Session 52

Unsupervised Learning II

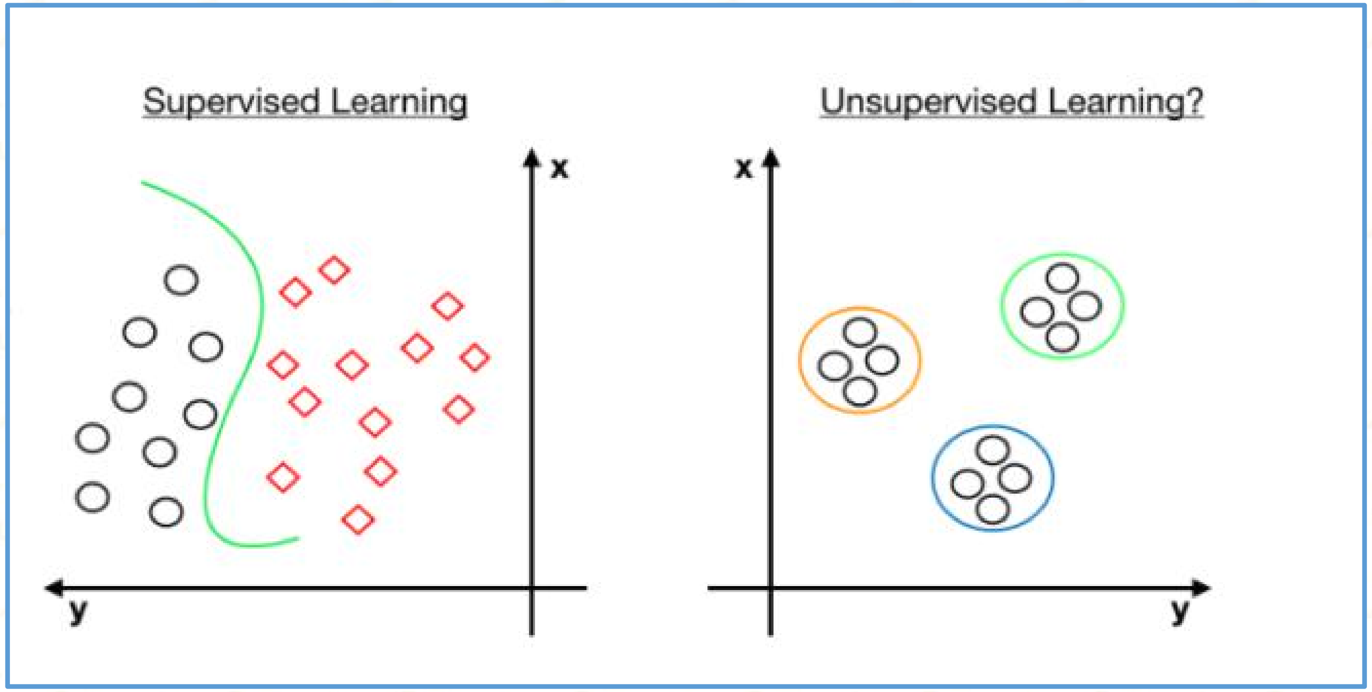
Table of Content

What will We Learn Today?

1. DBSCAN
2. Hierarchical Clustering

Supervised vs Unsupervised

- **Supervised** = Learn to predict the outcome.
  - We know the target label, so we make the model that try to predict the label.
- **Unsupervised** = Finding pattern/ characteristic from data.
  - We do not know our target label, so we make model that try to group the data.



Types of clustering algorithms

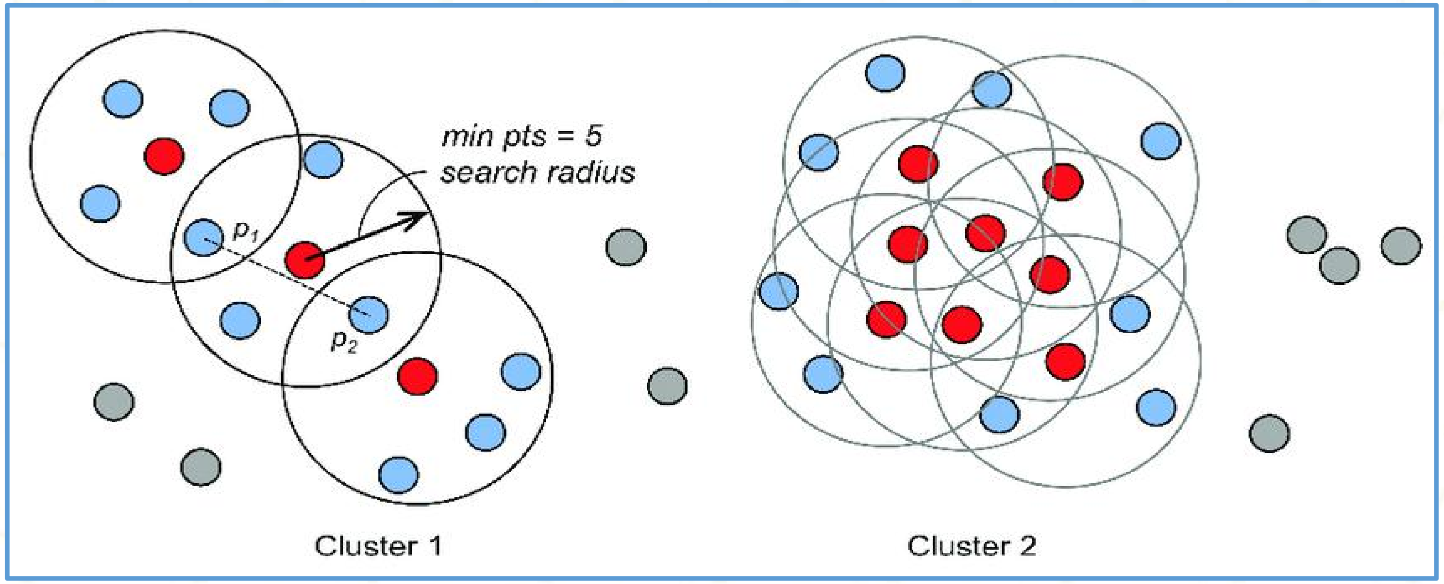
- Connectivity models
  - Based on the notion that the data points closer in data space exhibit more similarity to each other than the data points lying farther away. Example: hierarchical clustering algorithm.
- Centroid models:
  - The notion of similarity is derived by the closeness of a data point to the centroid of the clusters. Example : K-means
- Distribution models
  - The notion of how probable is it that all data points in the cluster belong to the same distribution (For example: Normal, Gaussian).
- Density models
  - These models search the data space for areas of varied density of data points in the data space. Example : DBSCAN

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

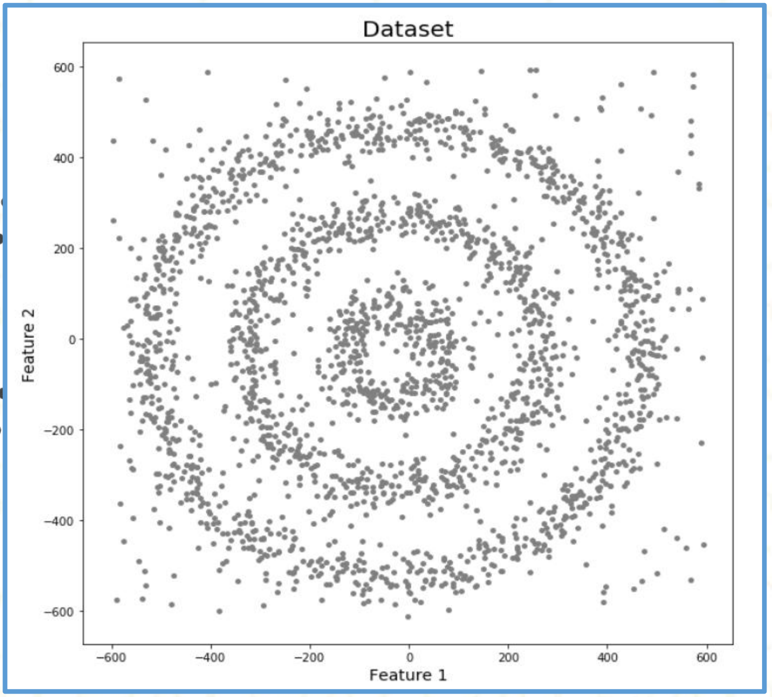
DBSCAN

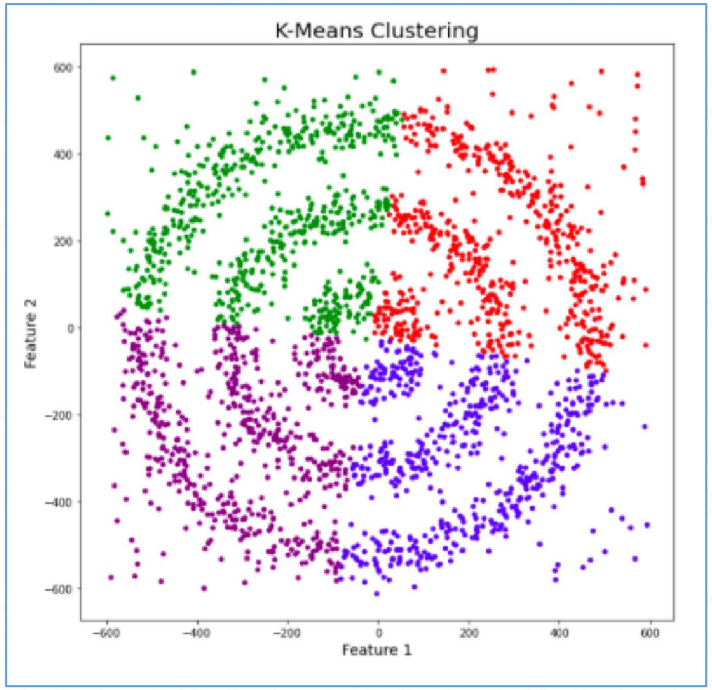
DBSCAN

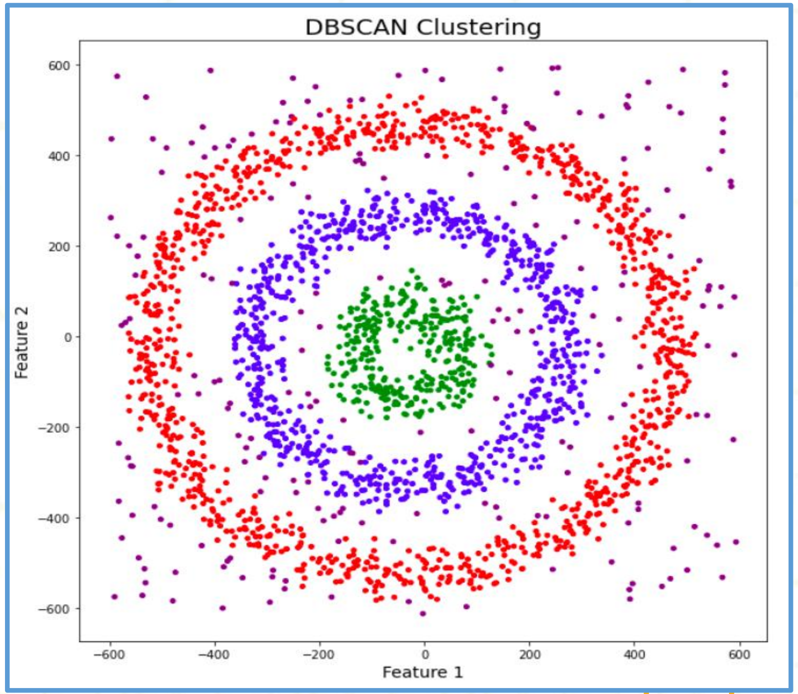
- Density-Based Spatial Clustering of Applications with Noise (DBSCAN) is
  - algorithm for density-based clustering
  - proposed by Martin Ester, Hans-Peter Kriegel, Jörg Sander and Xiaowei Xu in 1996



Why DBSCAN?







DBSCAN can cluster the data points correctly, and also detects noise

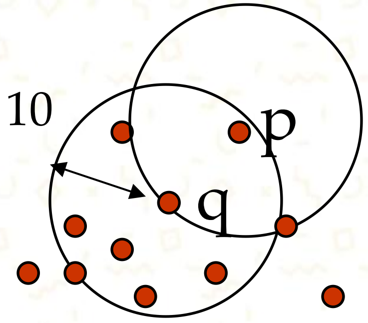
https://www.analyticsvidhya.com/blog/2020/09/how-dbscan-clustering-works/

How DBSCAN works?

- Group objects in dense region
- Major features
  - Discover clusters of arbitrary shape
  - Handle noise
  - One scan
  - Need density parameters as termination condition
- Density parameters
  - ***Radius*** ε : distance to determine the neighborhood
  - ***MinPts*** : Minimum number of points in neighborhood

Definitions

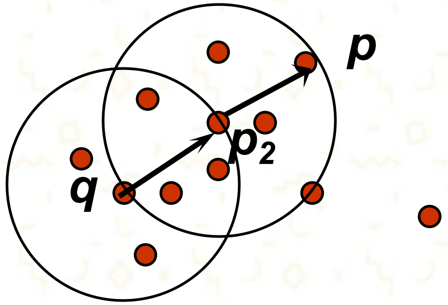
- Core object
  - ε-neighborhood contains ***MinPts*** objects
- Directly density-reachable
  - ***p*** is directly density-reachable from ***q*** if ***q*** is a core object, and ***p*** is ε-neighborhood of ***q***

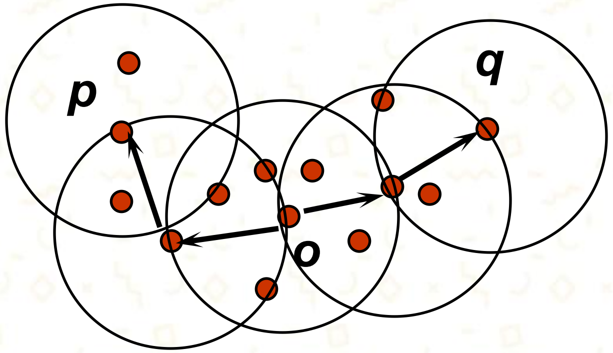


$\varepsilon = 10$
MinPts = 5

Definitions

- Density-reachable
  - p is density-reachable from q if
    there are objects $p_1, p_2, \ldots p_n$, $p_1 = q, p_n = p$ such that
    p_{i+1} is directly density-reachable from p_i
- Density-connected
  - p is density-connected to q if
    there is an object o
    such that
    p and q are density-reachable from o

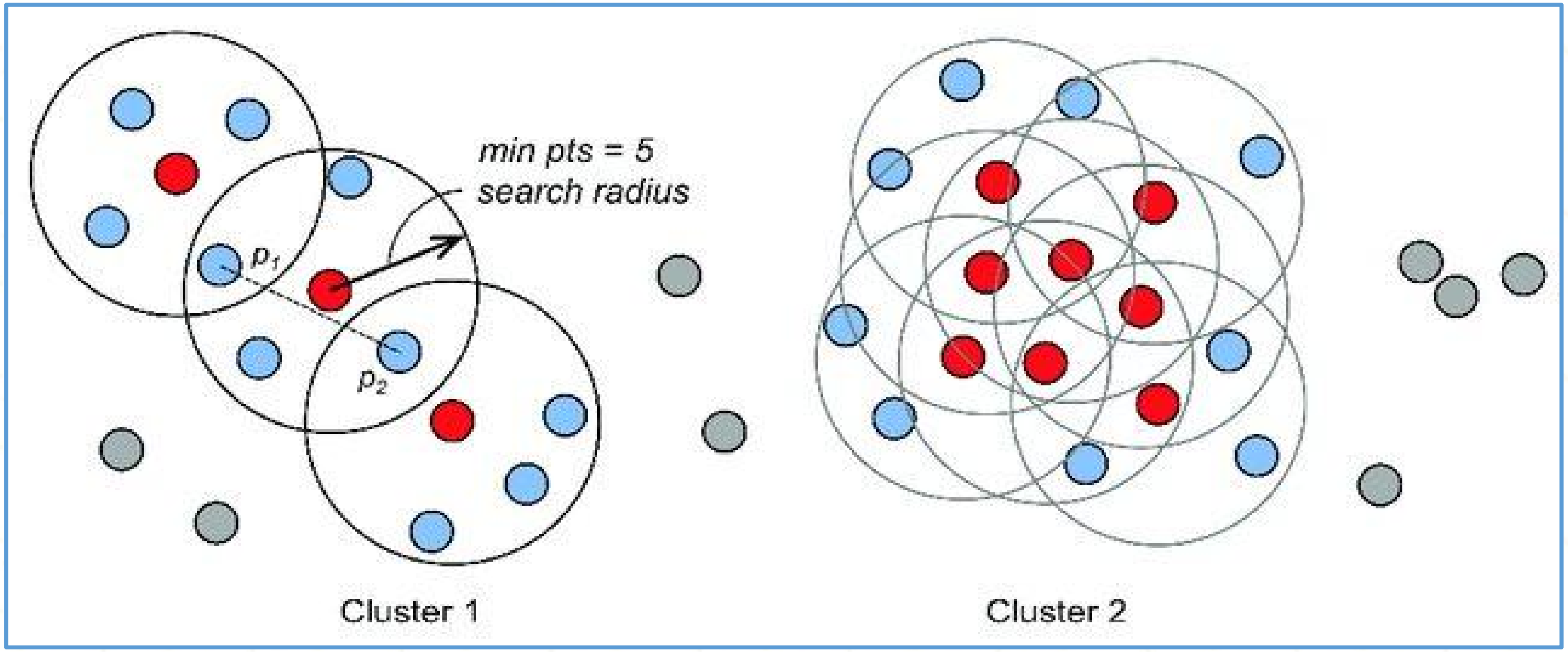




DBSCAN

- A *cluster* - a maximal set of density-connected points
- Discovers clusters of arbitrary shape in databases with noise
  1. Arbitrary select a point ***p***
  2. Retrieve all ε-neighborhood of ***p***
  3. If ***p*** is a core object, a cluster is formed
  4. From each core object ***p***, iteratively collects directly density-reachable objects (may merge clusters)
  5. Continue the process until no new points can be added
- Problem with DBSCAN
  - Selecting parameters ε and ***MinPts***

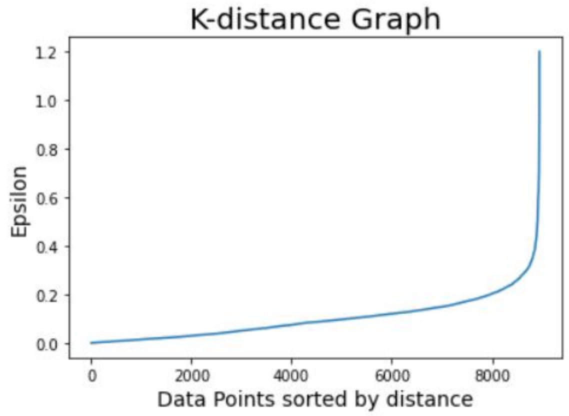
DBSCAN



- There are 3 types of point
  - core point (red)
  - border points (blue)
  - noise points (grey)

How to determine MinPts and eps

- MinPts
  - Using domain knowledge
  - The larger the data set, the larger the value of MinPts should be
  - If the data set is noisier, choose a larger value of MinPts
  - Generally, MinPts should be greater than or equal to the dimensionality of the data set
  - For 2-dimensional data, use DBSCAN's default value of MinPts = 4 (Ester et al., 1996).
  - If your data has more than 2 dimensions, choose MinPts = 2*dim, where dim= the dimensions of your data set (Sander et al., 1998).
- Epsilon
  - sorted k-dist graph

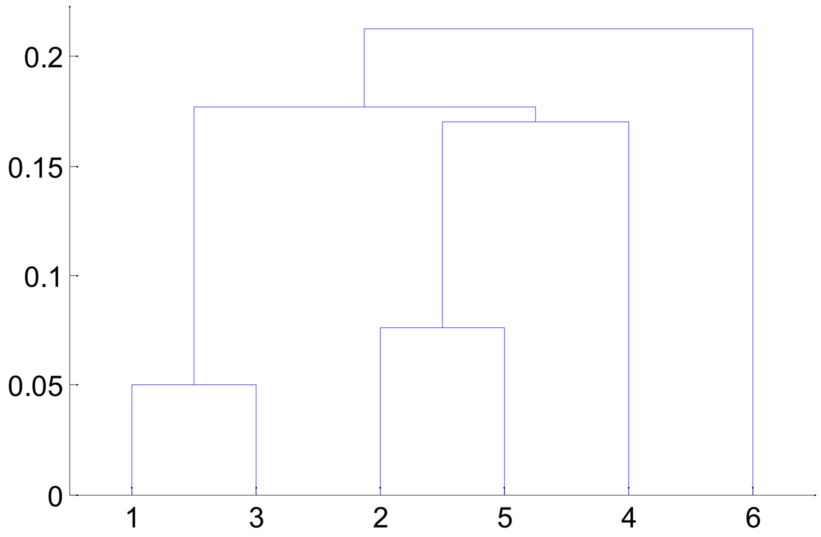


https://medium.com/@tarammullin/dbscan-parameter-estimation-ff8330e3a3bd

Hierarchical Clustering

Hierarchical clustering

- Produces a set of nested clusters organized as a hierarchical tree
- Can be visualized as a dendrogram
  - A tree like diagram that records the sequences of merges or splits



Strengths of Hierarchical Clustering

- Do not have to assume any particular number of clusters
  - Any desired number of clusters can be obtained by ‘cutting’ the dendogram at the proper level

- They may correspond to meaningful taxonomies
  - Example in biological sciences (e.g., phylogeny reconstruction, …)

Hierarchical clustering

- Two main types of hierarchical clustering
    - Agglomerative (bottom-up):
        - Start with the points as individual clusters
        - At each step, merge the closest pair of clusters until only one cluster (or k clusters) left
    - Divisive (top-down):
        - Start with one, all-inclusive cluster
        - At each step, split a cluster until each cluster contains a point (or there are k clusters)
- Traditional hierarchical algorithms use a similarity or distance matrix
    - Merge or split one cluster at a time

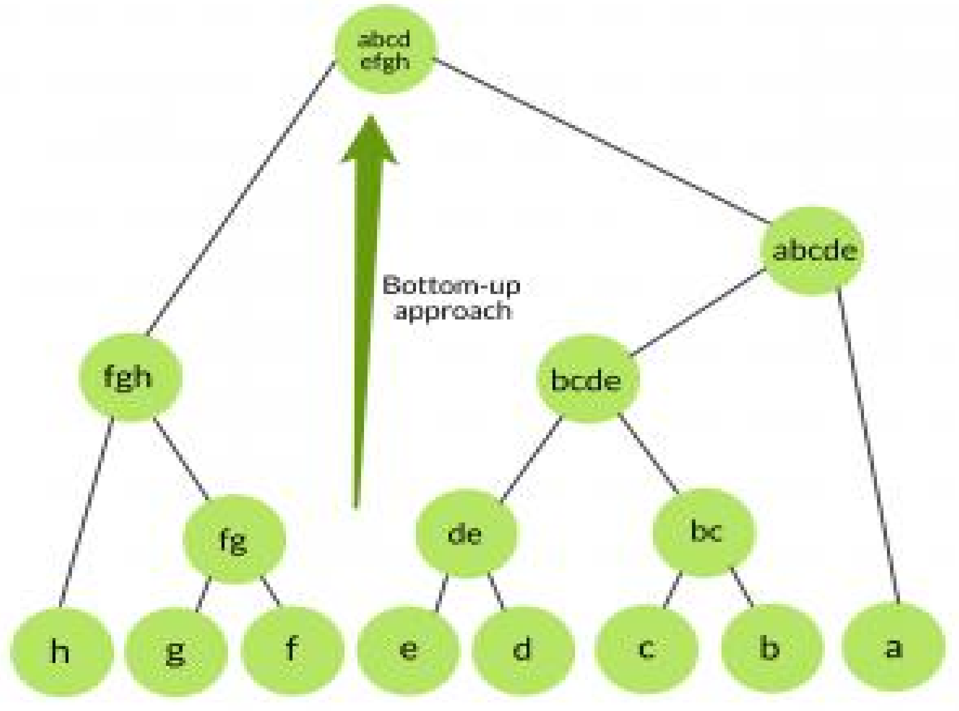
Agglomerative Clustering Algorithm

- More popular hierarchical clustering technique

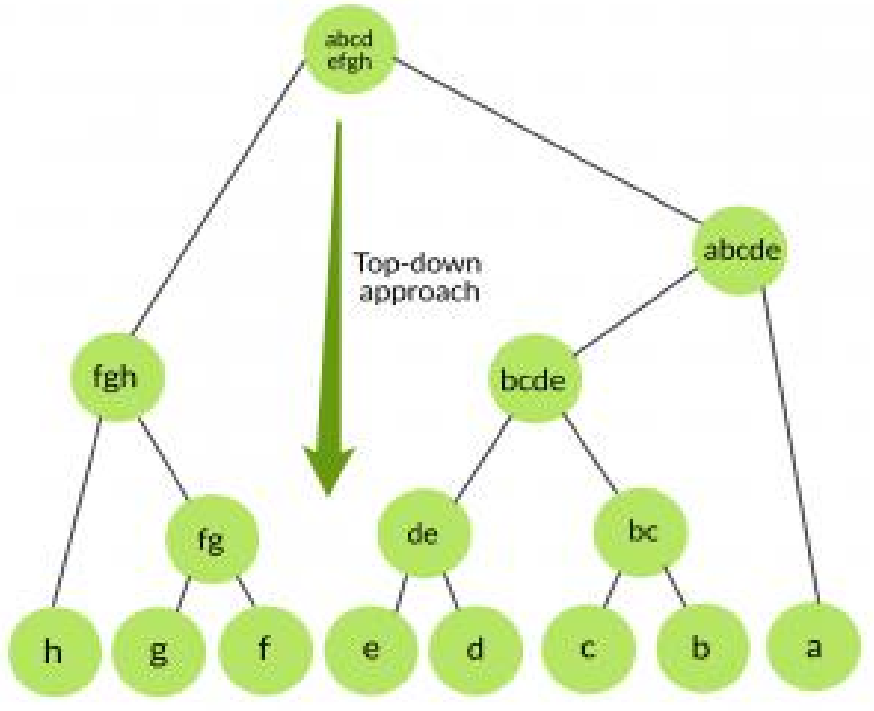
- Basic algorithm is straightforward
  1. Compute the proximity matrix
  2. Let each data point be a cluster
  3. **Repeat**
  4. Merge the two closest clusters
  5. Update the proximity matrix
  6. **Until** only a single cluster remains

- Key operation is the computation of the proximity of two clusters
  - Different approaches to defining the distance between clusters distinguish the different algorithms

Hierarchical clustering



Hierarchical agglomerative clustering



Hierarchical divisive clustering

https://www.geeksforgeeks.org/ml-hierarchical-clustering-agglomerative-and-divisive-clustering/

Lets Coding!

Digital Skola
Uncover The World Of Digital Skills With Us

Thank YOU