STA 35C: Statistical Data Science III

Lecture 24: Clustering & K-means Clustering

Dogyoon Song

Fall 2026, UC Davis

Announcements

Final exam: Wed, Dec 9, 8:00–10:00 AM

- **Location:** Olson Hall 205 (our usual classroom)
- **Cumulative coverage:** Lectures 1–26
- Arrive early; the exam ends at **10:00 AM sharp**
- Permitted materials
  - **Notes: Three (3)** handwritten, letter-sized (8.5×11 inches), double-sided sheets
  - **Calculator:** A simple non-graphing scientific calculator
  - No other reference materials or electronic devices
- **SDC accommodations:** Confirm exam scheduling with AES online

Upcoming schedule

- **No class:** Wed, Nov 25 and Fri, Nov 27
- **Remote lecture:** Mon, Nov 30, 9:00–9:50 AM PT on Zoom

Clarification on grading

Course grading scheme: see the syllabus

- **Homework (30%):** Average of the best 6 of 7 assignments
- **Midterms (30%):** Higher of the two midterm scores
- **Final exam (40%)**
- **Participation credit:** Up to 3 percentage points added to the course score

Letter grades

- Baseline cutoffs: 97% for A+, 93% for A, 90% for A-, etc.
- Cutoffs may be adjusted **at quarter's end** based on relative standing, **only in students' favor**
- Individual exam and homework scores will not be curved

C- safety net

- If your grade under the scheme above is **D+ or below**, we also calculate an alternative score: **30% homework + 70% final exam + participation credit**
- If this score **exceeds 65 out of 100**, your course grade will be raised to **C-**

* Please do not request individual rankings or preliminary grade cutoffs.
* No other grading adjustments or individual exceptions will be made.

Agenda

Quick recap: Principal component analysis (PCA)

- Objective: dimension reduction with minimal information loss
- Intuition: projection that retains maximum variance
- Proportion of variance explained & choosing number of PCs via scree plot

Today: Clustering

- Clustering problem
- Overview of two methods: K-means clustering & hierarchical clustering
- K-means clustering
  - Intuition & problem formulation
  - Algorithm
  - Illustration
  - Assessment

Recap: PCA

Problem setup:

- We have data of $X \in \mathbb{R}^p$, with potentially large dimension p
- **Goal:** reduce dimension from p to $r \ll p$ while retaining most of the "information"

PCA approach:

- Project data (X) onto an r-dimensional subspace (spanned by r vectors)
- These r principal components are chosen to capture maximum variance in X
  - **First PC:** a unit vector $\mathbf{u}_1 \in \mathbb{R}^p$ maximizing variance:

$$\mathbf{u}_1 = \underset{\|\mathbf{u}\|=1}{\operatorname{argmax}} \frac{1}{n} \sum_{i=1}^{n} \left(\mathbf{u} \cdot \mathbf{x}_i\right)^2$$

  - Subsequent PCs $\mathbf{u}_2, \ldots, \mathbf{u}_p$ are found similarly, each orthogonal to all previous PCs

Result:

- Often the first few PCs ($r \ll p$) capture most of the variation
- This allows dimension reduction by using only $(Z_{i1}, \ldots, Z_{ir})$ for observation i

Recap: PC scores, PVE, and choosing number of PCs

PC scores: PCA is a change of basis (=change of coordinate system)

- The k-th **PC score** of $\mathbf{x}_i$ is

$$Z_{ik} = \langle \mathbf{u}_k, \mathbf{x}_i \rangle = \sum_{j=1}^{p} u_{kj} x_{ij}.$$

- These Z_{ik} values become the coordinates of $\mathbf{x}_i$ in the new (PC) coordinate system

Proportion of variance explained (PVE):

- Total variance: $\mathrm{Var}(X) = \frac{1}{n}\sum_{i=1}^{n} \|\mathbf{x}_i\|^2 = \sum_{j=1}^{p} \mathrm{Var}(X_j) = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{p} x_{ij}^2$
- Variance explained by the k-th PC: $\mathrm{Var}\big(\langle \mathbf{u}_k, X \rangle\big) = \frac{1}{n}\sum_{i=1}^{n} Z_{ik}^2$
- $\mathrm{PVE}_k = \dfrac{\mathrm{Var}\big(\mathbf{u}_k \cdot X\big)}{\mathrm{Var}(X)}$ and $\mathrm{PVE}_{1:r} = \sum_{k=1}^{r} \mathrm{PVE}_k$

Choosing r: Use a scree plot or the cumulative PVE to decide how many PCs to keep

Clustering: Problem setup

Data: $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}, \qquad \mathbf{x}_i \in \mathbb{R}^p.$

Goal: Partition observations into clusters so that

- observations in the same cluster are similar,
- observations in different clusters are dissimilar.

Examples where clustering may be useful:

- **Cancer subtyping:** group patients or tumor samples using gene-expression profiles
- **Market segmentation:** group customers by purchasing patterns or behavior
- **Document analysis:** group articles by word usage
- **Image analysis:** group pixels or image patches by color/texture features

Important distinction from classification:

- Classification: labels/classes are observed in training data.
- Clustering: labels/classes are unknown; the algorithm discovers possible groups.

Caution: Clustering is exploratory. There may not be a single "true" clustering.

Illustration of clustering problem

Setup:

- Data: $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \subset \mathbb{R}^p$
- **Goal:** Partition the observations into clusters so that points within each cluster are "similar," while points in different clusters are "different"

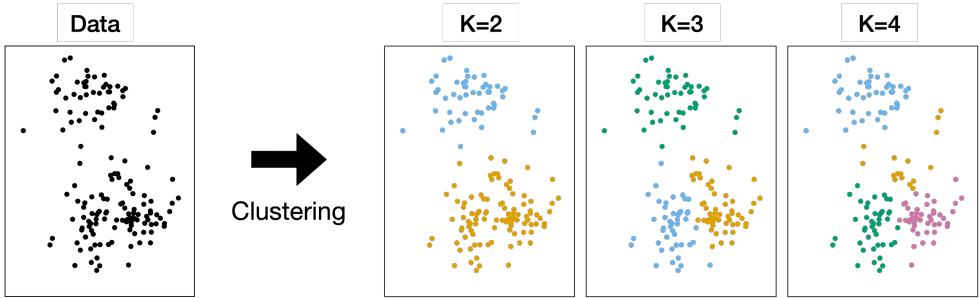


Figure: Illustration of clustering. Given a dataset of X (**Left**), we want to partition the observations into K distinct clusters (**Right**).

Distance and standardization matter

Clustering requires a notion of similarity. A common choice is Euclidean distance:

$$d(\mathbf{x}_i, \mathbf{x}_{i'}) = \|\mathbf{x}_i - \mathbf{x}_{i'}\|_2.$$

Scaling issue:

- Variables with larger numerical scales can dominate distances.
- Example: income in dollars may dominate age in years.

Common practice:

$$\widetilde{x}_{ij} = \frac{x_{ij} - \bar{x}_j}{s_j}.$$

- Standardize before clustering when variables have different units or scales.
- The choice of distance and scaling can substantially affect the clustering result.

Clustering: Overview of two algorithms

There are many clustering methods, but we focus on two well-known approaches:

- **K-means clustering** (Today)
  - Choose K in advance, then partition observations into K clusters
  - Simple & well-suited for relatively spherical clusters in a feature space
- **Hierarchical clustering** (next lecture)
  - We do not specify the number of clusters upfront
  - Observations are successively merged or split to form a hierarchical tree structure (*dendrogram*)
  - We can then *cut* the tree at various levels to obtain different numbers of clusters

K-means clustering: Basic idea

Goal: Partition the data $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \subset \mathbb{R}^p$ into K nonempty, non-overlapping clusters

- We specify the desired number of clusters K in advance
- Partition indices $\{1, \ldots, n\}$ into $C_1, \ldots, C_K$ with:
  - $C_1 \cup C_2 \cup \cdots \cup C_K = \{1, \ldots, n\}$: every $\mathbf{x}_i$ belongs to at least one of the K clusters
  - $C_k \cap C_{k'} = \emptyset$ for all $k \neq k'$; each $\mathbf{x}_i$ belongs to at most one cluster

Formulation of K-means clustering problem

$$\text{minimize}_{C_1, \cdots, C_K} \left\{ \sum_{k=1}^{K} W(C_k) \right\}$$

- The quantity $W(C_k)$ measures within-cluster variation
  - We want the clusters to be "tight," so $\sum_{k=1}^{K} W(C_k)$ to be as small as possible
- K-means clustering typically uses the squared (Euclidean) distance:

$$W(C_k) = \frac{1}{|C_k|} \sum_{i,i' \in C_k} \|\mathbf{x}_i - \mathbf{x}_{i'}\|^2 = \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} \left(x_{ij} - x_{i'j}\right)^2 = 2 \sum_{i \in C_k} \|\mathbf{x}_i - \bar{\mathbf{x}}_k\|^2.$$

Example: Within-cluster variation

Example

Let $\mathcal{X} = \{\mathbf{x}_1 = (-2, 1), \mathbf{x}_2 = (-1, 3), \mathbf{x}_3 = (2, 0), \mathbf{x}_4 = (3, -2)\} \subset \mathbb{R}^2$. Let $K = 2$ and

$$C_1 = \{1, 2\}, \ C_2 = \{3, 4\}.$$

Recall

$$W(C_k) = \frac{1}{|C_k|} \sum_{i, i' \in C_k} \|\mathbf{x}_i - \mathbf{x}_{i'}\|^2.$$

For each cluster, the within-cluster variation can be computed as

$$W(C_1) = \frac{1}{2}\left[\|(-2, 1) - (-1, 3)\|^2 + \|(-1, 3) - (-2, 1)\|^2\right] = \frac{1}{2} \times (5 + 5) = 5,$$
$$W(C_2) = \frac{1}{2}\left[\|(2, 0) - (3, -2)\|^2 + \|(3, -2) - (2, 0)\|^2\right] = \frac{1}{2} \times (5 + 5) = 5.$$

Therefore, the K-means clustering objective value is $W(C_1) + W(C_2) = 5 + 5 = 10$.

Pop-up quiz: K-means objective

Using the four points

$$\mathbf{x}_1 = (-2, 1), \quad \mathbf{x}_2 = (-1, 3), \quad \mathbf{x}_3 = (2, 0), \quad \mathbf{x}_4 = (3, -2),$$

compare two possible $K = 2$ clusterings:

$$\text{A: } C_1 = \{1, 2\},\ C_2 = \{3, 4\}, \qquad \text{B: } C_1 = \{1, 4\},\ C_2 = \{2, 3\}.$$

Question: Which clustering would the K-means objective prefer?

A) Clustering A, because points within each cluster are closer to their centroid.

B) Clustering B, because each cluster contains one left point and one right point.

C) Both are equally good because both have two clusters of size two.

D) Cannot be determined because no response variable Y is available.

Answer: A. The K-means objective prefers smaller within-cluster variation. Clustering A groups nearby points and has much smaller within-cluster variation than clustering B.

K-means clustering: Algorithm

K-means clustering is a hard combinatorial problem, so we use a heuristic algorithm:

K-means clustering algorithm

1. **Initialize:** Randomly assign the n observations to K nonempty clusters
2. **Iterate until assignments stop changing:**
   (a) **Update the centroids.** For each cluster C_k, compute the centroid

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

   (b) **Reassign.** For each observation i, reassign it to the cluster whose centroid is closest in squared Euclidean distance

- Within a run, the objective never increases, but the final solution may be suboptimal
- Often repeated from multiple random starts to choose the best result

Example: One iteration of K-means clustering

Example

Data: $\mathcal{X} = \big\{(-2,1),\ (-1,3),\ (2,0),\ (3,-2)\big\},\ K = 2$. The *initial* cluster assignment is

$$C_1 = \{1,4\}, \quad C_2 = \{2,3\}.$$

Step (a): Compute centroids.

$$\bar{\mathbf{x}}_1 = \tfrac{1}{2}[(-2,1) + (3,-2)] = (0.5,-0.5), \quad \bar{\mathbf{x}}_2 = \tfrac{1}{2}[(-1,3) + (2,0)] = (0.5,1.5).$$

Step (b): Reassign each point to the closer centroid.

- $\mathbf{x}_1$ is closer to $\bar{\mathbf{x}}_2$ because

$$\|\mathbf{x}_1 - \bar{\mathbf{x}}_1\|^2 = (-2.5)^2 + (1.5)^2 = 8.5 > \|\mathbf{x}_1 - \bar{\mathbf{x}}_2\|^2 = (-2.5)^2 + (-0.5)^2 = 6.5.$$

- Similarly, we observe $\mathbf{x}_2$ is closer to $\bar{\mathbf{x}}_2$, whereas $\mathbf{x}_3, \mathbf{x}_4$ are closer to $\bar{\mathbf{x}}_1$.

We get $C_1 = \{3,4\}, C_2 = \{1,2\}$.

Repeat (a) and **(b)** until the algorithm converges.

K-means clustering: Illustration of the algorithm iterations

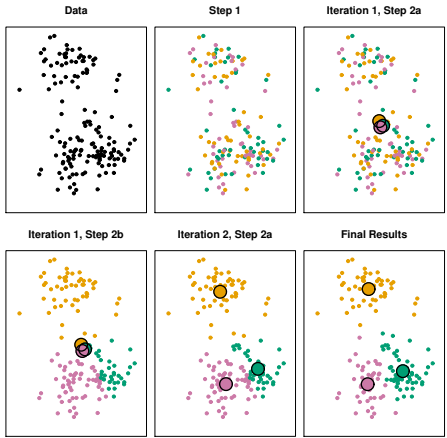


Figure: Selected stages of K-means with $K = 3$. Each iteration updates centroids (colored disks) and reassigns points [JWHT21, Figure 12.8].

Steps:

1. **Initialize** random cluster labels
2. **Iterate**:
   (a) Update cluster centroids
   (b) Reassign points to nearest centroid

Remarks:

- Each iteration does not increase the objective
- Final solution depends on initialization

K-means clustering: Local optima and multiple runs

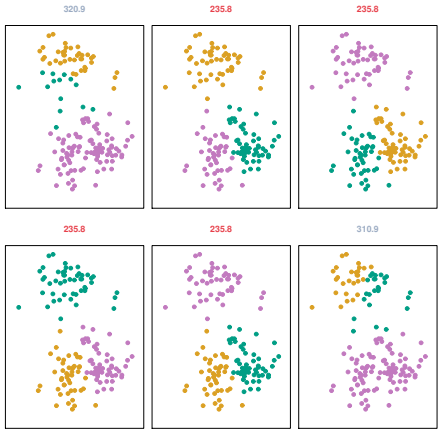


Figure: K-means with $K = 3$ repeated six times on the same data, each with a different random initial assignment. Above each plot is the final objective. Multiple local optima are found; the best has objective=235.8 [JWHT21, Figure 12.9].

Key points:

- K-means can converge to a suboptimal (local) solution
- Different initial cluster assignments can yield different final partitions
- Usually, re-run with multiple random starts and pick the best (lowest objective)

Choosing the number of clusters K

Challenge: K-means requires K to be specified in advance.

Important fact:

> The best achievable within-cluster variation cannot increase as K increases.

- If $K = n$, each point is its own cluster and within-cluster variation is 0.
- Minimizing this objective alone does not give a meaningful choice of K.

Common approaches:

- **Elbow plot:** look for a point where increasing K gives diminishing returns.
- **Domain knowledge:** choose K based on interpretability or scientific context.
- **Stability:** check whether clusters persist under resampling or perturbations.

K-means clustering: Strengths and limitations

Strengths:

- Simple and computationally fast, especially for large data
- Often yields sensible clusterings if K is well-chosen
- Easy to interpret: each cluster has a centroid

Limitations:

- Must pre-specify the number of clusters K
- Can converge to a *local* rather than global optimum
- Works best for roughly spherical clusters around centroids
- Sensitive to feature scaling, outliers, initialization, and non-spherical cluster shapes

Pop-up quiz: Choosing K

Suppose we compare the smallest achievable K-means objective on the same dataset for $K = 1, 2, \ldots, n$.

Question: Which statement is most accurate?

A) The within-cluster variation must increase as K increases.

B) The smallest achievable within-cluster variation cannot increase as K increases, but this does not mean the largest K is the best choice.

C) The best K is always $K = n$, because the objective becomes zero.

D) K-means chooses K automatically, so no decision is needed.

Answer: B. Splitting a cluster cannot increase the best achievable objective. However, $K = n$ gives one cluster per observation, so choosing a useful K also requires interpretation, elbow plots, stability, or domain knowledge.

Wrap-up: Takeaways

Clustering problem:

- We have feature vectors $\mathbf{x}_i \in \mathbb{R}^p$ (no response Y)
- **Goal:** partition observations into "clusters" so that points in the same cluster are similar, and points in different clusters are dissimilar

K-means clustering:

- Fix the number of clusters K in advance
- Define non-overlapping clusters $C_1, \ldots, C_K$ to *minimize* the total within-cluster variation
- **Algorithm:**
  i) *Initialize* random cluster assignments
  ii) Iteratively *(a) update centroids*, and *(b) reassign points* until convergence
- **Limitations:** can get stuck in local optima; requires K pre-specified

Suggested reading: [JWHT21, Ch. 12.4]

Next lecture: Hierarchical clustering: dendrograms and linkage criteria

References


Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani.
An Introduction to Statistical Learning: with Applications in R, volume 112 of *Springer Texts in Statistics*.
Springer, New York, NY, 2nd edition, 2021.