# Clustering

Course of Machine Learning
Master Degree in Computer Science
University of Rome “Tor Vergata”

Giorgio Gambosi

a.a. 2018-2019

### Problem

Given a dataset $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$, with $\mathbf{x}_i \in \mathbb{R}^d (i = 1, \ldots, n)$.

We wish to derive a set of clusters clusters (i.e. a partition of $\mathbf{X}$ into subsets of "near" elements). Clusters are represented by their prototypes $(\mathbf{m}_1, \ldots, \mathbf{m}_k)$, with $\mathbf{m}_j \in \mathbb{R}^d, j = 1, \ldots, k$.

### Rappresentation of a clustering

1. Cluster prototypes $(\mathbf{m}_1, \ldots, \mathbf{m}_k)$, with $\mathbf{m}_j \in \mathbb{R}^d (j = 1, \ldots, k)$
2. Element assignment to clusters: for each $\mathbf{x}_i$, $k$ binary flags $r_{ij} \in \{0, 1\}$, $j = 1, \ldots, k$. If $\mathbf{x}_i$ is assigned the $t$-th cluster, then $r_{it} = 1$ and $r_{ij} = 0$ for $j \neq t$

# Clustering types

### Partitional clustering

Given a set of items (points) $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, we wish to partition $\mathbf{X}$ by assigning each element to one out of $k$ clusters $C_1, \ldots, C_k$ in such a way to maximize (or minimize) a given cost $J$. The number $k$ of clusters could be given or should have to be computed.

### Hierarchical clustering

Given a set of items (points) $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, we wish to derive a set of nested partitions of $\mathbf{X}$, from the partition composed by all singletons (one cluster for each node) to the one composed by a single item (the whole set).

### Brute force methods

Check all partitions of a set of $n$ elements into $k$ subsets, selecting the one with minimum $J$. The number $P(n, k)$ of such partitions can be recursively defined as follows:

$$
\begin{aligned}
P(n+1, k) &= P(n, k-1) + kP(n, k) \\
P(n, 1) &= 1 \\
P(n, n) &= 1
\end{aligned}
$$

It is possible to prove that this results in the following closed form characterization:

$$P(n, k) = \frac{1}{k!} \sum_{j=0}^{k} (-1)^{k-j} \binom{k}{j} j^n$$

This is the Stirling number of the second type which is known to be at least $\frac{1}{2}(k^2 + k + 2)k^{n-k-1}$ for $n \geq 2$, $1 \leq k \leq n-1$.

## Sum of squares

Let us define the cost a clustering as follows:

$$J(R, M) = \sum_{i=1}^{k} \sum_{j=1}^{n} r_{ij} ||\mathbf{x}_j - \mathbf{m}_i||^2 = \sum_{i=1}^{k} \sum_{j=1}^{n} r_{ij} (\mathbf{x}_j - \mathbf{m}_i)^T (\mathbf{x}_j - \mathbf{m}_i)$$

where

- $R_{ij} = r_{ij}$, where $r_{is} = 1$ and $r_{ij} = 0$ for $j \neq s$ if $x_i$ is assigned to cluster $C_s$
- $M_i = \mathbf{m}_i$, $i = 1, \ldots, k$ is the prototype (centroid) of cluster $C_i$,

$$\mathbf{m}_i = \frac{1}{n_i} \sum_{j=1}^{n} r_{ij} \mathbf{x}_j$$

# k-means clustering

Dataset $\mathbf{X} = (x_1, \ldots, x_n)$, $x_i \in \mathbb{R}^d$: we wish to derive $k$ clusters with prototypes $\mathbf{m}_1, \ldots, \mathbf{m}_k$

Assignment of elements to cluster: for each $x_i$, $k$ binary flags $r_{ij}$ $(j = 1, \ldots, k)$

- if $x_i$ is assigned to cluster $s$, then $r_{is} = 1$, and $r_{ij} = 0$ for $j \neq k$

Cost: sum of the distances of each point from the prototype of the corresponding cluster

$$J(R, M) = \sum_{i=1}^{n} \sum_{j=1}^{k} r_{ij} \, ||x_i - \mathbf{m}_j||^2$$

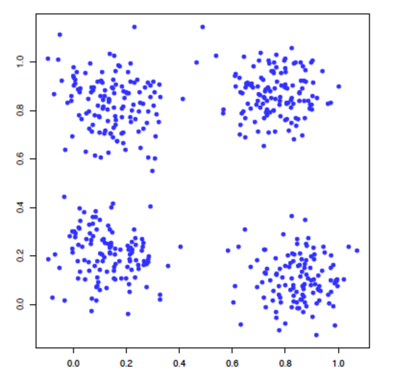
Objective: finding $r_{ij}$ and $\mathbf{m}_j$ $(i = 1, \ldots, n, \; j = 1, \ldots, k)$ to minimize $J(R, M)$

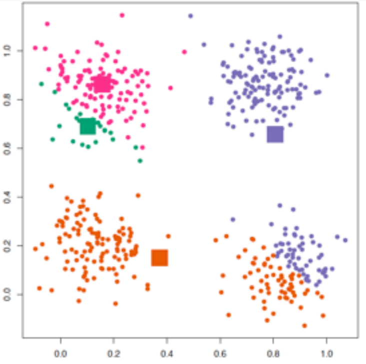
# Algorithm

1. Given a set of prototypes $\mathbf{m}_{ij}$, minimize wrt $r_{ij}$ (assigning elements to clusters).
   For each $x_i$, minimize $\sum_{j=1}^{k} r_{ij} \left|\left|x_i - \mathbf{m}_j\right|\right|^2$.
   The minimum is obtained for $r_{ik} = 1$ (and $r_{ij} = 0$ for $j \neq k$), where $\left|\left|x_i - \mathbf{m}_k\right|\right|^2$ is the minimum distance. That is, each point is assigned to the cluster of the nearest prototype.
2. Given a set of assignments $r_{ij}$, minimize wrt $\mathbf{m}_{ij}$ (defining new cluster prototypes)
   For each$\mathbf{m}_k$, $J = \sum_{i=1}^{n} \sum_{j=1}^{k} r_{ij} \left|\left|x_i - \mathbf{m}_j\right|\right|^2$ is a quadratic function of $\mathbf{m}_k$. By setting its derivative to zero, the values of $\mathbf{m}_k$ providing its minimum are obatined
   $$\frac{\partial J}{\partial \mathbf{m}_k} = 2 \sum_{i=1}^{n} r_{ik}(x_i - \mathbf{m}_k) = 0 \Longrightarrow \mathbf{m}_k = \frac{\sum_{i=1}^{n} r_{ik} x_i}{\sum_{i=1}^{n} r_{ik}}$$
   That is, the new prototype is the mean of the elements assigned to the cluster

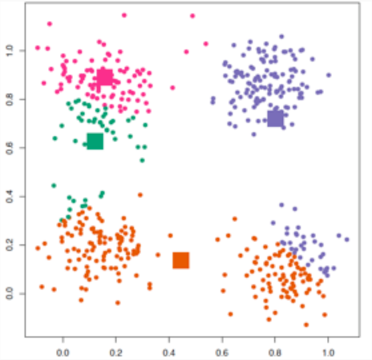
At each step, $J$ does not increase. There is a convergence to a local minimum.

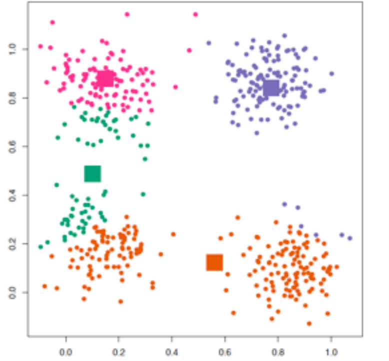
# Example of application of k-means



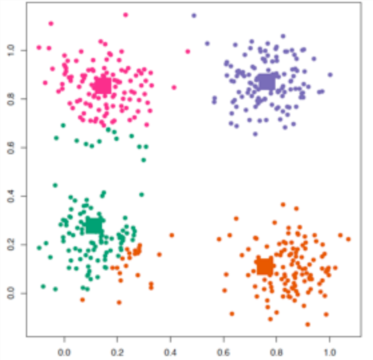



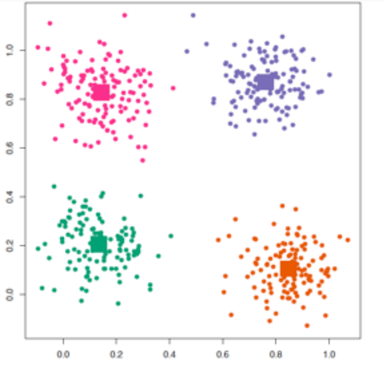
# Example of application of k-means



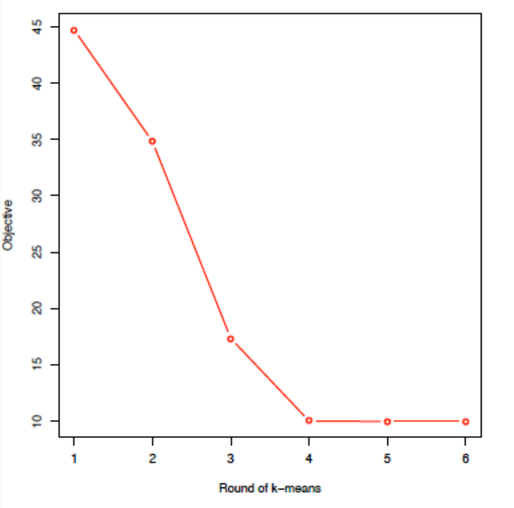



# Example of application of k-means



# Example of application of k-means



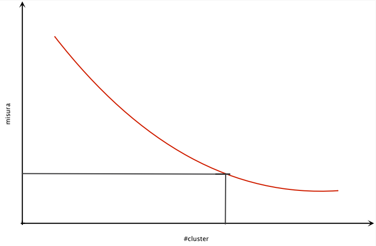
# How to choose $K$

## Cross validation

- Apply cross validation for different values of $K$, measuring the quality of the clustering obtained
- How to measure the quality of a clustering?
    1. mean distance of elements from the prototypes of their clusters
    2. log-likelihood of the elements wrt the resulting mixture model

## Note

Measures improves as $K$ increases (overfitting). A value such that further increases provide limited improvement should be found



# How to choose $K$

**Penalty**

Use of penalty terms wrt number of parameters

- Akaike Information Criterion (AIC)

$$\text{AIC} = 2K - 2\ln L$$

- Bayesian Information Criterion (BIC)

$$\text{BIC} = K\ln n - 2\ln L$$

where $L$ is the model likelihood

# Hierarchical clustering

**Aim**

Derivation of a binary tree. Node: cluster; arc: inclusion.

The tree specifies a set of pairwise merge of clusters.

- Aggregation, starting from $n$ singleton clusters
- Separation, starting from a single cluster of size $n$

**Requirements**

$k$-means requires:

- a number $K$ of clusters
- an initizl assignment
- a distance function between elements

Hierarchical clustering requires:

- a similarity function between clusters

## Algorithm

- define $n$ clusters (singleton)
- repeat
  - compute the matrix of distances between clusters
  - merge the pair of clusters which are “nearest”
- until “a single cluster has remained”

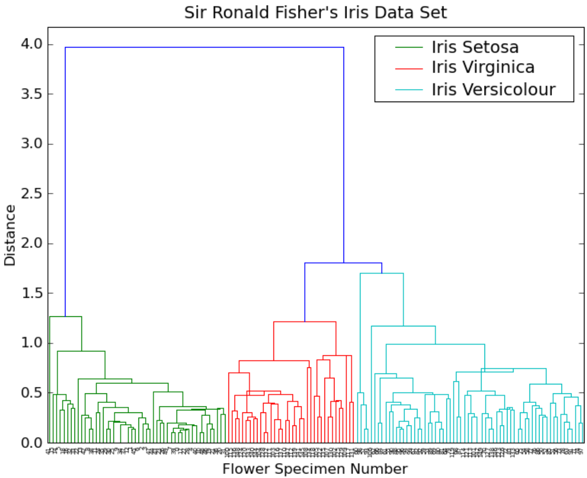
## Properties

- Each tree level is a partition of elements
- The algorithm provides a sequence of clusterings
- The best clustering has to be found
- Monotonicity: similarity between paired clusters decreases

## Dendrogram

- Tree of cluster pairings
- The height of the nodes is inversely proportional to the similarity of the paired clusters

# Dendrogramma



# Cluster similarity

Many measures. Most frequent ones:

- Similarity between nearest nodes (Single linkage)

$$d_{SL}(C_1, C_2) = \min_{\mathbf{x}_1 \in C_1, \mathbf{x}_2 \in C_2} d(x_i, x_j)$$

- Similarity between farthest nodes (Complete linkage)

$$d_{CL}(C_1, C_2) = \max_{\mathbf{x}_1 \in C_1, \mathbf{x}_2 \in C_2} d(x_i, x_j)$$

- Mean similarity (Group average)

$$d_{GA}(C_1, C_2) = \frac{1}{|C_1| \cdot |C_2|} \sum_{\mathbf{x}_1 \in C_1} \sum_{\mathbf{x}_2 \in C_2} d(x_i, x_j)$$

Different measures provide different dendrograms

# Dendrogram with complete linkage