CSE 435: Data Mining

Cluster Analysis: Concepts, Methods, and Evaluation

Md Atikuzzaman
Lecturer
Department of Computer Science & Engineering
atik@cse.green.edu.bd

GREEN UNIVERSITY OF BANGLADESH

Outline

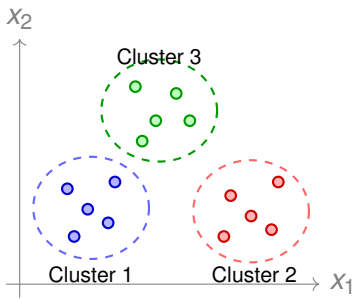
What is Cluster Analysis?

Core Idea

Cluster analysis groups data objects so that objects within the same group are highly similar, while objects in different groups are dissimilar.

Why do we use clustering?

- Discover hidden patterns without class labels.
- Summarize large datasets into meaningful groups.
- Support customer segmentation, image grouping, anomaly detection, document organization, and bioinformatics.



Clustering vs. Classification

Classification

- Supervised learning.
- Class labels are known during training.
- Goal: learn a function that predicts labels.

Example: Predict whether an email is spam or not spam.

Clustering

- Unsupervised learning.
- No predefined class labels.
- Goal: discover natural group structure.

Example: Group customers based on purchasing behavior.

In clustering, the meaning of each cluster is interpreted after the groups are formed.

Formal Clustering Problem

Input

A dataset $D = \{x_1, x_2, \ldots, x_n\}$, where each object x_i is described by a set of attributes.

Output

A partition of the data into clusters $C_1, C_2, \ldots, C_k$ such that:

$$C_i \cap C_j = \emptyset, \quad \bigcup_{i=1}^{k} C_i = D$$

High intra-cluster similarity:

$x_a, x_b \in C_i \Rightarrow sim(x_a, x_b)$ is high

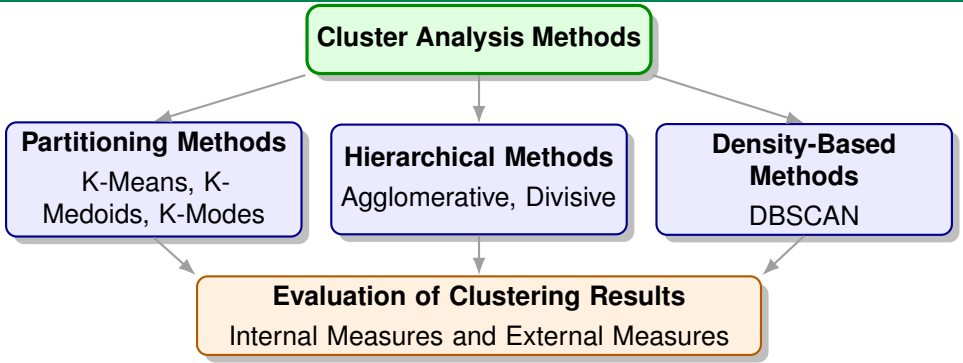
Low inter-cluster similarity:

$x_a \in C_i, x_b \in C_j \Rightarrow sim(x_a, x_b)$ is low

Similarity and Distance Measures

Measure	Formula	Best Use
Euclidean	$d(x, y) = \sqrt{\sum_i (x_i - y_i)^2}$	Numerical data with continuous features.
Manhattan	$d(x, y) = \sum_i \lvert x_i - y_i \rvert$	Grid-like movement or robust simple distance.
Cosine similarity	$\cos(x, y) = \frac{x \cdot y}{\lVert x \rVert \lVert y \rVert}$	Text mining and high-dimensional sparse data.
Hamming	Count of mismatched attributes	Binary or categorical strings.
Matching dissimilarity	Fraction of non-matching attributes	Categorical clustering such as K-Modes.

Major Types of Clustering Methods



Partitioning Methods: Basic Idea

Definition

Partitioning methods divide n objects into k clusters, where each object belongs to exactly one cluster.

Common Assumptions

- Number of clusters k is usually given.
- Clusters are represented by a center or representative object.
- The algorithm iteratively improves cluster assignments.

Popular Algorithms

- **K-Means:** mean as cluster center.
- **K-Medoids:** actual object as center.
- **K-Modes:** mode for categorical data.

K-Means Clustering: Objective Function

Goal

Divide data into k clusters by minimizing the within-cluster sum of squared errors.

$$J = \sum_{j=1}^{k} \sum_{x_i \in C_j} \|x_i - \mu_j\|^2$$

Where:

- C_j = cluster j
- μ_j = mean/centroid of cluster j
- k = number of clusters

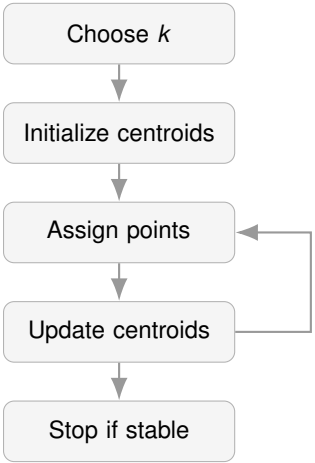
Interpretation

K-Means tries to make each cluster compact by keeping points close to their centroid.

K-Means Algorithm

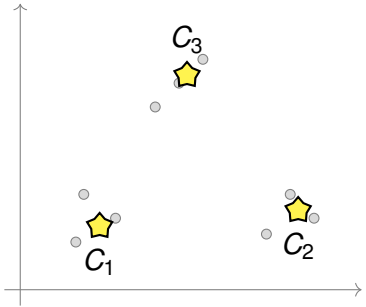
Steps

1. Choose the number of clusters k.
2. Initialize k centroids randomly.
3. Assign each data point to the nearest centroid.
4. Recalculate centroids using assigned points.
5. Repeat assignment and update until convergence.



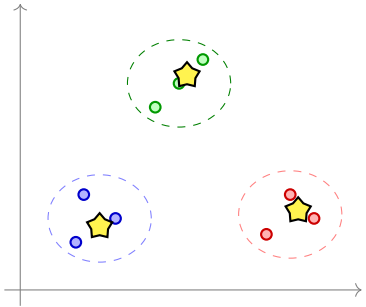
K-Means: Step-by-Step Visual Intuition

Step 1: Randomly choose initial centroids



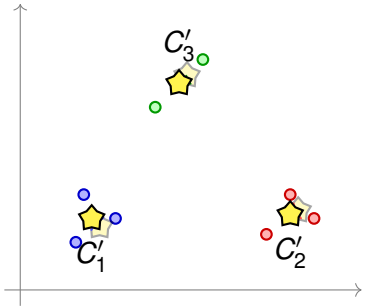
K-Means: Step-by-Step Visual Intuition

Step 2: Assign each data point to the nearest centroid



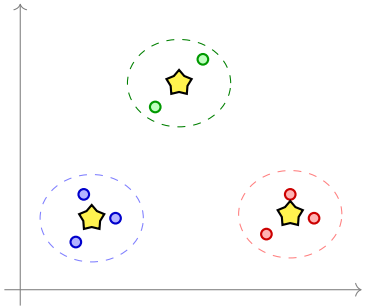
K-Means: Step-by-Step Visual Intuition

Step 3: Update centroid positions using cluster means



K-Means: Step-by-Step Visual Intuition

Step 4: Repeat until clusters become stable



Final clusters are formed after repeated assignment and update steps.

K-Means: Strengths and Limitations

Strengths

- Simple and easy to implement.
- Efficient for large numerical datasets.
- Works well for compact, spherical clusters.

Limitations

- Requires predefined k.
- Sensitive to initial centroids.
- Sensitive to outliers.
- Performs poorly for non-spherical or unequal-density clusters.
- Not suitable for categorical data directly.

K-Medoids Clustering

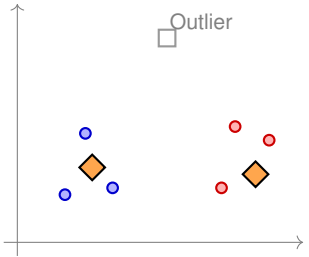
Core Idea

K-Medoids uses an actual data object as the representative of each cluster, called a **medoid**.

Objective

$$J = \sum_{j=1}^{k} \sum_{x_i \in C_j} d(x_i, m_j)$$

where m_j is the medoid of cluster C_j.



Why medoids?

- More robust to outliers than K-Means.
- Can use any distance measure.

PAM Algorithm for K-Medoids

PAM: Partitioning Around Medoids

PAM searches for a good set of medoids by repeatedly trying swaps between medoids and non-medoids.

1. Select k initial medoids.
2. Assign each object to the nearest medoid.
3. For each medoid and non-medoid pair, test a swap.
4. Compute total cost after the swap.
5. Keep the swap if it reduces the total cost.
6. Stop when no swap improves the clustering.

Key Point

PAM is more robust but usually slower than K-Means, especially for large datasets.

K-Modes Clustering for Categorical Data

Problem with K-Means

The mean is not meaningful for categorical attributes such as color, gender, department, or city.

K-Modes Solution

- Uses **mode** instead of mean.
- Uses matching dissimilarity instead of Euclidean distance.
- Updates cluster representatives by the most frequent category in each attribute.

Matching Dissimilarity

$$d(x, y) = \sum_{i=1}^{m} \delta(x_i, y_i)$$

where

$$\delta(x_i, y_i) = \begin{cases} 0, & x_i = y_i \\ 1, & x_i \neq y_i \end{cases}$$

K-Modes: Simple Example

Student	Dept.	Device	Internet Use
S1	CSE	Laptop	High
S2	CSE	Laptop	High
S3	EEE	Desktop	Medium
S4	CSE	Laptop	Medium
S5	BBA	Mobile	Low
S6	BBA	Mobile	Low

Possible Cluster Modes

- Cluster 1 mode: (CSE, Laptop, High/Medium)
- Cluster 2 mode: (BBA, Mobile, Low)

Comparison of Partitioning Methods

Method	Data Type	Representative	Main Advantage
K-Means	Numerical	Mean / Centroid	Fast and simple for large datasets.
K-Medoids	Numerical or mixed data	Actual object / Medoid	More robust to outliers.
K-Modes	Categorical	Mode	Suitable for categorical attributes.

Practical Rule

Use **K-Means** for clean numerical data, **K-Medoids** when outliers matter, and **K-Modes** for categorical attributes.

Hierarchical Clustering: Basic Concept

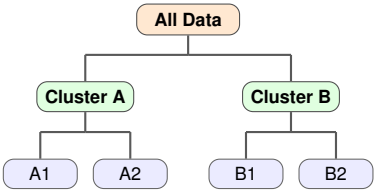
Definition

Hierarchical clustering creates a tree-like structure of nested clusters. The result is commonly visualized using a **dendrogram**.

Two Approaches

- **Agglomerative:** Bottom-up approach. Each object starts as a separate cluster.
- **Divisive:** Top-down approach. All objects start in one cluster and are split recursively.

Dendrogram View



Objects merge or split to form nested clusters

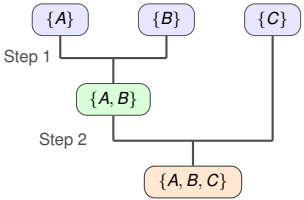
Key Idea

Agglomerative Hierarchical Clustering

Algorithm

1. Start with each object as its own cluster.
2. Compute distances between all clusters.
3. Merge the two closest clusters.
4. Update the distance matrix.
5. Repeat until one cluster remains or a stopping condition is met.

Bottom-Up Merging



Closest clusters are merged step by step

Key Idea

Agglomerative clustering follows a **bottom-up** process where small clusters are gradually merged into larger clusters.

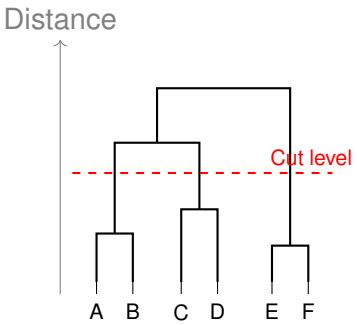
Linkage Criteria

Linkage	Cluster Distance	Behavior
Single linkage	Minimum pairwise distance	Can detect elongated shapes but may create chaining effect.
Complete linkage	Maximum pairwise distance	Produces compact clusters; sensitive to outliers.
Average linkage	Average pairwise distance	Balanced and commonly used.
Ward's method	Increase in within-cluster variance	Produces compact and spherical clusters.

Dendrogram and Cluster Selection

How to read a dendrogram

- Leaves represent data objects.
- Vertical height represents merge distance.
- Cutting the dendrogram at a chosen height gives the clusters.



Hierarchical Clustering: Pros and Cons

Advantages

- Does not require predefined k at the beginning.
- Dendrogram helps visualize cluster structure.
- Useful for small to medium datasets.

Limitations

- Computationally expensive for large datasets.
- Once a merge or split is made, it cannot be undone.
- Sensitive to distance measure and linkage choice.

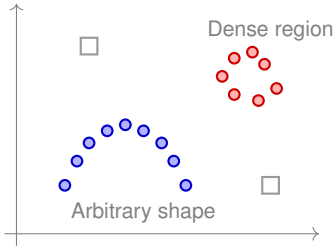
Density-Based Clustering: Motivation

Core Idea

Clusters are dense regions of data points separated by regions of lower density.

Why density-based methods?

- Can discover clusters of arbitrary shape.
- Can identify noise and outliers.
- Does not require specifying number of clusters in advance.



DBSCAN: Key Concepts

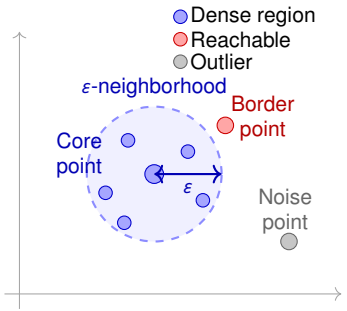
DBSCAN Parameters

- ε **(Eps):** Radius of the neighborhood around a point.
- **MinPts:** Minimum number of points required to form a dense region.

Point Types

- **Core point:** Has at least MinPts points within ε.
- **Border point:** Not a core point, but reachable from a core point.
- **Noise point:** Neither core nor border.

Density-Based Neighborhood



DBSCAN Algorithm

1. Select an unvisited point p.
2. Find all points within distance ϵ from p.
3. If p has at least MinPts neighbors, create a new cluster.
4. Expand the cluster by adding all density-reachable points.
5. If p does not satisfy the density condition, mark it as noise temporarily.
6. Continue until all points are visited.

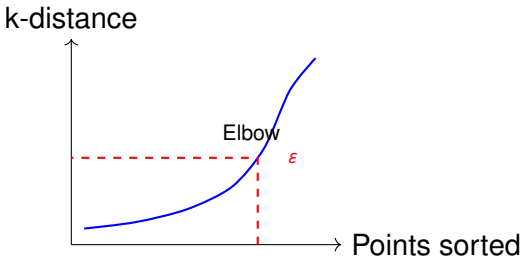
Important Note

A point initially marked as noise may later become a border point if it is found within the neighborhood of a core point.

Choosing ε and MinPts

Guidelines

- MinPts is often set based on dimensionality and domain knowledge.
- ε can be selected using a k-distance plot.
- The "elbow" in the sorted k-distance graph is a common choice for ε.



Parameter Sensitivity

Too small ε creates many noise points. Too large ε may merge different clusters.

DBSCAN: Strengths and Limitations

Strengths

- Finds clusters with arbitrary shapes.
- Detects noise and outliers.
- Does not require the number of clusters k.

Limitations

- Sensitive to ε and MinPts.
- Struggles when clusters have different densities.
- Distance measures become less meaningful in very high dimensions.

Why Evaluate Clustering?

Challenge

Since clustering is unsupervised, there may be no true labels. Evaluation checks whether the discovered groups are meaningful, compact, and well separated.

Evaluation Questions

- Are objects inside the same cluster close to each other?
- Are different clusters clearly separated?
- Is the number of clusters appropriate?
- Are the clusters meaningful for the domain problem?

Evaluation Types

- **Internal:** uses only the dataset.
- **External:** compares with known labels.
- **Relative:** compares different clustering results.

Internal Evaluation Measures

Measure	Main Idea	Interpretation
SSE	Sum of squared distances from points to centroids	Lower is better, but decreases as k increases.
Silhouette coefficient	Combines cohesion and separation	Range $[-1, 1]$; closer to 1 is better.
Davies-Bouldin index	Average similarity between clusters	Lower is better.
Dunn index	Ratio of minimum inter-cluster distance to maximum intra-cluster diameter	Higher is better.

Silhouette Coefficient

Definition

For a point i, let $a(i)$ be the average distance to points in its own cluster and $b(i)$ be the smallest average distance to another cluster.

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

$s(i) \approx 1$	$s(i) \approx 0$	$s(i) < 0$
Well matched to its cluster.	Lies near a cluster boundary.	Possibly assigned to the wrong cluster.

External Evaluation Measures

When to Use

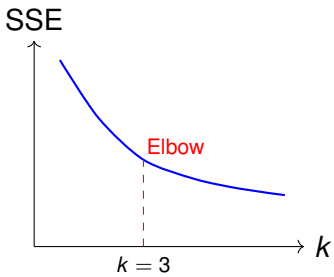
External measures are used when reference labels or ground truth classes are available.

Measure	Meaning
Purity	Measures how much each cluster contains a single class.
Rand Index	Measures agreement between clustering and true labels over object pairs.
Adjusted Rand Index	Corrects Rand Index for chance agreement.
Normalized Mutual Information	Measures shared information between clusters and true classes.

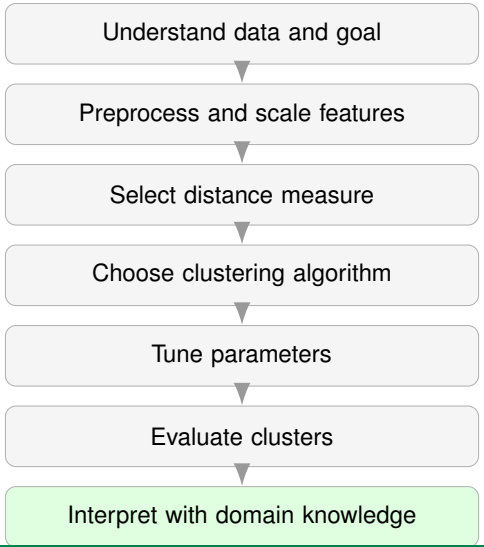
Choosing the Number of Clusters

Common Techniques

- **Elbow method:** choose k where SSE reduction slows down.
- **Silhouette method:** choose k with high average silhouette score.
- **Dendrogram cut:** choose a meaningful cut level.
- **Domain validation:** check whether clusters make practical sense.



Practical Clustering Workflow



Lecture Summary & Key Takeaways

- **Cluster analysis** discovers natural groups in unlabeled data.
- **Partitioning methods** such as K-Means, K-Medoids, and K-Modes create k clusters using different representatives.
- **Hierarchical clustering** builds nested clusters and represents them using a dendrogram.
- **DBSCAN** discovers dense regions, arbitrary-shaped clusters, and noise points.
- **Evaluation** is essential because clustering results must be checked for compactness, separation, stability, and domain meaning.

References

1. J. Han, M. Kamber, J. Pei, *Data Mining: Concepts and Techniques*, 4th Ed., Morgan Kaufmann, 2012.
2. A. K. Jain, M. N. Murty, P. J. Flynn, “Data clustering: A review,” *ACM Computing Surveys*, 1999.
3. L. Kaufman and P. J. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis*, Wiley, 1990.
4. M. Ester, H. P. Kriegel, J. Sander, X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise,” KDD, 1996.