

# **STA 35C: Statistical Data Science III**

## **Lecture 24: PCA (cont'd) + Clustering (k-means Clustering)**

Dogyoon Song

Spring 2025, UC Davis

# Announcement

---

**Final exam** on Fri, June 6 (1:00 pm–3:00 pm) in classroom

- **Be on time:** The exam starts at 1:00 pm and ends at 3:00 pm sharp
- **Three hand-written cheat sheets allowed:** Letter-size (8.5"×11"), double-sided, brief formulas/notes
- **Calculator:** A simple (non-graphing) scientific calculator is allowed
- **No other materials** beyond the single cheat sheet (no textbooks, etc.)
- **SDC accommodations:** Confirm scheduling to take on Thu, June 5 with AES ASAP

## Preparation:

- The exam is *cumulative* (Lectures 1–25)
- A practice final exam and brief answer key will be provided on the course webpage
- Office hours this week:
  - Instructor: Wed, June 4 (4:00–6:00pm, extended); no OH on Thu, June 5
  - TA: Mon, June 2 & Thu, June 5, 1–2pm

**Course evaluation:** Please share your feedback comments by Thu, June 5

# Today's topics

---

## Principal component analysis (PCA)

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

## Clustering

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

# Quick review: PCA overview

---

## 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:

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

- Subsequent PCs  $\mathbf{u}_2, \dots, \mathbf{u}_p$  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  $(Z_{i1}, \dots, Z_{ir})$  for observation  $i$

## Quick review: 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  $X_i$  is

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

- These  $Z_{ik}$  values become the coordinates of  $X_i$  in the new (PC) coordinate system

**Proportion of variance explained (PVE):**

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

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

## Example: $p = 3$ data reduced to $r = 1$

---

### Example

**Dataset:** Let  $X \in \mathbb{R}^3$ . Suppose we have five centered points:

$$\mathcal{X} = \{(0, 0, 0), (0, -1, 0), (0, 1, 0), (0, 0, -3), (0, 0, 3)\}.$$

One can verify  $\sum_i X_i = (0, 0, 0)$ , so these are already mean-centered.

**Step 1: Compute total variance.**

$$\begin{aligned}\text{Var}(X) &= \frac{1}{5} \sum_{i=1}^5 \|X_i\|^2 = \sum_{i=1}^5 (X_{i1}^2 + X_{i2}^2 + X_{i3}^2) \\ &= \frac{1}{5} (0^2 + (-1)^2 + 1^2 + (-3)^2 + 3^2) = \frac{1+1+9+9}{5} = \frac{20}{5} = 4.\end{aligned}$$

**Step 2: Identify the first principal component.**

A simple inspection shows the direction of greatest variance is along the z-axis:

$$\mathbf{u}_1 = (0, 0, 1).$$

Indeed, points  $(0, 0, \pm 3)$  have the largest spread among the three coordinates.

## Example: $p = 3$ data reduced to $r = 1$

---

### Example

**Step 3: Variance along  $\mathbf{u}_1$  and PVE.** Since  $\mathbf{u}_1 = (0, 0, 1)$ , the first PC score of  $X_i$  is equal to  $X_{i3}$ .

$$\text{Var}(\mathbf{u}_1 \cdot X) = \frac{1}{5} \sum_{i=1}^5 (\langle \mathbf{u}_1, X_i \rangle)^2 = \frac{1}{5} \sum_{i=1}^5 (x_{i3})^2 = \frac{1}{5} (0^2 + 0^2 + 0^2 + (-3)^2 + 3^2) = \frac{18}{5} = 3.6.$$

Hence the proportion of variance explained by the first PC is

$$\text{PVE}_1 = \frac{3.6}{4.0} = 0.9 \quad (\text{i.e., } 90\% \text{ of total variance}).$$

### Additional remarks.

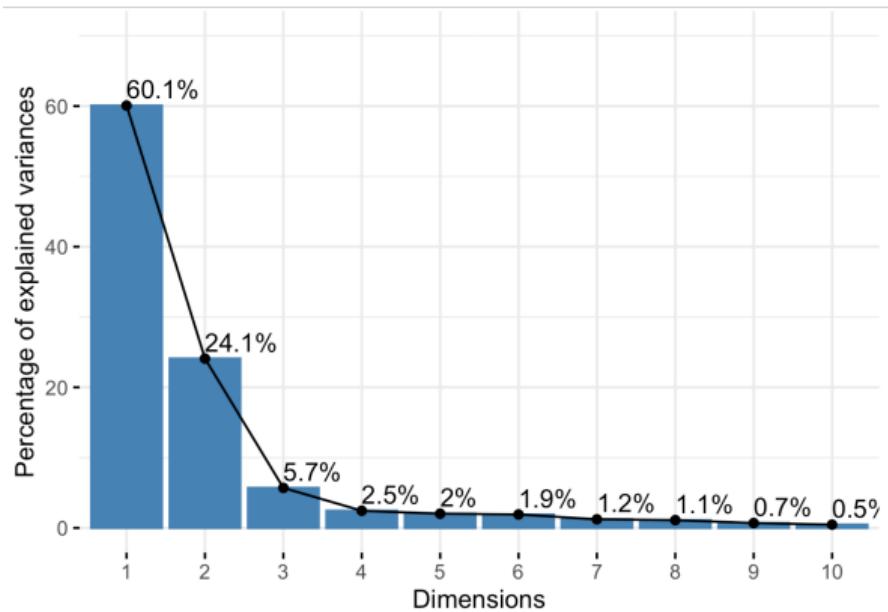
- Similarly, we can verify that the second PC direction is  $\mathbf{u}_2 = (0, 1, 0)$ .
- Hence,

$$\text{PVE}_2 = \frac{0.4}{4.0} = 0.1 \implies \text{PVE}_{1:2} = \text{PVE}_1 + \text{PVE}_2 = 1.$$

That is, all information about the dataset  $\mathcal{X}$  is explained by the first two PC scores.

# Choosing the number of PCs using scree plot example

---



## Trade-off:

- Smaller dimension  $r$  is easier to interpret and visualize
- Larger  $r$  retains more variance in the data

Figure: A scree plot from `mtcars` dataset in R. The elbow appears to occur at the third principal component, which suggests keeping the first three components (source: [Statistics Globe](#)).

## (Optional) Additional remarks on PCA

---

### Scaling variables?

- If predictors have very different scales (e.g. height in cm vs. income in \$), standardizing them to unit variance can drastically alter PCA directions
- Whether to scale depends on context: if raw scales matter, do not standardize; if you want each feature to contribute equally, do scale

### Uniqueness:

- Principal component directions are unique up to a sign ( $\mathbf{u}$  vs.  $-\mathbf{u}$ )
- This sign usually does not affect interpretation, so software packages pick a sign convention automatically

### Computation:

- Solve for eigenvectors/eigenvalues of the sample covariance (or correlation) matrix
- In R: `prcomp(..., scale=TRUE)` or `princomp(...)`

# PCA application in high-dimensional genomics

## Example: Genomics data [NJB<sup>+</sup>08]

- 1,387 individuals from Europe, each with genotype data at 197,146 loci
  - Apply PCA → reduce dimension from  $p = 197k$  to 2 principal components
  - Two PCs remarkably recapitulate Europe's geography in "genetic space," demonstrating how PCA can drastically compress data while still capturing meaningful structure

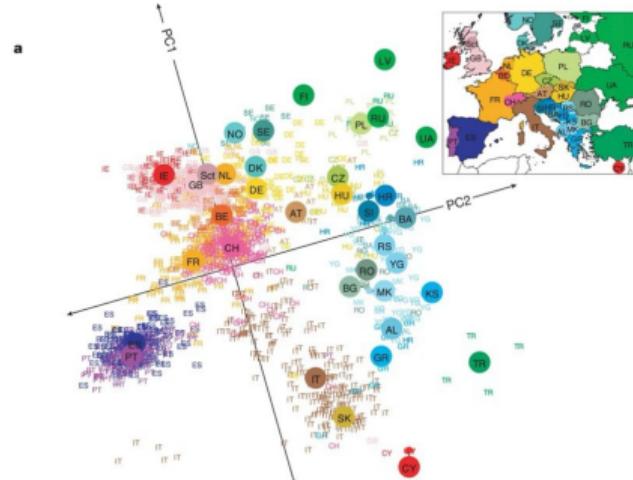


Figure: First two principal components of genetic variation among 1,387 Europeans. Small colored points are individuals; large dots mark country medians in PC1–PC2 space [NJB<sup>+</sup>08, 0Figure 1-a].

# PCA application in image compression

---

**Example:** Compressing a grayscale image via PCA

- Original image has  $372 \times 492$  pixels, each a grayscale intensity in  $[0, 255]$
- The image is partitioned into  $12 \times 12$  blocks, so each block is a  $12 \times 12 = 144$ -dimensional “vector”
- There are  $N = \frac{372}{12} \times \frac{492}{12} = 1271$  such vectors (observations)
- Apply PCA with rank  $r \in \{1, 3, 6, 16, 60\}$  for the dimension reduction

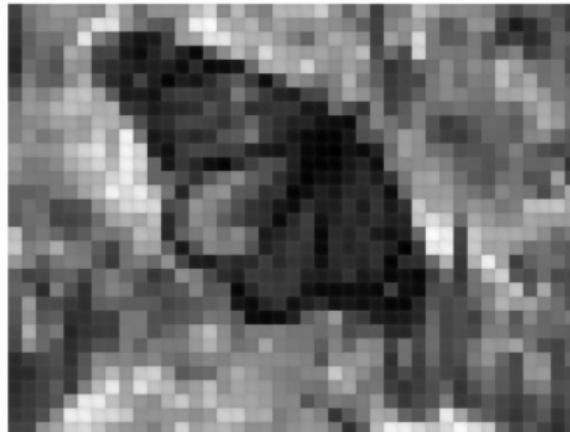
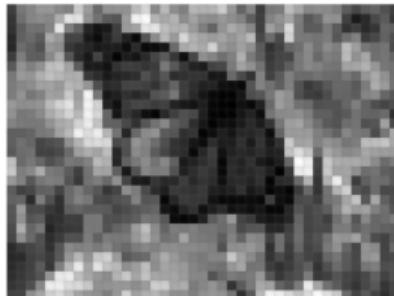


Figure: Compressing an image by PCA. **Left:** original image. **Right:** PCA rank-1 approximation. With  $r = 1$ , almost all details are lost, but the main global contrast is still visible.

## PCA application in image compression (cont'd)



Original



Rank-1 PCA ( $r = 1$ )



Rank-3 PCA ( $r = 3$ )



Rank-6 PCA ( $r = 6$ )



Rank-16 PCA ( $r = 16$ )



Rank-60 PCA ( $r = 60$ )

Figure: PCA-based image compression. Larger  $r$  yields better reconstruction quality.

# Summary of PCA

---

## Principal Component Analysis (PCA):

- Finds a few PC directions that capture maximum variance in the data
- The first few PCs often capture most of the total variation, enabling dimension reduction
- PCA is *unsupervised*, commonly used for exploratory analysis or as a pre-processing step

## Proportion of Variance Explained (PVE):

- Quantifies how much of the total variance is retained by a chosen number  $r$  of PCs
- A scree plot of PVE vs. PC index can guide how many PCs to keep

## Additional remarks:

- In R, use `prcomp(...)` or `princomp(...)`
- Predictor scaling can affect PCA
- Once you learn linear algebra & eigendecomposition, the definitions and details of PCA will become much clearer

# Clustering

---

## Problem setup:

- We have a dataset  $\mathcal{X} = \{X_1, \dots, X_n\}$  of  $p$ -dimensional features  $X_i \in \mathbb{R}^p$
- **Goal:** Partition the observations into distinct *clusters* such that points within each cluster are “similar,” and points in different clusters are “different”
  - Need a notion of (dis-)similarity to measure “similar” vs. “different”
  - This is similar to classification, but the classes are *not* known beforehand

## Examples:

- *Cancer subtyping*: cluster tissue cells with similar gene-expression profiles
- *Market segmentation*: group customers by their profiles and purchasing patterns

## Outcome:

- A few subgroups (clusters) of observations based on feature similarity
- (Conversely, we can also cluster features based on measurement similarity)

# Clustering: Overview of two algorithms

---

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

- **K-means clustering** (Today)
  - We specify a number of clusters  $K$  in advance
  - The algorithm assigns each observation to one of these  $K$  non-overlapping clusters, aiming to minimize within-cluster variation
  - 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  $\{X_1, \dots, X_n\} \subset \mathbb{R}^p$  into  $K$  non-overlapping clusters

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

**Formulation:**

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

- 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

$$W(C_k) = \frac{1}{|C_k|} \sum_{i, i' \in C_k} \|X_i - X_{i'}\|^2 = \frac{1}{|C_k|} \sum_{i, i' \in C_k} \sum_{j=1}^p (X_{ij} - X_{i'j})^2$$

## Example: Within-cluster variation

### Example

Let  $\mathcal{X} = \{(-2, 1), (-1, 3), (2, 0), (3, -2)\} \subset \mathbb{R}^2$ . Let  $K = 2$  and

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

Recall

$$W(C_k) = \frac{1}{|C_k|} \sum_{i, i' \in C_k} \|X_i - X_{i'}\|^2.$$

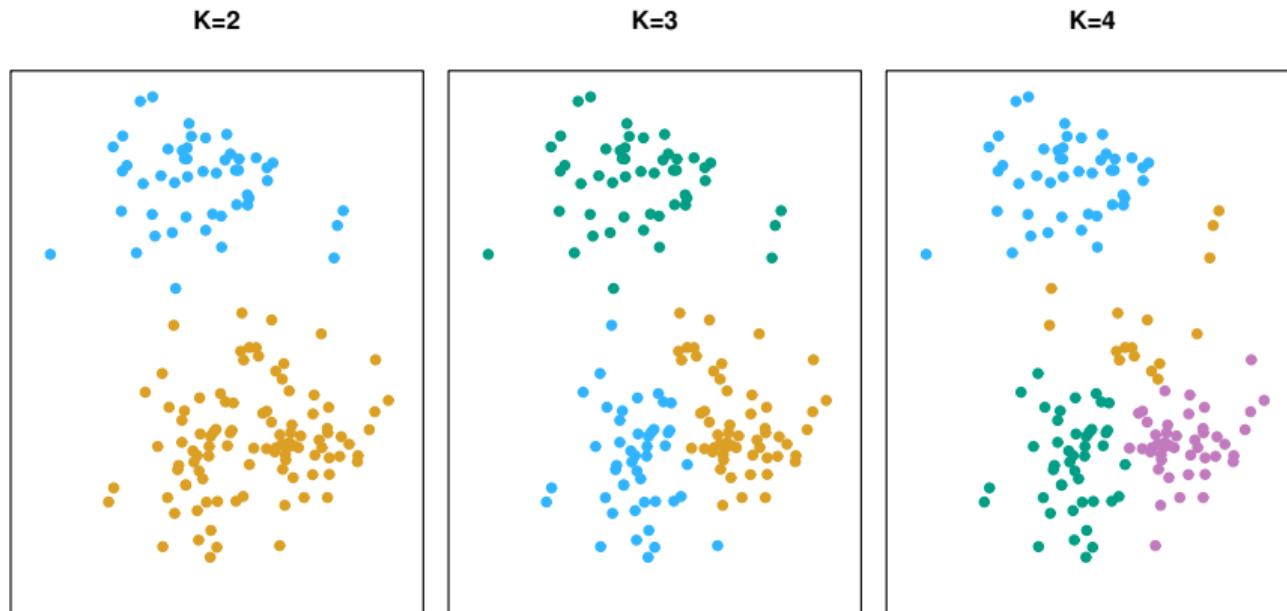
For each cluster, the within-cluster variation can be computed as

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

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

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

# K-means clustering: Visual illustration



**Figure:** A simulated data set with 150 observations in two-dimensional space. Panels show the results of applying K-means clustering with different values of  $K \in \{2, 3, 4\}$ . The color of each observation indicates the cluster to which it was assigned using the K-means clustering algorithm; note that there is no ordering of the clusters, so the cluster coloring is arbitrary. These cluster labels were not used in clustering; instead, they are the outputs of the clustering procedure [JWHT21, Figure 12.7].

# 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 each of the  $n$  observations to one of  $K$  clusters
- 2 **Iterate until assignments stop changing:**
  - (a) **Update the centroids.** For each cluster  $C_k$ , compute the centroid

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

- - (b) **Reassign.** For each observation  $i$ , reassign it to the cluster whose centroid is closest in squared Euclidean distance
- Each iteration reduces the objective but may converge to a local optimum
  - Often repeated from multiple random starts to choose the best result

# Example: One iteration of K-means clustering

## Example

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

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

### Step (a): Compute centroids.

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

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

- $X_1$  is closer to  $\bar{x}_2$  because

$$\|X_1 - \bar{x}_1\|^2 = (-2.5)^2 + (1.5)^2 = 8.5 > \|X_1 - \bar{x}_2\|^2 = (-2.5)^2 + (-0.5)^2 = 6.5.$$

- Similarly, we observe  $X_2$  is closer to  $\bar{x}_2$ , whereas  $X_3, X_4$  are closer to  $\bar{x}_1$ .

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

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

# K-means clustering: Visual illustration of the algorithm

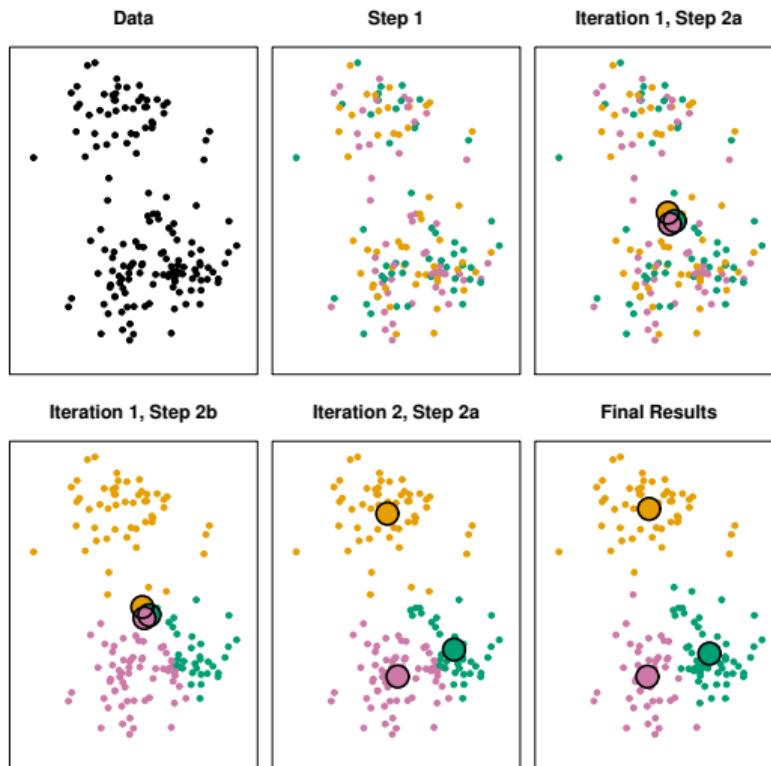


Figure: An example of the K-means algorithm for  $K = 3$  over 2 iterations [JWHT21, Figure 12.8].

# K-means clustering: Visual illustration of 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].

# 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
- Assumes clusters are roughly spherical around centroids
- Sensitive to outliers and rescaling of features

# Wrap-up: Takeaways

---

## Clustering problem:

- We have feature vectors  $X_i \in \mathbb{R}^p$  (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  $C_1, \dots, C_K$  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

## Next time:

- Hierarchical clustering (no need to specify  $K$ )
- Dendograms and various 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.
-  John Novembre, Toby Johnson, Katarzyna Bryc, Zoltán Kutalik, Adam R Boyko, Adam Auton, Amit Indap, Karen S King, Sven Bergmann, Matthew R Nelson, et al.  
Genes mirror geography within europe.  
*Nature*, 456(7218):98–101, 2008.