STAT 24620=FINM 34700, STAT 32950
Multivariate Data Analysis
Lecture 5: Clustering

Jingshu Wang

The University of Chicago

Outline

From continuous to discrete latent structure

- Lecture 1: covariance geometry and multivariate notation.
- Lectures 2–3: PCA summarizes variation through low-dimensional projections.
- Lecture 4: factor analysis models covariance via *continuous* latent variables.
- Today: what if latent structure is *discrete* rather than continuous?

Core clustering question:

Do observations belong to distinct, unobserved groups?

Looking ahead:

Clustering algorithms give *group assignments*;
mixture models will give a *probabilistic model* for these groups.

What is clustering?

- Input: observations $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^p$.
- Goal: partition observations into groups so that within-cluster similarity is high and between-cluster similarity is low.
- Output is usually exploratory rather than a definitive "truth."

Typical use cases

- customer segmentation: group users by purchasing behavior;
- genomics: cluster genes with similar expression patterns;
- finance: identify assets that move together (market sectors or regimes);
- text analysis: group documents by topic without labels;
- preprocessing: uncover structure before regression or classification.

Clustering is not classification

Classification (supervised)

- training data include labels (e.g., spam vs. not spam);
- goal: predict labels for new observations;
- evaluation is direct (accuracy, error rate).

Clustering (unsupervised)

- no labels are given;
- goal: discover structure or groups in the data;
- evaluation is indirect and context-dependent.

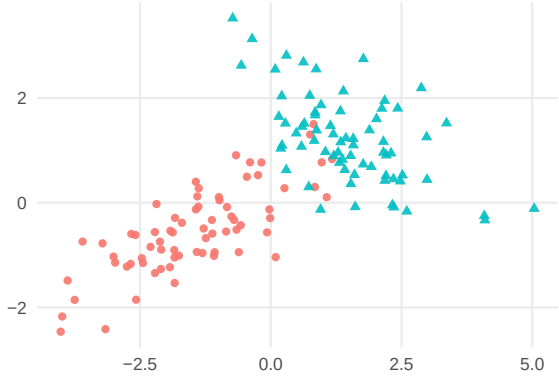
Important caution: there is no single "true" clustering; different methods impose different notions of a cluster.

Clustering v.s. Classification

Same data, different task

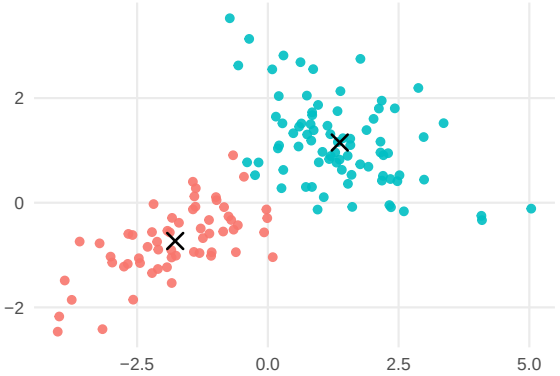
Classification

Labels are observed



Clustering

Labels are hidden; groups are inferred



Distance is the first modeling choice

For many clustering methods, the algorithm only sees a notion of pairwise dissimilarity.

- **Euclidean distance:**

$$d_E(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{\sum_{r=1}^{p} (x_{ir} - x_{jr})^2}.$$

- **Manhattan distance:**

$$d_M(\mathbf{x}_i, \mathbf{x}_j) = \sum_{r=1}^{p} |x_{ir} - x_{jr}|.$$

- Correlation-based dissimilarity can be useful when *direction* matters more than *magnitude*.

Takeaway: different distance choices can produce different clusterings even on the same data.

Distances

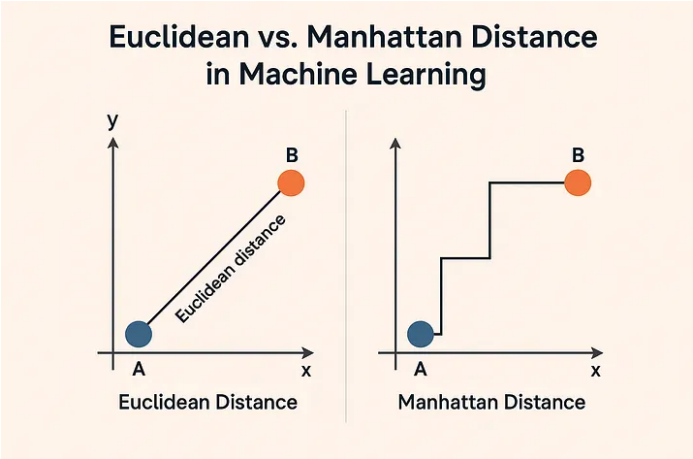


Figure: https://ai.plainenglish.io/euclidean-vs-manhattan-distance-in-machine-learning-9e93e1e16790

Scaling changes the notion of distance

- Many clustering methods rely on Euclidean distance.
- If one variable has much larger variance, it dominates the distance.
- Then clustering may reflect scale rather than meaningful structure.

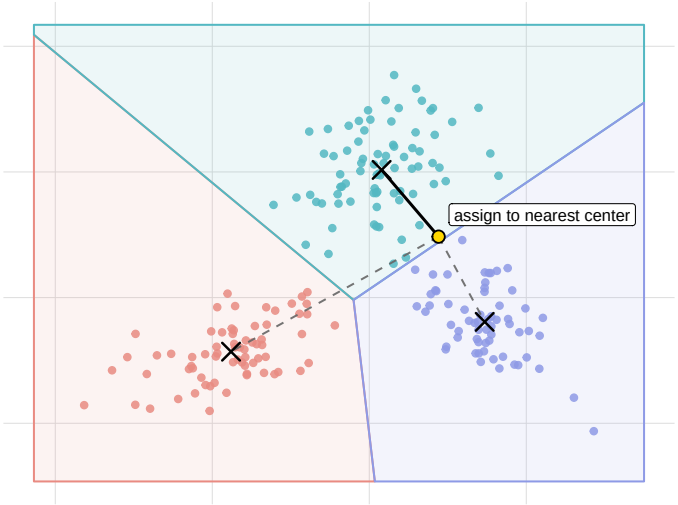
Implication:

- Standardizing variables changes the geometry of the data.
- As a result, it can lead to very different clusterings.

Takeaway: preprocessing is part of the model, not a neutral step.

A geometric picture of clustering

Centers are learned from the data, then each point is assigned to its nearest center

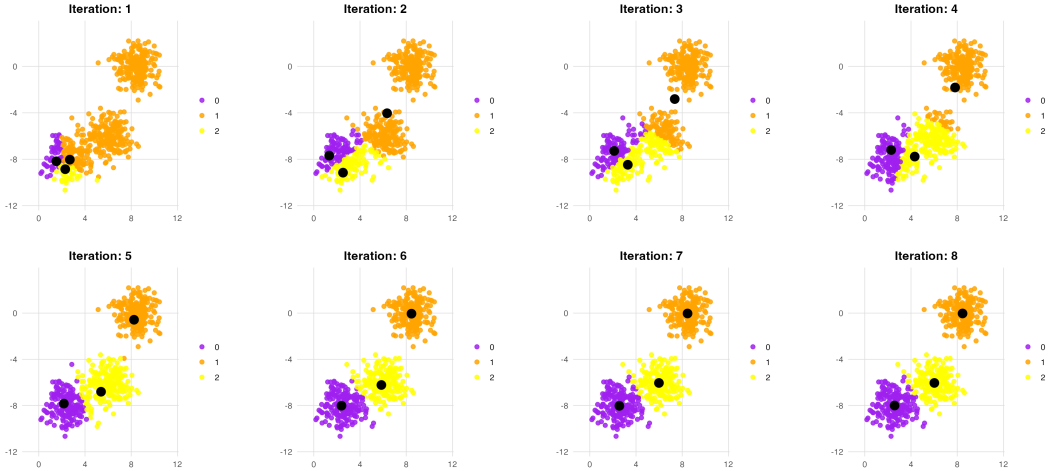


K-means: intuition

- Suppose we want to summarize the data using K representative points.
- Each representative (centroid) stands for a group of nearby observations.
- Assign each observation to its nearest centroid.
- Then update each centroid to be the average of its assigned points.
- Repeat until things stabilize.

Key idea: clustering alternates between *assignment* and *updating representatives*.

K-means in action



Black dots are the current centroids; colors show current assignments.

K-means objective function

Let $C_1, \ldots, C_K$ be a partition of $\{1, \ldots, n\}$. The K-means criterion is

$$\min_{C_1,\ldots,C_K} \sum_{k=1}^{K} \sum_{i \in C_k} \|\mathbf{x}_i - \bar{\mathbf{x}}_{C_k}\|_2^2,$$

where

$$\bar{\mathbf{x}}_{C_k} = \frac{1}{|C_k|} \sum_{i \in C_k} \mathbf{x}_i$$

is the centroid of cluster k.

- This is the **within-cluster sum of squares** (WCSS).
- Good clustering means small within-cluster variation relative to total variation.
- But the optimization is non-convex: local minima are common.

Why is the centroid the right representative?

Fix one cluster C. Consider minimizing

$$\sum_{i \in C} \|\mathbf{x}_i - \boldsymbol{\mu}\|_2^2 \quad \text{over } \boldsymbol{\mu} \in \mathbb{R}^p.$$

Take derivative with respect to $\boldsymbol{\mu}$:

$$-2 \sum_{i \in C} (\mathbf{x}_i - \boldsymbol{\mu}) = 0 \quad \Longrightarrow \quad \boldsymbol{\mu} = \frac{1}{|C|} \sum_{i \in C} \mathbf{x}_i.$$

Conclusion: given assignments, the best center is the sample mean of that cluster.

This is why the algorithm alternates naturally between **assignment** and **mean update** steps.

Lloyd's algorithm for K-means

1. Initialize K centroids.
2. **Assignment step:** assign each observation to the nearest centroid.
3. **Update step:** recompute each centroid as the mean of its assigned points.
4. Each iteration decreases the objective function.
5. Stop when assignments or objective value stabilize.

Practical implications

- Initialization matters: different starting points can lead to different local minima.
- In practice, use multiple random starts (e.g., `nstart` in R).
- Poor choices of K or weak separation can lead to unstable or degenerate solutions.

Other variants (e.g., MacQueen, Hartigan–Wong) differ in how they update assignments and centers.

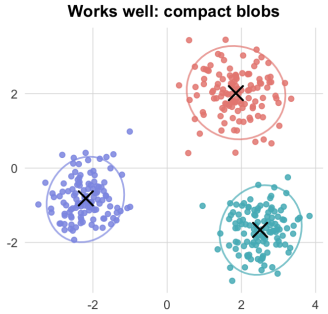
Strengths and limitations of K-means

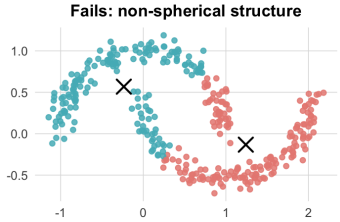
Strengths

- simple and fast;
- scales well to moderate/large n;
- easy to explain geometrically.

Limitations

- requires choosing K in advance;
- sensitive to scaling and outliers;
- struggles with non-spherical or unequal-density clusters.





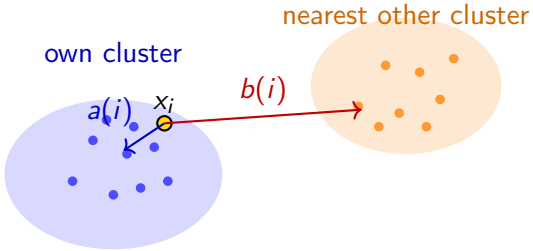
How do we choose K?

No single rule is universally correct. In practice, we combine several diagnostics.

- **Elbow method:** plot within-cluster sum of squares (WCSS) versus K and look for diminishing returns.
- **Silhouette width:** compares how close a point is to its own cluster versus the nearest other cluster.
- **Stability:** do clusters persist under resampling or small perturbations?
- **Interpretability:** does the chosen K answer a meaningful question?

Key message: choosing K is a modeling decision, guided by both data and context.

Silhouette intuition



$$a(i) = \frac{1}{|C(i)| - 1} \sum_{j \in C(i), j \neq i} d(x_i, x_j), \qquad b(i) = \min_{\ell \neq C(i)} \frac{1}{|C_\ell|} \sum_{j \in C_\ell} d(x_i, x_j)$$

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

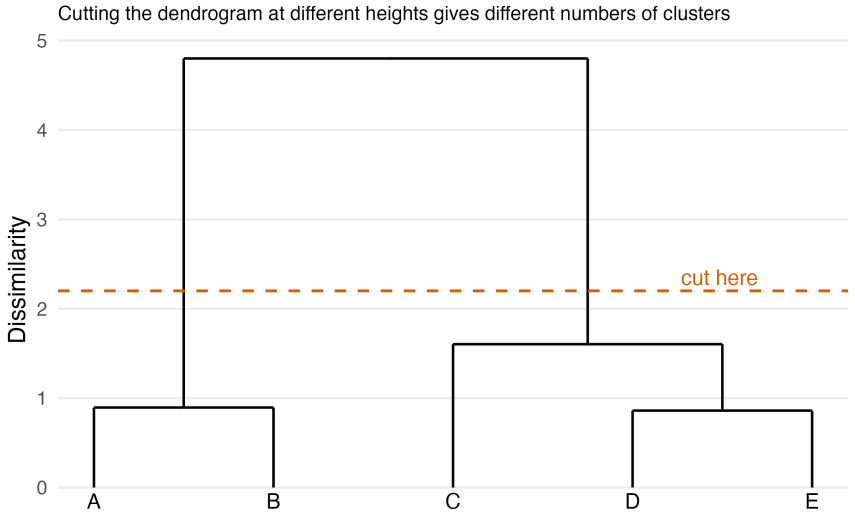
Interpretation: a good clustering has small $a(i)$, large $b(i)$, and hence silhouette close to 1.

Hierarchical clustering: a different philosophy

- Instead of choosing K in advance, we build *a hierarchy of clusterings*.
- Start with very fine groups and progressively combine them.
- **Agglomerative:** start with singletons and merge groups.
- **Divisive:** start with one group and split it.
- In practice, agglomerative methods are most common.

Key idea: clustering is viewed as a *tree structure*, not a single partition.
Output: a dendrogram records the order and dissimilarity of merges.

The dendrogram from hierarchical clustering



Linkage defines how clusters talk to each other

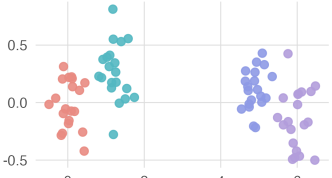
If A and B are two clusters, we need a notion of **dissimilarity** between sets of points.

- **Single linkage:**
$$d(A, B) = \min_{i \in A, j \in B} d(\mathbf{x}_i, \mathbf{x}_j)$$
(can connect clusters through nearby points, forming long chains)
- **Complete linkage:**
$$d(A, B) = \max_{i \in A, j \in B} d(\mathbf{x}_i, \mathbf{x}_j)$$
(prefers clusters where all points are close to each other)
- **Average linkage:** average pairwise distance across the two groups.
- **Ward's method:** merge the pair that increases within-cluster variation the least.

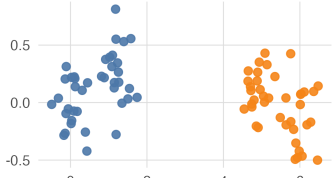
In practice, average linkage or Ward's method are often more stable choices.

Reading a dendrogram

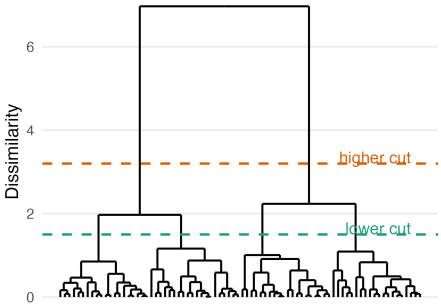
Lower cut
Finer partition: 4 clusters



Higher cut
Coarser partition: 2 clusters



Dendrogram
Different cut heights give different clusterings



- Lower cuts give finer subgroups; higher cuts give coarser groups.
- Merge height reflects dissimilarity at the time of merging.

Hierarchical clustering: pros and cons

Advantages

- no need to fix K at the start;
- dendrogram provides a multiscale summary;
- works with arbitrary dissimilarity matrices.

Drawbacks

- greedy merges cannot be undone (early mistakes persist);
- results depend strongly on the choice of linkage;
- can be expensive for large n (typically $O(n^2)$).

Remark: a dendrogram depends on both the chosen dissimilarity and linkage, not a "true" underlying tree.

Live R notebook

Notebook file: `Lecture5_demo.html`

What to keep in mind when clustering

- **Clustering is a modeling choice:** distance, linkage, and K all matter.
- **Different choices $\Rightarrow$ different answers:** there is no single "correct" clustering.
- **Interpret cautiously:** clusters are summaries of structure, not ground truth.
- **Look for stability:** do conclusions persist under reasonable changes?

Takeaway: clustering is most useful when it leads to a simple, stable, and interpretable summary of the data.

Clustering and mixture models

- Today we focus on algorithmic/geometric clustering methods.
- Next lecture: mixture models provide a probabilistic approach.
- Hard assignment now:

$$\text{observation } i \in \{1, \ldots, K\}.$$

- Soft assignment later:

$$\Pr(Z_i = k \mid \mathbf{x}_i), \quad k = 1, \ldots, K.$$

This bridge is important: mixture models can justify clustering through an explicit distributional model.

Lecture 5 summary

- Clustering is an unsupervised task: group observations by similarity.
- Distance choice and preprocessing are foundational modeling decisions.
- K-means is fast and interpretable but targets compact Euclidean clusters.
- Hierarchical clustering provides a multiscale view through dendrograms.

Next lecture: mixture models connect clustering to latent-variable probability models.

Suggested reading

- James, Witten, Hastie & Tibshirani (2nd edition), Chapter 12.4
- Johnson & Wichern (6th Edition), Chapter 12.1-12.4.