Chapter VII: Cluster analysis

Knowledge Discovery in Databases

Luciano Melodia M.A.
Evolutionary Data Management, Friedrich-Alexander University Erlangen-Nürnberg
Summer semester 2021

Chapter VII: Cluster analysis

Cluster analysis: basic concepts.

Partitioning methods.

Hierarchical methods.

Density-based methods.

Grid-based methods.

Evaluation of clustering.

Summary.

What is cluster analysis?

Cluster: A collection of data objects within a larger set that are.

Similar (or related) to one another within the same group and,
dissimilar (or unrelated) to the objects outside the group.

Cluster analysis (or clustering, data segmentation, . . .).

Define similarities among data based on the characteristics found in the data (input from user!).
Group similar data objects into clusters.

Unsupervised learning:

No predefined classes.
I.e., learning by observation (vs. learning by examples: supervised).

Typical applications:

As a stand-alone tool to get insight into data distribution.
As a preprocessing step for other algorithms.

Clustering for data understanding and applications

Biology:
Taxonomy of living things: kingdom, phylum, class, order, family, genus, and species.

Information retrieval:
Document clustering.

Land use:
Identification of areas of similar land use in an earth-observation database.

Marketing:
Help marketers discover distinct groups in their customer bases, and then use this knowledge to develop targeted marketing programs.

City planning:
Identifying groups of houses according to their house type, value, and geographical location.

Earthquake studies:
Observed earthquake epicenters should be clustered along continent faults.

Climate:
Understanding earth climate, find patterns of atmosphere and ocean.

Quality: what is good clustering?

A good clustering method will produce high-quality clusters.

- **High intra-class similarity:**
  - Cohesive within clusters.
- **Low inter-class similarity:**
  - Distinctive between clusters.

The quality of a clustering method depends on:

- the **similarity measure** used by the method,
- its implementation, and
- its ability to discover some or all of the hidden patterns.

Measure the quality of clustering

Dissimilarity/similarity metric:

Similarity is expressed in terms of a distance function, typically a metric: $d(x, y)$.
The definitions of distance functions are usually rather different for interval-scaled, Boolean, categorical, ordinal, ratio, and vector variables (see chapter 2).
Weights should be associated with different variables
based on applications and data semantics.

Quality of clustering:

There is usually a separate **"quality" function** that measures the "goodness" of a cluster.
It is hard to define "similar enough" or "good enough."
The answer is typically highly subjective.

Considerations for cluster analysis

Partitioning criteria:

Single level vs. hierarchical partitioning.
Often, multi-level hierarchical partitioning is desirable.

Separation of clusters:

Exclusive (e.g., one customer belongs to only one region) vs.
Non-exclusive (e.g., one document may belong to more than one class).

Similarity measure:

Distance-based (e.g., Euclidian, road network, vector) vs.
Connectivity-based (e.g., density or contiguity).

Clustering space:

Full space (often when low-dimensional) vs.
Subspaces (often in high-dimensional clustering).

Requirements and challenges

Scalability:

Clustering all the data instead of only on samples.

Ability to deal with different types of attributes:

Numerical, binary, categorical, ordinal, linked, and mixture of these.

Constraint-based clustering:

User may give inputs on constraints.

Use domain knowledge to determine input parameters.

Interpretability and usability.

Others:

Discovery of clusters with arbitrary shape.

Ability to deal with noisy data.

Incremental clustering and insensitivity to input order.

High dimensionality.

Major clustering approaches

Partitioning approach:

Construct various partitions and then evaluate them by some criterion.
E.g., minimizing the sum of square errors.
Typical methods: k-means, k-medoids, CLARA, CLARANS.

Hierarchical approach:

Create a hierarchical decomposition of the set of data (or objects) using some criterion.
Typical methods: AGNES, DIANA, BIRCH, CHAMELEON.

Density-based approach:

Based on connectivity and density functions.
Typical methods: DBSCAN, OPTICS, DENCLUE.

Grid-based approach:

Based on a multiple-level granularity structure.
Typical methods: STING, WaveCluster, CLIQUE.

Major clustering approaches (II)

Model-based approach:

A model is hypothesized for each of the clusters and tries to find the best fit of that model to each other.

Typical methods: EM, SOM, COBWEB.

Frequent-pattern-based approach:

Based on the analysis of frequent patterns.

Typical methods: p-Cluster.

User-guided or constraint-based approach:

Clustering by considering user-specified or application-specific constraints.

Typical methods: COD (obstacles), constrained clustering.

Link-based clustering:

Objects are often linked together in various ways.

Massive links can be used to cluster objects: SimRank, LinkClus.

Chapter VII: Cluster analysis

Cluster analysis: basic concepts.

Partitioning methods.

Hierarchical methods.

Density-based methods.

Grid-based methods.

Evaluation of clustering.

Summary.

Partitioning algorithms: basic concept

Partitioning method:

Partition a database D of n objects $o_j, j \in \{1, \ldots, n\}$ into a set of k-clusters C_i, $1 \leq i \leq k$ such that the sum of squared distances to c_i is minimized (where c_i is the **centroid** or **medoid** of cluster C_i):

$$\min \sum_{i=1}^{k} \sum_{o \in C_i} d(o, c_i)^2. \tag{1}$$

Given k, find a partition of k clusters that optimizes the chosen partitioning criterion.

Globally optimal: exhaustively enumerate all partitions.

Heuristic methods: k-means and k-medoids algorithms.

k-means (MacQueen'67, Lloyd'57/'82):

Each cluster is represented by the center of the cluster.

k-medoids or PAM (Partition around medoids) (Kaufman & Rousseeuw'87):

Each cluster is represented by one of the objects in the cluster.

The k-means clustering method

Given k, the k-means algorithm is implemented in four steps:

1. Partition the database into k non-empty subsets.

   E.g. the first $\frac{n}{k}$ objects, then the next $\frac{n}{k}$ objects, ...
2. Compute the centroids of the **clusters** of the current partitioning.

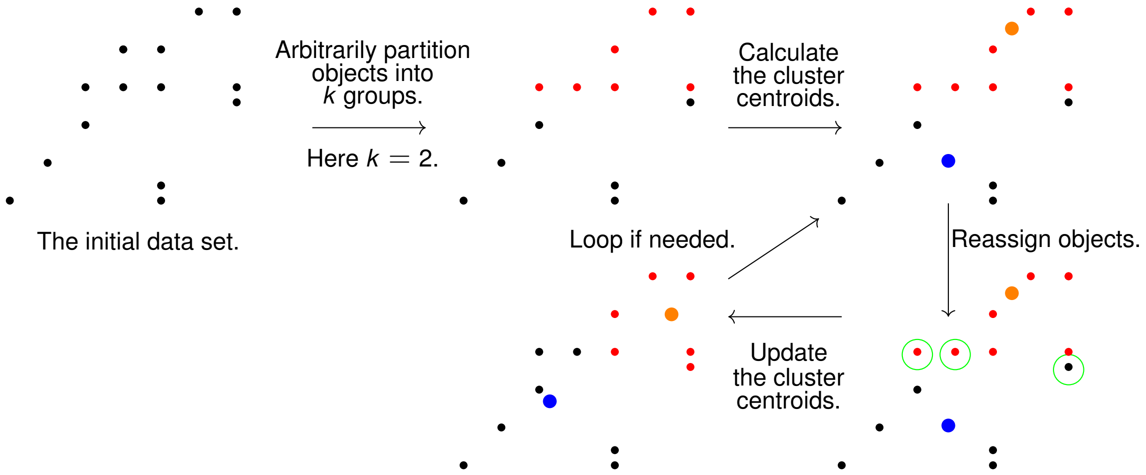
   The centroid is the center, i.e. mean point, of the cluster.
   For each attribute (or dimension), calculate the average value.
3. Assign each object to the cluster with the nearest centroid.

   That is, for each object calculate distance to each of the k centroids and pick the one with the smallest distance.
4. If any object has changed its cluster, go back to step 2. Otherwise stop.

Variant:

Start with arbitrarily chosen k objects as initial centroids in step 1.
Continue with step 3.

An example of *k*-means clustering



Comments on the *k*-means method

Strength:

- Efficient: $\mathcal{O}(tkn)$, where n is $\#$ objects, k is $\#$ of clusters, and t is the $\#$ of iterations. Normally, $k, t \ll n$.
- Comparing: PAM: $\mathcal{O}(k(n-k)^2)$, CLARA: $\mathcal{O}(ks^2 + k(n-k))$.
- Comment: Often terminates at a local optimum.

Weakness:

- Applicable only to objects in a continuous n-dimensional space.
  - Using the k-modes method for categorical data.
  - In comparison, k-medoids can be applied to a wide range of data.
- Need to specify k, the number of clusters, in advance.
  - There are ways to automatically determine the best k (see Hastie et al., 2009).
- Sensitive to noisy data and outliers.
- Not suitable to discover clusters with non-convex shapes.

Variations of the *k*-means method

Most of the variants of the *k*-means differ in:

- Selection of the initial *k* subsets (or centroids).
- Dissimilarity calculations.
- Strategies to calculate cluster centroids.

Handling categorical data: *k*-modes:

- Replacing centroids with modes.
  - See Chapter 2: mode = value that occurs most frequently in the data.
- Using new dissimilarity measures to deal with categorical objects.
- Using a frequency-based method to update modes of clusters.
- A mixture of categorical and numerical data: k-prototype method.

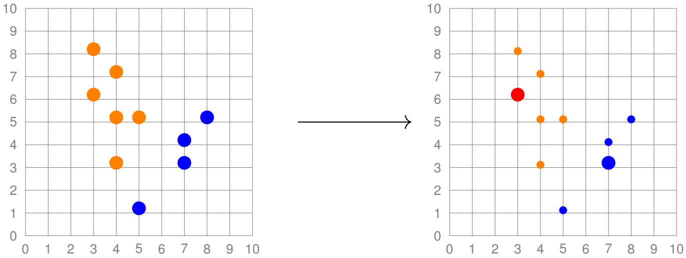
What is the problem of the *k*-means method?

The *k*-means algorithm is sensitive to outliers!

Since an object with an extremely large value may substantially distort the distribution of the data.

***k*-medoids:**

Instead of taking the mean value of the objects in a cluster as a reference point, medoids can be used, which is the most centrally located object in a cluster.



The *k*-medoids clustering method

k-medoids clustering:

Find representative objects (medoids) in clusters.

PAM (Partitioning Around Medoids, Kaufmann & Rousseeuw, 1987):

Starts from an initial set of k medoids and iteratively replaces one of the medoids by one of the non-medoids, if it improves the total distance of the resulting clustering.

PAM works effectively for small data sets, but does not scale well for large data sets (due to the computational complexity).

Efficiency improvement on PAM:

CLARA (Kaufmann & Rousseeuw, 1990): PAM on samples.

CLARANS (Ng & Han, 1994): Randomized re-sampling.

Thank you for your attention.

Any questions about the seventh chapter?

Ask them now, or again, drop me a line:

`luciano.melodia@fau.de`.