Clustering

Clustering is the process of grouping similar data objects into multiple groups. Clustering is an automatic classification, and can automatically find groupings. It is also called data segmentation because data is partitioned based on similarity. It can also be used for outlier detection.

The main requirements for cluster analysis are

1. Scalability
2. Ability to deal with different types of attributes
3. Discovery of clusters with arbitrary shape
4. Requirements for domain knowledge to determine input parameters
5. Ability to deal with noise
6. Incremental clustering and insensitivity to input order
7. Capability of clustering high dimensional data
8. Constraint based clustering
9. Interpretability and usability

Clustering methods are compared using

1. Partitioning criteria
2. Separation of clusters
3. Similarity measures
4. Clustering space

**Partitioning Methods**

Given a set of objects, we make k partitions of the data. Most partitioning methods are distance based. A partitioning method creates an initial partition and then uses an iterative relocation technique to improve the partition by moving objects from one group to another. Examples of this are k-means and k-medoids.

**Hierarchical Methods**

These create a hierarchical decomposition of the given set of objects. They can be

1. Agglomerative
   1. Bottom-up
   2. Starts with each object as a separate group
   3. Successively merge closest objects
2. Divisive
   1. Top down
   2. Starts with one cluster
   3. Divide into smaller clusters

These methods suffer from no backtracking, where once a split is done, it can never be undone.

**Density Based Methods**

These continue growing a cluster as long as the density of the neighborhood exceeds a certain threshold. This allows clusters to take up arbitrary shapes.

**Grid Based Methods**

These quantize the object space into a finite number of cells, forming a grid structure, and the clustering is done on this grid. The main advantage is its fast processing time, which is typically independent of the number of data objects and only on the number of cells in each dimension.

**K-Means Clustering**

Here, we represent k centroids, and associate points to each cluster by measuring the point’s distance to the nearest centroid, and then recomputing the centroid.

The quality of the cluster can be measured by the within-cluster distance, which is the distance between all objects in the cluster and the cluster’s centroid, which is squared, and represents the squared error.

Optimizing this is an NP Hard problem, and so greedy approaches are used.

This algorithm is however, not guaranteed to converge at a global minimum, and may depend on the initial choice of the clusters. The time complexity is given by O(nkt).

K-modes is a variation of this, where it replaces the mean of the clusters by the mode as the recompute.

The major disadvantage of this is to have knowledge of the number of clusters k in advance. Moreover, it is sensitive to noise and outlier points, and cannot approximate for clusters of different shapes.

This is efficient on large sets of data, and can be optimized by a filtering approach to save costs, by creating microclusters and then performing k means on the microclusters.

**K-Medoids**

K means is sensitive to outliers, because such objects are far away from the majority of the data.

So instead of taking mean value, we pick actual objects to represent the clusters, using one representative object per cluster. Each remaining object is assigned to the cluster of which the representative object is most similar. The partitions are done based on the dissimilarities between each object p and the corresponding representative object.

Here, the sum of all distances between each object and the representative object in a cluster is taken as the absolute error.

The representative object is chosen as the median of the cluster.

Partitioning Around Medoids (PAM) is a popular realisation of k-medoids, by tackling the problem in an iterative, greedy way. But here, we try and replace the representative object by other possible objects to increase clustering quality.

This method is more robust than k means in the presence of noise and outliers. However, the complexity is O(k(n-k)2).

For large datasets Clustering LARge Applications (CLARA) is used, where instead of taking the whole dataset, a random sample is taken and PAM is performed on it. It builds clusters from multiple such samples and returns the best clustering.

The effectiveness of CLARA depends on the sample size. CLARANS (CLA based on RANdomized Search) presents a tradeoff between cost and effectiveness of using samples

1. Select k objects as medoids
2. Randomly select a medoid and object y and see if y can replace x as the medoid
3. If yes replace
4. Do this randomized search l times, giving a local optimum
5. Repeat this m times and return the best local optimum

**Density Based Clustering Based on Connected Regions with High Density (DBSCAN)**

This method alleviates the problem of obtaining only spherical clusters. Density Based Spatial Clustering of Applications with Noise finds core objects, that is objects that have dense neighborhoods, and connects these core objects and their neighborhoods to form dense regions as clusters.

The density of an object is measured by the number of objects close to it.

A user specified parameter E > 0 is used to specify the radius of the neighborhood to consider. The E-neighborhood is measured simply as the number of objects in the neighborhood, and it is the space with a radius E centered at the object for an object o.

A density threshold MinPts specifies the density threshold, and an object is a core object only if its E-neighborhood contains at least MinPts objects.

For a core object q and an object p, we say p is directly density-reachable from q if p is within the E-neighborhood of q. But this is not symmetric.

To connect core objects, two objects are density connected with respect to E and MinPts if there is an object q such that both the objects are density-reachable from q. This is an equivalence relation.

If an object does not satisfy the neighborhood or MinPts, it is denoted as a noise point. If not, a new cluster is made for the object.