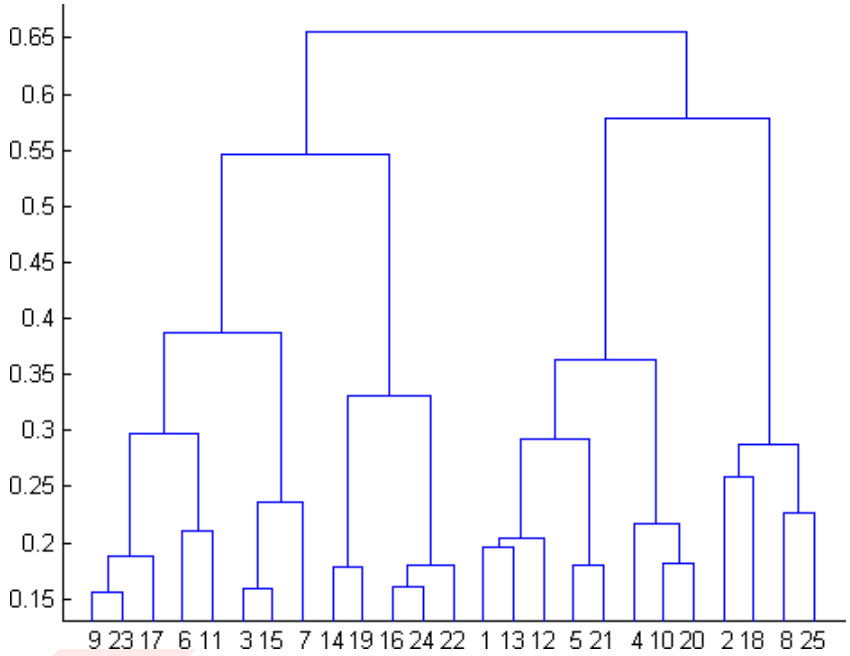
MACHINE LEARNING

Q1 to Q12 have only one correct answer. Choose the correct option to answer your question.

1. What is the most appropriate no. of clusters for the data points represented by the following dendrogram:



a) 2
b) 4
c) 6
d) 8

2. In which of the following cases will K-Means clustering fail to give good results?
   1. Data points with outliers
   2. Data points with different densities
   3. Data points with round shapes
   4. Data points with non-convex shapes

   Options:
   a) 1 and 2
   b) 2 and 3
   c) 2 and 4
   d) 1, 2 and 4

3. The most important part of____is selecting the variables on which clustering is based.
   a) interpreting and profiling clusters
   b) selecting a clustering procedure
   c) assessing the validity of clustering
   d) formulating the clustering problem

4. The most commonly used measure of similarity is the____or its square.
   a) Euclidean distance
   b) city-block distance
   c) Chebyshev's distance
   d) Manhattan distance

MACHINE LEARNING

5. ___is a clustering procedure where all objects start out in one giant cluster. Clusters are formed by dividing this cluster into smaller and smaller clusters.
   a) Non-hierarchical clustering
   b) Divisive clustering
   c) Agglomerative clustering
   d) K-means clustering

6. Which of the following is required by K-means clustering?
   a) Defined distance metric
   b) Number of clusters
   c) Initial guess as to cluster centroids
   d) All answers are correct

7. The goal of clustering is to-
   a) Divide the data points into groups
   b) Classify the data point into different classes
   c) Predict the output values of input data points
   d) All of the above

8. Clustering is a-
   a) Supervised learning
   b) Unsupervised learning
   c) Reinforcement learning
   d) None

9. Which of the following clustering algorithms suffers from the problem of convergence at local optima?
   a) K- Means clustering
   b) Hierarchical clustering
   c) Diverse clustering
   d) All of the above

10. Which version of the clustering algorithm is most sensitive to outliers?
    a) K-means clustering algorithm
    b) K-modes clustering algorithm
    c) K-medians clustering algorithm
    d) None

11. Which of the following is a bad characteristic of a dataset for clustering analysis-
    a) Data points with outliers
    b) Data points with different densities
    c) Data points with non-convex shapes
    d) All of the above

12. For clustering, we do not require-
    a) Labeled data
    b) Unlabeled data
    c) Numerical data
    d) Categorical data

Q13 to Q15 are subjective answers type questions, Answers them in their own words briefly.

13. How is cluster analysis calculated?

Cluster analysis or clustering is the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense) to each other than to those in other groups (clusters). **Clustering** is an unsupervised machine learning method of identifying and grouping similar data points in larger datasets

Determining the optimal number of clusters in a data set is a fundamental issue in partitioning

MACHINE LEARNING

clustering, such as k-means clustering, which requires the user to specify the number of clusters k to be generated. The optimal number of clusters is somehow subjective and depends on the method used for measuring similarities and the parameters used for partitioning.

Cluster analysis involves:

1. Calculating the distance
2. Link the clusters
3. Choose a solution by selecting the right number of clusters

Cluster analysis is done mainly using K-Means Cluster, Hierarchical Cluster, and Two-Step Cluster.

K-means cluster is a method to quickly cluster large data sets. The researcher define the number of clusters in advance. This is useful to test different models with a different assumed number of clusters.
K-Means is one of the most widely used and perhaps the simplest unsupervised algorithms to solve the clustering problems. Using this algorithm, we classify a given data set through a certain number of predetermined clusters or "k" clusters. Each cluster is assigned a designated cluster center and they are placed as much as possible far away from each other. Subsequently, each point belonging gets associated with it to the nearest centroid till no point is left unassigned. Once it is done, the centers are re-calculated and the above steps are repeated. The algorithm converges at a point where the centroids cannot move any further.

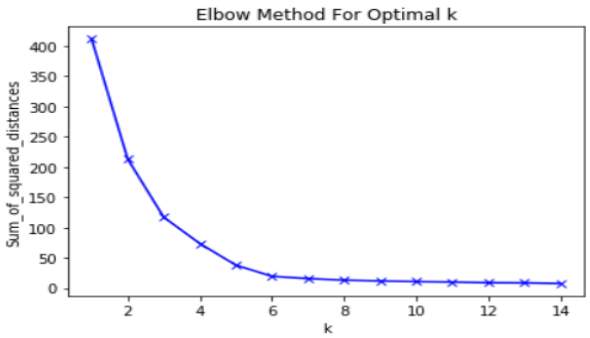
Hierarchical cluster is the most common method. It generates a series of models with cluster solutions from 1 (all cases in one cluster) to n (each case is an individual cluster). Hierarchical cluster also works with variables as opposed to cases; it can cluster variables together in a manner somewhat similar to factor analysis. In addition, hierarchical cluster analysis can handle nominal, ordinal, and scale data; however it is not recommended to mix different levels of measurement.

Two-step cluster analysis identifies groupings by running pre-clustering first and then by running hierarchical methods

Methods to find out optimal number of clusters:

1. Elbow Method:

In cluster analysis, the elbow method is a heuristic used in determining the number of clusters in a data set. The method consists of plotting the explained variation as a function of the number of clusters, and picking the elbow of the curve as the number of clusters to use.



In the plot above the elbow is at k=5 indicating the optimal k is 5

"Elbow" method helps data scientists select the optimal number of clusters by fitting the model with a

MACHINE LEARNING

range of values for K. If the line chart resembles an arm, then the “elbow” (the point of inflection on the curve) is a good indication that the underlying model fits best at that point.

2. Silhouette analysis

Silhouette analysis can be used to study the separation distance between the resulting clusters. The silhouette plot displays a measure of how close each point in one cluster is to points in the neighboring clusters and thus provides a way to assess parameters like number of clusters visually. This measure has a range of [-1, 1].

Silhouette coefficients (as these values are referred to as) near +1 indicates that the sample is far away from the neighboring clusters. A value of 0 indicates that the sample is on or very close to the decision boundary between two neighboring clusters and negative values indicates that those samples might have been assigned to the wrong cluster.

14. How is cluster quality measured?

The goal of clustering algorithms is to split the dataset into clusters of objects, such that:

- the objects in the same cluster are similar as much as possible, and
- the objects in different clusters are highly distinct

That is, we want the average distance within cluster to be as small as possible; and the average distance between clusters to be as large as possible.

In general, the validation methods can be categorized into two groups according to whether ground truth is available. Here, ground truth is the ideal clustering that is often built using human experts.

I. **Extrinsic Method**: If ground truth is available, it can be used by extrinsic methods, which compare the clustering against the group truth and measure.

Types of extrinsic methods:

1. Rand Index,
2. Mutual Information based scores .
3. Homogeneity, completeness and V-measure
4. Fowlkes-Mallows scores

II. **Intrinsic Method**: If the ground truth is unavailable, we can use intrinsic methods, which evaluate the goodness of a clustering by considering how well the clusters are separated.

Types of internal methods:

1. Silhouette analysis,
2. Dunn Index,
3. Davies-Bouldin index
4. Calinski-Harabasz index
5. Contingency Matrix
6. Pair confusion matrix

Ground truth can be considered as supervision in the form of “cluster labels.” Hence, extrinsic methods are also known as supervised methods, while intrinsic methods are unsupervised methods.

MACHINE LEARNING

Internal validation measures reflect often the compactness, the connectedness and separation of the cluster partitions.

- **Compactness measures** evaluate how close the objects within the same cluster are. A lower within-cluster variation is an indicator of a good compactness (i.e., a good clustering). The different indices for evaluating the compactness of clusters are based on distance measures such as the cluster-wise within average/median distances between observations.

- **Separation measures** determine how well-separated a cluster is from other clusters. The indices used as separation measures include:
  1. distances between cluster centers
  2. the pairwise minimum distances between objects in different clusters

- **Connectivity** corresponds to what extent items are placed in the same cluster as their nearest neighbors in the data space. The connectivity has a value between 0 and infinity and should be minimized.

Generally most of the indices used for internal clustering validation combine compactness and separation measures as follow:

Index= (α×Separation) (β×Compactness) Index= (α×Separation) (β×Compactness)

Where αα and ββ are weights.

We'll describe the two commonly used indices for assessing the goodness of clustering: **silhouette width and Dunn index.**

1. Silhouette analysis

Silhouette analysis measures how well an observation is clustered and it estimates the average distance between clusters. The silhouette plot displays a measure of how close each point in one cluster is to points in the neighboring clusters.

The Silhouette Coefficient (sklearn.metrics.silhouette_score) is an example of intrinsic evaluation, where a higher Silhouette Coefficient score relates to a model with better defined clusters. The

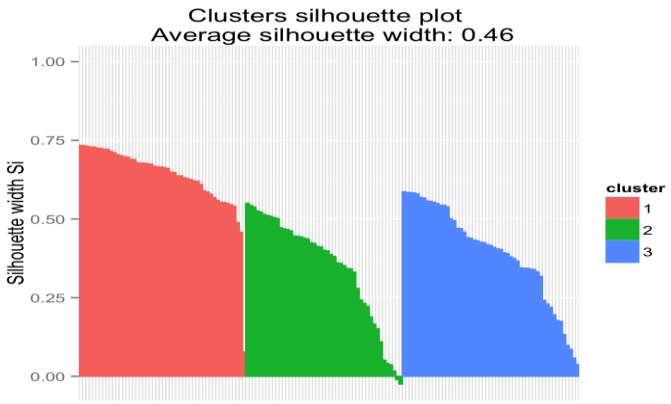
Silhouette Coefficient is defined for each sample and is composed of two scores:
a: The mean distance between a sample and all other points in the same class.
b: The mean distance between a sample and all other points in the next nearest cluster.

The Silhouette Coefficient s for a single sample is then given as:

$$s = \frac{b - a}{max(a, b)}$$

The best value is 1 and the worst value is -1. Values near 0 indicate overlapping clusters. Negative values generally indicate that a sample has been assigned to the wrong cluster, as a different cluster is more similar.

MACHINE LEARNING



Interpretation of silhouette width

Silhouette width can be interpreted as follow:

1. Observations with a large SiSi (almost 1) are very well clustered
2. A small SiSi (around 0) means that the observation lies between two clusters
3. Observations with a negative SiSi are probably placed in the wrong cluster.

2. Dunn index:

Dunn index is another internal clustering validation measure which can be computed as follow:

1. For each cluster, compute the distance between each of the objects in the cluster and the objects in the other clusters
2. Use the minimum of this pairwise distance as the inter-cluster separation (min.separation)
3. For each cluster, compute the distance between the objects in the same cluster.
4. Use the maximal intra-cluster distance (i.e. maximum diameter) as the intra-cluster compactness
5. Calculate Dunn index (D) as follow:

$$D = \frac{min.\,separation}{max.\,diameter}$$

If the data set contains compact and well-separated clusters, the diameter of the clusters is expected to be small and the distance between the clusters is expected to be large. Thus, Dunn index should be maximized.

3. Davies-Bouldin index

If the ground truth labels are not known Davies-Bouldin index(sklearn.metrics.davies_bouldin_score) can be used to evaluate the model, where a lower Davies- Bouldin index relates to a model with better separation between the clusters.

This index signifies the average 'similarity' between clusters, where the similarity is a measure that compares the distance between clusters with the size of the clusters themselves.
Zero is the lowest possible score. Values closer to zero indicate a better partition.

4. Calinski-Harabasz Index

MACHINE LEARNING

If the ground truth labels are not known, the Calinski-Harabasz index (sklearn.metrics.calinski_harabasz_score) - also known as the Variance Ratio Criterion - can be used to evaluate the model, where a higher Calinski-Harabasz score relates to a model with better defined clusters.
The index is the ratio of the sum of between-clusters dispersion and of within-cluster dispersion for all clusters (where dispersion is defined as the sum of distances squared)
The score is higher when clusters are dense and well separated, which relates to a standard concept of a cluster. The score is fast to compute.

5. Contingency matrix

Contingency matrix (sklearn.metrics.cluster.contingency_matrix) reports the intersection cardinality for every true/predicted cluster pair. The contingency matrix provides sufficient statistics for all clustering metrics where the samples are independent and identically distributed and one doesn't need to account for some instances not being clustered.

Here is an example:

```
>>> from sklearn.metrics.cluster import contingency_matrix
>>> x = ["a", "a", "a", "b", "b", "b"]
>>> y = [0, 0, 1, 1, 2, 2]
>>> contingency_matrix(x, y)
array([[2, 1, 0],
       [0, 1, 2]])
```

The first row of output array indicates that there are three samples whose true cluster is "a". Of them, two are in predicted cluster 0, one is in 1, and none is in 2. And the second row indicates that there are three samples whose true cluster is "b". Of them, none is in predicted cluster 0, one is in 1 and two are in 2.

6. The pair confusion matrix

The pair confusion matrix (sklearn.metrics.cluster.pair_confusion_matrix) is a 2x2 similarity matrix between two clustering's computed by considering all pairs of samples and counting pairs that are assigned into the same or into different clusters under the true and predicted clustering.

$$C = \begin{bmatrix} C_{00} & C_{01} \\ C_{10} & C_{11} \end{bmatrix}$$

It has the following entries:

C00 : number of pairs with both clusterings having the samples not clustered together
C10 : number of pairs with the true label clustering having the samples clustered together but the other clustering not having the samples clustered together
C01 : number of pairs with the true label clustering not having the samples clustered together but the other clustering having the samples clustered together
C11 : number of pairs with both clusterings having the samples clustered together
Considering a pair of samples that is clustered together a positive pair, then as in binary classification the count of true negatives is C00, false negatives is C10, true positives is C11 and false positives is C01.

Perfectly matching labeling have all non-zero entries on the diagonal regardless of actual label values:

MACHINE LEARNING

15. What is cluster analysis and its types?

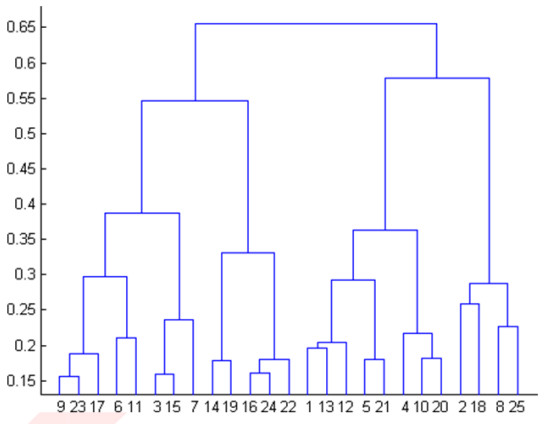
Cluster analysis or clustering is the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense) to each other than to those in other groups (clusters). It is a main task of exploratory data analysis, and a common technique for statistical data analysis, used in many fields, including pattern recognition, image analysis, information retrieval, bioinformatics, data compression, computer graphics and machine learning.
Cluster analysis is also called **classification analysis.**

The various types of clustering are:

1. Connectivity-based Clustering (Hierarchical clustering)
2. Centroids-based Clustering (Partitioning methods)
3. Distribution-based Clustering
4. Density-based Clustering (Model-based methods)
5. Fuzzy Clustering
6. Constraint-based (Supervised Clustering)

1. Connectivity-Based Clustering (Hierarchical Clustering)

Hierarchical Clustering is a method of unsupervised machine learning clustering where it begins with a pre-defined top to bottom hierarchy of clusters. It then proceeds to perform a decomposition of the data objects based on this hierarchy, hence obtaining the clusters. This method follows two approaches based on the direction of progress, i.e., whether it is the top-down or bottom-up flow of creating clusters. These are Divisive Approach and the Agglomerative Approach respectively.



Hierarchical clustering can be divided into two main types:

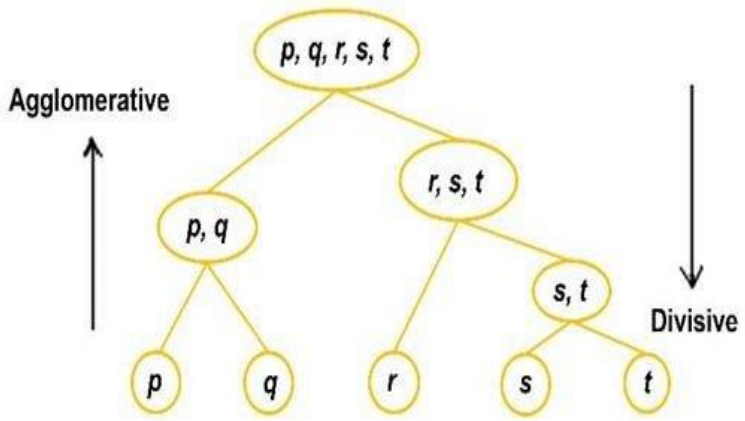
- Agglomerative
- Divisive.

Agglomerative clustering: It's also known as AGNES (Agglomerative Nesting). It works in a bottom-up manner. Agglomerative is quite the contrary to Divisive, where all the “N” data points are considered to be a single member of “N” clusters that the data is comprised into. We iteratively combine these numerous “N” clusters to fewer number of clusters, let’s say “k” clusters and hence assign the data points to each of these clusters accordingly. This approach is a bottom-up one, and also uses termination logic in combining the clusters.

Divisive hierarchical clustering: It's also known as DIANA (Divisive Analysis) and it works in a top-down manner where we consider that all the data points belong to one large cluster and try to divide

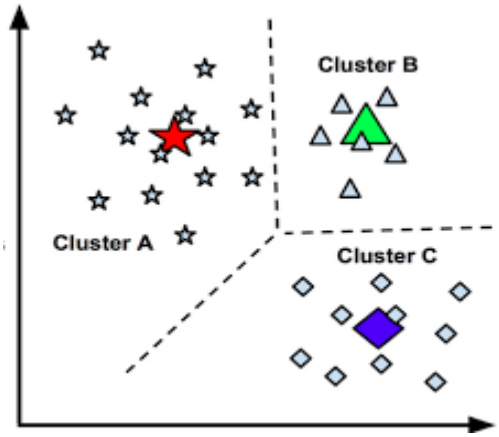
MACHINE LEARNING

the data into smaller groups based on a termination logic or, a point beyond which there will be no further division of data points.



2. Centroid Based Clustering

Centroid based clustering is considered as one of the most simplest clustering algorithms, yet the most effective way of creating clusters and assigning data points to it. The intuition behind centroid based clustering is that a cluster is characterized and represented by a central vector and data points that are in close proximity to these vectors are assigned to the respective clusters.
These groups of clustering methods iteratively measure the distance between the clusters and the characteristic centroids using various distance metrics. These are either of Euclidian distance, Manhattan Distance or Minkowski Distance.
The major setback here is that we should either intuitively or scientifically (Elbow Method) define the number of clusters, “k”, to begin the iteration of any clustering machine learning algorithm to start assigning the data points.

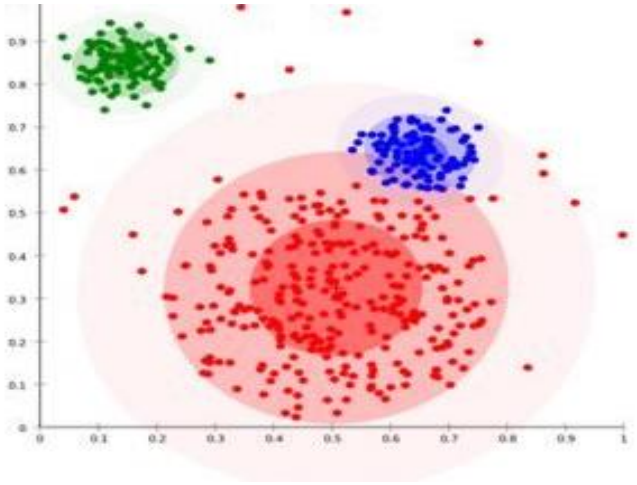


Centroid based clustering has proven its worth over Hierarchical clustering when working with large datasets. K-means is the most widely-used **centroid-based clustering** algorithm.

3. Distribution-Based Clustering

Until now, the clustering techniques as we know are based around either proximity (similarity/distance) or composition (density). There is a family of clustering algorithms that take a totally different metric into consideration – probability. Distribution-based clustering creates and groups’ data points based on their likely hood of belonging to the same probability distribution (Gaussian, Binomial etc.) in the data.
Distribution clustering identifies the probability that a point belongs to a cluster Distribution clustering is a great technique to assign outliers to clusters

MACHINE LEARNING



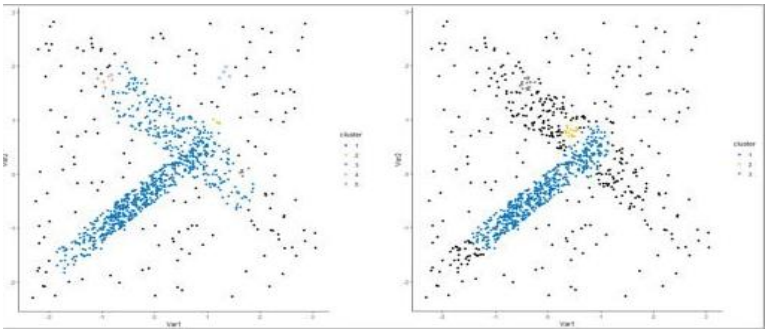
The distribution models of clustering are most closely related to statistics as it very closely relates to the way how datasets are generated and arranged using random sampling principles i.e., to fetch data points from one form of distribution. Clusters can then be easily be defined as objects that are most likely to belong to the same distribution.
Distribution based clustering has a vivid advantage over the proximity and centroid based clustering methods in terms of flexibility, correctness and shape of the clusters formed. The major problem however is that these clustering methods work well only with synthetic or simulated data or with data where most of the data points most certainly belong to a predefined distribution, if not, the results will over fit.

5. Density-based Clustering (Model-based Methods)

Both hierarchical and centroid based algorithms are dependent on a distance (similarity/proximity) metric. The very definition of a cluster is based on this metric. Density-based clustering methods take density into consideration instead of distances. Clusters are considered as the densest region in a data space, which is separated by regions of lower object density and it is defined as a maximal-set of connected points.

When performing most of the clustering, we take two major assumptions, one, the data is devoid of any noise and two, the shape of the cluster so formed is purely geometrical (circular or elliptical). The fact is, data always has some extent of inconsistency (noise) which cannot be ignored. Added to that, we must not limit ourselves to a fixed attribute shape, it is desirable to have arbitrary shapes so as to not to ignore any data points. These are the areas where density based algorithms have proven their worth!

Density-based algorithms can get us clusters with arbitrary shapes, clusters without any limitation in cluster sizes, clusters that contain the maximum level of homogeneity by ensuring the same levels of density within it, and also these clusters are inclusive of outliers or the noisy data.
Density clustering will not assign an outlier to a cluster.



MACHINE LEARNING

5. Fuzzy Clustering

The general idea about clustering revolves around assigning data points to mutually exclusive clusters, meaning, a data point always resides uniquely inside a cluster and it cannot belong to more than one cluster. Fuzzy clustering methods change this paradigm by assigning a data-point to multiple clusters with a quantified degree of belongingness metric.

The data-points that are in proximity to the center of a cluster, may also belong in the cluster that is at a higher degree than points in the edge of a cluster. The possibility of which an element belongs to a given cluster is measured by membership coefficient that varies from 0 to 1.

Fuzzy clustering can be used with datasets where the variables have a high level of overlap. It is a strongly preferred algorithm for Image Segmentation, especially in bioinformatics where identifying overlapping gene codes makes it difficult for generic clustering algorithms to differentiate between the image's pixels and they fail to perform a proper clustering.

6. Constraint-based (Supervised Clustering)

The clustering process, in general, is based on the approach that the data can be divided into an optimal number of "unknown" groups. The underlying stages of all the clustering algorithms is to find those hidden patterns and similarities, without any intervention or predefined conditions. However, in certain business scenarios, we might be required to partition the data based on certain constraints. Here is where a supervised version of clustering machine learning techniques comes into play.

A constraint is defined as the desired properties of the clustering results, or a user's expectation on the clusters so formed – this can be in terms of a fixed number of clusters, or, the cluster size, or, important dimensions (variables) that are required for the clustering process.

Parameters	Hierarchical Clustering	Partitioning methods	Distribution-based Clustering	Density-based Clustering (Model-based methods)	Fuzzy Clustering	Constraint Based (Supervised Clustering)
Description	Based on top-to-bottom hierarchy of the data points to create clusters	Based on centroids and data points are assigned into a cluster based on its proximity to the cluster centroid	Based on the probability distribution of the data, clusters are derived from various metrics like mean, variance etc.	Based on density of the data points, also known as model based clustering	Based on Partitioning Approach but data points can belong to more than one cluster	Clustering is directed and controlled by user constraints
Advantages	Easy to implement, the number of clusters need not be specified apriori, dendrograms	Easy to implement, faster processing, can work on larger data, easy to interpret	Number of clusters need not be specified apriori, works on real-time data,	Can handle noise and outliers, need not specify number of clusters in the start,	Can work on highly overlapped data, a higher rate of convergence	Creates a perfect decision boundary, can automatically determine the outcome classes based on constraints,

MACHINE LEARNING

	are easy to interpret.	the outputs	metrics are easy to understand and tune	clusters that are created are highly homogenous, no restrictions on cluster shapes.		future data can be classified based on the training boundaries
Disadvantages	Cluster assignment is strict and cannot be undone, high time complexity, cannot work for a larger dataset	We need to specify the number of cenrtroids apriori, clusters that get created are of inconsistent t sizes and densities, Effected by noise and outliers	Complex algorithm and slow, cannot be scaled to larger data	Complex algorithm and slow, cannot be scaled to larger data	We need to specify the number of centroids apriori, Effected by noise and outliers, Slow algorithm and cannot be scaled	Overfitting, high level of misclassification errors, cannot be trained on larger datasets
Algorithms	DIANA, AGNES, hclust etc.	k-means, k-medians, k-modes	Gaussian Mixed Models, DBCLASD	DENCAST, DBSCAN	Fuzzy C Means, Rough k means	Decision Trees, Random Forest, Gradient Boosting

Thank-You.