Lecture 2: Classification, Clustering

STATS 202: Data mining and analysis

Rajan Patel

Classification problem

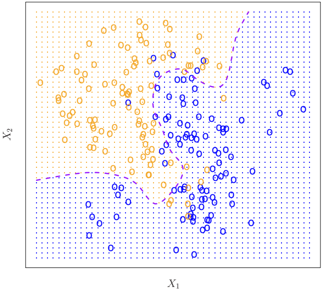


Figure: *

Figure 2.13

Recall:

- $X = (X_1, X_2)$ are inputs.
- Color $Y \in \{$Yellow , Blue$\}$ is the output.
- (X, Y) have a joint distribution.
- Purple line is *Bayes boundary* — the best we could do if we knew the joint distribution of (X, Y)

K-nearest neighbors

To assign a color to the input $\times$, we look at its $K = 3$ nearest neighbors. We predict the color of the majority of the neighbors.

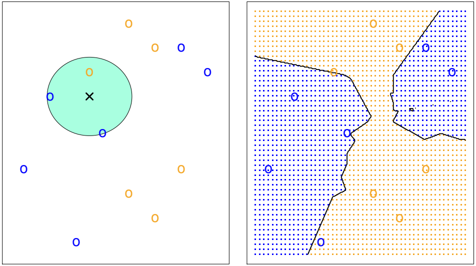


Figure: *

Figure 2.14

K-nearest neighbors also has a decision boundary

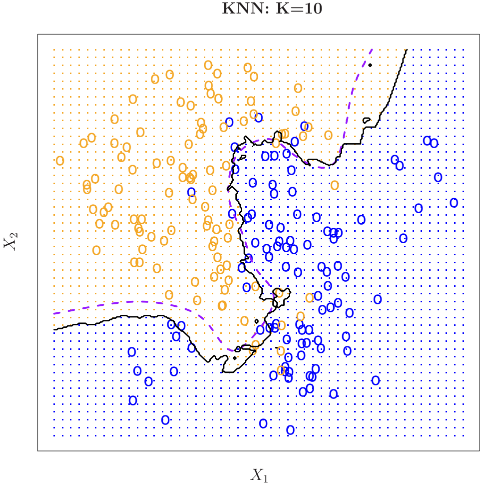


Figure: *

Figure 2.15

The higher K, the smoother the decision boundary

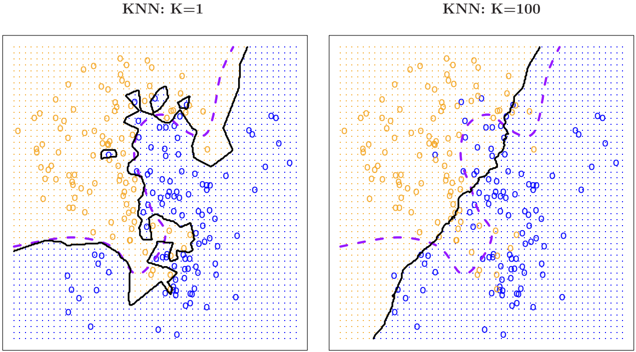


Figure: *

Figure 2.16

Clustering

As in **classification**, we assign a class to each sample in the data matrix. However, the class *is not an output variable*; we only use input variables.

Clustering is an **unsupervised** procedure, whose goal is to find homogeneous subgroups among the observations.

We will discuss 2 algorithms:

- K-means clustering
- Hierarchical clustering

K-means clustering

- K is the number of clusters and must be fixed in advance.
- The goal of this method is to maximize the similarity of samples within each cluster:

$$\min_{C_1,\dots,C_K} \sum_{\ell=1}^{K} W(C_\ell) \quad ; \quad W(C_\ell) = \frac{1}{|C_\ell|} \sum_{i,j \in C_\ell} \text{Distance}^2(x_{i,:}, x_{j,:}).$$

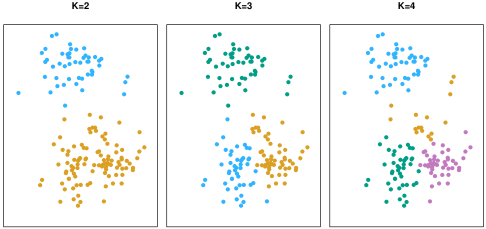


Figure: *

Figure 10.5

K-means clustering algorithm

1. Assign each sample to a cluster from 1 to K arbitrarily, e.g. at random.
2. Iterate these two steps until the clustering is constant:
   - Find the *centroid* of each cluster ℓ; i.e. the average $\overline{x}_{\ell,:}$ of all the samples in the cluster:
     $$x_{\ell,j} = \frac{1}{|C_\ell|} \sum_{i \in C_\ell} x_{i,j} \quad \text{for } j = 1, \ldots, p.$$
   - Reassign each sample to the nearest centroid.

K-means clustering algorithm

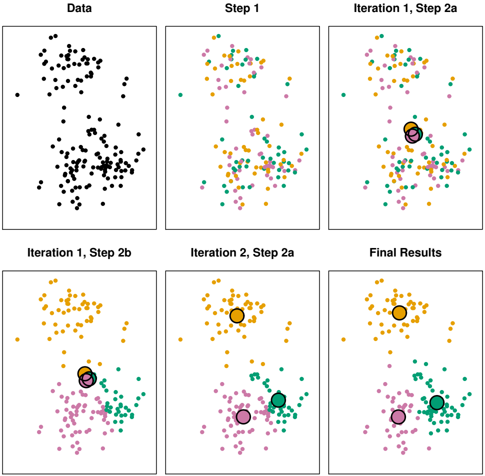


Figure: *

Figure 10.6

Properties of K-means clustering

- The algorithm always converges to a local minimum of

$$\min_{C_1,\ldots,C_K} \sum_{\ell=1}^{K} W(C_\ell) \quad ; \quad W(C_\ell) = \frac{1}{|C_\ell|} \sum_{i,j \in C_\ell} \mathsf{Distance}^2(x_{i,:}, x_{j,:}).$$

- Each initialization could yield a different minimum.

Example: K-means output with different initializations

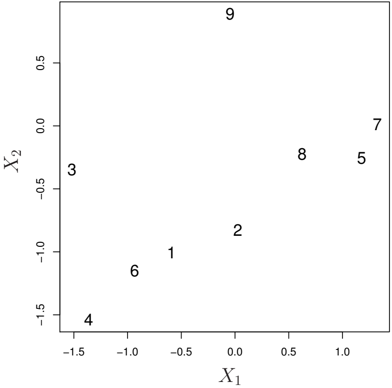
Figure: *

Figure 10.7

In practice, we start from many random initializations and choose the output which minimizes the objective function.

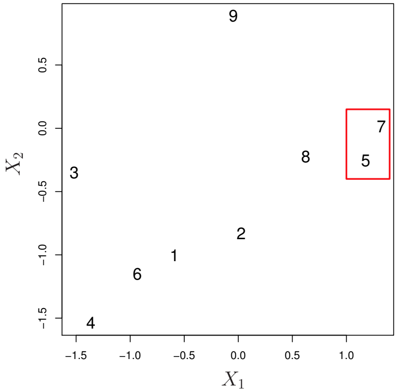
Hierarchical clustering

Most algorithms for hierarchical clustering are *agglomerative*.



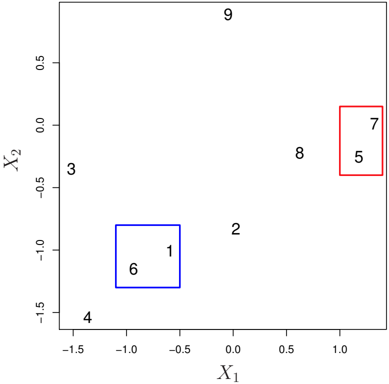
Hierarchical clustering

Most algorithms for hierarchical clustering are *agglomerative*.



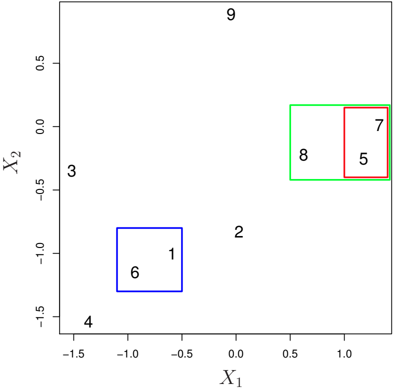
Hierarchical clustering

Most algorithms for hierarchical clustering are *agglomerative*.



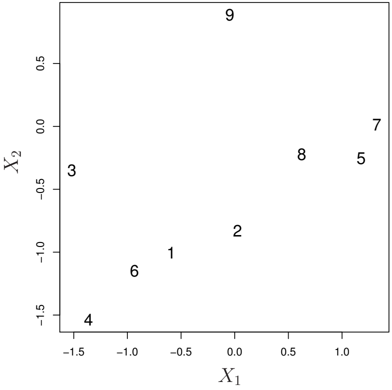
Hierarchical clustering

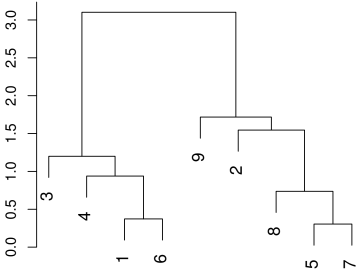
Most algorithms for hierarchical clustering are *agglomerative*.



Hierarchical clustering

Most algorithms for hierarchical clustering are *agglomerative*.

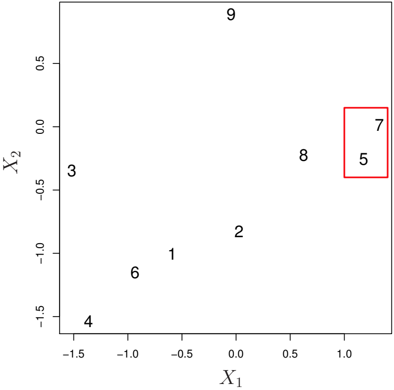


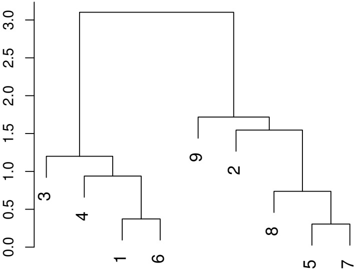


The output of the algorithm is a *dendogram*.

Hierarchical clustering

Most algorithms for hierarchical clustering are *agglomerative*.

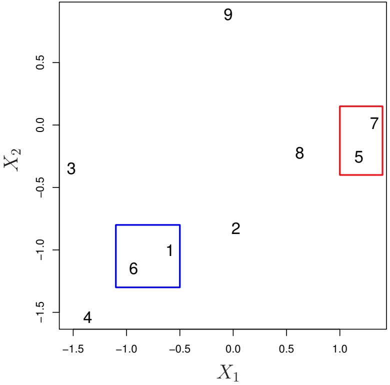


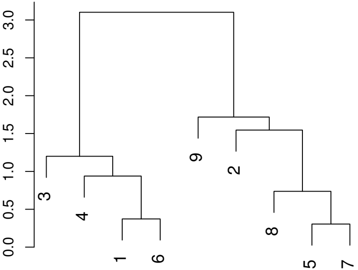


The output of the algorithm is a *dendogram*.

Hierarchical clustering

Most algorithms for hierarchical clustering are *agglomerative*.

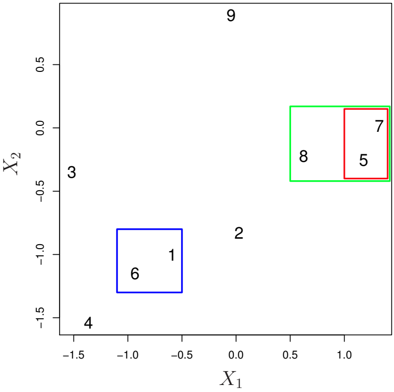


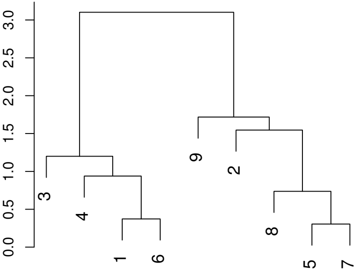


The output of the algorithm is a *dendogram*.

Hierarchical clustering

Most algorithms for hierarchical clustering are *agglomerative*.

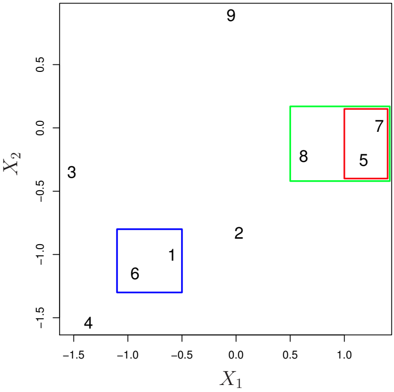


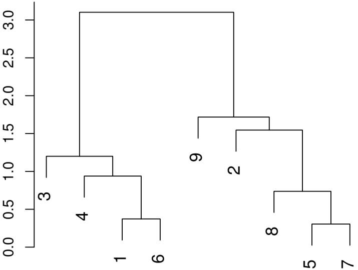


The output of the algorithm is a *dendogram*.

Hierarchical clustering

Most algorithms for hierarchical clustering are *agglomerative*.





We must be careful about how we interpret the dendogram.

Hierarchical clustering

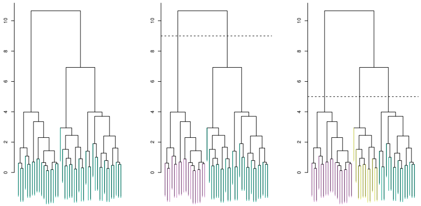


Figure: *

Figure 10.9

- The number of clusters is not fixed.

Hierarchical clustering

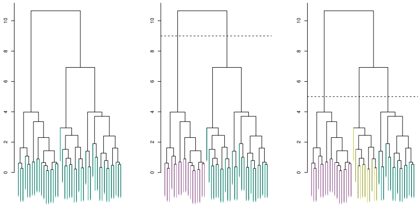


Figure: *

Figure 10.9

- The number of clusters is not fixed.
- Hierarchical clustering is not always appropriate.

Hierarchical clustering

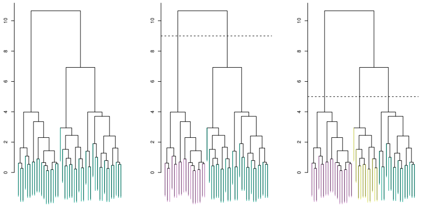


Figure: *

Figure 10.9

- The number of clusters is not fixed.
- Hierarchical clustering is not always appropriate.

  e.g. Market segmentation for consumers of 3 different nationalities.
  - Natural 2 clusters: gender
  - Natural 3 clusters: nationality

  These clusterings are not nested or hierarchical.

Notion of distance between clusters

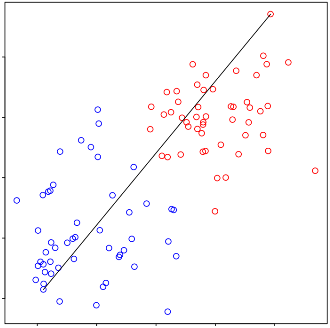
At each step, we link the 2 clusters that are “closest” to each other.

Hierarchical clustering algorithms are classified according to the notion of distance between clusters.

Notion of distance between clusters

At each step, we link the 2 clusters that are "closest" to each other.

Hierarchical clustering algorithms are classified according to the notion of distance between clusters.

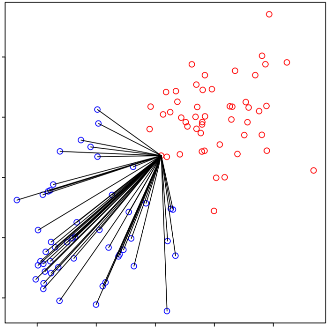


Complete linkage:
The distance between 2 clusters is the *maximum* distance between any pair of samples, one in each cluster.

Notion of distance between clusters

At each step, we link the 2 clusters that are "closest" to each other.

Hierarchical clustering algorithms are classified according to the notion of distance between clusters.

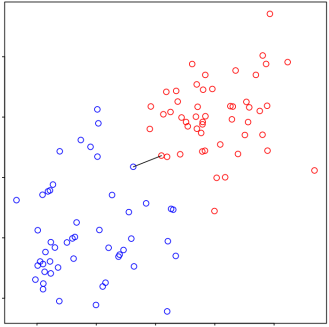


Average linkage:
The distance between 2 clusters is the average of all pairwise distances.

Notion of distance between clusters

At each step, we link the 2 clusters that are "closest" to each other.

Hierarchical clustering algorithms are classified according to the notion of distance between clusters.



Single linkage:
The distance between 2 clusters is the *minimum* distance between any pair of samples, one in each cluster.

Suffers from the *chaining phenomenon*

Example

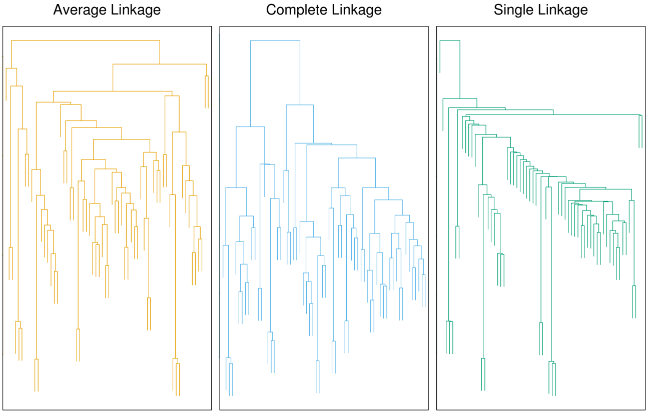


Figure: *

Figure 10.12

Clustering is riddled with questions and choices

- Is clustering appropriate? i.e. Could a sample belong to more than one cluster?
  - Mixture models, soft clustering, topic models.
- How many clusters are appropriate?
  - Choose subjectively — depends on the inference sought.
  - There are formal methods based on gap statistics, mixture models, etc.
- Are the clusters robust?
  - Run the clustering on different random subsets of the data. Is the structure preserved?
  - Try different clustering algorithms. Are the conclusions consistent?
  - Most important: temper your conclusions.

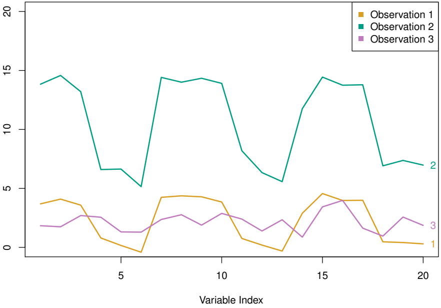
Clustering is riddled with questions and choices

- Should we scale the variables before doing the clustering.
  - Variables with larger variance have a larger effect on the Euclidean distance between two samples.
- Does Euclidean distance capture dissimilarity between samples?

Correlation distance

Example: Suppose that we want to cluster customers at a store for market segmentation.

- Samples are customers
- Each variable corresponds to a specific product and measures the number of items bought by the customer during a year.



Correlation distance

- Euclidean distance would cluster all customers who purchase few things (orange and purple).
- Perhaps we want to cluster customers who purchase *similar* things (orange and teal).
- Then, the **correlation distance** may be a more appropriate measure of dissimilarity between samples.