5. Models based on distances

Distance is the amount of space between two samples. Formally, a distance is a function with the following characteristics:

- It is not negative. $D(x,y) \geq 0,\ \forall x,y$
- It is symmetric. $D(x,y) = D(y,x),\ \forall x,y$
- It satisfies the triangle inequality. $D(x,y) \leq D(x,z) + D(z,y),\ \forall x,y,z$
- The distance between a sample and itself is 0. $D(x,x) = 0,\ \forall x$

Some distances are:

Vectors	
Euclidean distance	$\left\|\|A-B\right\|\|_2 = \sqrt{\sum_i (A_i - B_i)^2}$
Manhattan distance	$\left\|\|A-B\right\|\|_1 = \sum_i \|A_i - B_i\|$
Maximum distance	$\left\|\|A-B\right\|\|_\infty = \max_i \|A_i - B_i\|$
Mahalanobis distance	$D_{Mahalanobis}(A,B) = \sqrt{(A-B)^T \Sigma^{-1} (A-B)}$ Σ is the covariance matrix
Cosine similarity	$\text{Cos\_sim}(A,B) = \frac{A.B}{\left\|\|A\right\|\|\left\|\|B\right\|\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i}\sqrt{\sum_{i=1}^n B_i}}$
Words	
Hamming distance	Levenshtein distance
Probability distributions	
Kullback – Leibler divergence	$D_{KL}(P,Q) = \sum_i P(i) \ln\left(\frac{P(i)}{Q(i)}\right)$

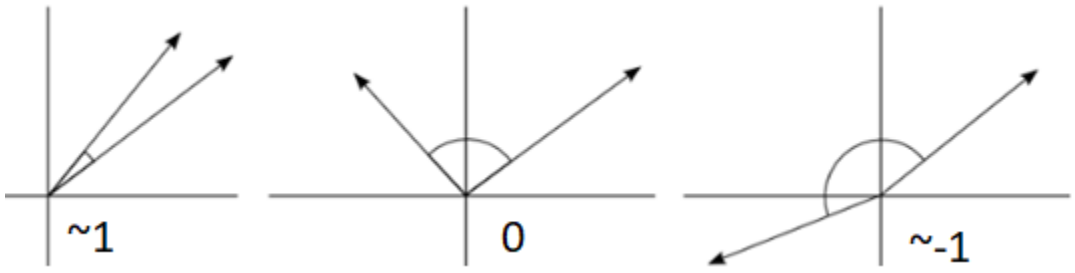
Cosine similarity

It measures the cosine of the angle between two vectors A and B. In other words, it measures the similarity between vector directions.

$$\cos(\theta) = \frac{A.B}{\left\|\|A\right\|\|\left\|\|B\right\|\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i}\sqrt{\sum_{i=1}^n B_i}}$$

The value can be between -1 and 1:

- -1, it means the vectors are opposite
- 0, it means the vectors are orthogonal
- 1, it means the vectors have the same direction



Mahalanobis distance

Explanation with an example:

Imagine that a fisher wants to measure the similarity among salmons because he wants to classify them into two groups for selling the bigger ones at a higher price. For each salmon, he measures the width and the length. Each salmon can be represented as a vector whose entries are these measures $\overrightarrow{x_i} = [x_{1i}, x_{2i}]^T$.

The length is a random variable with values between 50 and 100cm, whereas the width values are between 10 and 20cm. If the fisher uses a Euclidean distance, the length will have more importance than the width. For that reason, he decides to use the following equation:

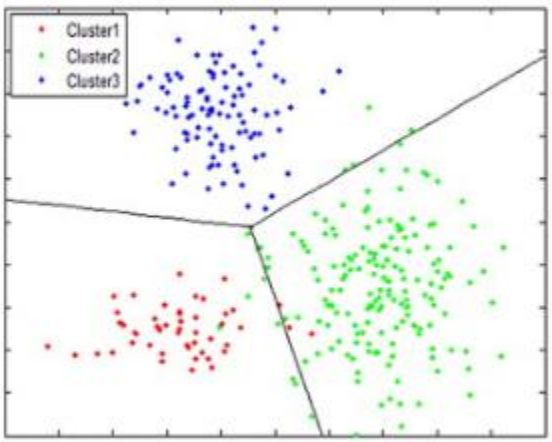
$$Mahalanobis\_distance(x_1, x_2) = \sqrt{\left(\frac{x_{11} - x_{12}}{\sigma_1}\right)^2 + \left(\frac{x_{21} - x_{22}}{\sigma_2}\right)^2} = \sqrt{(\overrightarrow{x_i} - \overrightarrow{x_2})^T S^{-1} (\overrightarrow{x_i} - \overrightarrow{x_2})}$$

where $S = \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix}$

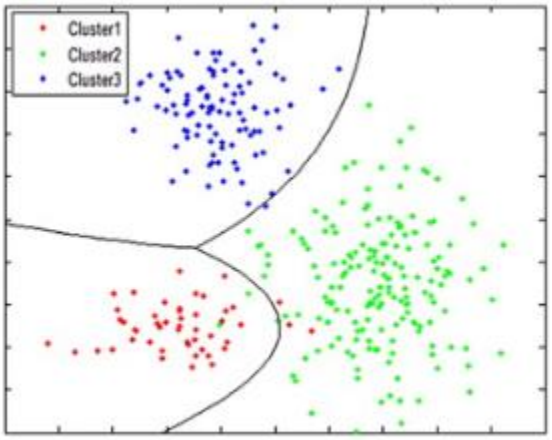
In general, the equation of Mahalanobis distance is:

$$Mahalanobis\_distance(x_1, x_2) = \sqrt{(\overrightarrow{x_i} - \overrightarrow{x_2})^T \Sigma^{-1} (\overrightarrow{x_i} - \overrightarrow{x_2})}$$

where Σ is the covariance matrix



(a) Clusters' boundaries by Euclidean distance



(b) Clusters' boundaries by Mahalanobis distance

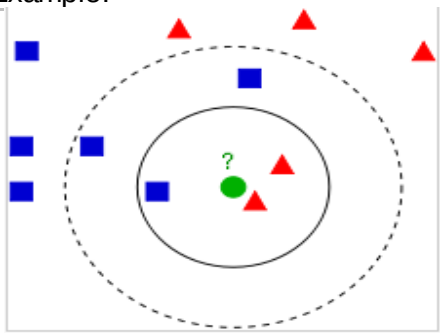
5.1 k - Nearest Neighbors (KNN)

Supervised learning: Classification
Variable type: all

It is a simple classifier that assigns the label that corresponds to the mode of the k nearest neighbors. It is sensible to the value of k.
Disadvantage: It has high complexity. To predict a label, it calculates the distance against the sample and all the training samples to find the k nearest neighbors.

Input: samples, k (number of neighbors) and the point to be predicted
Begin
  Get the k nearest neighbors to the point to be predicted
  Return the label that corresponds to the mode of the k nearest neighbors' labels
End

Example:



3 – nearest neighbors
  The circle gets the label of red triangle

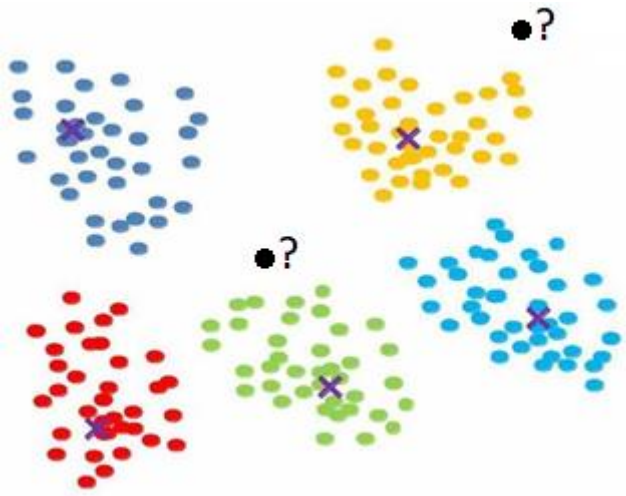
5 – nearest neighbors
  The circle gets the label of blue square

11 – nearest neighbors
  The circle gets the label of red triangle

5.2 Nearest Centroid

Unsupervised learning: Classification
Variable type: continuous

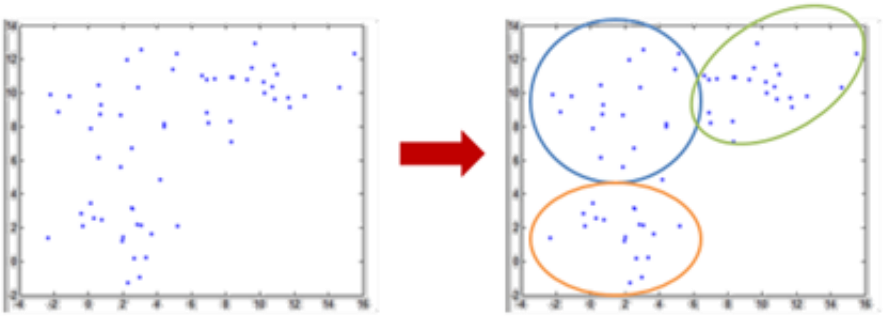
It is a simple classifier that represents each class by the centroid of its samples. It assigns the label corresponding to the class whose centroid is the nearest to the sample to be predicted.
Disadvantage: The problem is that it assumes unimodal distributions on the classes.



5.3 Hierarchical clustering

Unsupervised learning: Clustering
Variable type: all

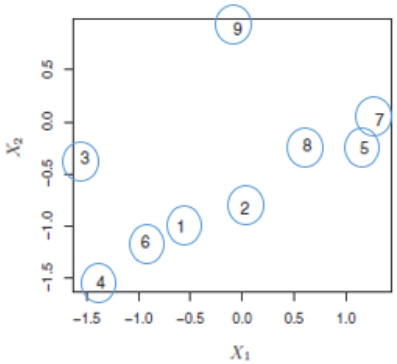
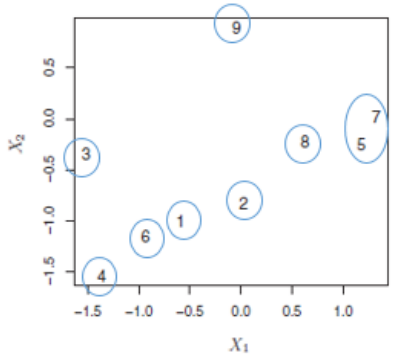
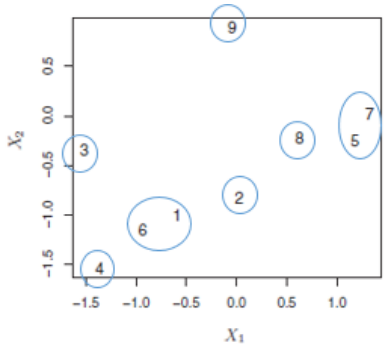
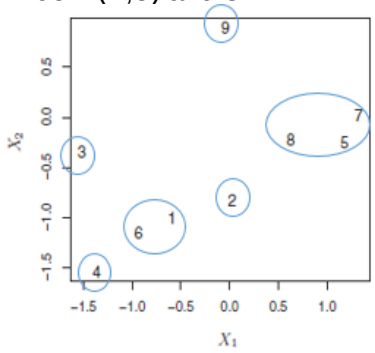
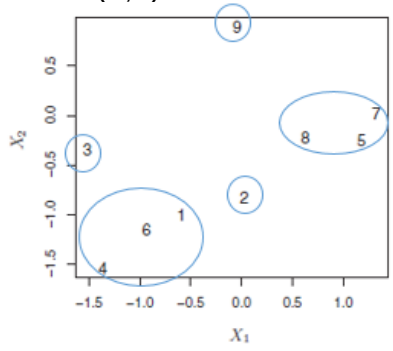
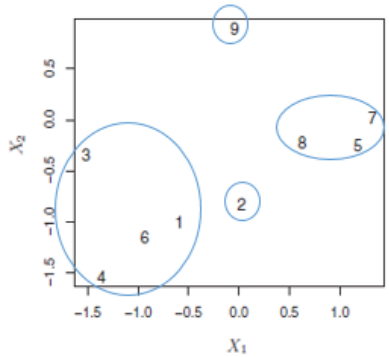
Remembering, clustering consists of group samples based on the features.



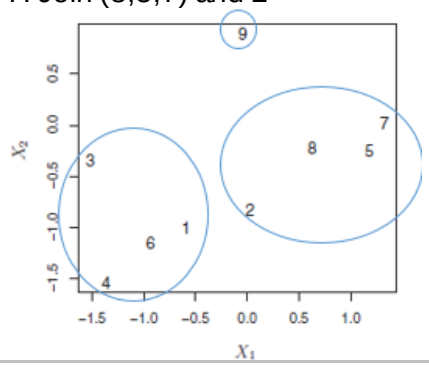
Hierarchical clustering works iteratively. The algorithm is:

Input: samples
Begin
  Each sample is a cluster
  Repeat until there is only one cluster
    Join the nearest two clusters
End

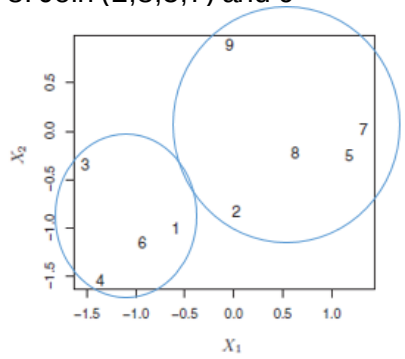
Example:

1. All the samples are different clusters 	2. Join 5 and 7 	3. Join 1 and 6 
4. Join (7,5) and 8 	5. Join (1,6) and 4 	6. Join (1,6,4) and 3 

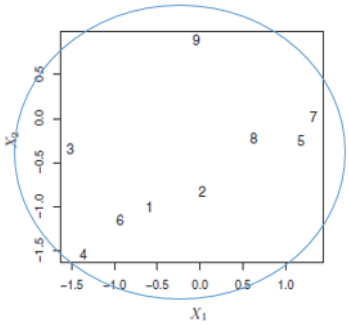
7. Join (8,5,7) and 2



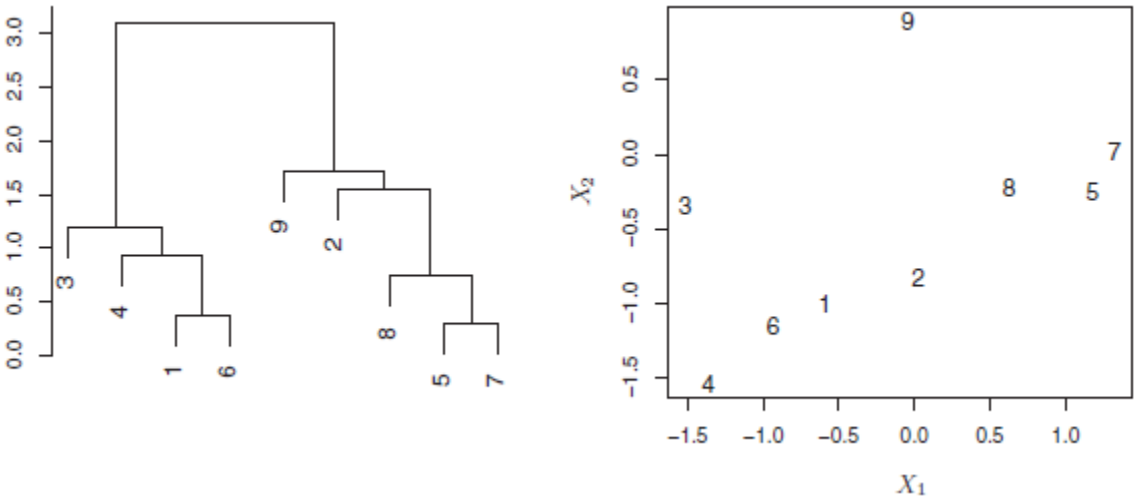
8. Join (2,8,5,7) and 9



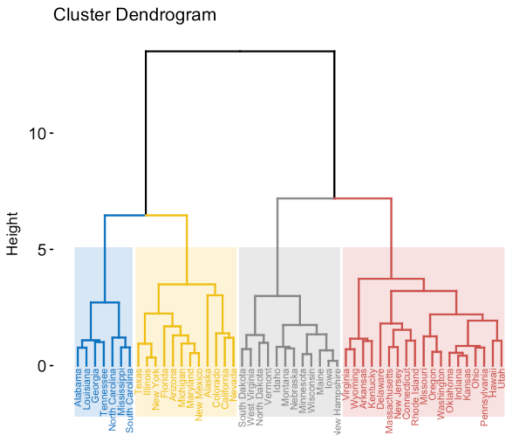
9. Join all



We can represent the clustering process with a dendrogram that is a binary tree where the length of the branch represents the distance where the samples were joined.



The dendrogram can be used to analyze the number of clusters. In addition, by pruning the tree, the clusters can be found.

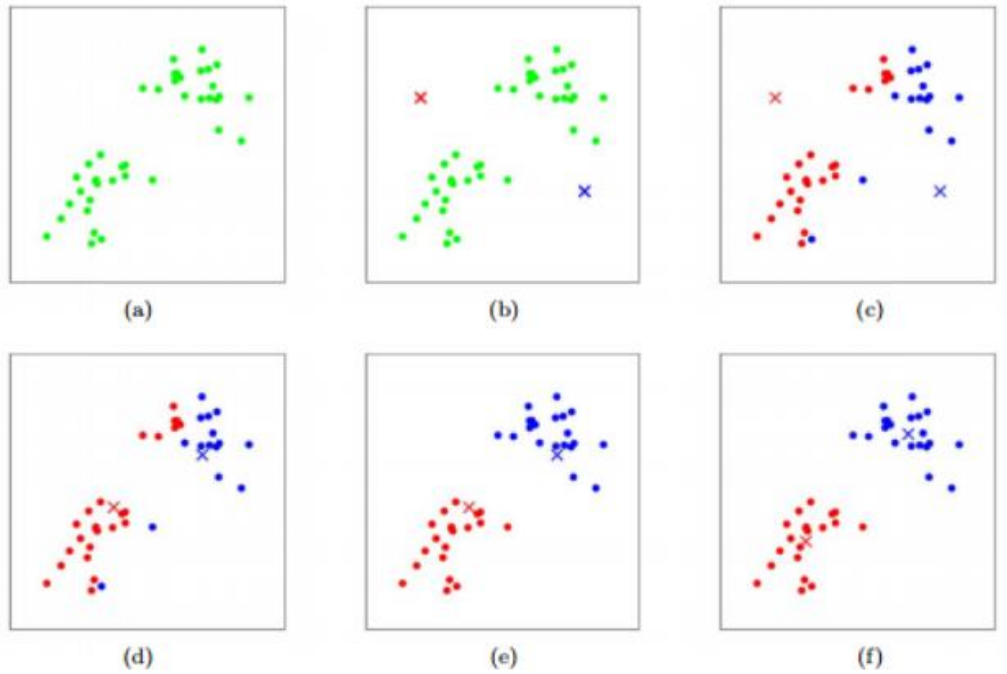


5.4 k - Means

Unsupervised learning: Clustering
Variable type: continuous

K-means finds k groups in the unlabeled data. An important parameter is the number of clusters we expect to find, called k. The algorithm is:

Input: samples and k (the number of clusters)
Begin
  Randomly select k prototypes
  Repeat until the prototypes do not move
    Assign the samples to the nearest prototype
    Update the prototypes as the centroid of the samples
End



For example, we can use k-means for grouping the image's colors