UNIVERSIDADE CATÓLICA PORTUGUESA

BRAGA

Behavior Analysis Technologies

Session 7

Text Clustering

Applied Data Science

2024/2025

Clustering

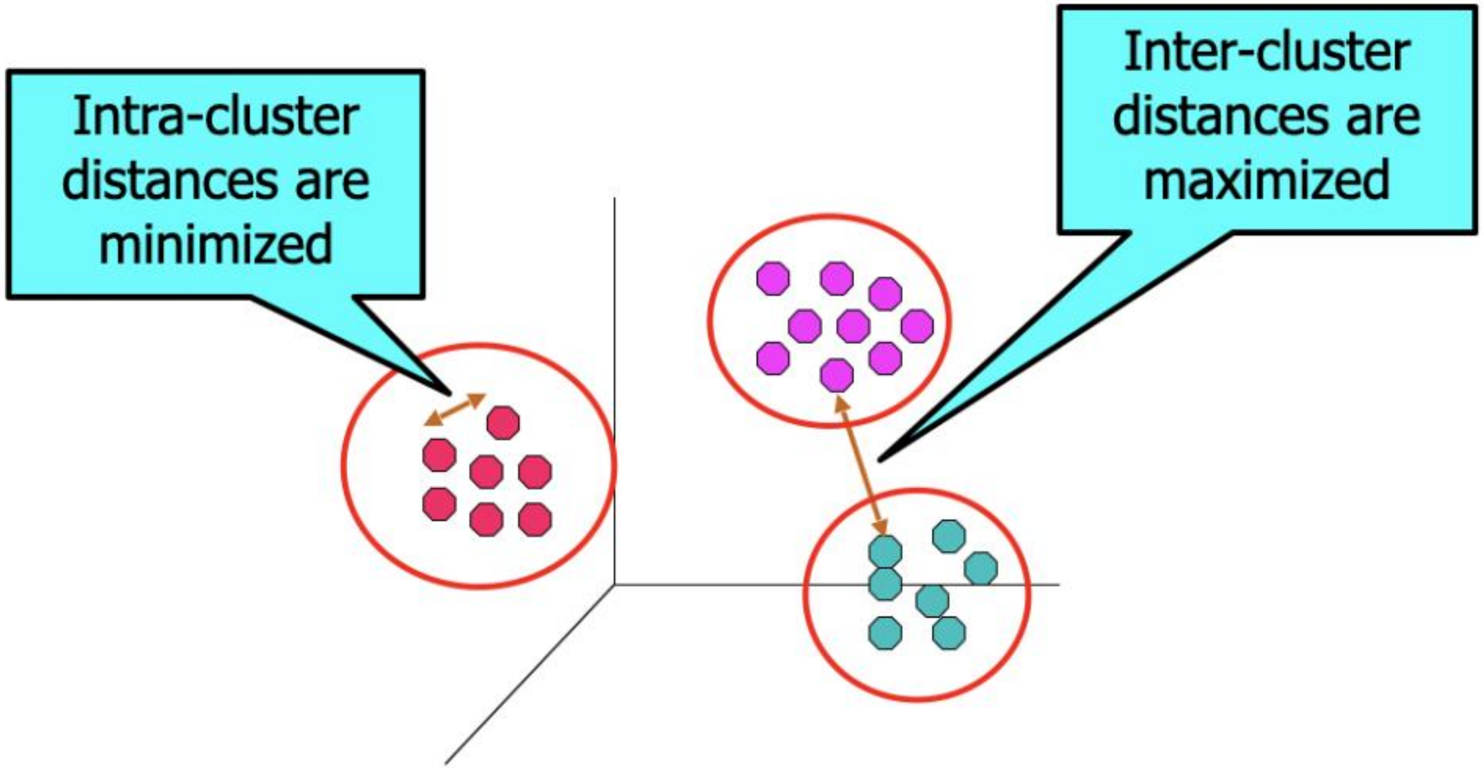
- **Definition:** Grouping similar objects together (e.g., documents, sentences, words, users, etc.)

- **Purpose:** A key data mining technique for exploring large datasets
  - Uncovers hidden patterns or structures in large datasets
  - Facilitates data exploration, content organization, and redundancy detectionl

- **Type:** An unsupervised learning problem

Clustering

- **Objective:** Identify groups of objects where:
  - Objects within a group are similar (or related) to each other
  - Objects in different groups are dissimilar (or unrelated)



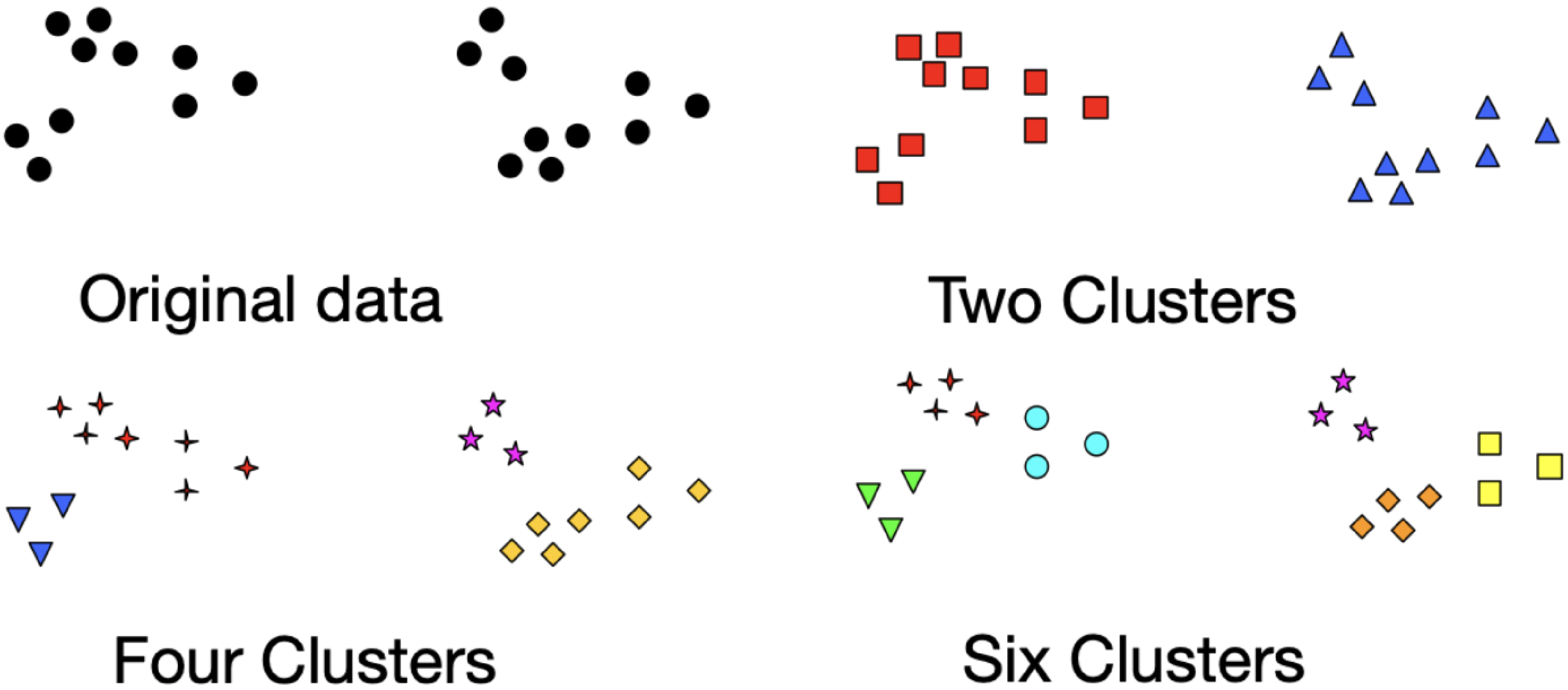
Clustering

- How many clusters should be formed?

Original data

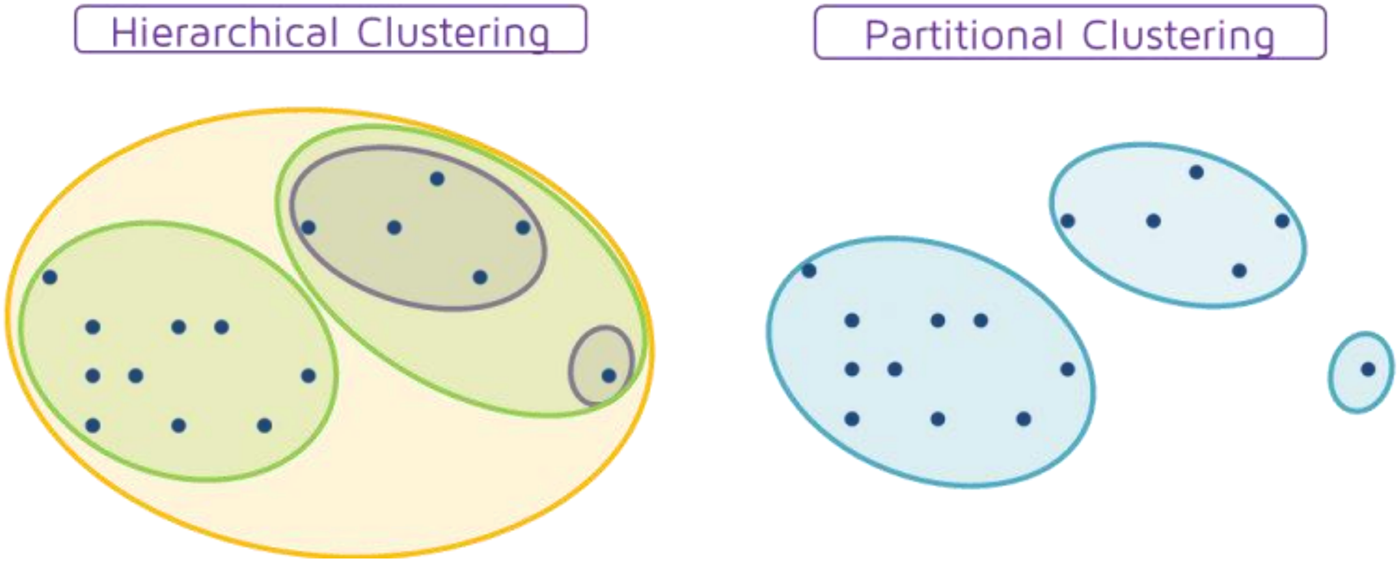
Clustering

- The notion of a cluster can be ambiguous!



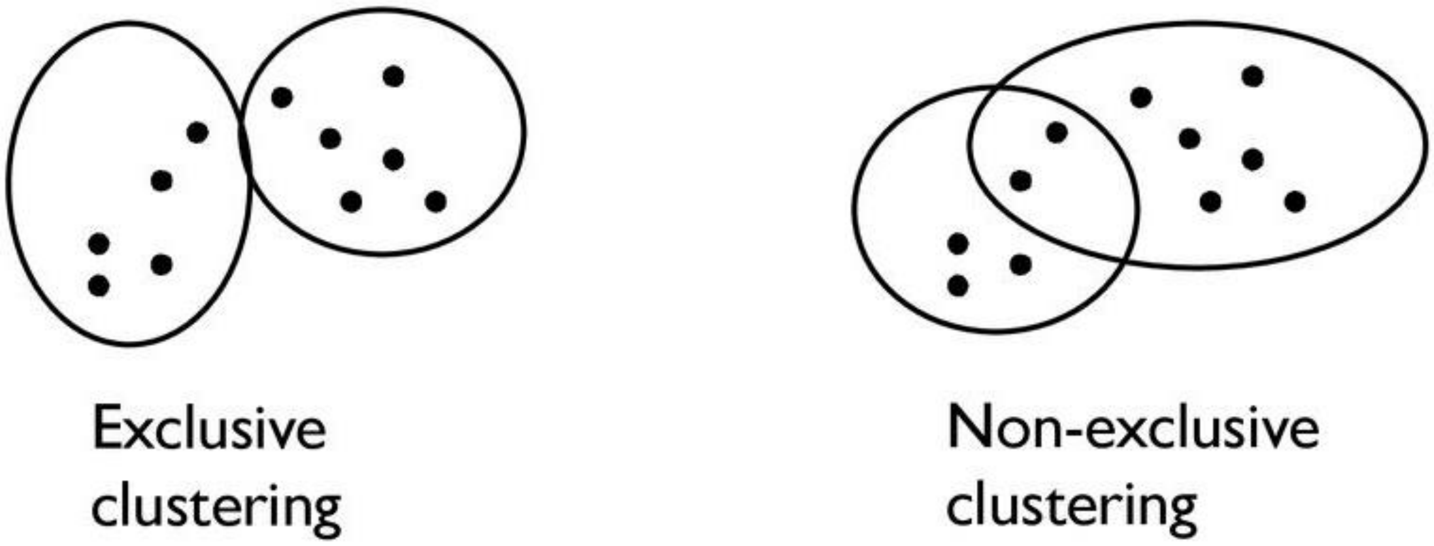
Types of Clustering

- **Partitional vs. Hierarchical:**
  - **Partitional:** Creates non-overlapping clusters; each data object belongs to exactly one cluster.
  - **Hierarchical**: Forms a nested structure of clusters organized in a tree format.



Types of Clustering

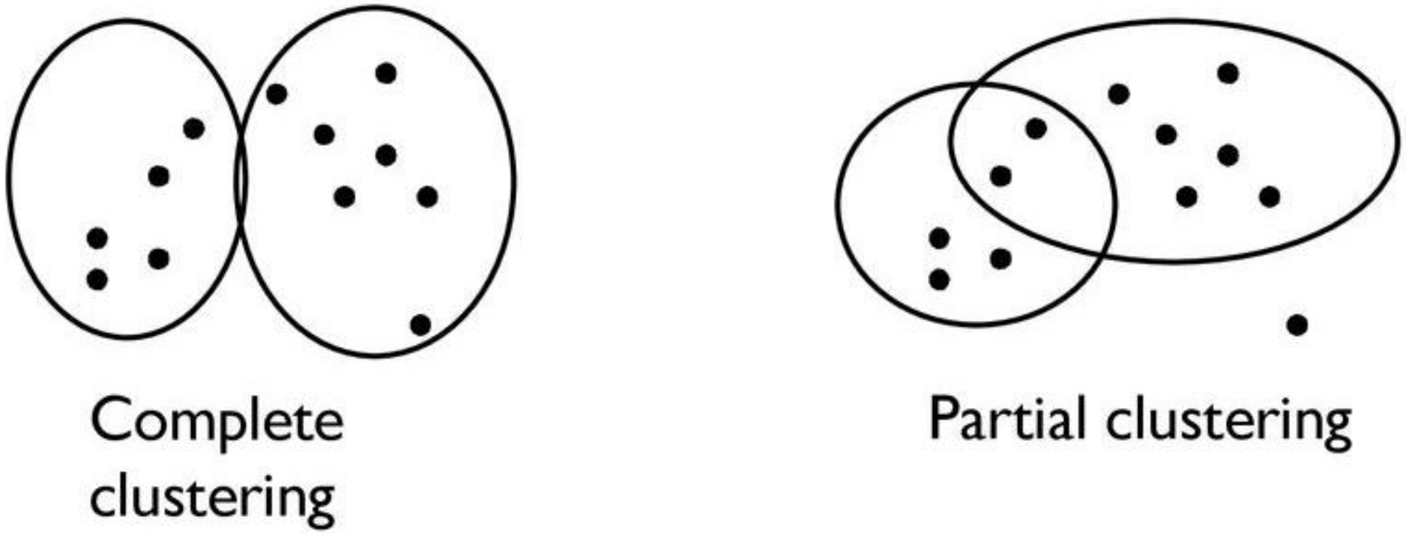
- **Exclusive vs. Non-exclusive**:
  - **Exclusive**: Objects belong to a single cluster.
  - **Non-exclusive**: Objects can belong to multiple clusters.



Types of Clustering

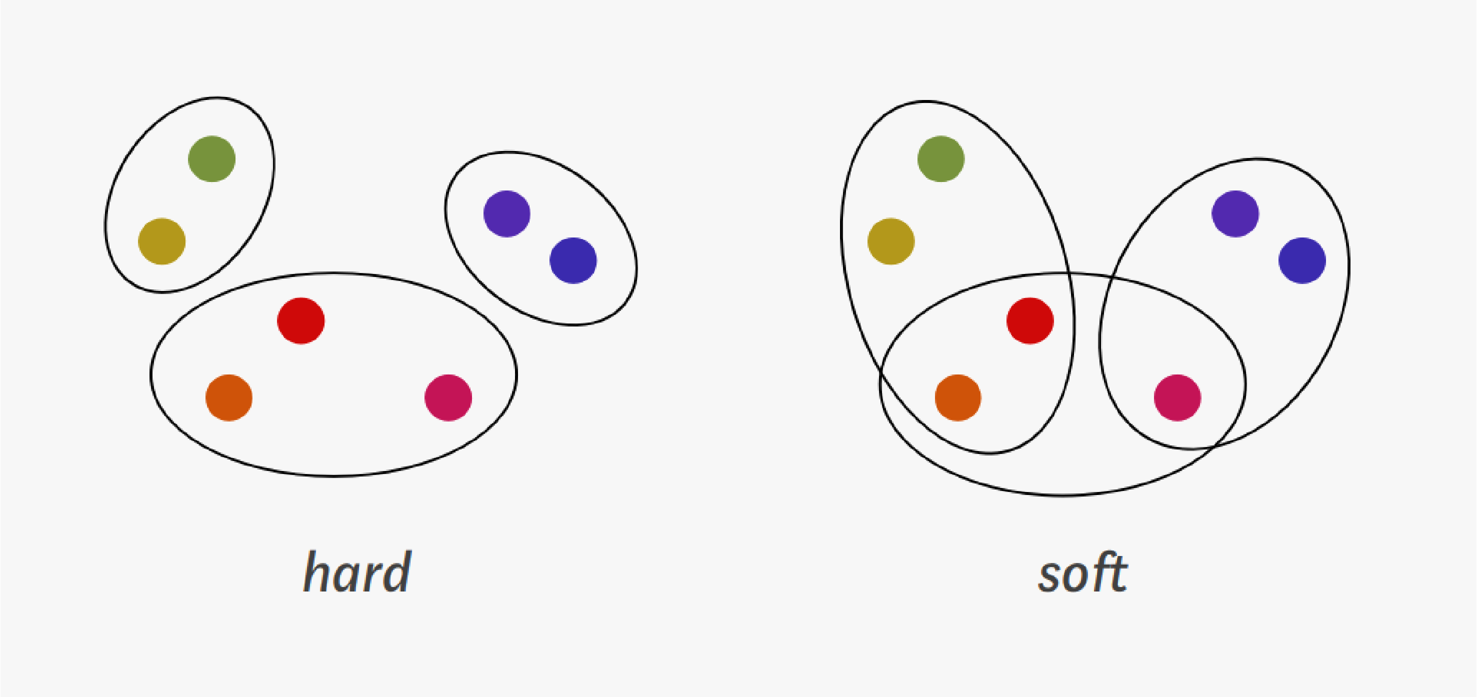
UNIVERSIDADE CATÓLICA PORTUGUESA
BRAGA

- **Partial vs. Complete**:
    - **Partial**: Clustering focuses on a subset of the data.



Types of Clustering

- **Hard vs. Soft:**
  - **Hard Clustering:** Each object belongs to one cluster only.
  - **Soft (Fuzzy) Clustering:** Objects can belong to multiple clusters with varying probabilities.



Clustering Techniques

- **Similarity-based Clustering:** Utilizes a similarity function; each object belongs to one cluster (hard clustering).
  - **Agglomerative Clustering:** Merges similar objects incrementally to form clusters (bottom-up approach).
  - **Divisive Clustering:** Divides the entire dataset into smaller clusters (top-down approach).

- **Model-based Techniques**: Rely on probabilistic models to capture the underlying structure of data.
  - Typically represent soft clustering, allowing objects to belong to multiple clusters with associated probabilities.

Similarity-Based Clustering

UNIVERSIDADE CATÓLICA PORTUGUESA
BRAGA

- Both agglomerative and divisive clustering methods rely on a **document-document similarity** measure, denoted as: $sim(d_1, d_2)$

- Requirements for the Similarity Measure:
  - **Symmetric:** $sim(d_1, d_2) = sim(d_2, d_1)$
  - **Normalized:** $sim(d_1, d_2) \in [0, 1]$

- The choice of similarity measure is closely linked to the representation of documents.

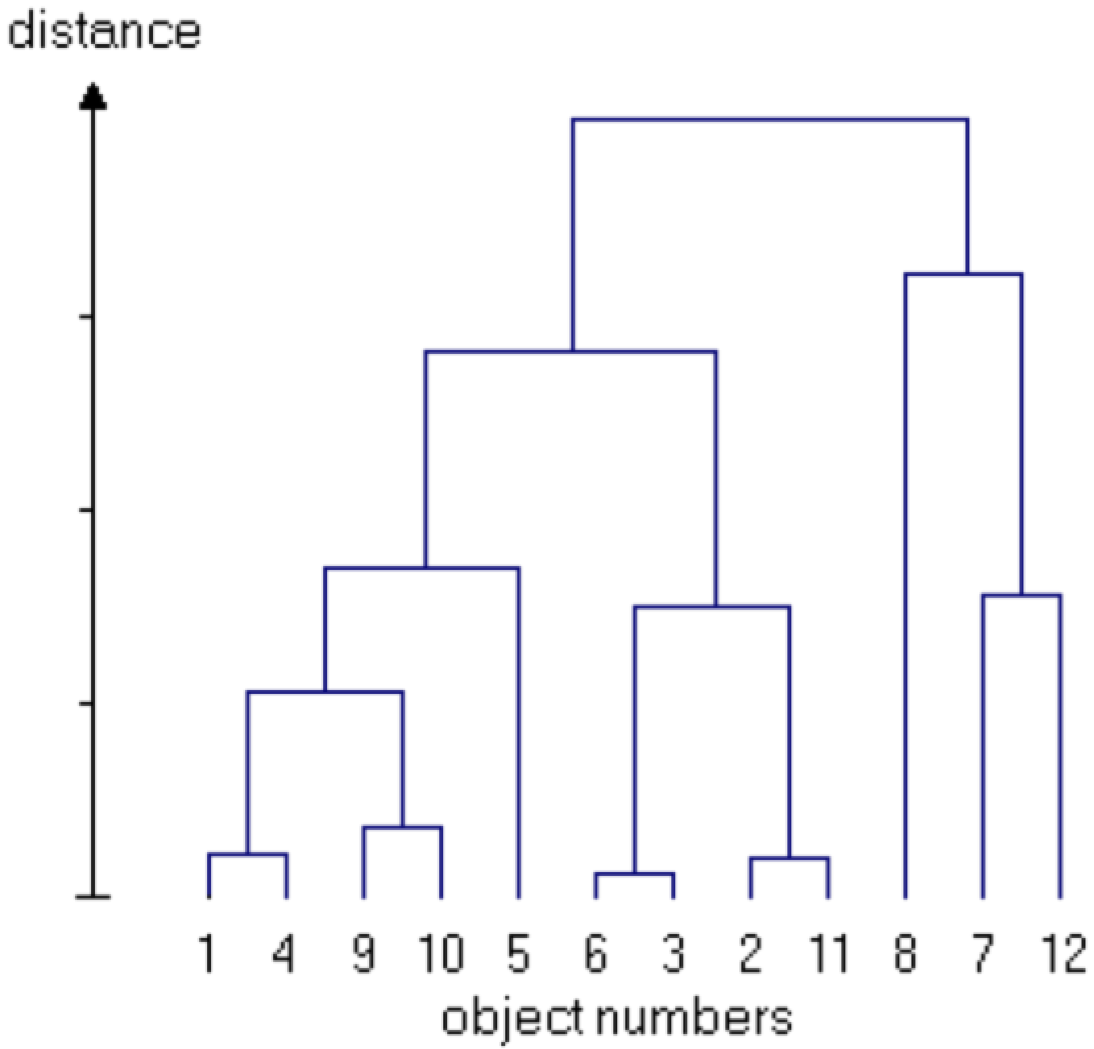
Agglomerative Hierarchical Clustering

- Progressively builds clusters to create a hierarchy of merged groups (bottom-up approach).

- **Process:**
  - Begin with each document as its own cluster.
  - Gradually merge clusters into larger groups until only one cluster remains.

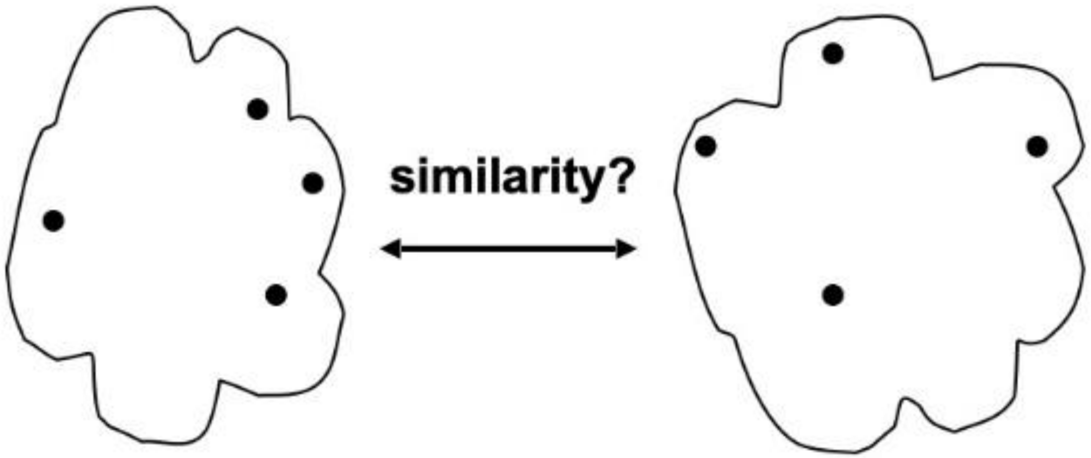
- **Output:** This series of merges results in a dendrogram.

- **Cluster Selection:**
  - The dendrogram can be segmented to obtain the desired number of clusters
  - Alternatively, merging can be stopped once the target number of clusters is reached.



Measuring Inter-Clustering Similarity

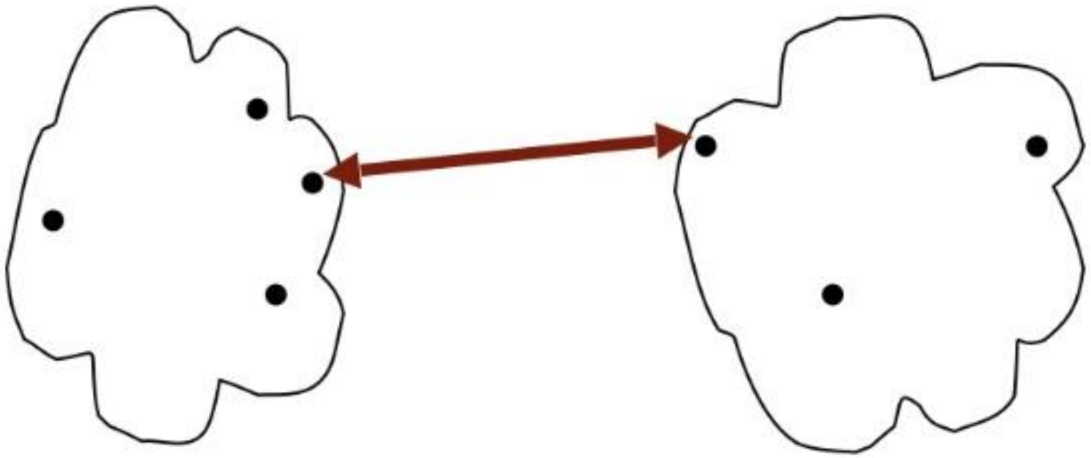
- Sinlge-link
- Complete-link
- Average-link
- Prototype-based (centroid)



Single-link ("min")

UNIVERSIDADE CATÓLICA PORTUGUESA BRAGA

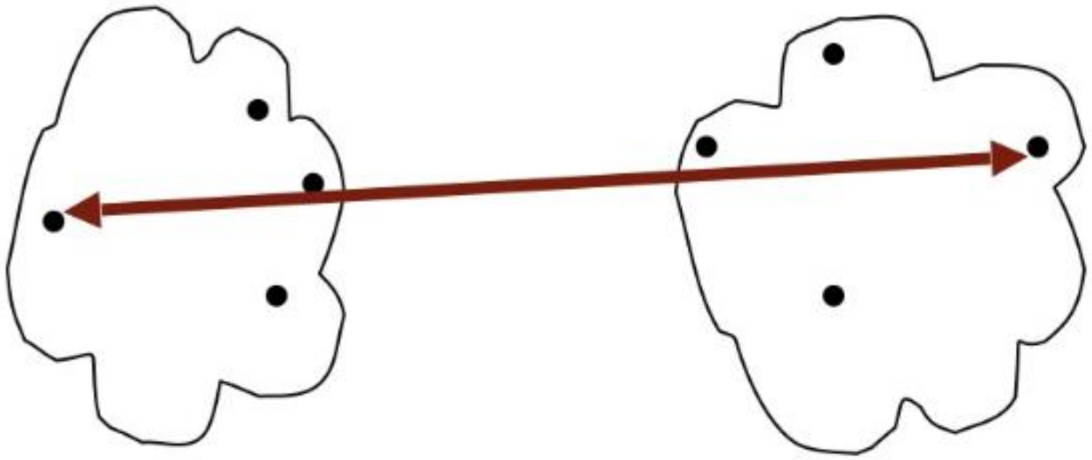
- Similarity of two clusters is based on the most similar (closest) points in the different clusters.
  - Results in "looser" clusters, as it can result in elongated shapes.



Complete-link ("max")

UNIVERSIDADE CATÓLICA PORTUGUESA
BRAGA

- Similarity of two clusters is based on the two least similar (most distant) points in the different clusters.
  - Results in “tight” and “compact” clusters (tends to break large clusters)

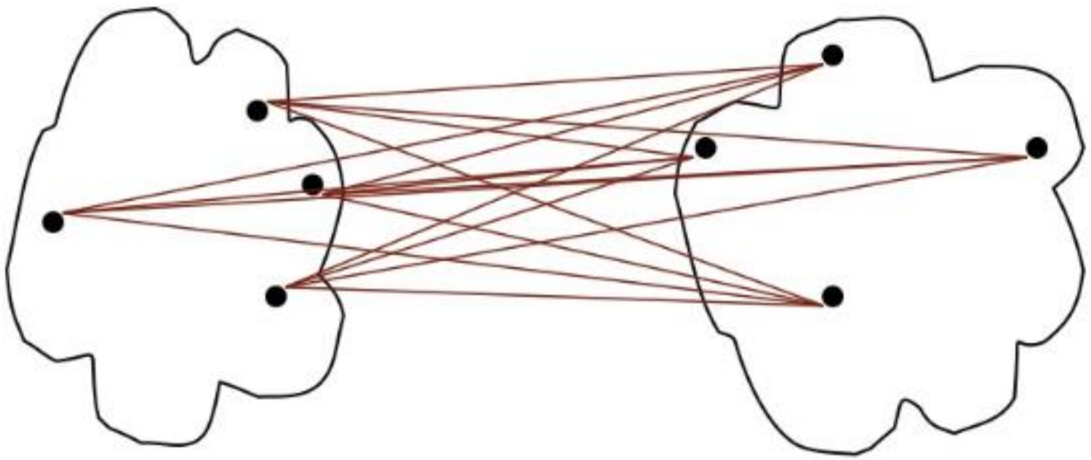


Average-link ("avg")

- Similarity of two clusters is the average of pairwise similarity between points in the two clusters
  - Less susceptible to noise and outliers than single- and complete-link

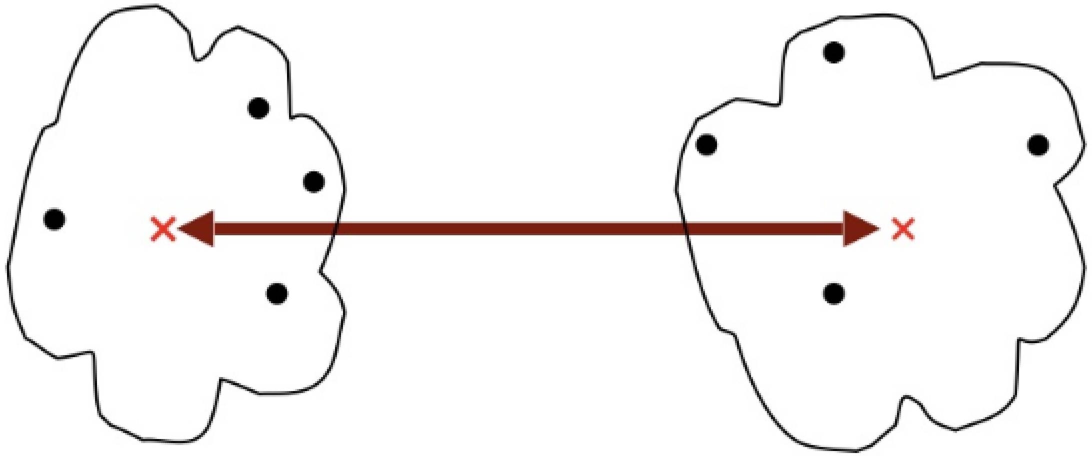
$$sim(C_i, C_j) = \frac{\sum_{x \in C_i, y \in C_j} sim(x, y)}{|C_i| \times |C_j|}$$

cardinality of a set
(number of elements
in a mathematical set)



Prototype-based (centroid)

- Represent clusters by their centroids and base their similarity on the similarity of the centroids.
  - To find the centroid, we need to compute the (arithmetic) mean of the points' positions separately for each dimension



K-means Clustering

UNIVERSIDADE CATÓLICA PORTUGUESA
BRAGA

- A form of **divisive clustering**.
- **Process:** Begins with an initial clustering and iteratively refines it until a stopping criterion is met.
- Each cluster is represented by a **centroid**, which is the average of all members' values within the cluster.
- Identifies a user-specified number of clusters, denoted as K.

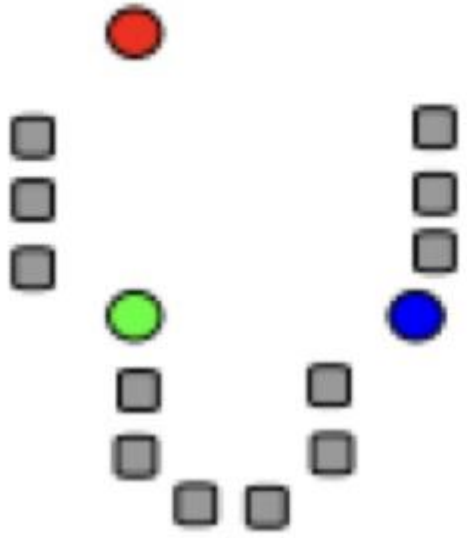
Basic K-means Algorithm

UNIVERSIDADE CATÓLICA PORTUGUESA
BRAGA

1. Select K points as initial centroids
2. **Repeat**
   2.1 Form K clusters by assigning wach point to its closest centroid
   2.2 Recompute the centroid of each cluster
3. **Until** centroids do not change

Basic K-means Algorithm

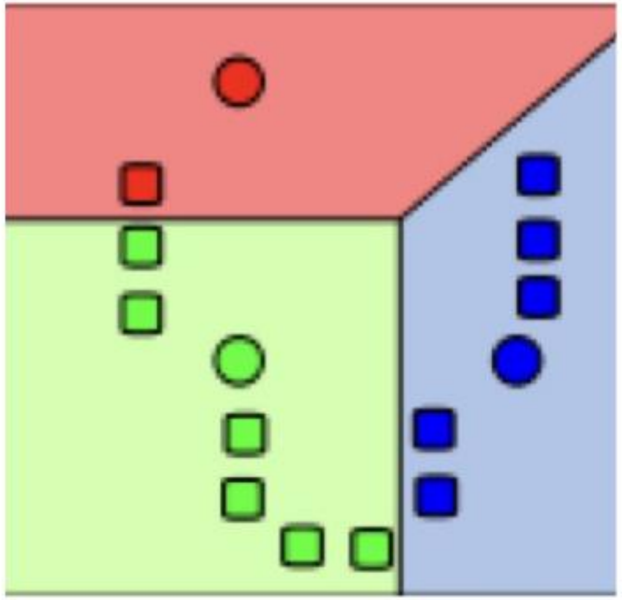
1. Select K points as initial centroids
2. Repeat
   2.1 Form K clusters by assigning wach point to its closest centroid
   2.2 Recompute the centroid of each cluster
3. **Until** centroids do not change



Basic K-means Algorithm

UNIVERSIDADE CATÓLICA PORTUGUESA
BRAGA

1. Select K points as initial centroids
2. **Repeat**
   2.1 **Form K clusters by assigning wach point to its closest centroid**
   2.2 Recompute the centroid of each cluster
3. **Until** centroids do not change



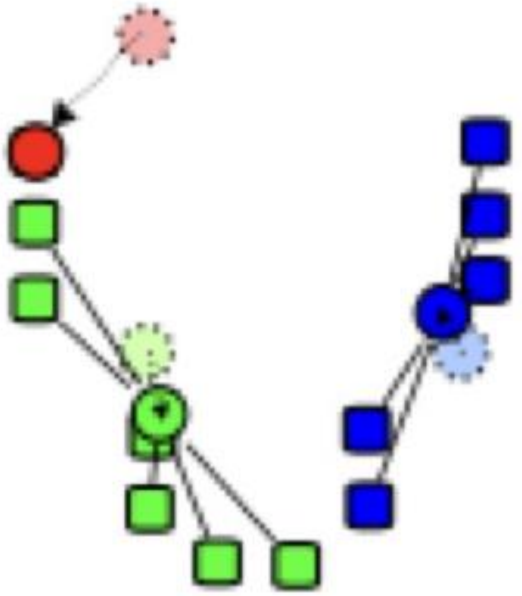
Basic K-means Algorithm

1. Select K points as initial centroids
2. **Repeat**

    2.1 Form K clusters by assigning wach point to its closest centroid

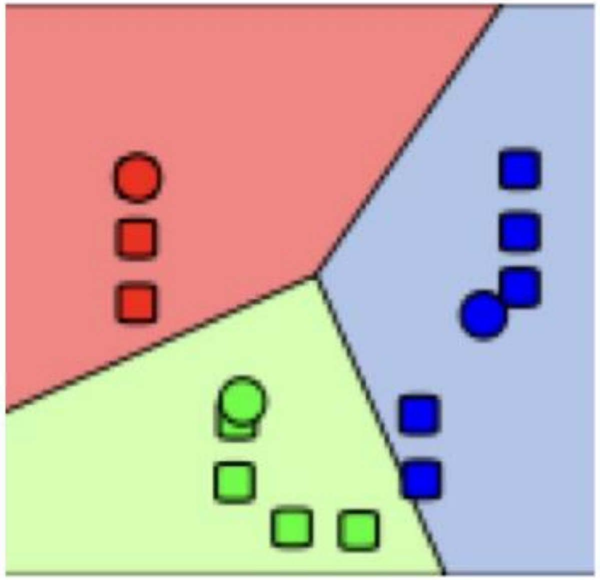
    2.2 Recompute the centroid of each cluster

3. **Until** centroids do not change



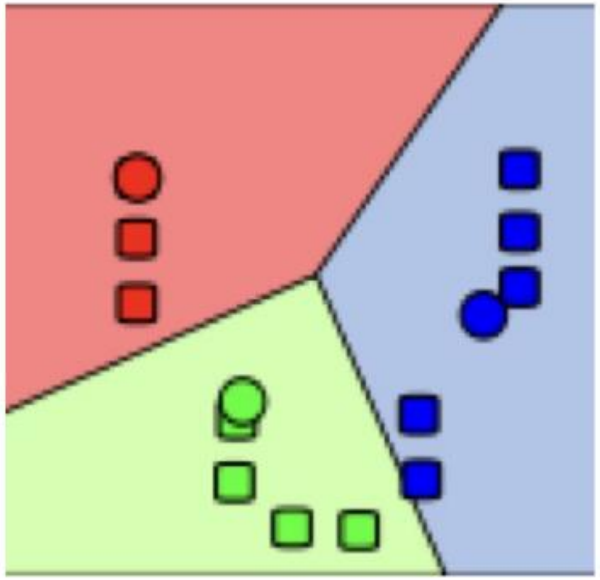
Basic K-means Algorithm

1. Select K points as initial centroids
2. **Repeat**
   2.1 Form K clusters by assigning wach point to its closest centroid
   2.2 Recompute the centroid of each cluster
3. **Until** centroids do not change



Basic K-means Algorithm

1. Select K points as initial centroids
2. **Repeat**
   2.1 Form K clusters by assigning wach point to its closest centroid
   2.2 Recompute the centroid of each cluster
3. **Until centroids do not change**



Components for text-data clustering