3 Clustering

Clustering refers to a broad set of techniques for finding *subgroups* in a data set.

We seek to partition observations into distinct groups so that

- observations within a group are similar
- observations in different groups are dissimilar

need to define depends on the domain!

For instance, suppose we have a set of n observations, each with p features. The n observations could correspond to tissue samples for patients with breast cancer and the p features could correspond to measurements collected for each tissue sample.

- clinical measurements, e.g. tumor stage or grade.
- gene expression measurements.

We may have reason to believe there is heterogeneity (diverse in character) among the n observations.

e.g. different unknown subtype of cancer.

This is *unsupervised* because

We are trying to discover structure (distinct clusters).

This is different from a supervised problem (no goal of prediction).

Both clustering and PCA seek to simplify the data via a small number of summaries.

- PCA – find low dimensional representation of observations that explain a good fraction of variability
- Clustering = find homogeneous subgroups among observations.

Since clustering is popular in many fields, there are many ways to cluster.

we focus on the 2 best-known clustering approaches.

- K-means clustering

  we seek to partition the observations into a pre-specified # of clusters.

- Hierarchical clustering

  We do not know in advance how many clusters we want.

  We obtain clusterings for $1, \ldots, n$ clusters and view these in a dendrogram.

In general, we can (1) cluster observations on the basis of features or (2) we can cluster features on the basis of observations.

(1) identify subgroups among observation.

(2) discover subgroups among features.

We will focus on (1), but can perform (2) just by transposing our data matrix.

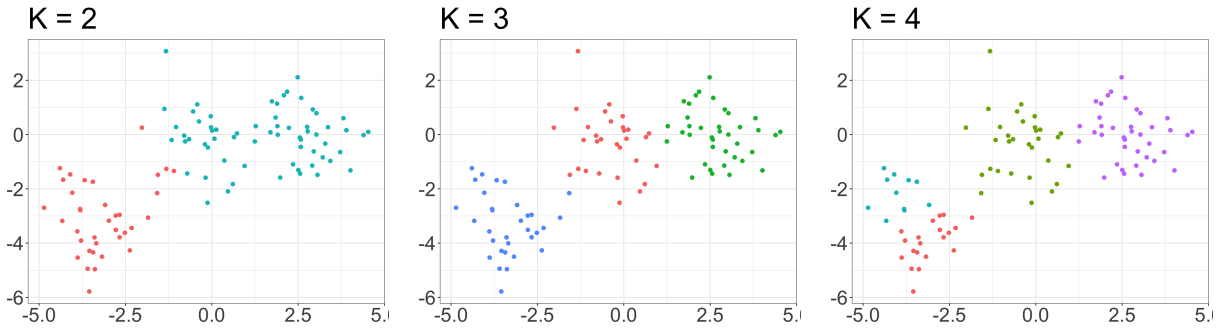
3.1 K-Means Clustering

Simple and elegant approach to partition a data set into K distinct, non-overlapping clusters.

We must first specify how many clusters K.

then K-means assigns each observation to one of K clusters.

e.g. clustering $n = 100$ observations into K clusters using $p = 2$ features.



The K-means clustering procedure results from a simple and intuitive mathematical problem. Let $C_1, \ldots, C_K$ denote sets containing the indices of observations in each cluster. These satisfy two properties:

e.g. if obs i is in cluster k, $i \in C_k$

clusters define a partition.

1. $C_1 \cup C_2 \cup \cdots \cup C_K = \{1, \ldots, n\}$
   each observation belongs to one of the K clusters.

2. $C_k \cap C_{k'} = \emptyset \quad \forall\ k \neq k'$
   the clusters are nonoverlapping

Idea: good clustering is one for which the within-cluster-variation is as small as possible.

The *within-cluster variation* for cluster C_k is a measure of the amount by which the observations within a cluster differ from each other.

Call this $W(C_k)$.

Then want to solve the problem

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

← Want to partition observations into K clusters such that total within-cluster variation is minimized.

To solve this, we need to define within-cluster variation.

Many ways we could do this.

Most common way: squared euclidean distance.

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

$|C_k|$ ↑ # obs in k^{th} cluster

This results in the following optimization problem that defines K-means clustering:

$$\underset{C_1,\dots,C_K}{\text{minimize}} \left\{ \sum_{k=1}^{K} \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 \right\}$$

"objective"

This is very difficult to solve exactly: $\approx K^n$ ways to partition n observations into k clusters!

A very simple algorithm has been shown to find a local optimum to this problem: ("pretty good solution")

1. randomly assign a number from 1 to K to each of the observations. these will be the initial cluster assignments for each observation.

2. iterate until cluster assignments stop changing:
   (a) for each of the K cluster compute the cluster centroid — vector of p feature means for observations in each cluster.
   (b) assign each observation to closest centroid cluster.
   ↳ euclidean distance.

Algorithm is guaranteed to decrease value of objective at each step.

When cluster assignments stop changing this is a local minimum.

↳ not necessarily global ⇒ clustering depends on (random) initial cluster values (step 1).

⇒ run the algorithm multiple times from different initial configurations and choose clustering w/ smallest objective function.

Problem: We must choose K! more later...

3.2 Hierarchical Clustering

One potential disadvantage of K-means clustering is that it requires us to specify the number of clusters K. *Hierarchical clustering* is an alternative that does not require we commit to a particular K. ahead of time.

also hierarchical clustering results in a tree-based representation of observations.

clusters getting larger

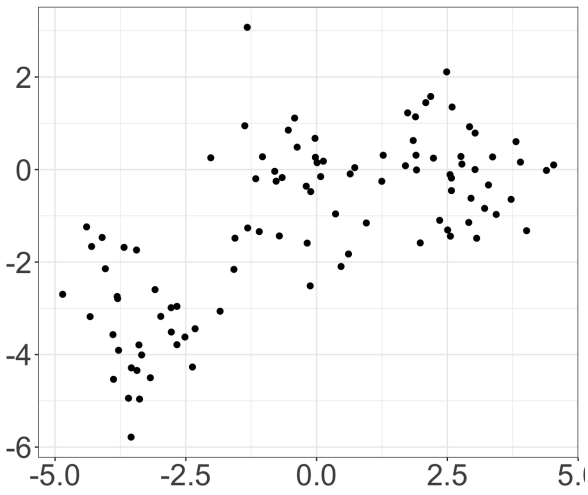
We will discuss *bottom-up* or "*agglomerative*" clustering.

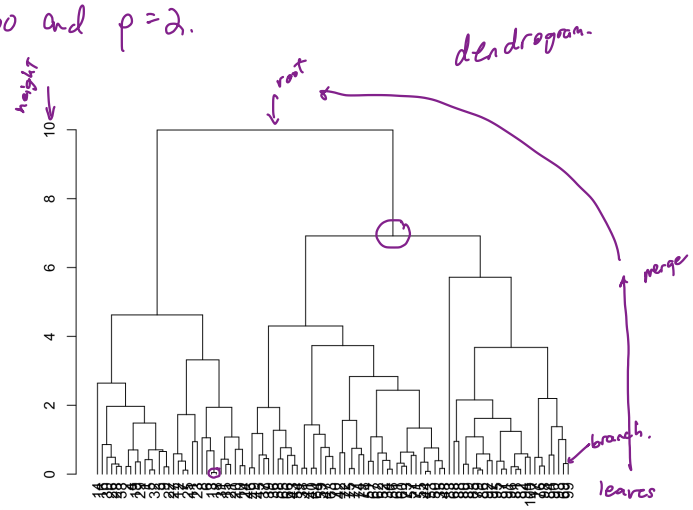
start with every observation in its own cluster and merge clusters until all observations are in a single cluster (n clusters → 1 cluster).

"bottom-up" refers to the tree representation w/ leaves on bottom.

3.2.1 Dendrograms

Same simulated as before w/ n=100 and p=2.





Each *leaf* of the dendrogram represents one of the 100 simulated data points.

As we move up the tree, leaves begin to fuse into branches, which correspond to observations that are similar to each other.

- as we move higher up the tree, branches fuse w/ other branches
- the lower the fusion occurs, the more similar the observations are
- observations that fuse high up in the tree can be quite different.

more precisely:

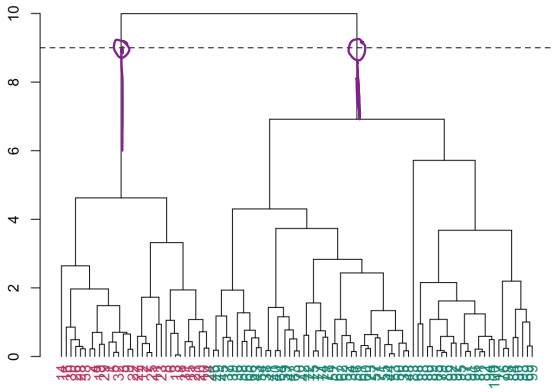
For any two observations, we can look for the point in the tree where branches containing those two observations are first fused.

The height of the first fusion indicates how different those points are.

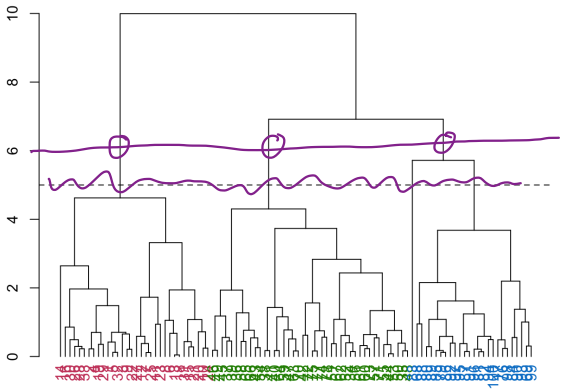
We draw conclusions about the similarity of two observations based on the location on the vertical axis where branches containing those observations are first fused.

How do we get clusters from the dendrogram?

We can make a horizontal "cut" across the dendrogram.



This cut at height = 9 results in 2 clusters.



This cut at height 6 results in 3 clusters

We can cut at a height corresponding to 1, ..., n clusters (i.e. height of cut is similar to K in K-means).

⇒ a single dendrogram can be used to obtain any number of clusters!

In practice: people look at dendrogram and choose where to cut based on height of fusion and # cuts resulting (subjective).

The term *hierarchical* refers to the fact that clusters obtained by cutting the dendrogram at a given height are necessarily nested within the clusters obtained by cutting the dendrogram at a greater height.

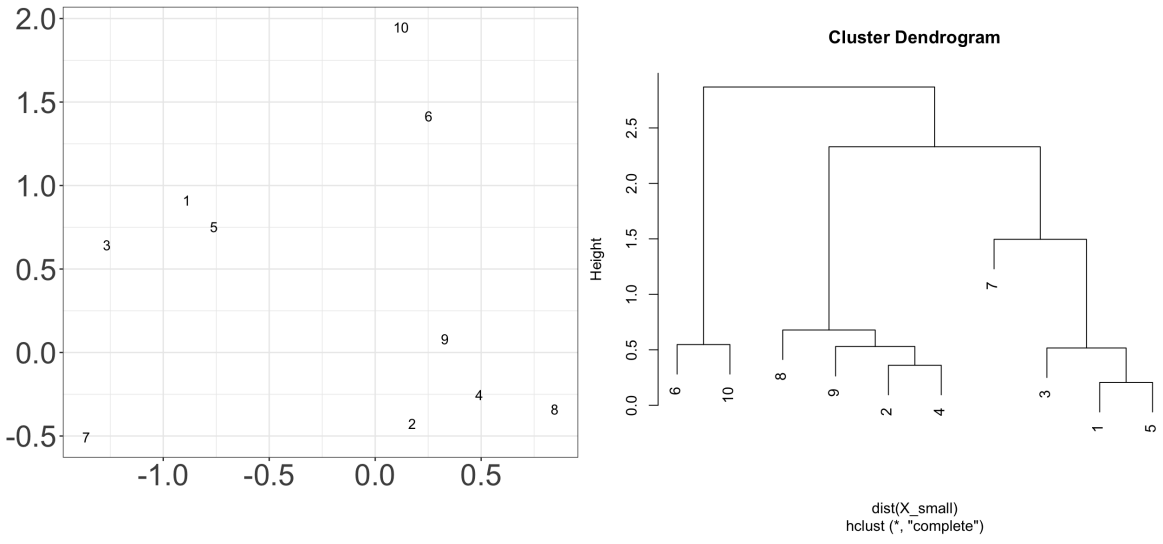
This hierarchical assumption may or may not be realistic.

e.g. Suppose we have a group of observations 50-50 split M/F and evenly split English, Japanese, French speakers.
Maybe result in 2 clusters by sex
Maybe result in 3 clusters by language ⇒ not nested.

3.2.2 Algorithm

First, we need to define some sort of *dissimilarity* metric between pairs of observations.

Then the algorithm proceeds iteratively.



More formally,

One issue has not yet been addressed.

How do we determine the dissimilarity between two clusters if one or bother of them contains multiple observations?

1.

2.

3.

4.

3.2.3 Choice of Dissimilarity Metric

3.3 Practical Considerations in Clustering

In order to perform clustering, some decisions should be made.

-
-
-

Each of these decisions can have a strong impact on the results obtained. What to do?