Clustering*

February 15, 2023

1 What is clustering?

A loose definition of clustering could be "the process of organizing objects into groups whose members are similar in some way". A cluster is therefore a collection of objects which are "similar to each other and are "dissimilar" to the objects belonging to other clusters. Cluster analysis is also used to form descriptive statistics to ascertain whether or not the data consists of a set distinct subgroups, each group representing objects with substantially different properties. The latter goal requires an assessment of the degree of difference between the objects assigned to the respective clusters.

Clustering is a process which partitions a given data set into homogeneous groups based on given features such that similar objects are kept in a group whereas dissimilar objects are in different groups. It is the most important unsupervised learning problem. It deals with finding structure in a collection of unlabeled data.

Central to clustering is to decide what constitutes a good clustering. This can only come from subject matter considerations and there is no absolute "best" criterion which would be independent of the final aim of the clustering. For example, we could be interested in finding representatives for homogeneous groups (data reduction), in finding "natural clusters" and describe their unknown properties ("natural" data types), in finding useful and suitable groupings ("useful" data classes) or in finding unusual data objects (outlier detection).

Centrally, all clustering methods use the same approach i.e. first we calculate similarities and then we use it to cluster the data points into groups or batches. Thus, there are two important components of cluster analysis: the similarity (distance) measure between two data samples and the clustering algorithm.

2 Distance measure

Different formula in defining the distance between two data points can lead to different clustering results. Domain knowledge must be used to guide the formulation of a suitable distance measure for each particular application. For high dimensional data, a popular measure is the Minkowski Metric:

*References

- Andrew Ng. Machine Learning (Stanford course)
- Unknown author(s). Data clustering algorithms
- Satoru Hayasaka. What is Clustering and How Does it Work?. June 21, 2021

$$d\left(x_i, x_j\right) = \left(\sum_{k=1}^{d} \left|x_{i,k} - x_{j,k}\right|^p\right)^{\frac{1}{p}}, \tag{1}$$

where d is the dimensionality of the data. In practice, two special cases of p are commonly used:

- $p = 2$: Euclidean distance
- $p = 1$: Manhattan distance

However, there are no general theoretical guidelines for selecting a measure for any given application. In the case that the components of the data feature vectors are not immediately comparable, such as the days of the week, domain knowledge must be used to formulate an appropriate measure.

3 Overview of clustering algorithms

Clustering algorithms may be classified into the following categories:

- Exclusive clustering. In exclusive clustering data are grouped in an exclusive way, so that a certain datum belongs to only one definite cluster. K-means clustering is one example of the exclusive clustering algorithms.
- Overlapping clustering. The overlapping clustering uses fuzzy sets to cluster data, so that each point may belong to two or more clusters with different degrees of membership. The fuzzy c-means clustering algorithm is one of this type.
- Hierarchical clustering. Hierarchical clustering algorithm has two versions: agglomerative clustering and divisive clustering

  Agglomerative clustering is based on the union between the two nearest clusters. The beginning condition is realized by setting every datum as a cluster. After a few iterations it reaches the final clusters wanted. Basically, it works in the bottom-up manner.

  Divisive clustering starts from one cluster containing all data items. At each step, clusters are successively split into smaller clusters according to some dissimilarity. Basically, it works in the top-down manner.
- Density based clustering. Density based clustering algorithm has played a vital role in finding non linear shape structures based on the density. DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is the most widely used density based algorithm.
- Probabilistic clustering. Probabilistic clustering, e.g. Mixture of Gaussian or Expectation Maximization, uses a completely probabilistic approach.

Based on their applicability to different types of data, clustering methods can be classified into:

- Linear clustering algorithms, such as k-means, fuzzy c-means[1], hierarchical clustering, and Gaussian (EM).
- Non-linear clustering algorithms, such as MST (minimum spanning tree) based clustering, kernel k-means clustering[2], and density based clustering.

4 K-Means clustering

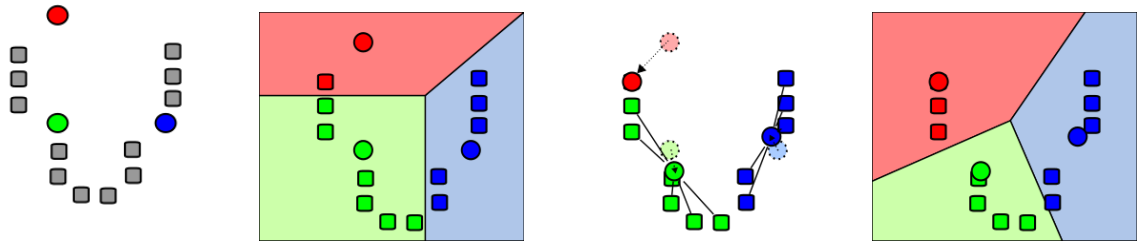
k-means is one of the simplest unsupervised learning algorithms that solves the clustering problem. The procedure follows a simple and easy way with the objective to classify a given data set $S = \{s_1, s_2, ...s_N\}$ into a certain number of clusters (assume initial clusters) fixed a priori. The idea is to define initial centroids, one for each cluster $c_i (1 \leq i \leq k)$. The procedure is:

1) Initialize the k clusters $\ell^o = \{c_1^0, c_2^0 \dots c_k^0\}$ in a way such that the initial centroids are placed as far as possible from each other.
2) Calculate the centroids of the clusters: $u_j^i = \frac{1}{|c_j^i|} \sum_{x \in c_j^i} x$, where $j = 1, ..., k$ and i denotes the i-th iteration.
3) Take each point belonging to a given data set and associate it to the nearest centroid:

$$\begin{aligned} c_j^{i+1} &= \left\{ x \mid d\left(x, u_j^i\right) \leq d\left(x, u_{j'}^i\right), \forall j', 1 \leq j' \leq k \right\} \\ \ell^{i+1} &= \left\{ c_j^{i+1} \mid 1 \leq j \leq k \right\} \end{aligned} . \tag{2}$$

4) Repeat steps 2 and 3 until no more changes can be made to the clusters, i.e., $\ell^{i+1} = \ell^i$. In other words, centroids do not move any more.

Figure 1 demonstrates the k-means clustering algorithm.



(a) (b) (c) (d)

Figure 1: Demonstration of the k-means clustering algorithm. (a) k initial "means" (in this case k=3) are randomly generated within the data domain (shown in color). (b) k clusters are created by associating every observation with the nearest mean. The partitions here represent the Voronoi diagram generated by the means. (c) The centroid of each of the k clusters becomes the new mean. (d) Steps 2 and 3 are repeated until convergence has been reached.

[1]Fuzzy c-means create k clusters and then assign each data to each cluster, but their will be a factor that will define how strongly the data belongs to that cluster.

[2]The kernel k-means clustering algorithm applies the same trick as k-means but with one difference that here in the calculation of distance, kernel method is used instead of the Euclidean distance.

Objective function. The above steps of k-means can be formulated as an optimization problem, i.e., minimizing an objective function, in this case a squared error function:

$$J = \sum_{i=1}^{k} \sum_{x \in c_i} \|x - u_i\|^2 , \tag{3}$$

where J measures the sum of squared distances between each data sample x and the cluster centroid u_i to which x has been assigned.

Convergence. It can be shown that k-means is exactly coordinate descent on J. Specifically, the inner-loop (i.e., step [2] and step [3]) of k-means repeatedly minimizes J with respect to the cluster centroids u while holding the the assignments c fixed, and then minimizes J with respect to the assignments c while holding the cluster centroids u fixed. Thus, J must monotonically decrease, and the value of J must converge. In other words, which means that updating each centroid to the sample mean of its cluster will never increase the value of J, so each iteration therefore improves on the previous result and k-means clustering usually converges quickly.

Local minima. Although the k-means algorithm will always converge/terminate, the algorithm does not necessarily find the most optimal configuration, corresponding to the global minimum of the objective function. It might get stuck in a local minimum.

The algorithm is also significantly sensitive to the initial randomly selected cluster centers. To get out of local minimum, the k-means algorithm can be run multiple times from different initial clustering, so as to empirically determine good clusters. Yet, even then, k-means may produce questionable results when seeded with inappropriate centroids. Arbitrary initialization are considered harmful and intelligent initializations are the topic of ongoing research.

Advantages:

- Fast, robust and easier to understand.
- Relatively efficient: $O(tknd)$, where n is the number of objects, k is the number of clusters, d is the dimension of each object, and t is the number of iterations. Normally, $k, t, d \ll n$.
- Gives best result when data set are distinct or well separated from each other.

Disadvantages:

- The learning algorithm requires a priori specification of the number of cluster centers.
- The use of Exclusive Assignment - If there are two highly overlapping data then k-means will not be able to resolve that there are two clusters.
- The learning algorithm is not invariant to non-linear transformations i.e. with different representation of data we get different results (data represented in form of cartesian coordinates and polar coordinates will give different results).
- Euclidean distance measures can unequally weight underlying factors.
- The learning algorithm provides the local optima of the squared error function.

- Randomly choosing of the cluster center cannot lead us to the fruitful result.
- Applicable only when mean is defined i.e. fails for categorical data.
- Unable to handle noisy data and outliers.
- Algorithm fails for non-linear data set (see Figure 2 for two non-linear examples of data sets where the k-means algorithm fails).

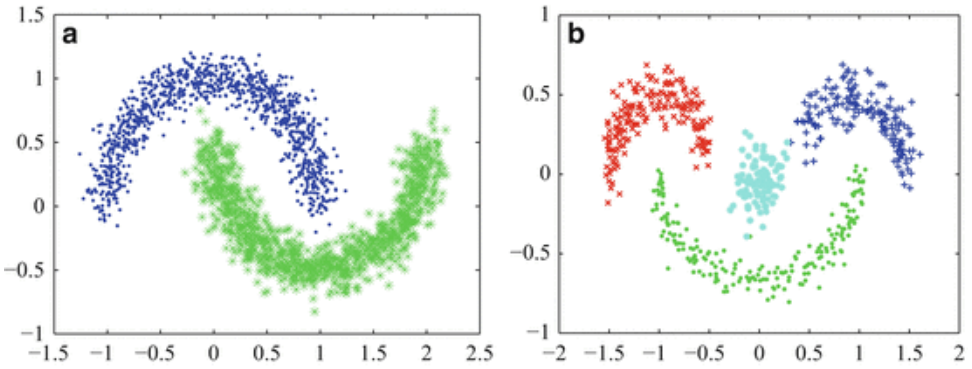


Figure 2: Two examples of data sets consisting of nonlinear separable clusters.

5 Hierarchical clustering

This algorithm works by grouping the data one by one on the basis of the nearest distance measure of all the pairwise distance between the data points. Again distance between the data point is recalculated but which distance to consider when the groups has been formed? For this there are many available methods. Some of them are: 1) single-nearest distance or single linkage. 2) complete-farthest distance or complete linkage. 3) average distance or average linkage.

Given a set of N objects $S = \{s_1, s_2, ..., s_N\}$ to be clustered and a function of distance between two clusters c_i and c_j, build a hierarchy tree on S such that for every $c_i, c_j \in S$, $c_i \cap c_j = \emptyset$. The basic process of hierarchical clustering is as follows:

1) Start by assigning each object to a cluster $c_i = s_i (i = 1, ..., N)$, so that if you have N objects, you have N clusters $\ell = \{c_1, c_2, ..., c_N\}$, each containing just one item.

2) Find the pair of clusters (c_i, c_j) such that $D(c_i, c_j) \leq D(c_{i'}, c_{j'})$, $\forall c_{i'} \neq c_{j'} \in \ell$ and merge them into a single cluster $c_k = c_i \cup c_j$. Delete c_i and c_j from ℓ and insert c_k into ℓ so that now you have one cluster less.

3) Compute distances (similarities) between the new cluster and each of the old clusters.

4) Repeat steps 2) and 3) until all items are clustered into a single cluster of size N.

This way we go on grouping the data until one cluster is formed. Now on the basis of dendrogram graph we can calculate how many number of clusters should be actually present. An example of how the hierarchical algorithm leads to long clusters is shown in Figure 3.

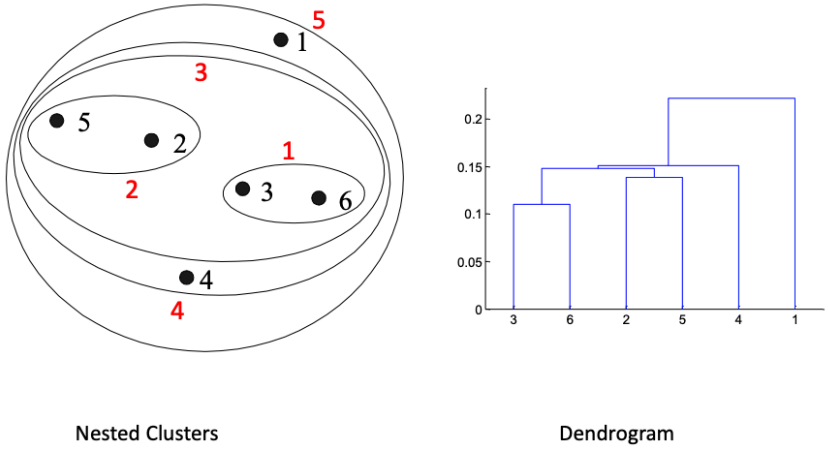


Figure 3: An example of hierarchical clustering.

Step 3) in the hierarchical algorithm can be done in different ways, which is what distinguishes *single-linkage* from *complete-linkage* and *average-linkage* clustering.

- **Single-linkage clustering**: the distance between one cluster and another cluster is equal to the *shortest* distance from any member of one cluster to any member of the other cluster, i.e., $D(c_i, c_j) = \min d(a, b), \forall a \in c_i$ and $b \in c_j$. It is obvious that:

$$D(c_k, c_l) = \min \{D(c_i, c_l), D(c_j, c_l)\}, \text{ for } c_k = c_i \cup c_j. \tag{4}$$

- **Complete-linkage clustering**: the distance between one cluster and another cluster is equal to the *greatest* distance from any member of one cluster to any member of the other cluster, i.e., $D(c_i, c_j) = \max d(a, b), \forall a \in c_i$ and $b \in c_j$.

- **Average-linkage clustering**: the distance between one cluster and another cluster is equal to the *average* distance from any member of one cluster to any member of the other cluster, i.e., $D\left(c_i, c_j\right) = \frac{1}{|c_i||c_j|} \sum_{a \in c_i, b \in c_j} d(a, b)$. It is obvious that

$$D\left(c_k, c_l\right) = \frac{|c_i|}{|c_k|} D\left(c_i, c_l\right) + \frac{|c_j|}{|c_k|} D\left(c_j, c_l\right), \text{ for } c_k = c_i \cup c_j. \tag{5}$$

Advantages:

- No a priori information about the number of clusters required.
- Easy to implement and gives best result in some cases.

Disadvantages:

- Algorithm can never undo what was done previously.
- Time complexity of at least $O(n^2 \log n)$ is required, where n is the number of data points.
- Based on the type of distance matric chosen for merging, algorithms can suffer with one or more of the following:

  - Sensitivity to noise and outliers.
  - Breaking large clusters.
  - Difficulty handling different sized clusters and convex shapes.

- No objective function is directly minimized.
- Sometimes it is difficult to identify the correct number of clusters by the dendrogram.

6 Density based clustering algorithm

Density-Based clustering refers to unsupervised learning methods that identify distinctive groups/clusters in the data, based on the idea that a cluster in data space is a contiguous region of high point density, separated from other such clusters by contiguous regions of low point density. These methods play a vital role in finding non-linear shape structures based on the density.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is the most widely used density based algorithm. It uses the concept of density reachability and density connectivity:

- **Density reachability**: a point p is said to be density reachable from a point q if point p is within ϵ distance from point q and q has at least $minPts$ points in its neighbors which are within distance ϵ.
- **Density connectivity**: a point p and q are said to be density connected if there exist a point r which has at least $minPts$ points in its neighbors and both the points p and q are within the ϵ distance from r. This is chaining process. So, if q is neighbor of r, r is neighbor of s, s is neighbor of t which in turn is neighbor of p implies that q is neighbor of p. So density connectivity involves a transitivity based chaining-approach to determine whether points are located in a particular cluster.

Let $X = \{x_1, x_2, ..., x_N\}$ be the set of data points. DBSCAN requires two parameters: ϵ and $minPts$. ϵ is a distance measure that will be used to locate the points in the neighborhood of any point, and $minPts$ is the minimum number of points required to form a cluster. The basic process of DBSCAN clustering is as follows:

1) Start with an arbitrary starting point that has not been visited.
2) Extract the neighborhood of this point using ϵ (i.e., all points within the ϵ distance are neighborhood).
3) If there are at least $minPts$ neighborhood points around this point, the clustering process starts and all these points are considered to be part of the same cluster, and these points are marked as visited. Otherwise, this point is labeled as noise (Later this point can become the part of the cluster).
4) If a point is found to be a part of the cluster then its ϵ neighborhood is also the part of the cluster and the above procedure from step 2) is repeated for all ϵ neighborhood points. This is repeated until all points in the cluster is determined.

5) A new unvisited point is retrieved and processed, leading to the discovery of a further cluster or noise.

6) This process continues until all points are marked as visited.

Advantages:

- Does not require a-priori specification of number of clusters.
- Able to identify noise data while clustering.
- DBSCAN algorithm is able to find arbitrarily size and arbitrarily shaped clusters.

Disadvantages:

- DBSCAN algorithm fails in case of varying density clusters.
- Fails in case of neck type of dataset.

7 Applications of clustering algorithms

In identifying cancerous data. Clustering algorithm can be used in identifying the cancerous data set. Initially we take known samples of cancerous and non cancerous data set. Label both the samples data set. We then randomly mix both samples and apply different clustering algorithms into the mixed samples data set (this is known as learning phase of clustering algorithm) and accordingly check the result for how many data set we are getting the correct results (since this is known samples we already know the results beforehand) and hence we can calculate the percentage of correct results obtained. Now, for some arbitrary sample data set if we apply the same algorithm we can expect the result to be the same percentage correct as we got during the learning phase of the particular algorithm. On this basis we can search for the best suitable clustering algorithm for our data samples.

It has been found through experiment that cancerous data set gives best results with unsupervised non linear clustering algorithms and hence we can conclude the non linear nature of the cancerous data set.

In search engines. Clustering algorithm is the backbone behind the search engines. Search engines try to group similar objects in one cluster and the dissimilar objects far from each other. It provides result for the searched data according to the nearest similar object which are clustered around the data to be searched. Better the clustering algorithm used, better are the chances of getting the required result on the front page. Hence, the definition of similar object play a crucial role in getting the search results, better the definition of similar object better the result is. Most of the brainstorming activities needs to be done for defining the criteria to be used for similar object.

In academics and education. The ability to monitor the progress of students' academic performance has been the critical issue for the academic community of higher learning. Clustering algorithm can be used to monitor the students' academic performance. Based on the students' score they are grouped into different-different clusters (using k-means for example), where the clusters denote the different levels of performance. By knowing the number of students' in each cluster we can know the average performance of a class as a whole.

8 Summary

With the advent of many data clustering algorithms in the recent few years and its extensive use in wide variety of applications, including image processing, computational biology, mobile communication, medicine and economics, has lead to the popularity of this algorithms. Main problem with the data clustering algorithms is that it cannot be standardized. Algorithm developed may give best result with one type of data set but may fail or give poor result with data set of other types. Although there has been many attempts for standardizing the algorithms which can perform well in all case of scenarios but till now no major accomplishment has been achieved. Many clustering algorithms have been proposed so far. However, each algorithm has its own merits and demerits and cannot work for all real situations.

9 Useful interactive tools

k-means:
https://www.naftaliharris.com/blog/visualizing-k-means-clustering/
DBSCAN:
https://www.naftaliharris.com/blog/visualizing-dbscan-clustering/