initially plasec each point into a cluster of its own (C_0). Like the k-means algorithm it also does not handle noise, thus, in the model (P_1 = NULL, $D_0 = D_1$).

After creating the initial clustering each cluster is investigated and those two clusters are merged (P_4), that are the nearest in a certain manner. The distance of two clusters can be measured by the distance of the two nearest object of the cluster, the distance of the two center object of the cluster, the distance of the two

farther objects of the clusters and so on (P_3). The merging step is executed (merging two clusters results in C_i) until all clusters are merged into one single group. The termination condition can be the number of clusters. The clustering in this phase is the resulting clusters (C_r). This algorithms also try to minimize the RMS of the clustering.

4.3 DBSCAN

The DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a density-based algorithm. It creates clusters based on two parameters, MinPts and **ε**. **ε** determines the investigated environment of an object if MinPts number of other objects can be found in this environment then the object is a core object. Those objects are considered as noise that have less objects in their **ε** neighbourhod than minPts Those object belongs to same cluster that are density-reachable or density-connected to the core object..

The basic steps of the algorithm are as follows:

- Select an arbitrary object o. This is the initial clustering C_0. It means that the selected object belongs to a cluster, and the other ones are not clustered yet.
- Mark all object to this first created cluster with the core object o that are density-reachable from o. When an investigated object is marked as the new object of the cluster, then a new actual clustering is created, C_i.
- When no more objects can be reached, a new object is selected from those points, that are not clustered yet (C_{i+1}).
- When an object has not enough neighbour, mark it as noise (P_1).
- When all objects are marked, finish, C_r.

Conclusion

In this paper the problem of comparing the different existing clustering algorithms is investigated by a developed novel general framework. The main benefit of the framework is to divide the whole process of the clustering problem into smaller, well-defined, good separable sub steps that can be applied on the algorithms. We introduced certain sub steps of the model, and validated it by explaining, how to adopt the model to the best-known clustering algorithms. It can be proved, that the original algorithm and its adopted form provide same clustering result.

References

[1] R. T. Ng and J. Han, “Clarans: A method for clustering objects for spatial data mining,” IEEE Transactions on Knowledge and Data Engineering, 14(5), pp. 1003–1016, 2002

[2] G. Karypis, E.-H. S. Han, and V. K. NEWS, "Chameleon: Hierarchical clustering using dynamic modeling," Computer, 32(8), pp. 68-75, 1999

[3] T. Zhang, R. Ramakrishnan, and M. Livny, "BIRCH: an efficient data clustering method for very large databases," pp. 103-114, 1996

[4] S. Guha, R. Rastogi, and K. Shim, "Cure: An efficient clustering algorithm for large databases," in SIGMOD 1998, Proceedings ACM SIGMOD International Conference on Management of Data, June 2-4, 1998, Seattle, Washington, USA (L. M. Haas and A. Tiwary, eds.), pp. 73-84, ACM Press, 1998

[5] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise.," in KDD, pp. 226–231, 1996

[6] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander, "Optics: ordering points to identify the clustering structure," in SIGMOD '99: Proceedings of the 1999 ACM SIGMOD international conference on Management of data, (New York, NY, USA), pp. 49–60, ACM Press, 1999

[7] W. Wang, J. Yang, and R. Muntz, "Sting: A statistical information grid approach to spatial data mining," 1997

[8] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan, "Automatic subspace clustering of high dimensional data for data mining applications," pp. 94– 105, 1998

[9] G. Sheikholeslami, S. Chatterjee, and A. Zhang, "WaveCluster: A multi-resolution clustering approach for very large spatial databases," in Proc. 24th Int. Conf. Very Large Data Bases, VLDB, pp. 428– 439, 24–27 1998

[10] C. Fraley and A. Raftery, "Mclust: Software for model-based cluster and discriminant analysis," 1999

[11] J. C. Bezdeck, R. Ehrlich, and W. Full, "Fcm: Fuzzy c-means algorithm," Computers and Geoscience, vol. 10, no. 2-3, pp. 191–203, 1984

[12] L. Kaufman, P. J. Rousseeuw, Finding Groups in Data: An introduction to Cluster Analysis, John Wiley and Sons, 1990