Clustering

15th Feb 2021

Contents

0.1 Introduction to Clustering

Clustering is the task of dividing the population or data points into a number of groups such that data points in the same groups are more similar to other data points in the same group than those in other groups. In simple words, the aim is to segregate groups with similar traits and assign them into clusters.

It is basically a type of unsupervised learning method . An unsupervised learning method is a method in which we draw references from datasets consisting of input data without labelled responses. Generally, it is used as a process to find meaningful structure, explanatory underlying processes, generative features, and groupings inherent in a set of examples.

Clustering is the task of dividing the population or data points into a number of groups such that data points in the same groups are more similar to other data points in the same group and dissimilar to the data points in other groups. It is basically a collection of objects on the basis of similarity and dissimilarity between them.

Clustering is very much important as it determines the intrinsic grouping among the unlabeled data present. There are no criteria for a good clustering. It depends on the user, what is the criteria they may use which satisfy their need. For instance, 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). This algorithm must make some assumptions which constitute the similarity of points and each assumption make different and equally valid clusters.

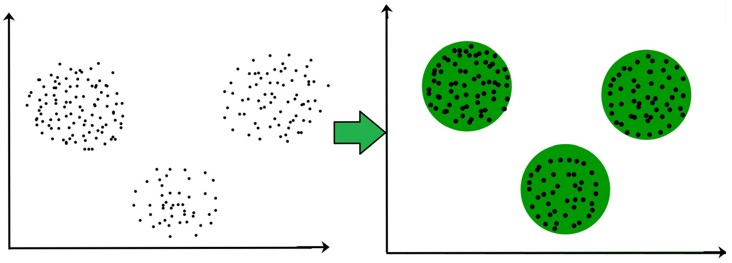


Figure 1: Clustering

0.2 Types of Clustering

Broadly speaking, clustering can be divided into two subgroups :

- **Hard Clustering**: In hard clustering, each data point either belongs to a cluster completely or not. For example, in the above example each customer is put into one group out of the 10 groups.
- **Soft Clustering**: In soft clustering, instead of putting each data point into a separate cluster, a probability or likelihood of that data point to be in those clusters is assigned. For example, from the above scenario each costumer is assigned a probability to be in either of 10 clusters of the retail store.

0.3 Types of clustering algorithms

Since the task of clustering is subjective, the means that can be used for achieving this goal are plenty. Every methodology follows a different set of rules for defining the 'similarity' among data points. In fact, there are more than 100 clustering algorithms known. But few of the algorithms are used popularly, let's look at them in detail:

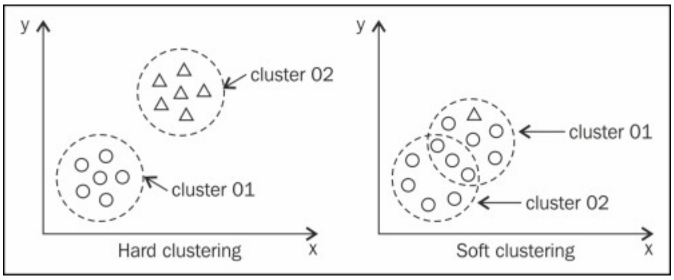


Figure 2: Hard and Soft clustering

- **Connectivity models**: As the name suggests, these models are based on the notion that the data points closer in data space exhibit more similarity to each other than the data points lying farther away. These models can follow two approaches. In the first approach, they start with classifying all data points into separate clusters then aggregating them as the distance decreases. In the second approach, all data points are classified as a single cluster and then partitioned as the distance increases. Also, the choice of distance function is subjective. These models are very easy to interpret but lacks scalability for handling big datasets. Examples of these models are hierarchical clustering algorithm and its variants.

- **Centroid models**: These are iterative clustering algorithms in which the notion of similarity is derived by the closeness of a data point to the centroid of the clusters. K-Means clustering algorithm is a popular algorithm that falls into this category. In these models, the no. of clusters required at the end have to be mentioned beforehand, which makes it important to have prior knowledge of the dataset. These models run iteratively to find the local optima.

- **Distribution models**: These clustering models are based on the notion of how probable is it that all data points in the cluster belong to the same distribution (For example: Normal, Gaussian). These models often suffer from overfitting. A popular example of these models is Expectation-maximization algorithm which uses multivariate normal distributions.

- **Density Models**: These models search the data space for areas of varied density of data points in the data space. It isolates various different density regions and assign the data points within these regions in the same cluster. Popular examples of density models are DBSCAN and OPTICS

.

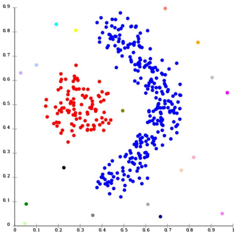


Figure 3: Connectivity clustering

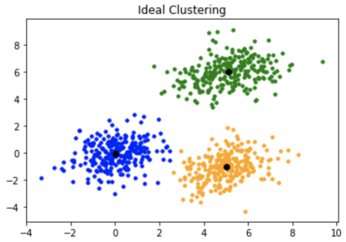


Figure 4: Centroid clustering

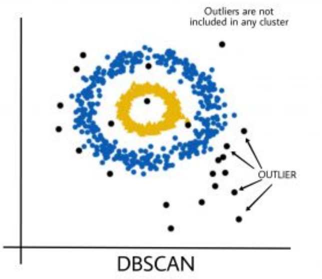


Figure 5: Density clustering

0.4 Applications of Clustering

Clustering has a large no. of applications spread across various domains. Some of the most popular applications of clustering are:

- **Recommendation engines**

A recommendation engine is a system that suggests products, services, information to users based on analysis of data. Notwithstanding, the recommendation can derive from a variety of factors such as the history of the user and the behaviour of similar users.

- **Market segmentation**

  Market segmentation is the process of dividing a target market into smaller, more defined categories. It segments customers and audiences into groups that share similar characteristics such as demographics, interests, needs, or location.

- **Social network analysis**

  Social network analysis (SNA) is the process of investigating social structures through the use of networks and graph theory. It characterizes networked structures in terms of nodes (individual actors, people, or things within the network) and the ties, edges, or links (relationships or interactions) that connect them.

- **Search result grouping**

  Result Grouping groups documents with a common field value into groups and returns the top serch result for each group.

- **Medical imaging**

  Clustering and classification methods are known as the learning methods, which do not use any spatial or shape information.

- **Image segmentation**

  image segmentation is the process of partitioning a digital image into multiple segments (sets of pixels, also known as image objects). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze.[1][2] Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain characteristics.

- **Anomaly detection**

  Anomaly detection (aka outlier analysis) is a step in data mining that identifies data points, events, and/or observations that deviate from a dataset's normal behavior. Anomalous data can indicate critical incidents, such as a technical glitch, or potential opportunities, for instance a change in consumer behavior. Machine learning is progressively being used to automate anomaly detection.

0.5 Introduction to Agglomerative Clustering

In data mining and statistics, hierarchical clustering (also called hierarchical cluster analysis or HCA) is a method of cluster analysis which seeks to build a hierarchy of clusters. Strategies for hierarchical clustering generally fall into two types:

- **Agglomerative**: This is a ”bottom-up” approach: each observation starts in its own cluster, and pairs of clusters are merged as one moves up the hierarchy.
- **Divisive**: This is a ”top-down” approach: all observations start in one cluster, and splits are performed recursively as one moves down the hierarchy.

In general, the merges and splits are determined in a greedy manner. The results of hierarchical clustering are usually presented in a dendrogram. For example, suppose this data is to be clustered,

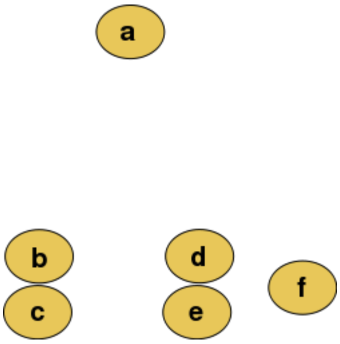


Figure 6: Raw Data

and the Euclidean distance is the distance metric. The hierarchical clustering dendrogram would be as such:

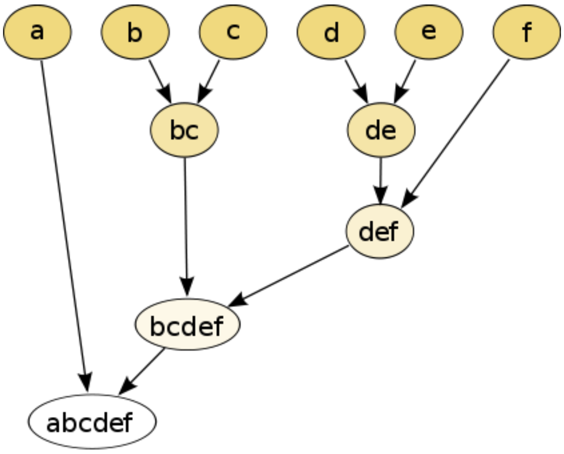


Figure 7: Dendogram

Cutting the tree at a given height will give a partitioning clustering at a selected precision. In this example, cutting after the second row (from the top) of the dendrogram will yield clusters a b c d e f. Cutting after the third row will yield clusters a b c d e f, which is a coarser clustering, with a smaller number but larger clusters.

This method builds the hierarchy from the individual elements by progressively merging clusters. In our example, we have six elements a b c d e and f. The first step is to determine which elements to merge in a cluster. Usually, we want to take the two closest elements, according to the chosen distance.

Optionally, one can also construct a distance matrix at this stage, where the number in the i-th row j-th column is the distance between the i-th and j-th elements. Then, as clustering progresses, rows and columns are merged as the clusters are merged and the distances updated. This is a common way to implement this type of clustering, and has the benefit of caching distances between clusters. A simple agglomerative clustering algorithm is described in the single-linkage clustering page; it can easily be adapted to different types of linkage.

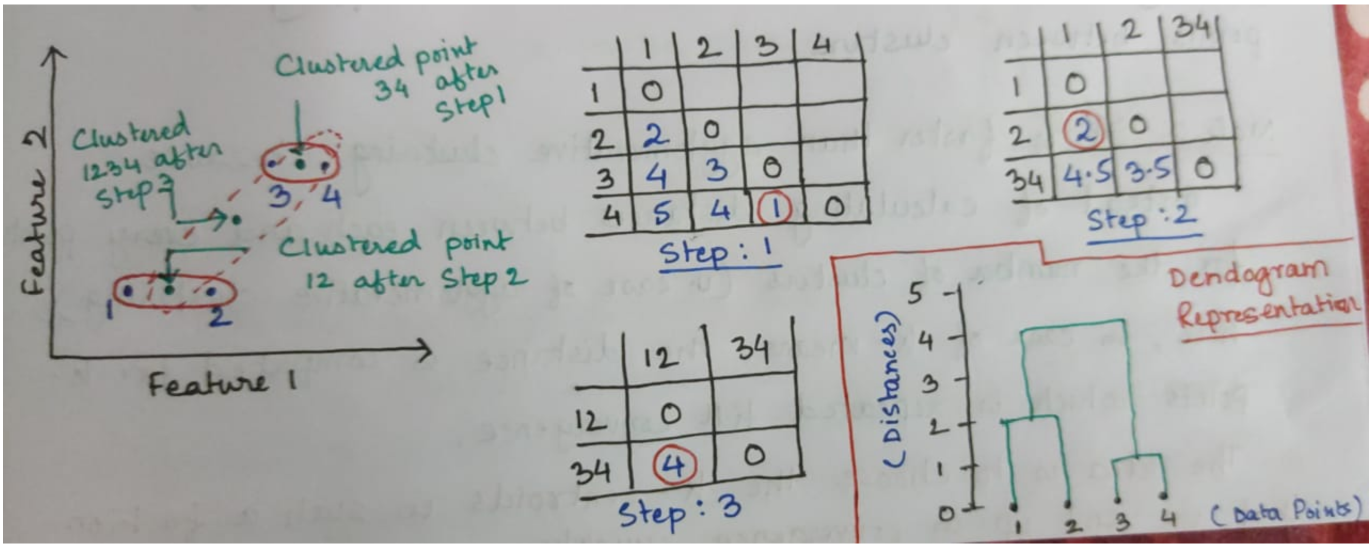
Suppose we have merged the two closest elements b and c, we now have the following clusters a, b, c, d, e and f, and want to merge them further. To do that, we need to take the distance between a and b c, and therefore define the distance between two clusters. Usually the distance between two clusters A and B is one of the following:

- The maximum distance between elements of each cluster (also called complete-linkage clustering):
- The minimum distance between elements of each cluster (also called single-linkage clustering)
- The mean distance between elements of each cluster (also called average linkage clustering, used e.g. in UPGMA):
    - The sum of all intra-cluster variance.
    - The increase in variance for the cluster being merged (Ward's method)
    - The probability that candidate clusters spawn from the same distribution function (V-linkage).

In case of tied minimum distances, a pair is randomly chosen, thus being able to generate several structurally different dendrograms. Alternatively, all tied pairs may be joined at the same time, generating a unique dendrogram.

One can always decide to stop clustering when there is a sufficiently small number of clusters (number criterion). Some linkages may also guarantee that agglomeration occurs at a greater distance between clusters than the previous agglomeration, and then one can stop clustering when the clusters are too far apart to be merged (distance criterion). However, this is not the case of, e.g., the centroid linkage where the so-called reversals (inversions, departures from ultrametricity) may occur.

0.6 Agglomerative Clustering Algorithm



	1	2	3	4
1	0			
2	2	0		
3	4	3	0	
4	5	4	1	0

	1	2	34
1	0		
2	2	0	
34	4.5	3.5	0

	12	34
12	0	
34	4	0

Figure 8: Cluster Formation

Algorithm Agglomerative

1. For a given dataset, we compute distances between each and every points. By this way, we create a distance matrix (n x n) of n data points.**Note:** The matrix will be a symmetric matrix along its diagonal, with the diagonal elements as 0 - distance between each point to itself.
2. compute the minimum distance in the distance matrix and cluster the points as a single cluster
3. Repeat the Step 1 and Step 2 thereby creating a dendogram.

0.7 Clustering is NOT Classification

Clustering is unsupervised learning based in parameters in the dataset. Clusters are formed based on similarity in that dataset. Whereas, in classification, we try to associate data points to a label. However, in real world machine learning approaches, clustering and classification can go hand in hand. For example - we may want to cluster our data before classification.

0.8 Introduction to K-Means Clustering

K-means clustering is an unsupervised machine learning algorithm, used to group similar data points together and discover underlying patterns. To achieve this objective, K-means looks for a fixed number (K) of clusters in a dataset.

Some Notations:
K = number of clusters
m = number of examples
$c^{(i)}$ = index of cluster (1,2,..,K) to which example $x^{(i)}$ is currently assigned.
μ_k = cluster centroid k ($\mu_k \in R^n$)

Let the dataset be: $\{x^{(1)}, x^{(2)}, ..., x^{(m)}\}$

Optimization Objective

$$J(c^{(1)}, .., c^{(m)}, \mu_1, .., \mu_K) = \frac{1}{m}\sum_{i=1}^{m}(|| \; x^{(i)} - \mu_j \; ||)^2$$

Here, we have to find the values of parameters $c^{(1)}, .., c^{(m)}$ and $\mu_1, .., \mu_k$ such that the cost function J is minimized

Algorithm KMeans
The K-means clustering algorithm is as follows:

1. Initialize cluster centroids $\mu_1, \mu_2, ..., \mu_k \in R^n$
2. Repeat until convergence
   (a) For every i, set $c^{(i)} := min_j(|| \; x^{(i)} - \mu_j \; ||)^2$
   (b) For each j, set $\mu_j := \frac{\sum_{i=1}^{m}\{c^{(i)}=j\}x^{(i)}}{\sum_{i=1}^{m}\{c^{(i)}=j\}}$

Step a: **Cluster assignment step**
This step is minimizing J(..) w.r.t $c^{(1)}, c^{(2)}, .., c^{(n)}$ holding $\mu_1, \mu_2, ..., \mu_k$ fixed.

Step b: **Move centroid step**
Here, we choose μ that minimizes J(..) w.r.t $\mu_1, \mu_2, .., \mu_k$

0.9 Initializing number of clusters

The most obvious condition can be k < m,
One way to initialize k-means is to pick k distinct random integers $i_1, i_2, .., i_k$ from $\{1, .., m\}$.Set $\mu_1 = x^{(i_1)}, \mu_2 = x^{(i_2)}, .., \mu_k = x^{(i_k)}$

In order to get a good set of clusters, K-means algorithm should be run several times, typically 50 to 1000 times. The modified algorithm will be as follows:

Modified k-means:
For i=1 to 100

1. Randomly initialize k-means
2. Run k-means. Get $c^{(1)}, .., c^{(m)}, \mu_1, .., \mu_k$
3. compute cost function J($c^{(1)}, .., c^{(m)}, \mu_1, ..\mu_k$)

After running k-means a 100 times, pick the clustering that gave the lowest cost J($c^{(1)}, .., c^{(m)}, \mu_1, .., \mu_k$)

The time complexity of kmeans algorithm is O(K × m × d × I), where,
K = number of clusters

m = total number of data points
d = dimensions of each data point
I = number of iterations

0.10 Choosing the number of clusters

There is no thumb rule for choosing the number of clusters. Although one way is to try for different values of k and plot a graph as one shown below:

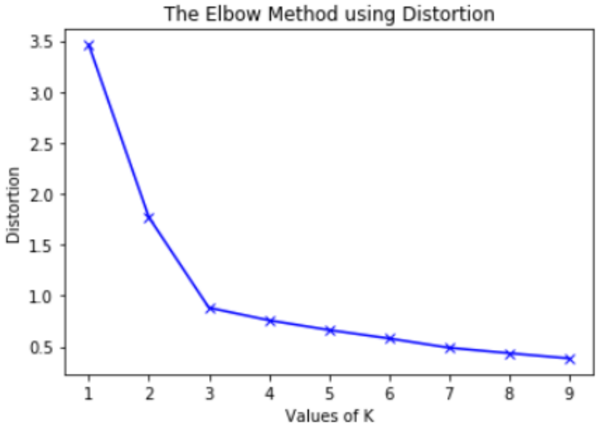


Figure 9: Elbow method

From the above figure we infer that the cost function decreases steeply in the beginning and then it becomes almost constant. Here, k=3 can be a good choice for number of clusters.

If while running k-means algorithm the cost for k=3 is less than that when k=5, then it implies that, when k=5 k-means got stuck in a bad local minima. Therefore, to avoid such cases, k-means should be run with multiple random initializations.

0.11 KMeans++

Some of the drawbacks of kmeans clustering are:

- No point associated with a cluster
- More than one centroids might be initialized into the same cluster
- More than one cluster might end up linked with a single centroid

In order to overcome these limitations due to poor initialization of centroids, KMeans++ was introduced,in 2007 by David Arthur and Sergei Vassilvitskii, as an algorithm for choosing the initial centroids.

The KMeans++ algorithm is defines as follows:

Algorithm KMeans++

```
1. Randomly choose one of the observation to be a cluster center
2. For each observation x find
    (a) d(x,$\mu_i$) - distance of x from each of the available centroids
    (b) d(x) - min{d(x,$\mu_i$)} $\forall$ i $\in$ centroid set
3. Choose the next cluster centroid by:
    (a) Create a distribution by doing $(d(x))^2$
    (b) Select the next cluster centroid by sampling from $(d(x))^2$
4. Repeat steps 2 and 3 till required number of clusters are formed
```

The time complexity remains same i.e O(K × m × d × I)

0.12 Applications of k-means clustering

Some of the applications of k-means clustering are as follows:

- **Document Classification**:
  Cluster documents in multiple categories based on tags, topics, and the content of the document. The initial processing of the documents is needed to represent each document as a vector and uses term frequency to identify commonly used terms that help classify the document. the document vectors are then clustered to help identify similarity in document groups.

- **Customer Segmentation**:
  Clustering helps marketers improve their customer base, work on target areas, and segment customers based on purchase history, interests, or activity monitoring. Ex - Telecom providers can cluster pre-paid customers to identify patterns in terms of money spent in recharging, sending sms, and browsing the internet and can target specific clusters of customers for specific campaigns.

- **Insurance Fraud Detection**:
  Machine learning has a critical role to play in fraud detection and has numerous applications in automobile, healthcare, and insurance fraud detection. utilizing past historical data on fraudulent claims, it is possible to isolate new claims based on its proximity to clusters that indicate fraudulent patterns. Since insurance fraud can potentially have a multi-million dollar impact on a company, the ability to detect frauds is crucial.

- **Cyber-profiling Criminals**:
  Cyber-profiling is the process of collecting data from individuals and groups to identify significant co-relations. The idea of cyber profiling is derived from criminal profiles, which provide information on the investigation division to classify the types of criminals who were at the crime scene.

Bibliography

[1] Lecture by Vineet sir.

[2] Coursera Machine Learning by Andrew Ng

[3] https://en.wikipedia.org/wiki/K-means

[4] https://towardsdatascience.com/understanding-k-means-clustering-in-machine-learning-6a6e67336aa1

[5] https://www.analyticsvidhya.com/blog/2016/11/an-introduction-to-clustering-and-different-methods-of-clustering/

[6] https://www.geeksforgeeks.org/clustering-in-machine-learning/

[7] https://medium.com/voice-tech-podcast/a-simple-way-to-explain-the-recommendation-engine-in-ai-d1a609f59d97

[8] https://blog.alexa.com/types-of-market-segmentation/

[9] https://en.wikipedia.org/wiki/Socialnetworkanalysis

[10] https://www.ijltet.org/journal/149976233816-Vasavi-medical

[11] https://en.wikipedia.org/wiki/Imagesegmentation

[12] https://www.anodot.com/blog/what-is-anomaly-detection/

[13] https://medium.com/@jorgesleonel/clustering-d2895d9e264c

[14] https://www.researchgate.net/figure/Connectivity-model-in-clusteringfig1332053160

[15] https://stackoverflow.com/questions/13619272/misplaced-centroid-in-clusters

[16] https://www.geeksforgeeks.org/dbscan-clustering-in-ml-density-based-clustering/

[17] https://www.geeksforgeeks.org/ml-k-means-algorithm/

[18] https://dzone.com/articles/10-interesting-use-cases-for-the-k-means-algorithm

[19] Rokach, Lior, and Oded Maimon. ”Clustering methods.” Data mining and knowledge discovery handbook. Springer US, 2005. 321-352.

[20] Frank Nielsen (2016). ”Chapter 8: Hierarchical Clustering”. Introduction to HPC with MPI for Data Science. Springer.