Unsupervised Learning: Clustering 2

Ninghao Liu

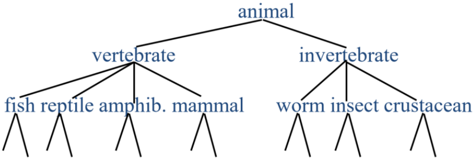
University of Georgia

February 15, 2024

Some contents adopted from "Data Mining", Chapter 10, by Jiawei Han et al.

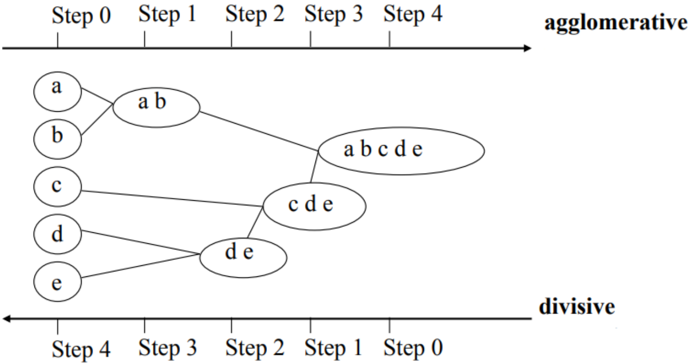
1. Hierarchical Clustering: Basic Concepts

- In some situations, we may want to partition our data into groups **at different levels** such as in a hierarchy.
- Grouping data objects into a “tree” of clusters.



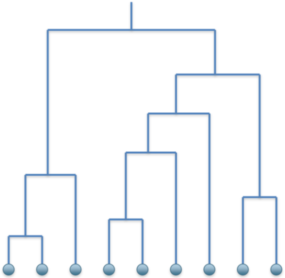
- Two types of methods:
  - **Agglomerative**: Start with individual objects as clusters, which are iteratively merged to form larger clusters.
  - **Divisive**: Initially let all the given objects form one cluster, which they iteratively split into smaller clusters.

1. Hierarchical Clustering: Basic Concepts



1. Hierarchical Clustering: Basic Concepts

- The tree structure is called a **dendrogram**, which represents the process of hierarchical clustering.
- Clusters are obtained by cutting the dendrogram at a desired level.
  - Each connected component forms a cluster.



2. Hierarchical Agglomerative Clustering

Algorithm sketch:

- Starts with each instance as a separate cluster.
- Repeatedly joins the **closest pair** of clusters as a single one.
- Stops when there is only one cluster.

The history of merging forms a binary tree or hierarchy.

How to define “closeness” between two clusters?

- Easy for two instances.
- Undefined for two groups of instances.

2.1 Closeness between Clusters

There are many ways to define the “closest pair” of clusters.

The similarity between two clusters is defined based on the similarity of instances between the two clusters.

- **Single-link**: Similarity of the most-similar instances.
- **Complete-link**: Similarity of the "furthest" instances.
- **Average-link**: Average similarity between all pairs of instances.

2.2 Single-Link

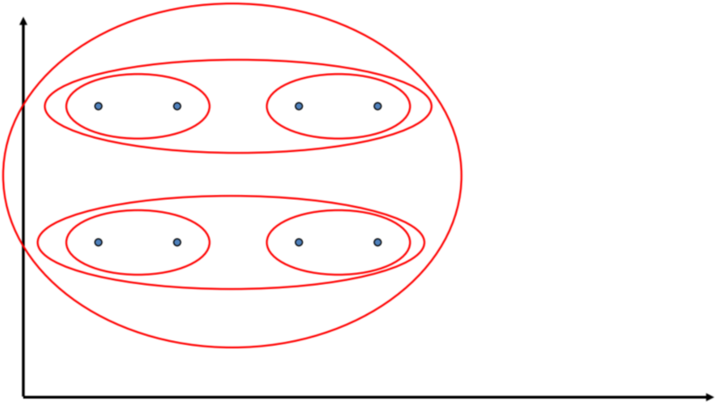
- Use maximum similarity of pairs:

$$sim(c_i, c_j) = \max_{x \in c_i, y \in c_j} sim(x, y)$$

- Can result in "straggly" (long and thin) clusters due to chaining effect.
- After merging c_i and c_j, the similarity of the resulting cluster to another cluster, c_k, is:

$$sim((c_i \cup c_j), c_k) = \max(sim(c_i, c_k), sim(c_j, c_k))$$

2.2 Single-Link



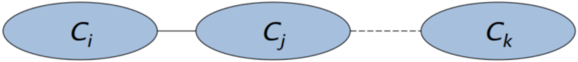
2.3 Complete Link

- Use minimum similarity of pairs:

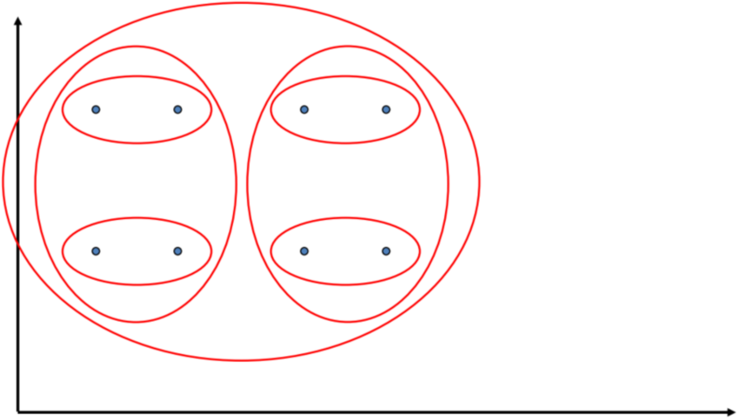
$$sim(c_i, c_j) = \min_{x \in c_i, y \in c_j} sim(x, y)$$

- Makes "tighter," spherical clusters that are typically preferable.
- After merging c_i and c_j, the similarity of the resulting cluster to another cluster, c_k, is:

$$sim((c_i \cup c_j), c_k) = \min(sim(c_i, c_k), sim(c_j, c_k))$$



2.3 Complete Link



2.4 Average-Link: Group Average

- Similarity of two clusters = average similarity of all pairs within merged cluster.

$$sim(c_i, c_j) = \frac{1}{|c_i \cup c_j|(|c_i \cup c_j| - 1)} \sum_{\vec{x} \in (c_i \cup c_j)} \sum_{\vec{y} \in (c_i \cup c_j): \vec{y} \neq \vec{x}} sim(\vec{x}, \vec{y})$$

- Compromise between single and complete link.
- Two options:
  - Averaged across all ordered pairs in the merged cluster
  - Averaged over all pairs *between* the two original clusters
- No clear difference in efficacy

3. Hierarchical Divisive Clustering

- Inverse order of agglomerative clustering.
- Practically, use heuristics in partitioning, because there are $\sim 2^n$ possible ways to partition a set of n objects into two exclusive subsets.
- For example, you can apply k-means multiple times, to iteratively partition a large set into multiple smaller subsets.
- There are many more agglomerative methods than divisive methods.