Section 11: K-means clustering

STA 35C – Statistical Data Science III

Instructor: Akira Horiguchi

Fall Quarter 2025 (Sep 24 – Dec 12)
MWF, 12:10 PM – 1:00 PM, Olson 158
University of California, Davis

Overview

Based on Chapter 12 of ISL book James et al. (2021).

- For more R code examples, see R Markdown files in `https://www.statlearning.com/resources-second-edition`

1 *K*-means clustering

Unsupervised learning

Supervised data: predictors $X_1, \dots, X_p$ and a response Y measured on n observations.

Unsupervised data: predictors $X_1, \dots, X_p$ measured on n observations, but no response.

- Still useful to analyze the association between the predictors $X_1, \dots, X_p$.
- Often performed as part of an exploratory data analysis.
- Harder to assess the results from an unsupervised learning method;
  there is no “truth” to compare to.
  (In contrast, in supervised learning the “truth” is the response Y.)

Clustering

Common unsupervised learning task: find homogeneous subgroups (i.e., *clusters*) among observations.

- "Market segmentation" aims to identify subgroups of people who might be more receptive to certain kind of advertisements/products etc.
- Flow cytometry: group cells based on their biomarker values.

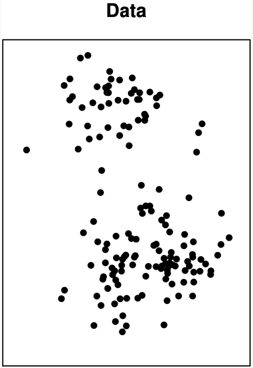

Figure 1: From James et al. (2021).

Cluster definition

If we index the n observations by the integers $1, 2, 3, \dots, n$, then

$$\text{cluster } n \text{ observations} \iff \text{cluster the integers } 1, 2, 3, \dots, n$$

In other words, we want to partition the set $\{1, 2, 3, \dots, n\}$.

Definition (Cluster)

Clusters are sets $C_1, \dots, C_K$ with the following features:

- $C_1 \cup C_2 \cup \cdots \cup C_K = \{1, \dots, n\}$ (each observation belongs to at least one cluster);
- $C_k \cap C_l = \emptyset$ for all $k \neq l$ (no observation belongs to more than one cluster).

There are almost K^n ways to partition n observations into K clusters.

- How to select “best” partition of given data?
- A common algorithmic technique: *K-means clustering*

K-means clustering

What is a "good" partition?

A partition is "good" if the observations in each cluster are close to each other, i.e., if each cluster has a relatively small *within-cluster variation*.

Definition (Within-cluster variation)

Given a cluster C consisting of $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^p$, the cluster's within-cluster variation is

$$W(C) := \frac{1}{\#C} \sum_{i,i' \in C} \|\mathbf{x}_i - \mathbf{x}_{i'}\|_2^2 = \frac{1}{\#C} \sum_{i,i' \in C} \sum_{j=1}^{p} \left(x_{ij} - x_{i'j}\right)^2 , \tag{1}$$

where $\#C$ denotes the number of observations in cluster C. If $p = 1$, then (1) becomes

$$W(C) := \frac{1}{\#C} \sum_{i,i' \in C} (x_i - x_{i'})^2 .$$

Example (Compute the within-cluster variation of cluster $C = \{1, 3, 5, 7\}$)

How does it compare to the within-cluster variation of cluster $\{1, 4, 7, 10\}$?

Idea of K-means clustering

The user chooses K (the number of clusters) before performing *K-means clustering*. For any clusters $C_1, \ldots, C_K$, we can compute

$$\sum_{k=1}^{K} W(C_k). \tag{2}$$

Goal: We want to find clusters $C_1, \ldots, C_K$ that produce the smallest value of (2).

- There are almost K^n ways to partition n observations into K clusters.
- Hence it is very difficult to find the *global* minimizer.
- The following *K-means clustering* algorithm provides a *local* minimizer.

The algorithm defines the *centroid* of a cluster as the *mean* of all points in the cluster. (Hence the name of the approach.)

Example (Draw & compute the centroid of the points $(1,2), (2,1), (3,2), (1,0)$)

Algorithm for *K*-*means* clustering

1. Randomly assign a number from 1 to K (K is pre-defined) to each observation.
2. Iterate steps (a) and (b) until the cluster assignments stop changing:
   (a) Compute each cluster's *centroid* (see future slide).
   (b) Assign each observation to the cluster whose centroid is the closest.

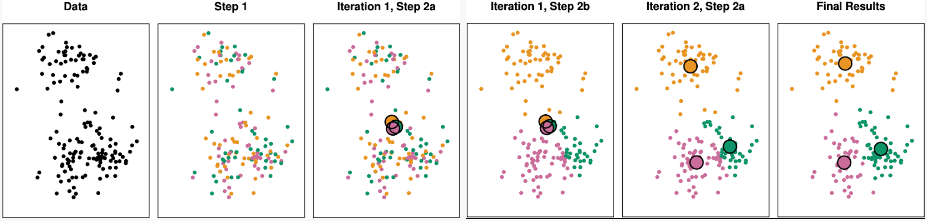

Figure 2: From James et al. (2021). 3-means clustering and 10 iterations.

Comments:

- Each iteration of Step 2 will produce a partition; its value of (2) will be smaller than the previous iteration's if this partition is different from the previous iteration's.
- The final cluster assignments will produce a *local minimum* of (2). Which local minimum is obtained depends on the random cluster assignment from Step 1.
- To reduce probability of choosing a "bad" local minimum, we should run the algorithm many times, and then choose the partition with smallest value of (2).

Local minima

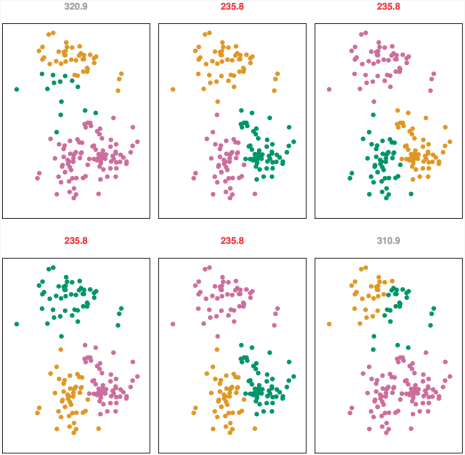

Figure 3: From James et al. (2021). 3-means clustering performed six times on the same data, each time with a different random assignment of the observations in Step 1 of the *K*-means algorithm. Above each plot is the value of the objective (2). Three different local minima were obtained, one of which resulted in a smaller value of the objective and provides better separation between the clusters. Those labeled in red all achieved the same best solution, with an objective value of 235.8.

Issues in clustering

Issues in clustering

- Should the features first be standardized in some way? E.g. maybe scale them to have standard deviation one?
- *K*-means clustering: how many clusters should we look for?

It is challenging to validate obtained clusters.

- Do obtained clusters represent true subgroups in the data, or are they a result of clustering noise?
- Outside scope of class; more details found in “sequel” book *The Elements of Statistical Learning*
- In practice, try several different choices, and look for the one with the most useful or interpretable solution.
- Also, see `kmeans-clustering.qmd`