Lecture 20: K-means clustering & hierarchical clustering

BIOS635

04/02/2020

Some slides taken/modified from other people's lectures online.

Clustering

- *Clustering* refers to a very broad set of techniques for finding *subgroups*, or *clusters*, in a data set.
- We seek a partition of the data into distinct groups so that the observations within each group are quite similar to each other,
- It make this concrete, we must define what it means for two or more observations to be *similar* or *different*.
- Indeed, this is often a domain-specific consideration that must be made based on knowledge of the data being studied.

Distance metrics

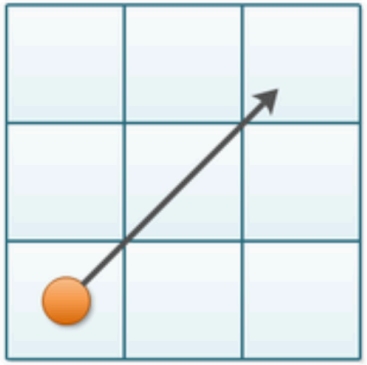
$$\text{Dist}(a,b) = \left(\sum_i |a_i - b_i|^p\right)^{1/p}$$

$p = 1$, *Manhattan Distance*

$p = 2$, *Euclidean Distance*

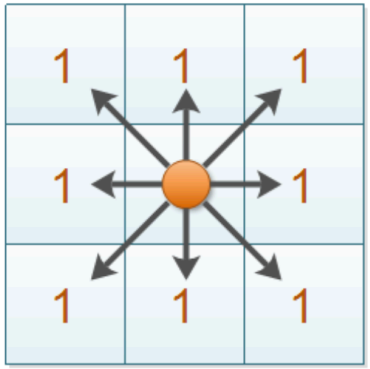
$p = \infty$, *Chebychev Distance*

Euclidean Distance



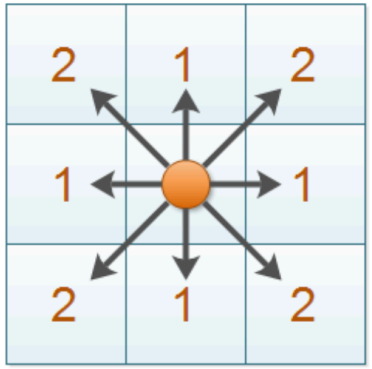
$$\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

Chebyshev Distance



$$\max(|x_1 - x_2|, |y_1 - y_2|)$$

Manhattan Distance



$$|x_1 - x_2| + |y_1 - y_2|$$

PCA v.s. clustering

- PCA looks for a low-dimensional representation of the observations that explains a good fraction of the variance.
- Clustering looks for homogeneous subgroups among the observations.

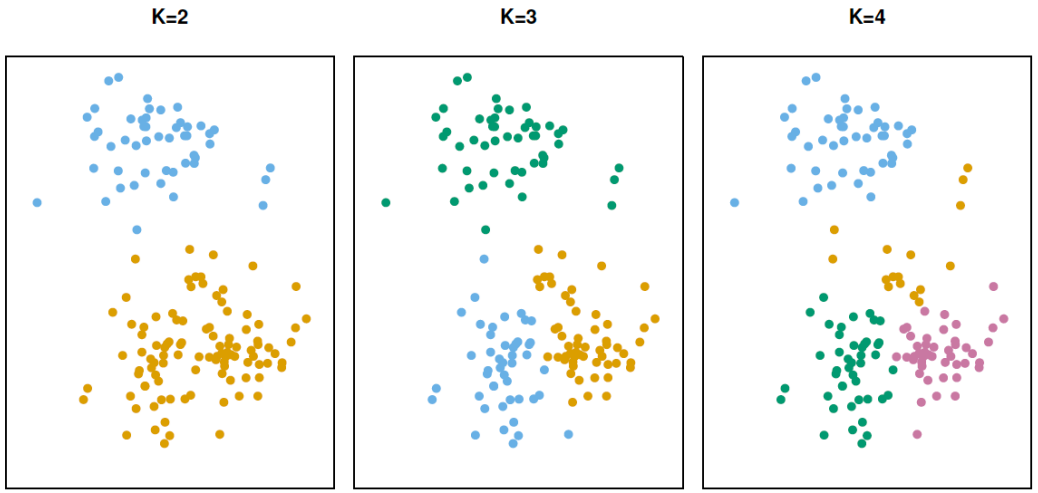
Clustering for marketing segmentation

- Suppose we have access to a large number of measurements (e.g. median household income, occupation, distance from nearest urban area, and so forth) for a large number of people.
- Our goal is to perform *market segmentation* by identifying subgroups of people who might be more receptive to a particular form of advertising, or more likely to purchase a particular product.
- The task of performing market segmentation amounts to clustering the people in the data set.

Two clustering methods

- In *K-means clustering*, we seek to partition the observations into a pre-specified number of clusters.
- In *hierarchical clustering*, we do not know in advance how many clusters we want; in fact, we end up with a tree-like visual representation of the observations, called a *dendrogram*, that allows us to view at once the clusterings obtained for each possible number of clusters, from 1 to n.

K-means clustering



A simulated data set with 150 observations in 2-dimensional space. Panels show the results of applying K-means clustering with different values of K, the number of clusters. 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.

Details of K-means clustering

Let $C_1, \ldots, C_K$ denote sets containing the indices of the observations in each cluster. These sets satisfy two properties:

1. $C_1 \cup C_2 \cup \ldots \cup C_K = \{1, \ldots, n\}$. In other words, each observation belongs to at least one of the K clusters.
2. $C_k \cap C_{k'} = \emptyset$ for all $k \neq k'$. In other words, the clusters are non-overlapping: no observation belongs to more than one cluster.

For instance, if the ith observation is in the kth cluster, then $i \in C_k$.

Details of K-means clustering

- The idea behind K-means clustering is that a *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 $\mathrm{WCV}(C_k)$ of the amount by which the observations within a cluster differ from each other.
- Hence we want to solve the problem

$$\underset{C_1,\ldots,C_K}{\text{minimize}} \left\{ \sum_{k=1}^{K} \mathrm{WCV}(C_k) \right\}. \tag{1}$$

- In words, this formula says that we want to partition the observations into K clusters such that the total within-cluster variation, summed over all K clusters, is as small as possible.

How to define within-cluster variation?

- Typically we use Euclidean distance

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

where $|C_k|$ denotes the number of observations in the kth cluster.

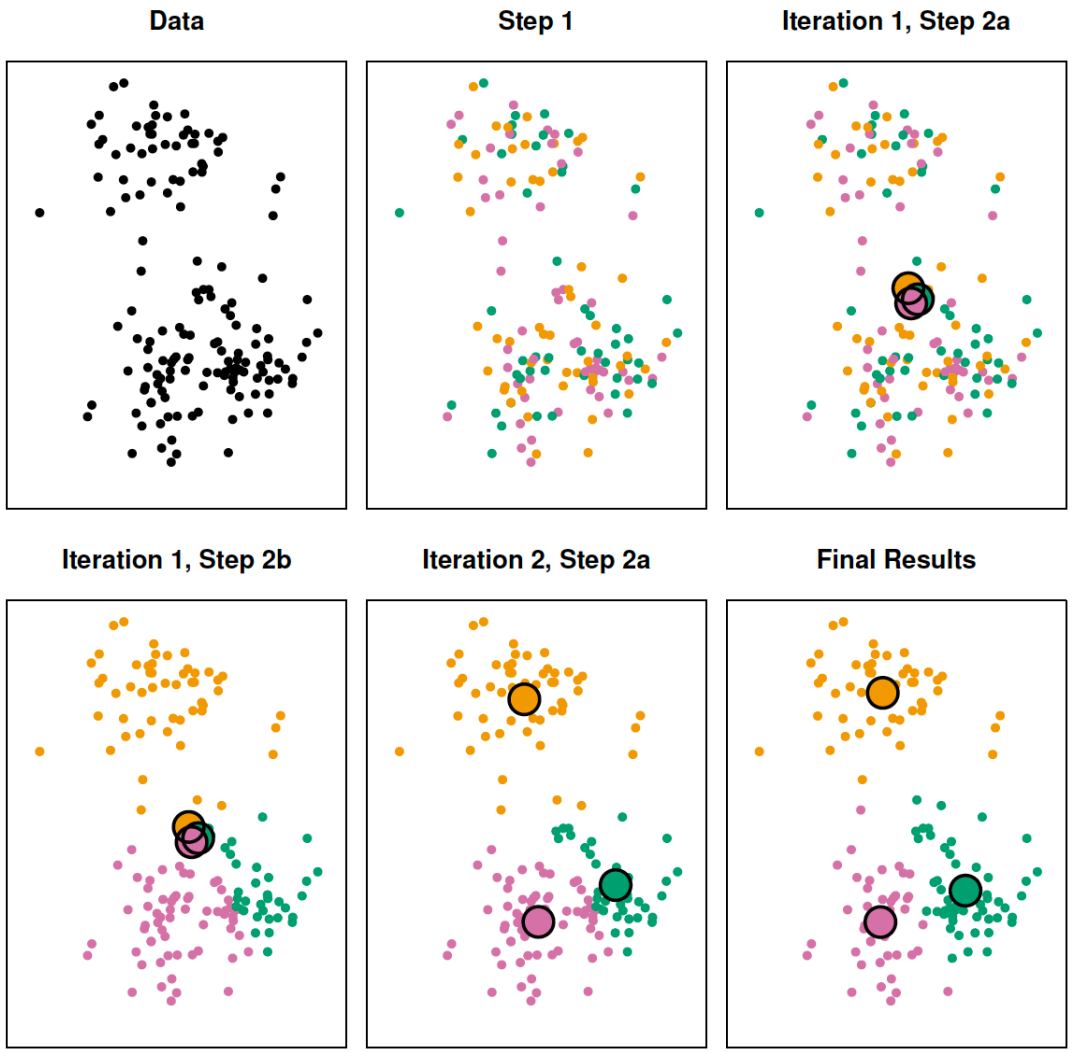
- Combining (1) and (2) gives the optimization problem that defines K-means clustering,

$$\underset{C_1,\ldots,C_K}{\text{minimize}} \left\{ \sum_{k=1}^{K} \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 \right\}. \qquad (3)$$

K-means clustering algorithm

1. Randomly assign a number, from 1 to K, to each of the observations. These serve as initial cluster assignments for the observations.
2. Iterate until the cluster assignments stop changing:
   - 2.1 For each of the K clusters, compute the cluster *centroid*. The kth cluster centroid is the vector of the p feature means for the observations in the kth cluster.
   - 2.2 Assign each observation to the cluster whose centroid is closest (where *closest* is defined using Euclidean distance).

Example


Data
Step 1
Iteration 1, Step 2a
Iteration 1, Step 2b
Iteration 2, Step 2a
Final Results

Properties of the algorithm

- This algorithm is guaranteed to decrease the value of the objective (3) at each step.

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

  where $\bar{x}_{kj} = \frac{1}{|C_k|}\sum_{i \in C_k} x_{ij}$ is the mean for feature j in cluster C_k.

- however it is not guaranteed to give the global minimum.

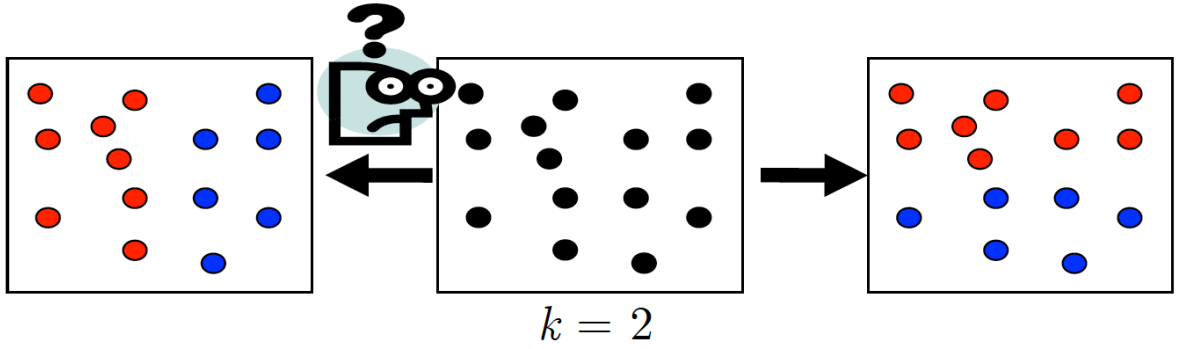
Example: different starting values

Details of the previous figure

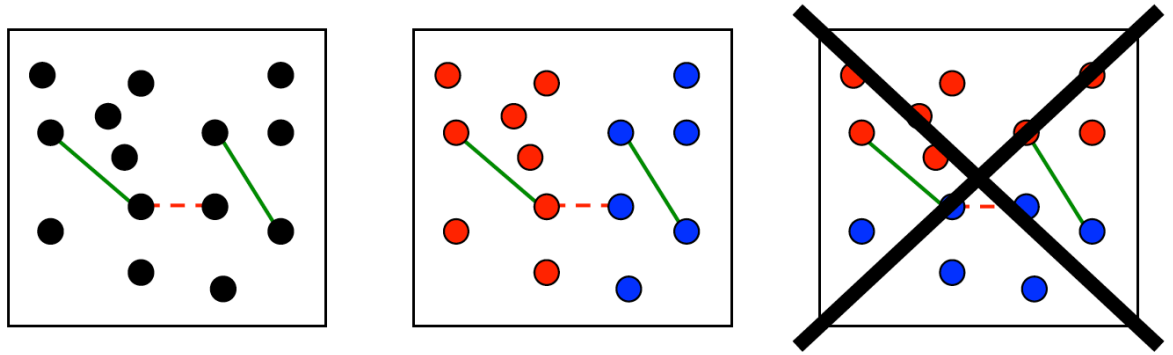
K-means clustering performed six times on the data from previous figure with $K = 3$, each time with a different random assignment of the observations in Step 1 of the K-means algorithm.
Above each plot is the value of the objective (4).
Three different local optima were obtained, one of which resulted 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

Semi-supervised k-means

- How do you tell it which clustering you want?

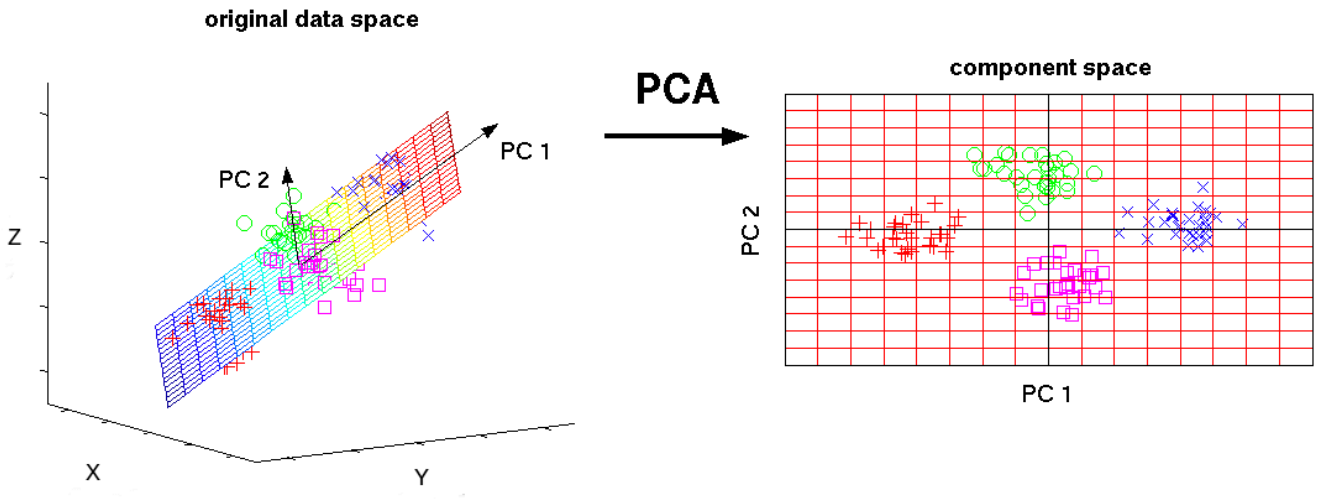


Constrained clustering techniques (semi-supervised)

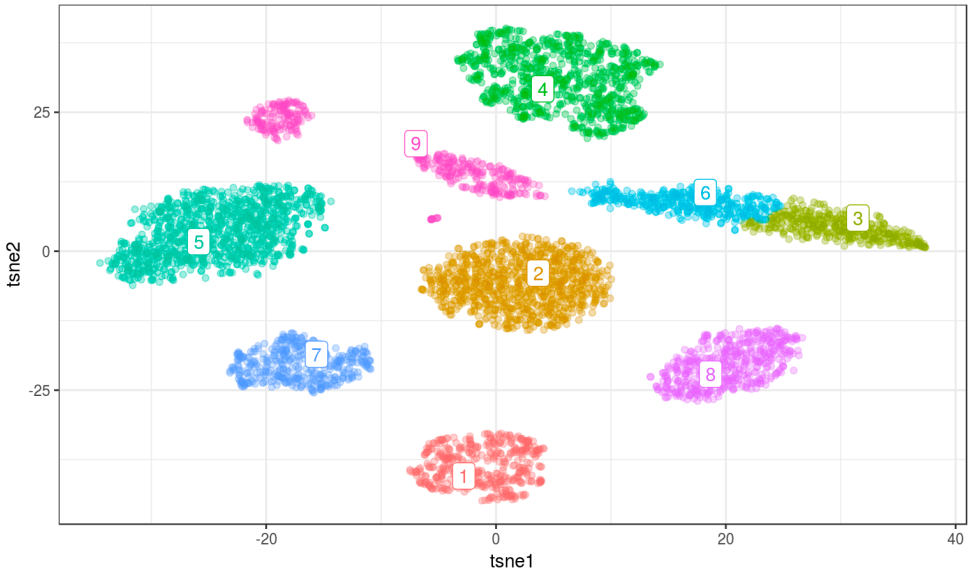


Dimension reduction + k-means

PCA + K-means



tSNE + K-means

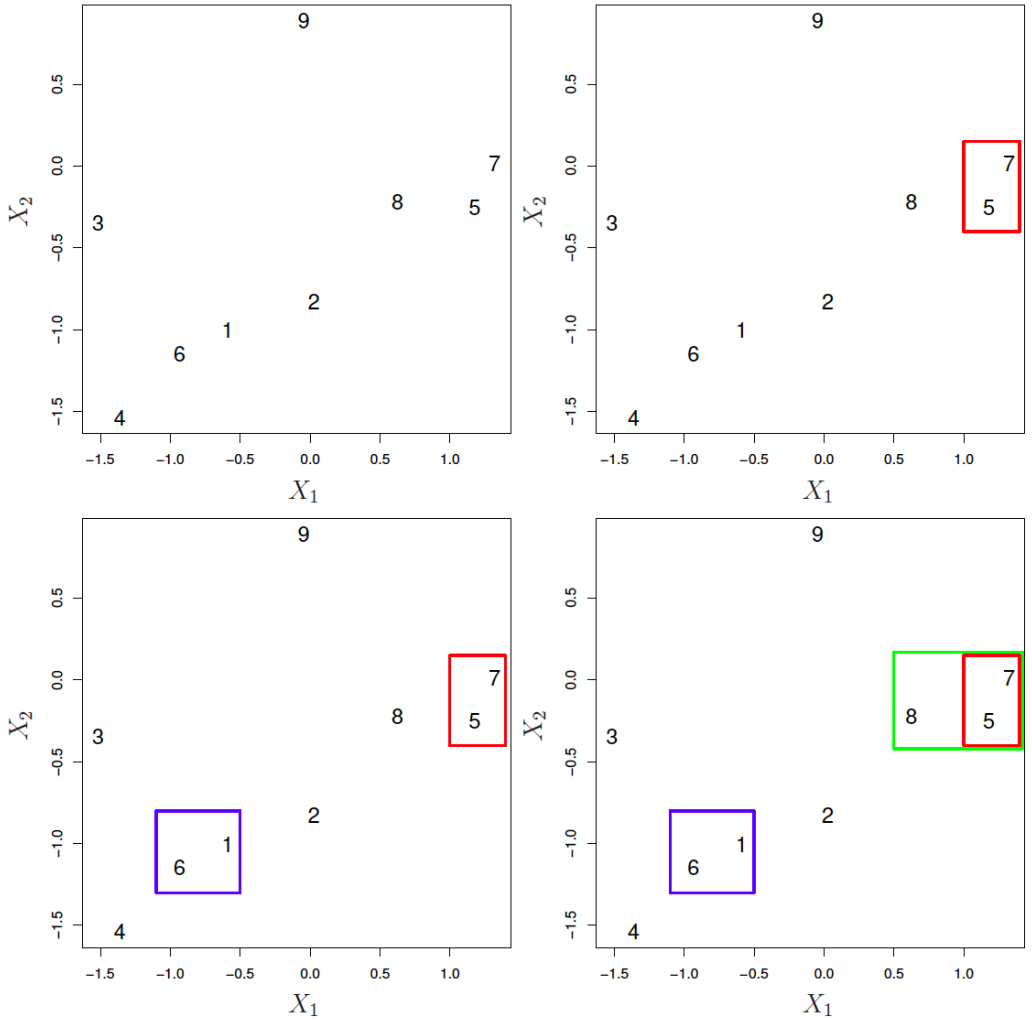


Hierarchical clustering

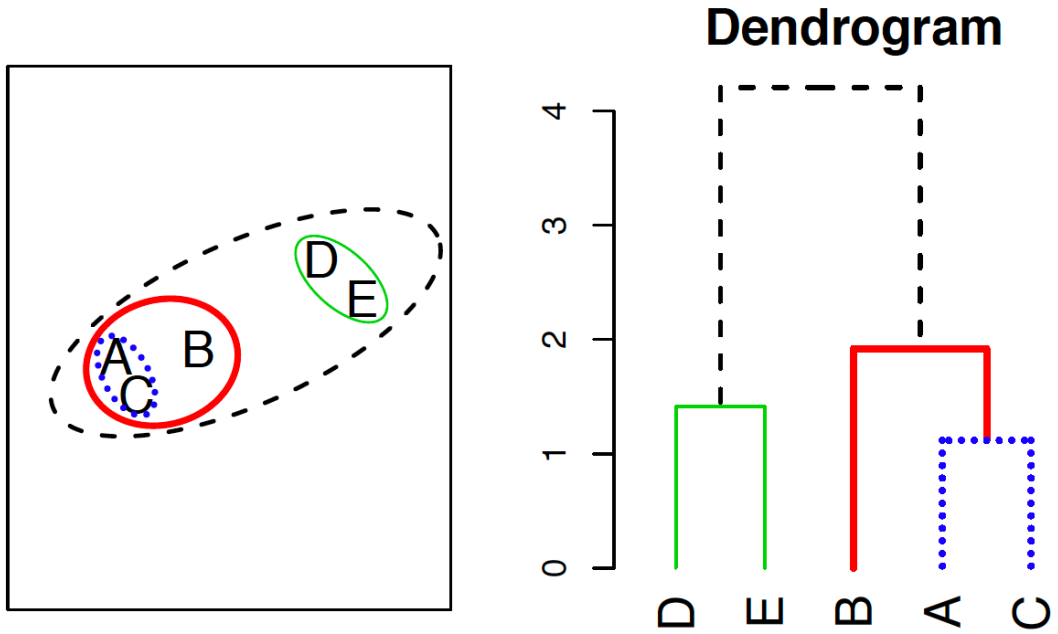
- K-means clustering requires us to pre-specify the number of clusters K. This can be a disadvantage (later we discuss strategies for choosing K)
- *Hierarchical clustering* is an alternative approach which does not require that we commit to a particular choice of K.
- In this section, we describe *bottom-up* or *agglomerative* clustering. This is the most common type of hierarchical clustering, and refers to the fact that a dendrogram is built starting from the leaves and combining clusters up to the trunk.

Merges in previous example

Builds a hierarchy in a "bottom-up" fashion...



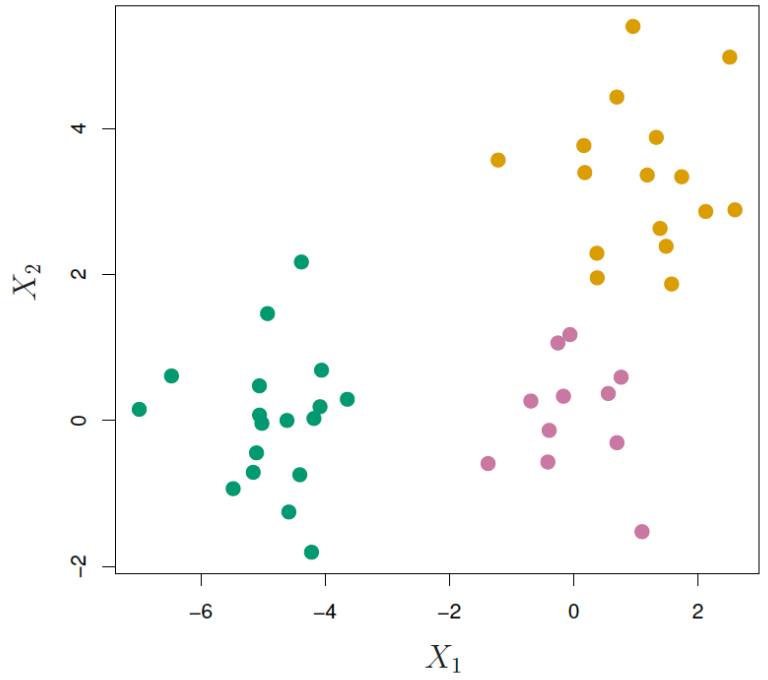
Hierarchical clustering algorithm



The approach in words:

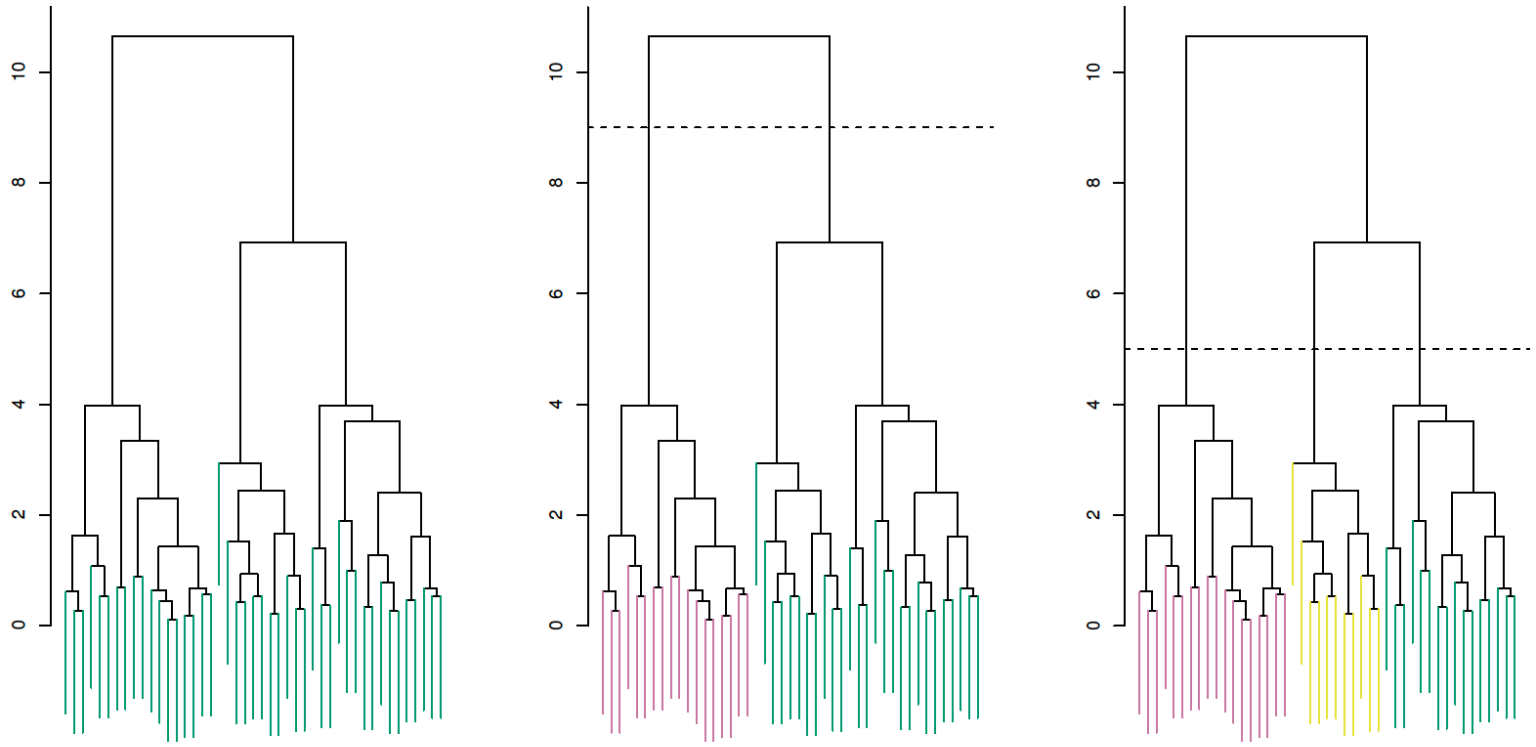
- Start with each point in its own cluster.
- Identify the closest two clusters and merge them.
- Repeat.
- Ends when all points are in a single cluster.

Example



45 observations generated in 2-dimensional space. In reality there are three distinct classes, shown in separate colors. However, we will treat these class labels as unknown and will seek to cluster the observations in order to discover the classes from the data.

Application of hierarchical clustering



Details of the previous figure

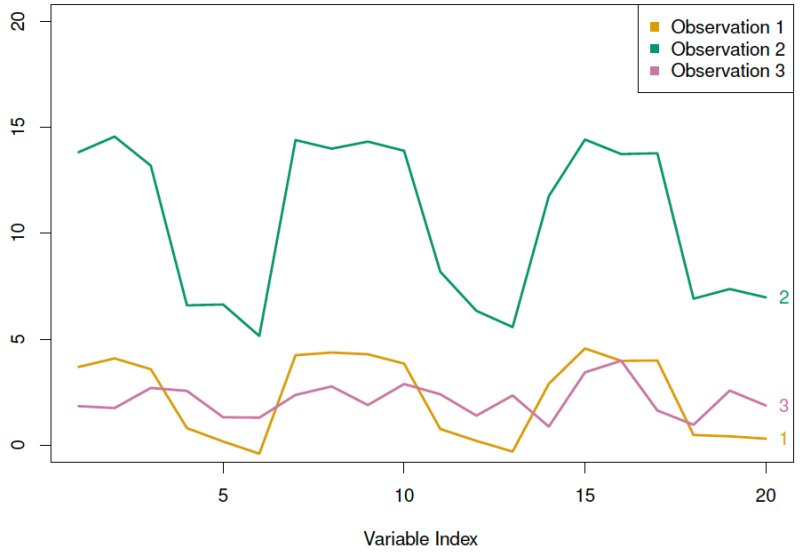
- *Left:* Dendrogram obtained from hierarchically clustering the data from previous slide, with complete linkage and Euclidean distance.
- *Center:* The dendrogram from the left-hand panel, cut at a height of 9 (indicated by the dashed line). This cut results in two distinct clusters, shown in different colors.
- *Right:* The dendrogram from the left-hand panel, now cut at a height of 5. This cut results in three distinct clusters, shown in different colors. Note that the colors were not used in clustering, but are simply used for display purposes in this figure

Type of linkage

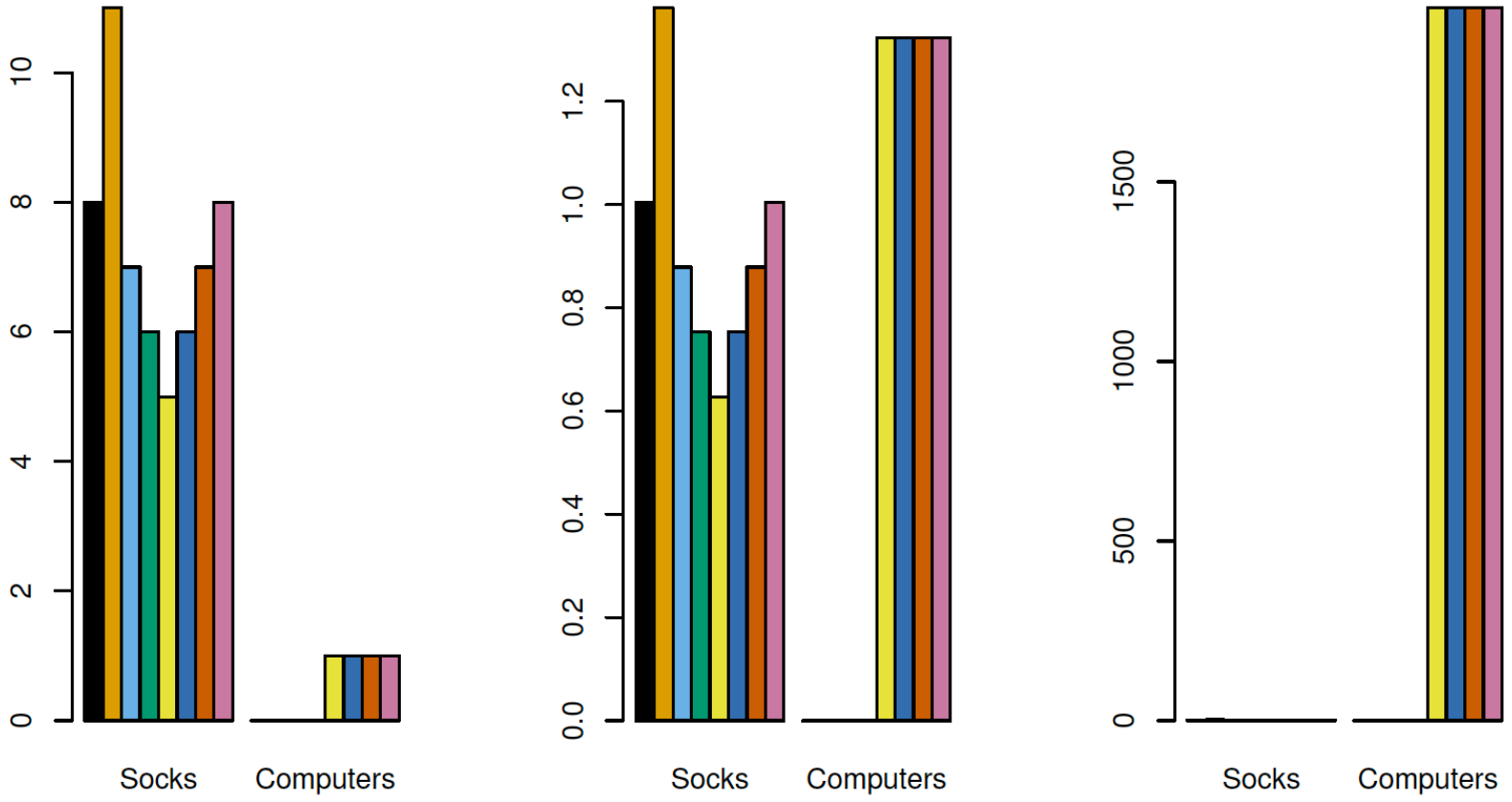
Linkage	*Description*
Complete	Maximal inter-cluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *largest* of these dissimilarities.
Single	Minimal inter-cluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *smallest* of these dissimilarities.
Average	Mean inter-cluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *average* of these dissimilarities.
Centroid	Dissimilarity between the centroid for cluster A (a mean vector of length p) and the centroid for cluster B. Centroid linkage can result in undesirable *inversions*.

Choice of dissimilarity measure

- So far have used Euclidean distance.
- An alternative is *correlation-based distance* which considers two observations to be similar if their features are highly correlated.
- This is an unusual use of correlation, which is normally computed between variables; here it is computed between the observation profiles for each pair of observations.



Scaling of the variables matter

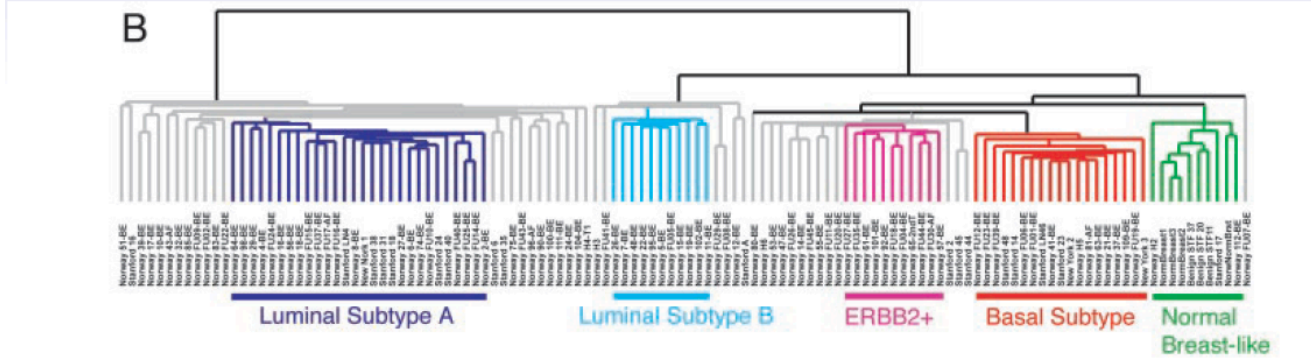


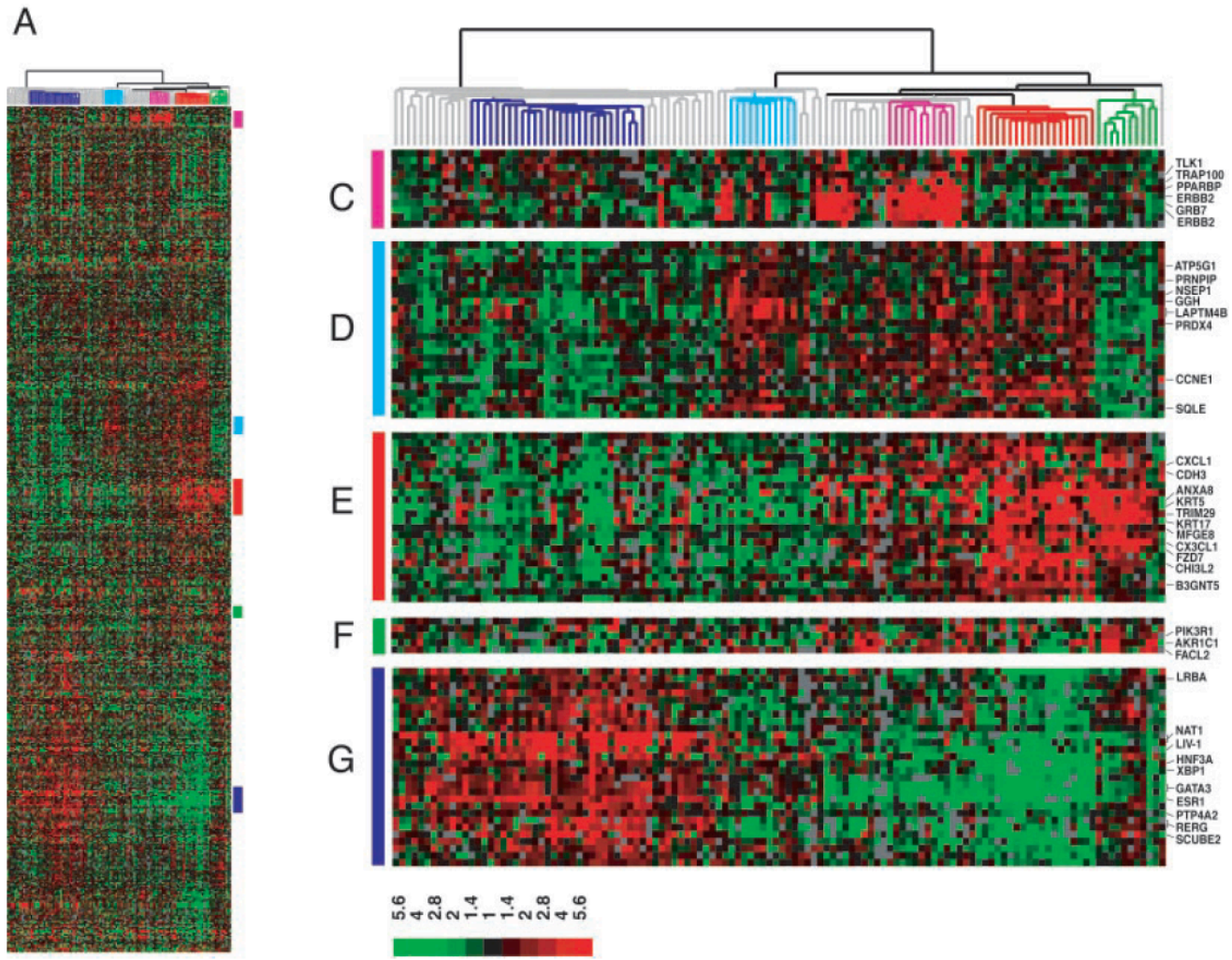
Practical issues

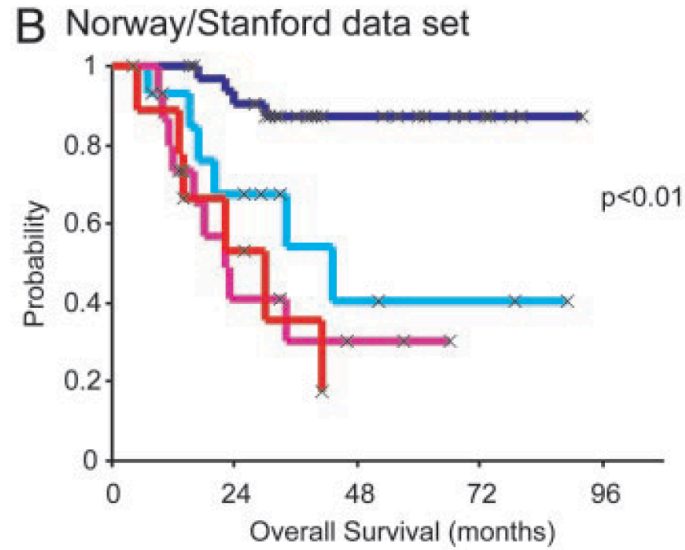
- Should the observations or features first be standardized in some way? For instance, maybe the variables should be centered to have mean zero and scaled to have standard deviation one.
- In the case of hierarchical clustering,
  - What dissimilarity measure should be used?
  - What type of linkage should be used?
- How many clusters to choose? (in both K-means or hierarchical clustering). Difficult problem. No agreed-upon method. See Elements of Statistical Learning, chapter 13 for more details.

Example: breast cancer microarray study

- "Repeated observation of breast tumor subtypes in independent gene expression data sets;" Sorlie at el, PNAS 2003
- Average linkage, correlation metric
- Clustered samples using 500 *intrinsic genes:* each woman was measured before and after chemotherapy. Intrinsic genes have smallest within/between variation.


B
Luminal Subtype A
Luminal Subtype B
ERBB2+
Basal Subtype
Normal
Breast-like


A
C
D
E
F
G
TLK1
TRAP100
PPARBP
ERBB2
GRB7
ERBB2
ATP5G1
PRNPIP
NSEP1
GGH
LAPTM4B
PRDX4
CCNE1
SQLE
CXCL1
CDH3
ANXA8
KRT5
TRIM29
KRT17
MFGE8
CX3CL1
FZD7
CHI3L2
B3GNT5
PIK3R1
AKR1C1
FACL2
LRBA
NAT1
LIV-1
HNF3A
XBP1
GATA3
ESR1
PTP4A2
RERG
SCUBE2
5.6
4
2.8
2
1.4
1
1.4
2
2.8
4
5.6


B Norway/Stanford data set
Probability
1
0.8
0.6
0.4
0.2
0
0
24
48
72
96
Overall Survival (months)
p<0.01

Conclusions

- *Unsupervised learning* is important for understanding the variation and grouping structure of a set of unlabeled data, and can be a useful pre-processor for supervised learning
- It is intrinsically more difficult than *supervised learning* because there is no gold standard (like an outcome variable) and no single objective (like test set accuracy)
- It is an active field of research, with many recently developed tools such as *self-organizing maps*, *independent components analysis* and *spectral clustering*.