MSc Data Science for Business

Introduction to Machine Learning

Map 534

Julie Josse

Hierarchical Clustering - K-Means

J. Josse

Previous lecture: exploration with PCA

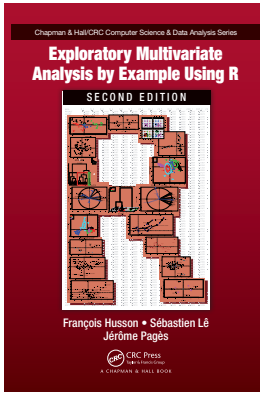
- similarities between observations
- linear relationship between variables
- description of observations with variables
- small number of synthetic variables
- Vizualise/ **Dimension reduction**: represent data in a lower dimension space, keeping information as much as possible

Today: Clustering. Hierarchical Clustering - K-means

- Finding *homogeneous groups*, or clusters, in a data set

Both approaches share similarities: study links between objects - variability of the data. Different representation.
Unsupervised learning setting: no target variable
We aim at uncovering (learning) what is hidden in the data

References



Exploratory Multivariate Analysis by Example using R, Husson, Le, Pages (2017), Chapman & Hall

Youtube: playlist

G. James, D. Witten, T. Hastie, and R. Tibshirani (2013) An Introduction to Statistical Learning with Applications in R Springer Series in Statistics.

Motivations for clustering

- **Marketing:** find groups of customers with similar behavior who might be more receptive to a particular form of advertising, or more likely to purchase a particular product.

  Ex: L'OREAL: 100 000 women different countries - 300 questions
  • questionnaire: life style, skin and hair characteristics, care and consumer habits
  • Clinical assessments by a dermatologist: facial skin complexion, wrinkles, scalp dryness, greasiness
  • Hair assessments by a hair dresser: abundance, volume, breakage

Motivations for clustering

- **Marketing:** find groups of customers with similar behavior who might be more receptive to a particular form of advertising, or more likely to purchase a particular product.

  Ex: L'OREAL: 100 000 women different countries - 300 questions
  • questionnaire: life style, skin and hair characteristics, care and consumer habits
  • Clinical assessments by a dermatologist: facial skin complexion, wrinkles, scalp dryness, greasiness
  • Hair assessments by a hair dresser: abundance, volume, breakage

- **Biology:** tissue samples for patients with breast cancer described by gene expression measurements. There may be different unknown types of cancer which we could discover

- **Internet:** document classification, clustering weblog data to discover groups of similar access patterns

Clustering

Data $D_I = \{x_{1.}, \ldots, x_{I.}\}$ with $x_i \in \mathbb{R}^K$

Goal : retrieve groups = clusters = classes of individuals
2 individuals within a group must be as similar as possible
2 individuals of different groups must be as different as possible
Construct a map $f : \mathbb{R}^K \to \{1, \ldots, Q\}$ which affects a cluster number to each $x_{i.}$: $f : x_{i.} \mapsto q_i$
No ground truth for q_i... **Warning:** Choice of Q is hard

Clustering

Data $D_I = \{x_{1.}, \ldots, x_{I.}\}$ with $x_i \in \mathbb{R}^K$

Goal : retrieve groups = clusters = classes of individuals
2 individuals within a group must be as similar as possible
2 individuals of different groups must be as different as possible
Construct a map $f : \mathbb{R}^K \to \{1, \ldots, Q\}$ which affects a cluster number to each $x_{i.}$: $f : x_{i.} \mapsto q_i$
No ground truth for q_i... **Warning:** Choice of Q is hard

Strengths: can be applied to objects with various features. Here: objects are statistical obs described by variables.

Clustering

Data $D_I = \{x_{1.}, \ldots, x_{I.}\}$ with $x_i \in \mathbb{R}^K$

Goal : retrieve groups = clusters = classes of individuals
2 individuals within a group must be as similar as possible
2 individuals of different groups must be as different as possible
Construct a map $f : \mathbb{R}^K \to \{1, \ldots, Q\}$ which affects a cluster number to each $x_{i.}$: $f : x_{i.} \mapsto q_i$
No ground truth for q_i... **Warning:** Choice of Q is hard

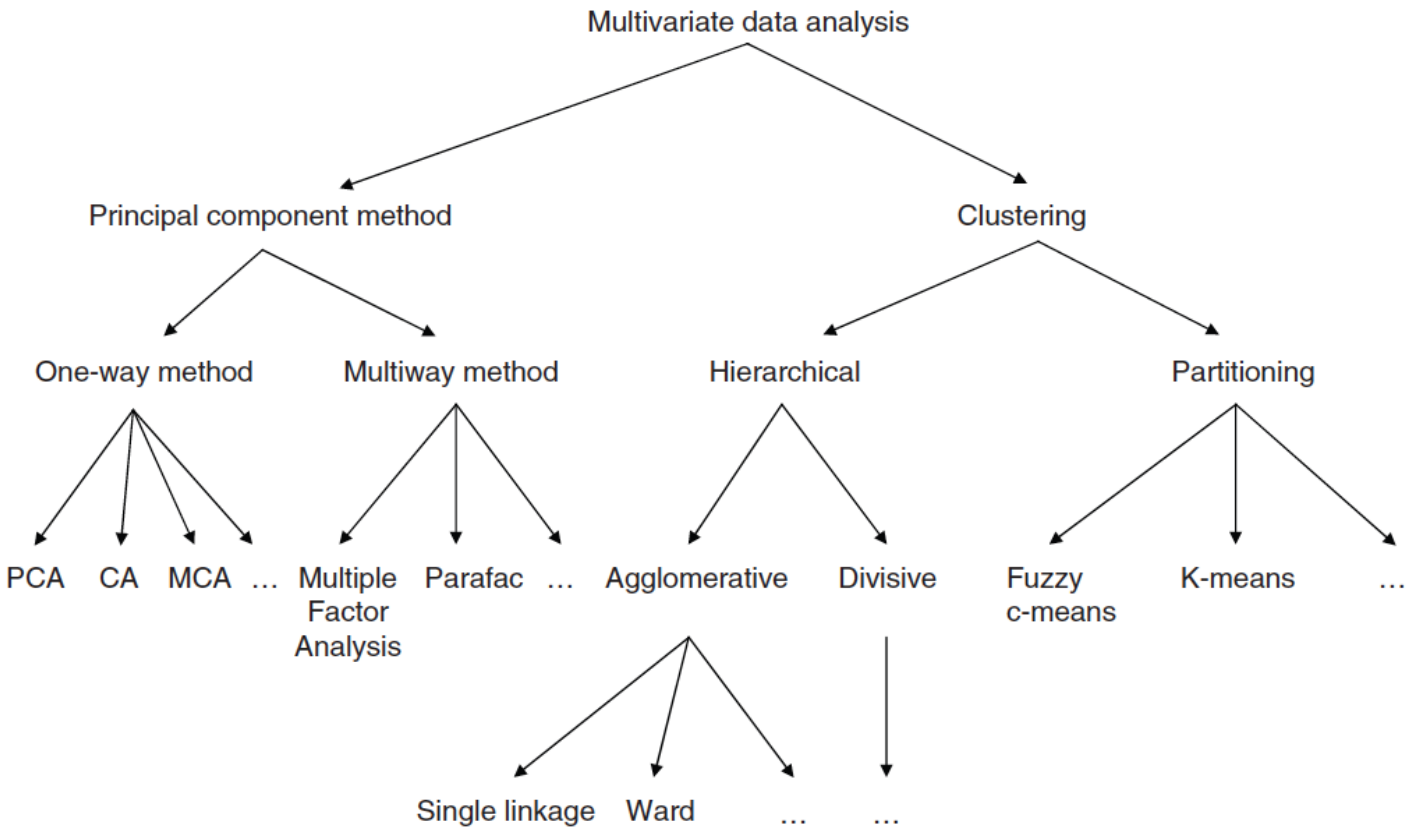
Strengths: can be applied to objects with various features. Here: objects are statistical obs described by variables.

There are many different types of clustering methods. We will concentrate on two of the most commonly used approaches

- Hierarchical Clustering - Partition-based
- Model-based (Mixture models - EM algo) (Stats in action)

Outline

Hierarchical example: unsupervised methods



Outline

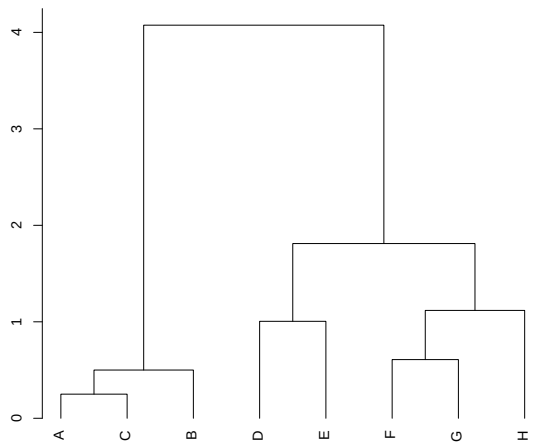
What data? What goals?

Data tables: rows of individuals, columns of quantitative variables

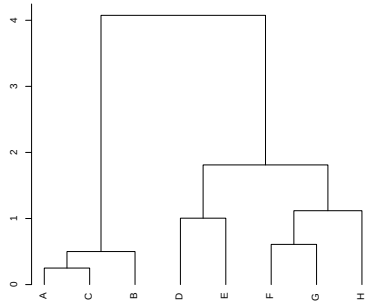
	Janv	Févr	Mars	Avri	Mai	Juin	juil	Août	Sept	Octo	Nove	Déce	Lati	Long
Bordeaux	5.6	6.6	10.3	12.8	15.8	19.3	20.9	21	18.6	13.8	9.1	6.2	44.5	-0.34
Brest	6.1	5.8	7.8	9.2	11.6	14.4	15.6	16	14.7	12	9	7	48.24	-4.29
Clermont	2.6	3.7	7.5	10.3	13.8	17.3	19.4	19.1	16.2	11.2	6.6	3.6	45.47	3.05
Grenoble	1.5	3.2	7.7	10.6	14.5	17.8	20.1	19.5	16.7	11.4	6.5	2.3	45.1	5.43
Lille	2.4	2.9	6	8.9	12.4	15.3	17.1	17.1	14.7	10.4	6.1	3.5	50.38	3.04
Lyon	2.1	3.3	7.7	10.9	14.9	18.5	20.7	20.1	16.9	11.4	6.7	3.1	45.45	4.51
Marseille	5.5	6.6	10	13	16.8	20.8	23.3	22.8	19.9	15	10.2	6.9	43.18	5.24
Montpellier	5.6	6.7	9.9	12.8	16.2	20.1	22.7	22.3	19.3	14.6	10	6.5	43.36	3.53
Nantes	5	5.3	8.4	10.8	13.9	17.2	18.8	18.6	16.4	12.2	8.2	5.5	47.13	-1.33
Nice	7.5	8.5	10.8	13.3	16.7	20.1	22.7	22.5	20.3	16	11.5	8.2	43.42	7.15
Paris	3.4	4.1	7.6	10.7	14.3	17.5	19.1	18.7	16	11.4	7.1	4.3	48.52	2.2
Rennes	4.8	5.3	7.9	10.1	13.1	16.2	17.9	17.8	15.7	11.6	7.8	5.4	48.05	-1.41
Strasbourg	0.4	1.5	5.6	9.8	14	17.2	19	18.3	15.1	9.5	4.9	1.3	48.35	7.45
Toulouse	4.7	5.6	9.2	11.6	14.9	18.7	20.9	20.9	18.3	13.3	8.6	5.5	43.36	1.26
Vichy	2.4	3.4	7.1	9.9	13.6	17.1	19.3	18.8	16	11	6.6	3.4	46.08	3.26

Goals: build a tree structure called a Dendogram that

- shows hierarchical links between individuals or groups of individuals
- detects a “natural” number of classes in the population



Indexed hierarchy



Simple interpretation: Each leaf represents one of the observations. As we move up the tree, some leaves fuse into branches. These correspond to observations that are similar to each other. As we move higher up the tree, branches themselves fuse, either with leaves or other branches.

Height of fusing/merging (on vertical axis) indicates how similar the points are: 2 objects are similar if, to move from one to the other, it is not necessary to go too far back in the tree. (A & C) more similar than (F & G). H more similar to (D & E) than to (A & B).

Rk: the positions of the two fused branches could be swapped without affecting the meaning of the tree. (See exercice.)

Hierarchical clustering: principles

Starting from the leaves and combining clusters up to the trunk

- Start with I initial clusters, each one being one observation. Agglomerate closest observations: smallest $d(i, i')$
- Sequentially merge the two closest clusters into one new cluster.
- Stop when you have only one cluster.

Rk: AHC first groups most similar objects and then groups of object. Divisive clustering

Hierarchical clustering: principles

Starting from the leaves and combining clusters up to the trunk

- Start with I initial clusters, each one being one observation. Agglomerate closest observations: smallest $d(i, i')$
- Sequentially merge the two closest clusters into one new cluster.
- Stop when you have only one cluster.

$\rightarrow$ How do we know which clusters are the closest from each other?

We need to

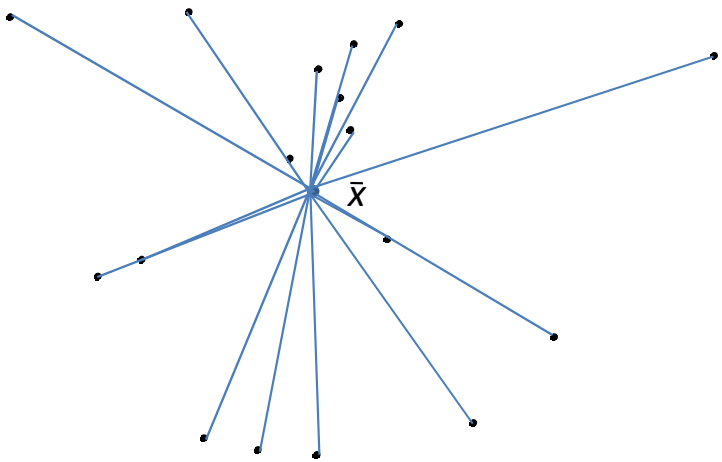
- measure similarity of individuals (Euclidian to start, other?) representations, which are also Euclidean
- measure similarity between groups of individuals (for instance distance between $\{5, 8\}$ and 3?): linkage

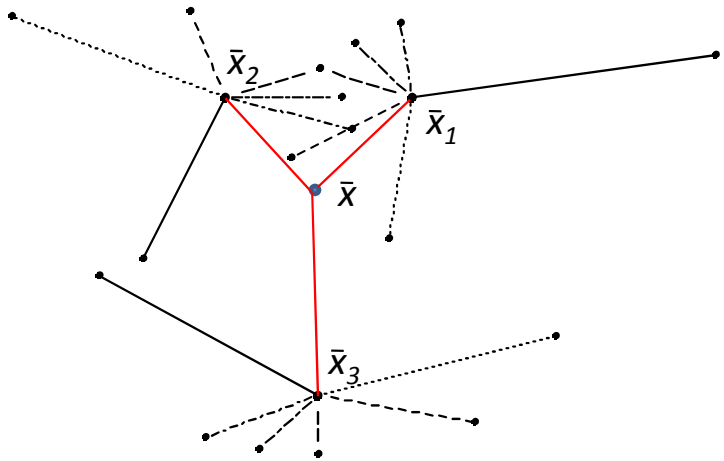
Rk: AHC first groups most similar objects and then groups of object. Divisive clustering

Ward's method: Euclidian distances + Ward - Based on Inertia

Huygens theorem: $\bar{x}_k$ the mean of variable k $\bar{x}_k = \frac{1}{I}\sum_{i=1}^{I} x_{ik}$,
$\bar{x}_{qk}$ the mean of variable k for the observations in cluster q
$\bar{x}_{qk} = \frac{1}{|C_q|}\sum_{i\in C_q} x_{ik} = \frac{1}{I_q}\sum_{i\in C_q} x_{ik}$

$$\underbrace{\frac{1}{I}\sum_{k=1}^{K}\sum_{q=1}^{Q}\sum_{i\in C_q}(x_{ik}-\bar{x}_k)^2}_{\text{total inertia}} = \underbrace{\frac{1}{I}\sum_{k=1}^{K}\sum_{q=1}^{Q}\sum_{i\in C_q}(x_{ik}-\bar{x}_{qk})^2}_{\text{within-class inertia}} + \underbrace{\frac{1}{I}\sum_{k=1}^{K}\sum_{q=1}^{Q}\sum_{i\in C_q}(\bar{x}_{qk}-\bar{x}_k)^2}_{\text{between-class inertia}}$$





Rk: Total inertia equal the sum of the variance of the variables
Hope: find a partition that minimizes W (maximizes B, T)

Ward's method

Init: 1 class = 1 obs $\Longrightarrow$ Between-class inertia = Total inertia

Euclidean distance between individuals

Distance between cluster $d(C_1, C_2) = \frac{I_{C_1} I_{C_2}}{I_{C_1}+I_{C_2}} d(m_{C_1}, m_{C_2})^2$.

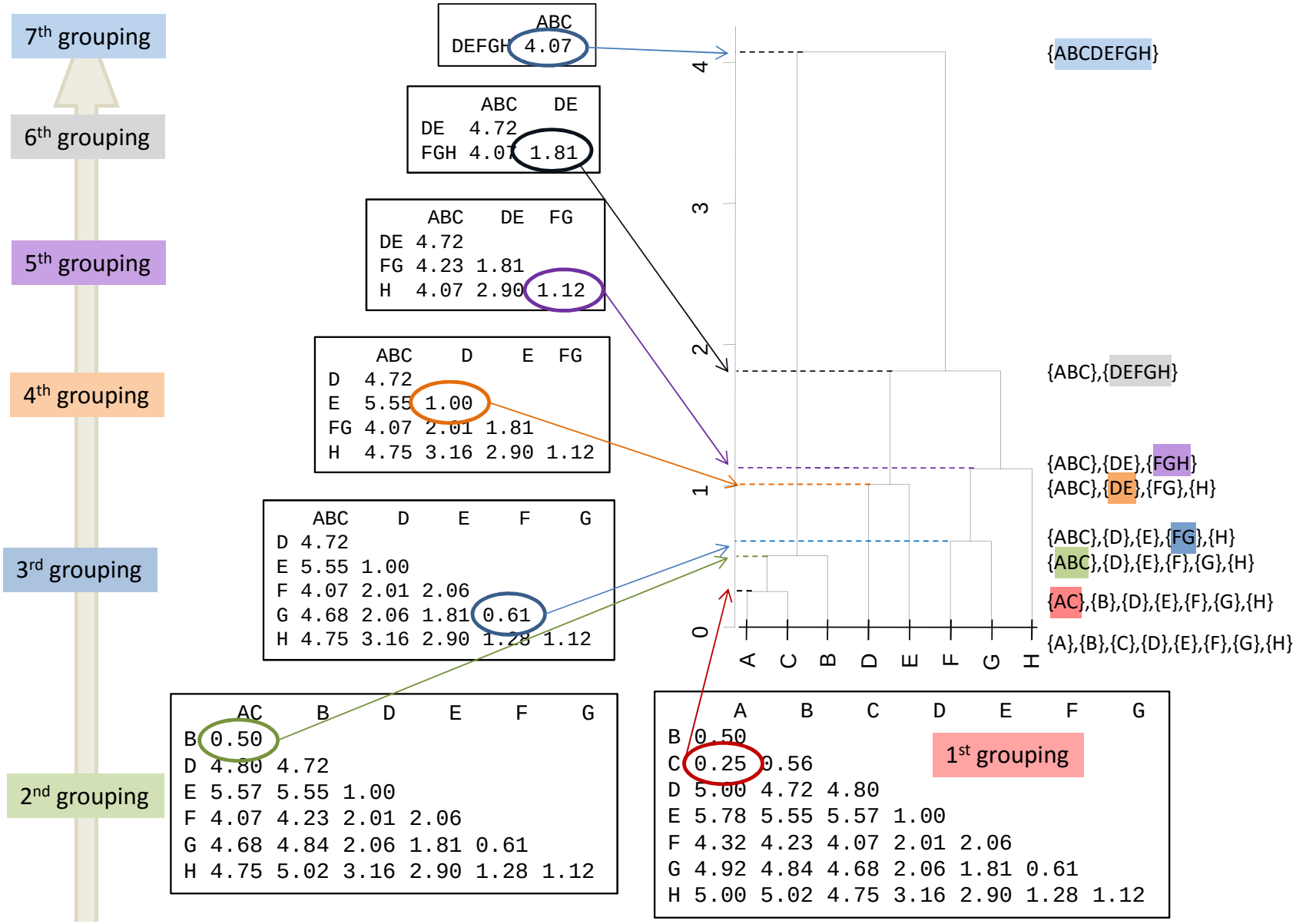
$$
\begin{aligned}
d(C_1, C_2) = & \sum_{i \in C_1 \cup C_2} \|x_{i.} - m_{C_1 \cup C_2}\|^2 - \sum_{i \in C_1} \|x_{i.} - m_{C_1}\|^2 \\
& - \sum_{i \in C_2} \|x_{i.} - m_{C_2}\|^2 \\
= & \frac{I_{C_1} I_{C_2}}{I_{C_1} + I_{C_2}} \|m_{C_1} - m_{C_2}\|^2.
\end{aligned}
$$

$W(C_1 \cup C_2) = W(C_1) + W(C_2) + d(C_1, C_2)$ (Proof exercice) how much the sum of squares increases when merging two clusters.

The merge is (locally) the merge minimising W.

Group objects with small weights (ex:1-1/50-50) and similar centers of gravity.

Algorithm - Exercice



Linkage criterion between sets C_q and C'_q

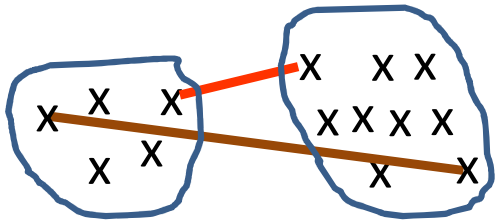
Similarity between groups of individuals:

- Single linkage/Nearest neighbor (smallest distance):

$$d(C_q, C'_q) = \min\{d(x,z), x \in C_q, z \in C'_q\}$$

- Complete linkage/Furthest neighbor (largest distance):

$$d(C_q C'_q) = \max\{d(x,z), x \in C_q, z \in C'_q\}$$



- Average linkage (average distance):

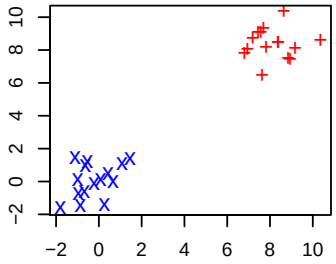
$$d(C_q, C'_q) = \frac{1}{|C_q||C'_q|} \sum_{x \in C_q, z \in C'_q} d(x,z)$$

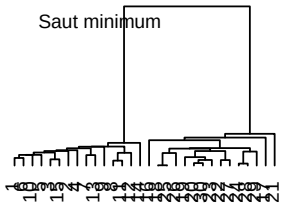
Complete linkage preferred over single linkage, as it tends to yield more balanced dendrograms

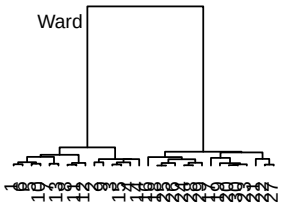
No principal approach to choose the linkage criterion from the data

Impact of the different approaches

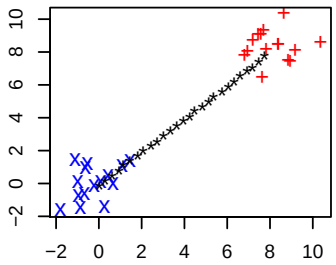
Single linkage: chain effects

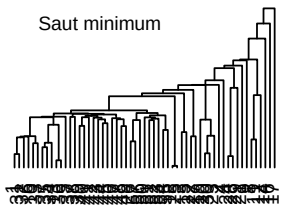


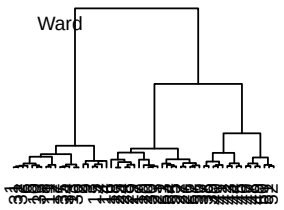




Group together classes with similar centers of gravity







Direct use for clustering

$\Rightarrow$ Small decisions with big consequences? (Scaling before)

$\Rightarrow$ Depends on the context (Read ex Element of Stat book)

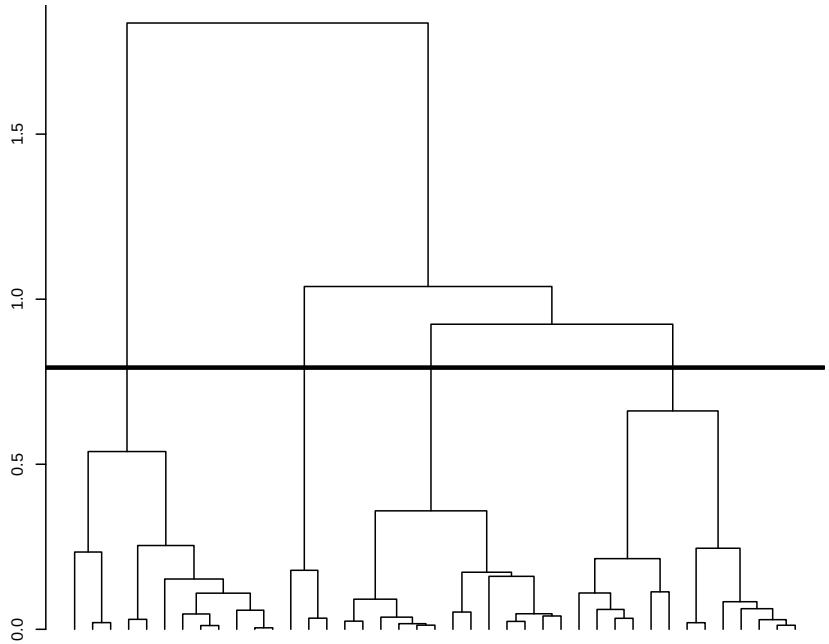
Try several different choices, and look for the one with the most useful or interpretable solution.

$\Rightarrow$ There is no single right answer

Trees and partitions

Trees always end up . . . cut through!

Choosing a height to cut at gives a partition
Different cut, different partitions



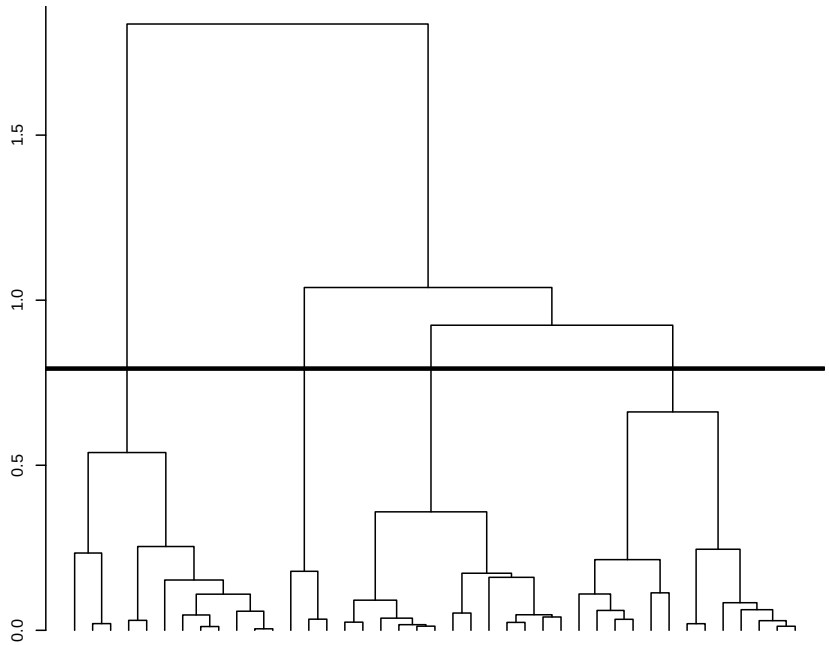
Hierarchical tree: sequence of **nested partitions** from most precise (each obs is a class) to most general (only 1 class)

Advantage: don't know in advance how many clusters - view at once the clusterings obtained from 1 to I clusters.

Trees and partitions

Trees always end up . . . cut through!

Choosing a height to cut at gives a partition
Different cut, different partitions



Hierarchical tree: sequence of **nested partitions** from most precise (each obs is a class) to most general (only 1 class)

Advantage: don't know in advance how many clusters - view at once the clusterings obtained from 1 to *I* clusters.
Rk: the partition is interesting but not optimal

Partition quality

When is a partition a good one?

Partition quality

When is a partition a good one?

- If individuals placed in the same class are close to each other
- If individuals in different classes are far from each other

Partition quality

When is a partition a good one?

- If individuals placed in the same class are close to each other
- If individuals in different classes are far from each other

Mathematically speaking?

- small within-class variability
- large between-class variability

Partition quality

When is a partition a good one?

- If individuals placed in the same class are close to each other
- If individuals in different classes are far from each other

Mathematically speaking?

- small within-class variability
- large between-class variability

$$\underbrace{\frac{1}{I}\sum_{k=1}^{K}\sum_{q=1}^{Q}\sum_{i\in C_q}(x_{ik}-\bar{x}_k)^2}_{\text{total inertia}} = \underbrace{\frac{1}{I}\sum_{k=1}^{K}\sum_{q=1}^{Q}\sum_{i\in C_q}(x_{iqk}-\bar{x}_{qk})^2}_{\text{within-class inertia}} + \underbrace{\frac{1}{I}\sum_{k=1}^{K}\sum_{q=1}^{Q}\sum_{i\in C_q}(\bar{x}_{qk}-\bar{x}_k)^2}_{\text{between-class inertia}}$$

Partition quality

Partition quality is measured by:

$$0 \leq \frac{\text{between-class inertia}}{\text{total inertia}} \leq 1$$

Partition quality

Partition quality is measured by:

$$0 \leq \frac{\text{between-class inertia}}{\text{total inertia}} \leq 1$$

$$\frac{\text{between inertia}}{\text{total inertia}} = 0 \Longrightarrow \forall k, \forall q, \bar{x}_{qk} = \bar{x}_k$$

by variable, classes have the same means

Doesn't allow us to classify

Partition quality

Partition quality is measured by:

$$0 \leq \frac{\text{between-class inertia}}{\text{total inertia}} \leq 1$$

$\frac{\text{between inertia}}{\text{total inertia}} = 0 \Longrightarrow \forall k, \forall q, \bar{x}_{qk} = \bar{x}_k$

by variable, classes have the same means

Doesn't allow us to classify

$\frac{\text{between inertia}}{\text{total inertia}} = 1 \Longrightarrow \forall k, \forall q, \forall i, x_{iqk} = \bar{x}_{qk}$

individuals in the same class are identical

Ideal for classifying

Partition quality

Partition quality is measured by:

$$0 \leq \frac{\text{between-class inertia}}{\text{total inertia}} \leq 1$$

$\frac{\text{between inertia}}{\text{total inertia}} = 0 \Longrightarrow \forall k, \forall q, \bar{x}_{qk} = \bar{x}_k$

by variable, classes have the same means

Doesn't allow us to classify

$\frac{\text{between inertia}}{\text{total inertia}} = 1 \Longrightarrow \forall k, \forall q, \forall i, x_{iqk} = \bar{x}_{qk}$

individuals in the same class are identical

Ideal for classifying

It corresponds with 1 variable to the η^2 (or R^2 in analysis of variance) $\rightarrow$ percentage of variability explained by the partition (categorical variable).

Partition quality

Partition quality is measured by:

$$0 \leq \frac{\text{between-class inertia}}{\text{total inertia}} \leq 1$$

$\frac{\text{between inertia}}{\text{total inertia}} = 0 \Longrightarrow \forall k, \forall q, \bar{x}_{qk} = \bar{x}_k$

by variable, classes have the same means

Doesn't allow us to classify

$\frac{\text{between inertia}}{\text{total inertia}} = 1 \Longrightarrow \forall k, \forall q, \forall i, x_{iqk} = \bar{x}_{qk}$

individuals in the same class are identical

Ideal for classifying

It corresponds with 1 variable to the η^2 (or R^2 in analysis of variance) $\rightarrow$ percentage of variability explained by the partition (categorical variable). Warning: don't just accept this criteria at face value: it depends on the number of individuals and clusters

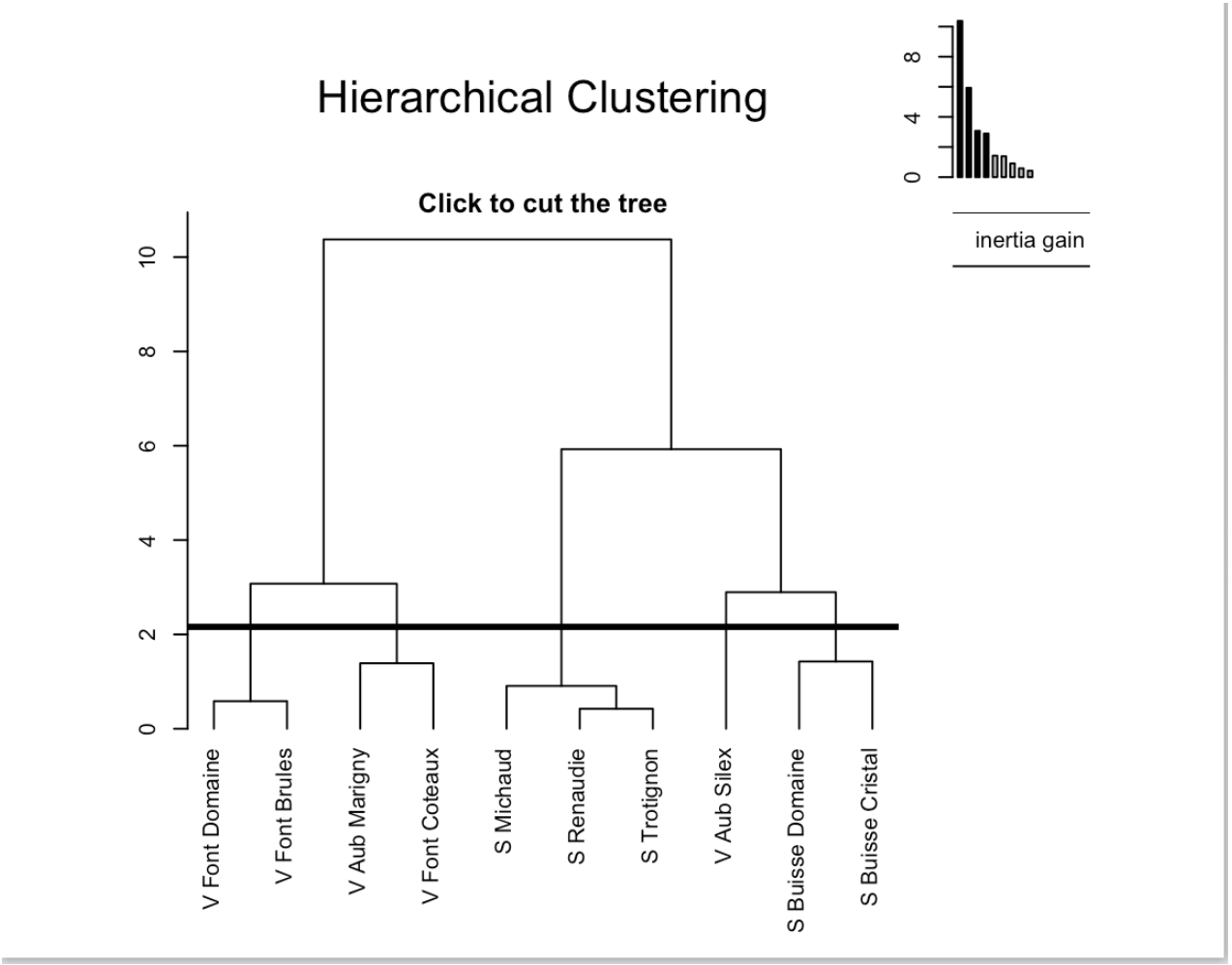
Outline

Wine data

- 10 observations (rows): white wines from Val de Loire
- 27 variables (columns): sensory descriptors

	O.fruity	O.passion	O.citrus	...	Sweetness	Acidity	Bitterness	Astringency	Aroma.intensity	Aroma.persistency	Visual.intensity	Odor.preference	Overall.preference	Label
S Michaud	4,3	2,4	5,7	...	3,5	5,9	4,1	1,4	7,1	6,7	5,0	6,0	5,0	Sauvignon
S Renaudie	4,4	3,1	5,3	...	3,3	6,8	3,8	2,3	7,2	6,6	3,4	5,4	5,5	Sauvignon
S Trotignon	5,1	4,0	5,3	...	3,0	6,1	4,1	2,4	6,1	6,1	3,0	5,0	5,5	Sauvignon
S Buisse Domaine	4,3	2,4	3,6	...	3,9	5,6	2,5	3,0	4,9	5,1	4,1	5,3	4,6	Sauvignon
S Buisse Cristal	5,6	3,1	3,5	...	3,4	6,6	5,0	3,1	6,1	5,1	3,6	6,1	5,0	Sauvignon
V Aub Silex	3,9	0,7	3,3	...	7,9	4,4	3,0	2,4	5,9	5,6	4,0	5,0	5,5	Vouvray
V Aub Marigny	2,1	0,7	1,0	...	3,5	6,4	5,0	4,0	6,3	6,7	6,0	5,1	4,1	Vouvray
V Font Domaine	5,1	0,5	2,5	...	3,0	5,7	4,0	2,5	6,7	6,3	6,4	4,4	5,1	Vouvray
V Font Brûlés	5,1	0,8	3,8	...	3,9	5,4	4,0	3,1	7,0	6,1	7,4	4,4	6,4	Vouvray
V Font Coteaux	4,1	0,9	2,7	...	3,8	5,1	4,3	4,3	7,3	6,6	6,3	6,0	5,7	Vouvray

Hierarchical clustering of wine

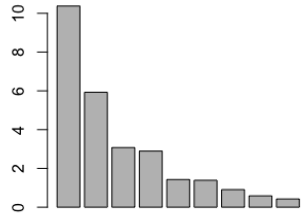


Selecting the number of clusters

- Using the tree, depends of the use - the overall appearance of the tree
- The number of clusters, which must not be too high so as not to impede the concise nature of the approach.
- Interpretability of the clusters
- Inertia barplot

In FactoMineR: suggests a division into Q clusters when the increase of between-clusters inertia when going from a $Q+1$ to a Q clusters partition is much greater than that from a Q to a $Q+1$ clusters partition.

Inertia plot



Loss in between-inertia when going from

- 2 clusters to 1: 10.37
- 3 clusters to 2: 5.92
- 4 clusters to 3: 3.07
- ...

Important loss when going from 4 clusters to 3 clusters thus we prefer to keep 4 clusters

```
 res.hcpc$call$t$inert.gain
[1] 10.3743969  5.9271832  3.0749287  2.8940507  1.4259050
[6]  1.3885472  0.9069387  0.5844212  0.4236284

sum(res.hcpc$call$t$inert.gain)
27
```

Outline

Partition method: K-means

- In practice AHC is used to get partition. Why not looking for a partition directly?

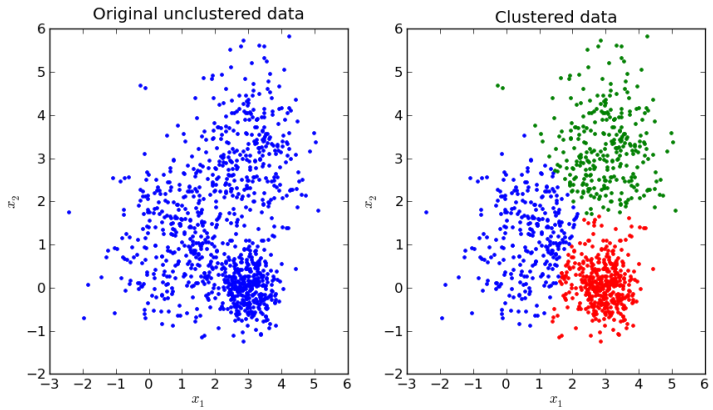
Outline

Partition method: K-Means

Simple and elegant approach for partitioning a data set into Q non-overlapping clusters.

To perform K-means:

- Specify the desired number of clusters Q.
- Similar: small Euclidian distance
- Objective: a good clustering is one for which the within-cluster variation is as small as possible.



Objective

- Fix $Q \geq 2$, I data points $x_{i\cdot} \in \mathbb{R}^K$
- Find partition $C_1, \ldots, C_Q$ such that minimizes the quantification error

$$\sum_{q=1,\ldots,Q} \sum_{i \in C_q} \sum_{k=1}^{K} (x_{ik} - \bar{x}_{qk})^2$$

$\Rightarrow$ We want to partition the observations into Q clusters such that the total within-cluster variation, summed over all Q clusters, is as small as possible.

- Impossible to find the exact solution!

kmeans algorithm

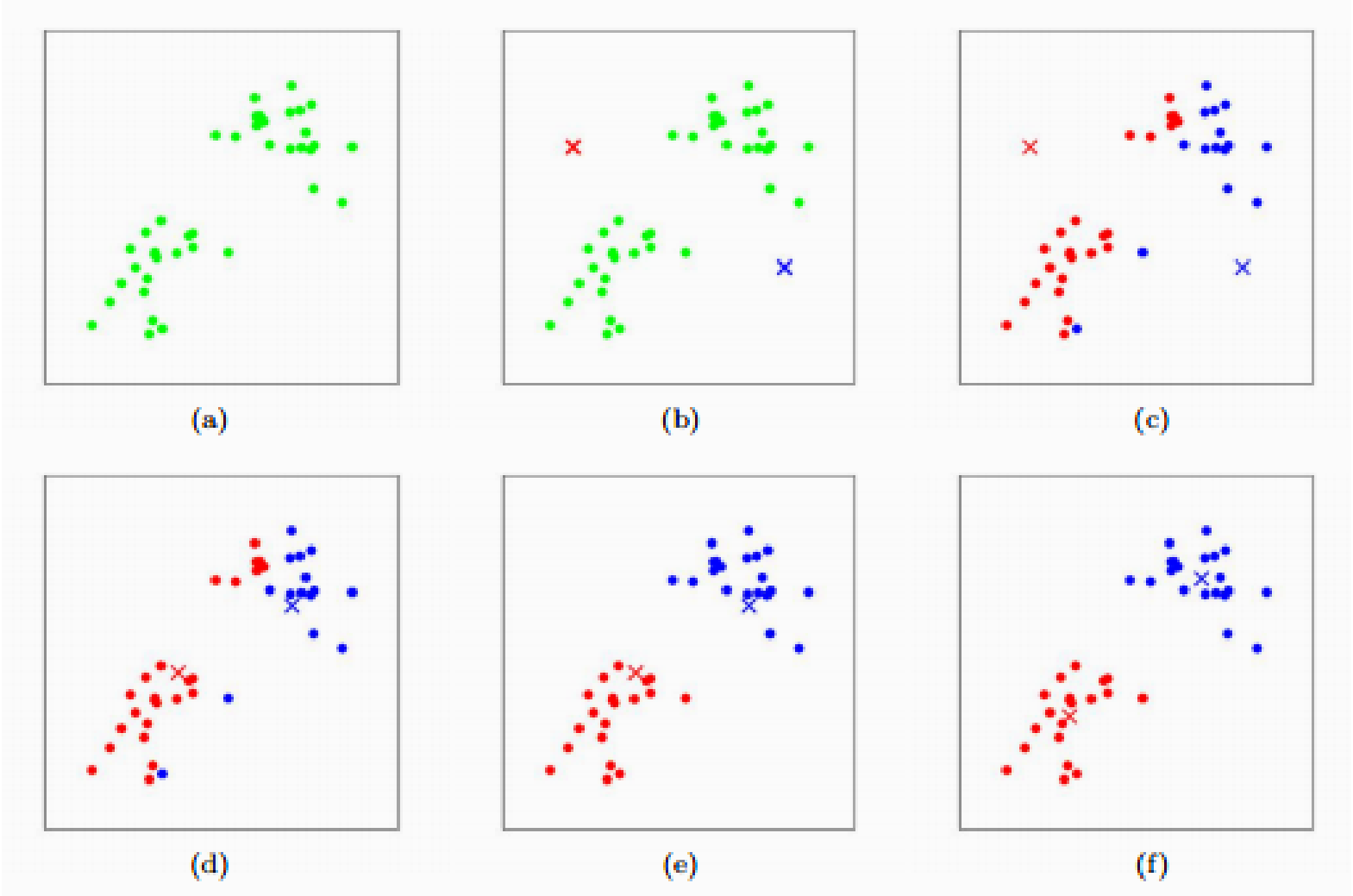
Lloyd (1981) proposes a way of finding local solutions

- Choose at random Q centroids $\{c_1, \ldots, c_Q\}$ K clusters
- For each $q \in \{1, \ldots, Q\}$, find the set C_q of points that are closer to m_q than any $m_{q'}$ for $q' \neq q$
- Update the centroids (with means, hence the name)

$$m_{q\cdot} = \frac{1}{|C_q|} \sum_{i \in C_q} x_{i\cdot}$$

- Repeat the two previous steps until the sets C_q don't change

kmeans: illustration



Outline

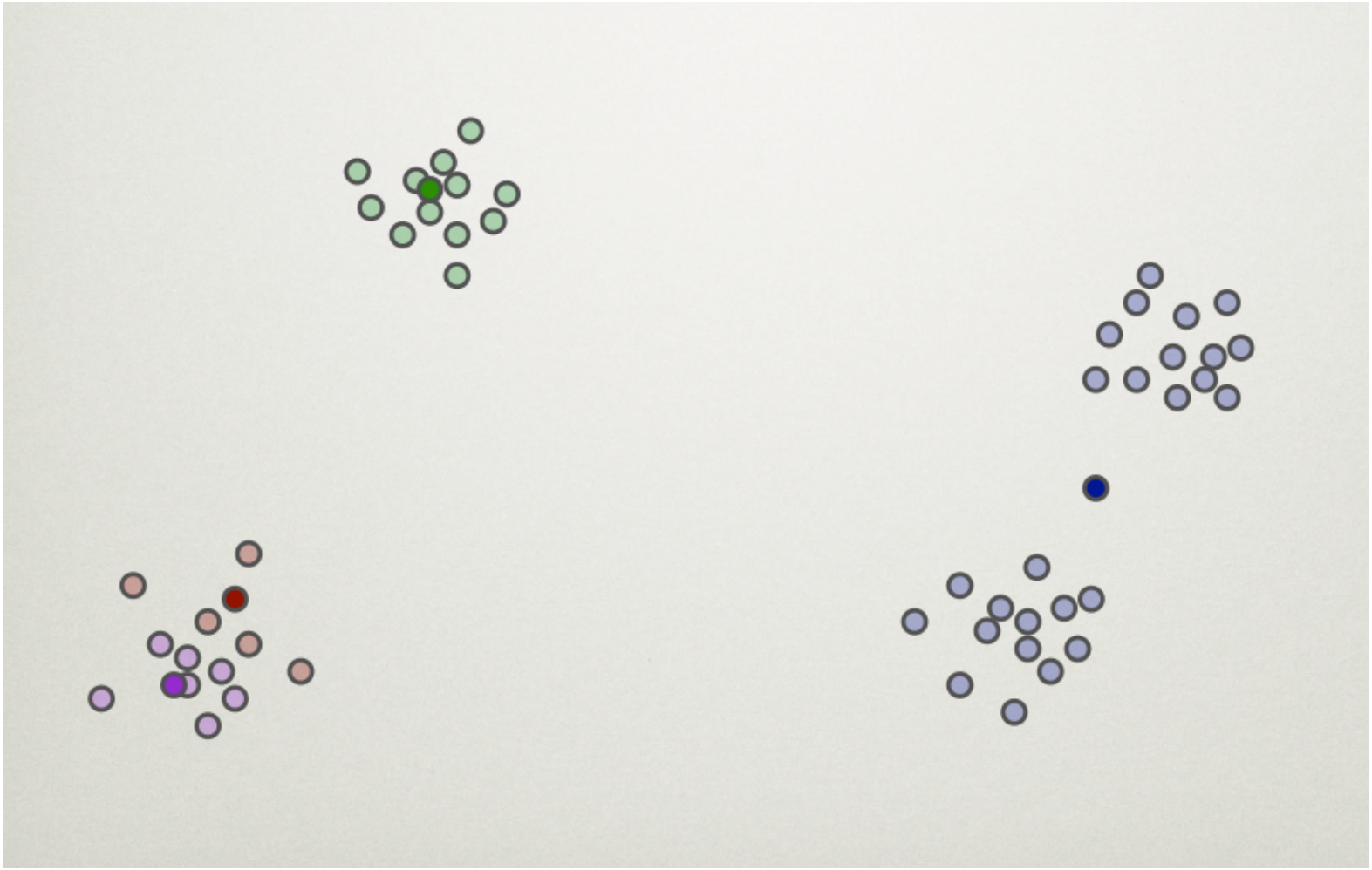
Local minimum

- The objective will never increase (Exercice). When the result no longer changes, a local optimum has been reached
- Local rather than a global optimum: the results depend on the initial (random) cluster assignment of each observation

$\Rightarrow$ Important to run the algorithm multiple times from different random then one selects the best solution, *i.e.* that for which the objective is smallest. (Illustration Lab).

Initial values

K-means very sensitive to the choice of initial points

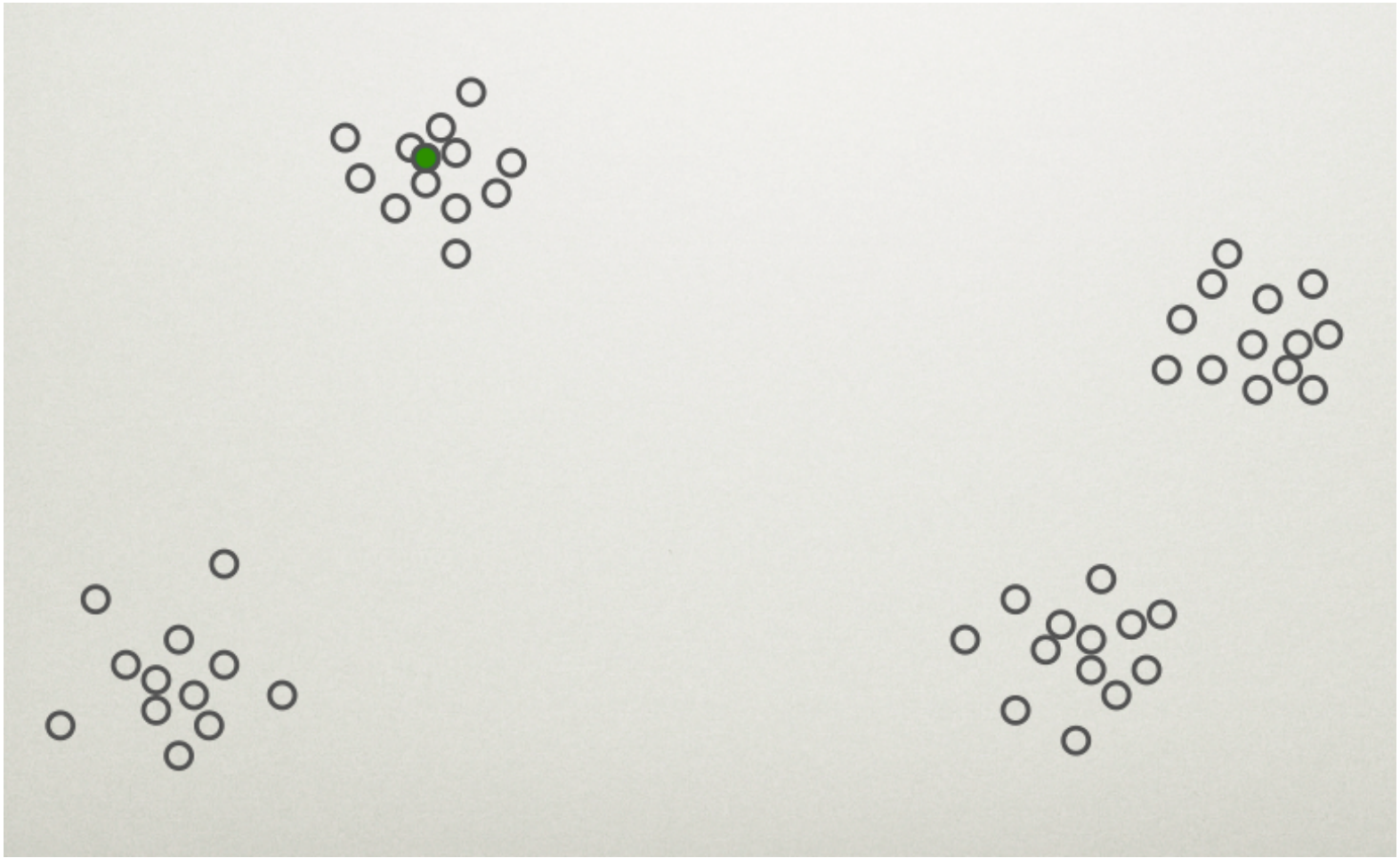


Random Initialization

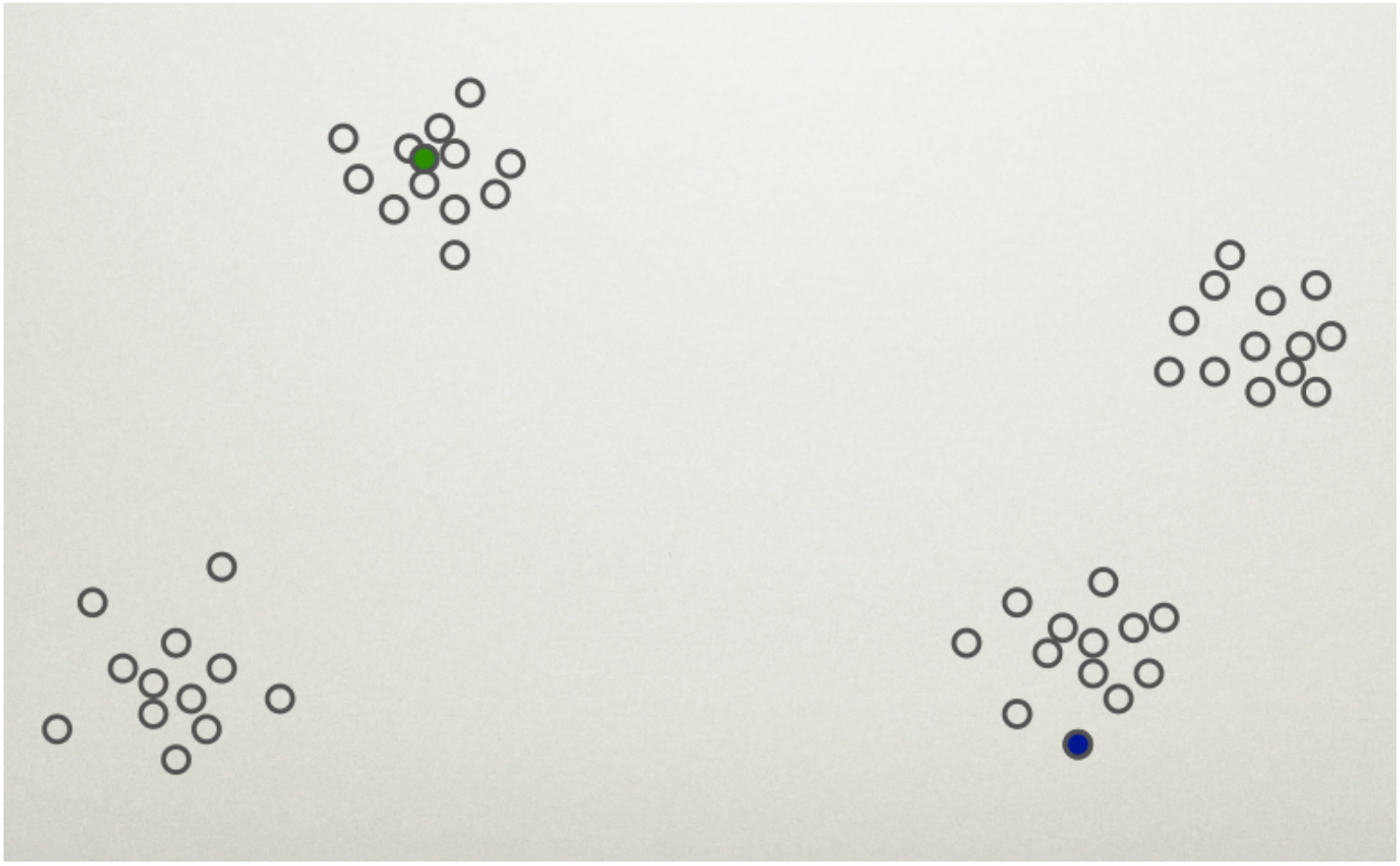
An easy solution:

- Pick a first point at random
- Choose the next initial point the farthest from the previous ones

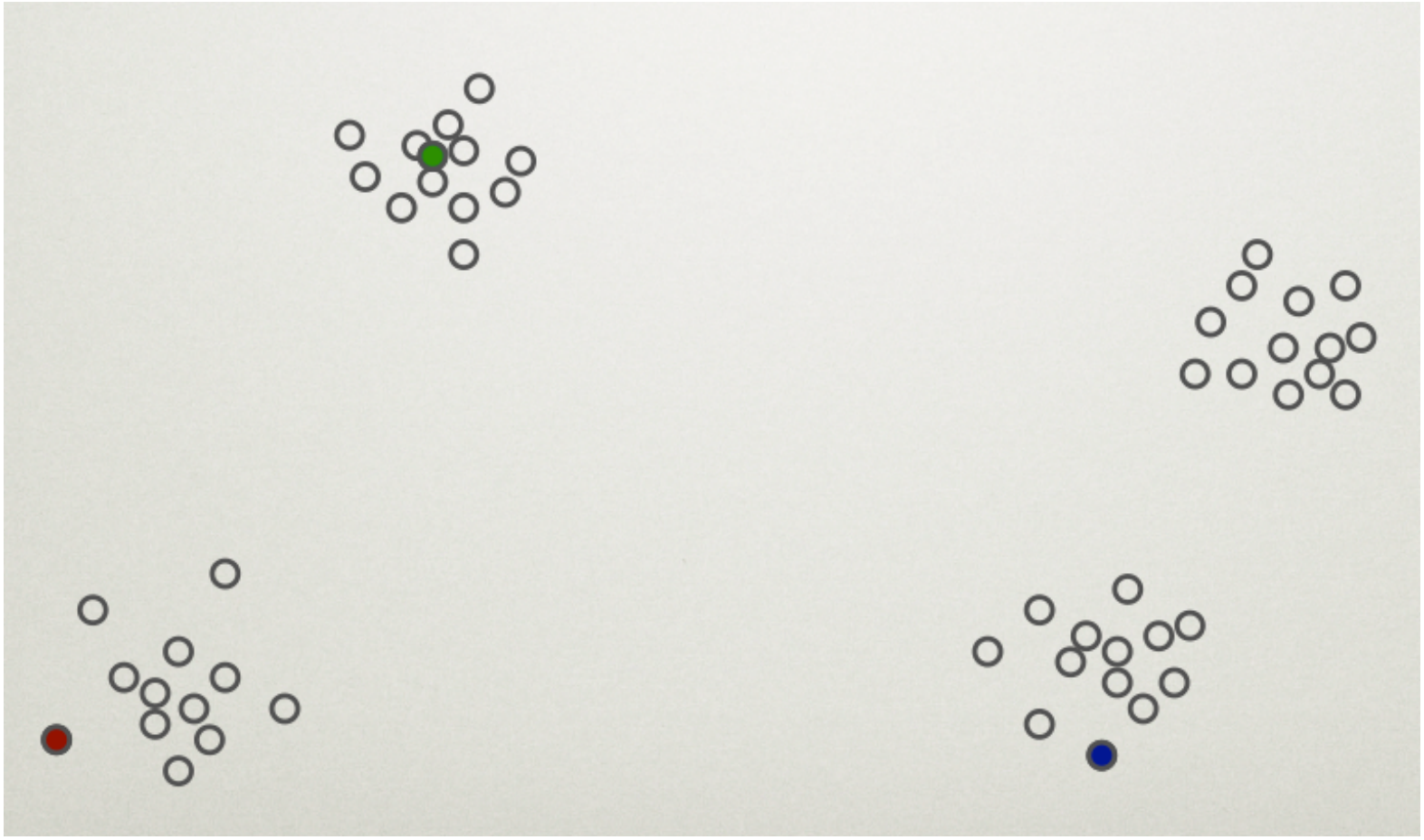
Farthest point initialization



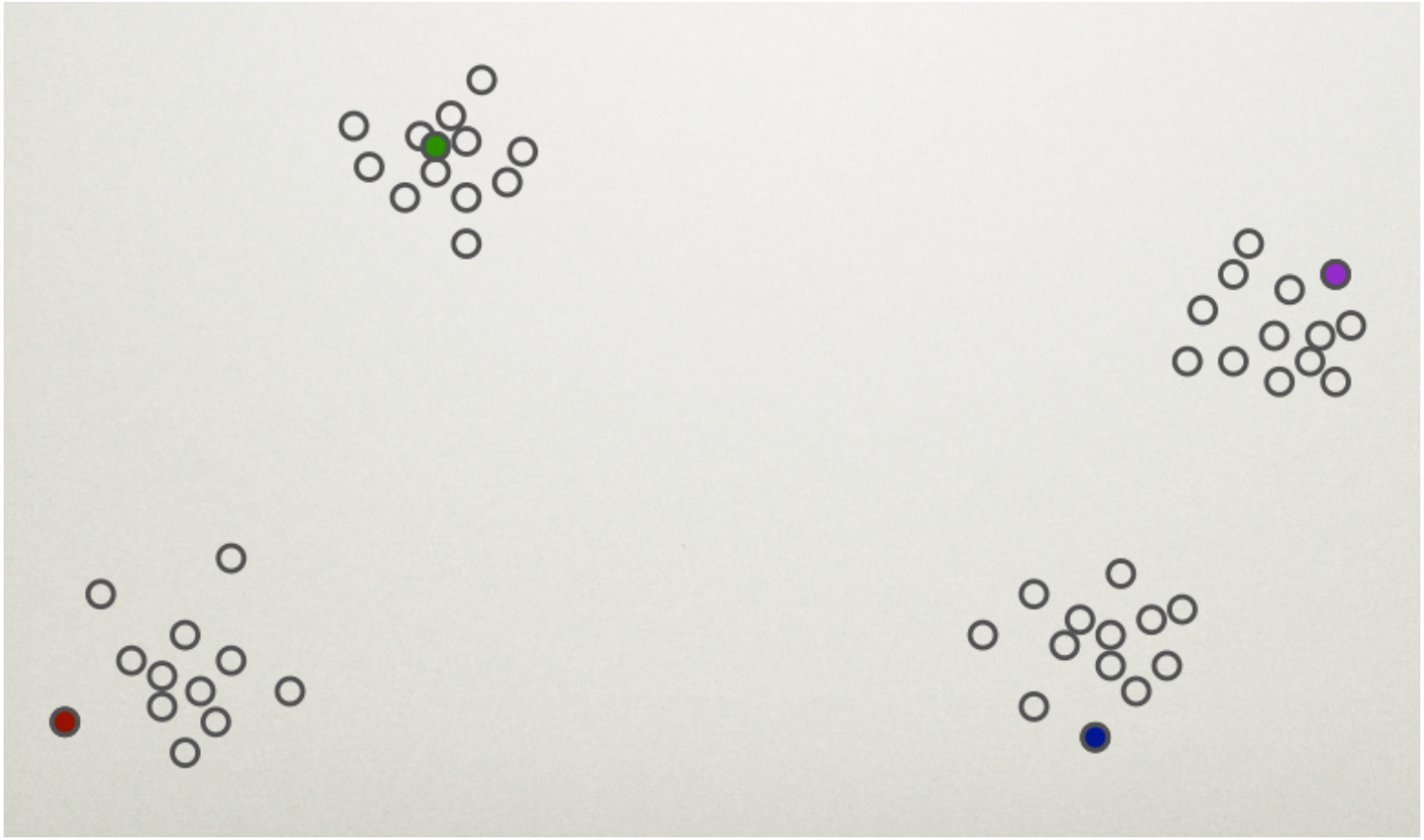
Farthest point initialization



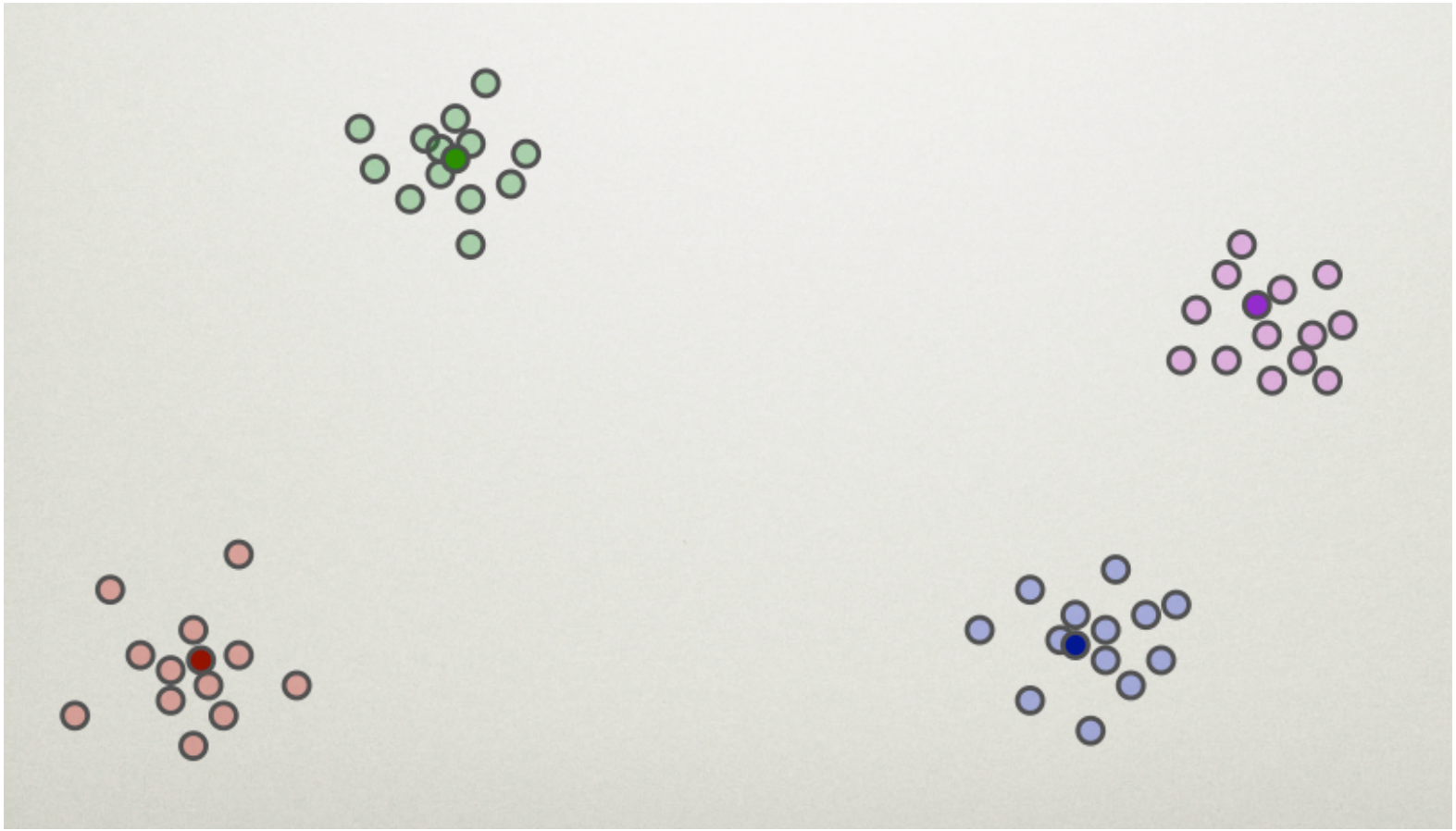
Farthest point initialization



Farthest point initialization

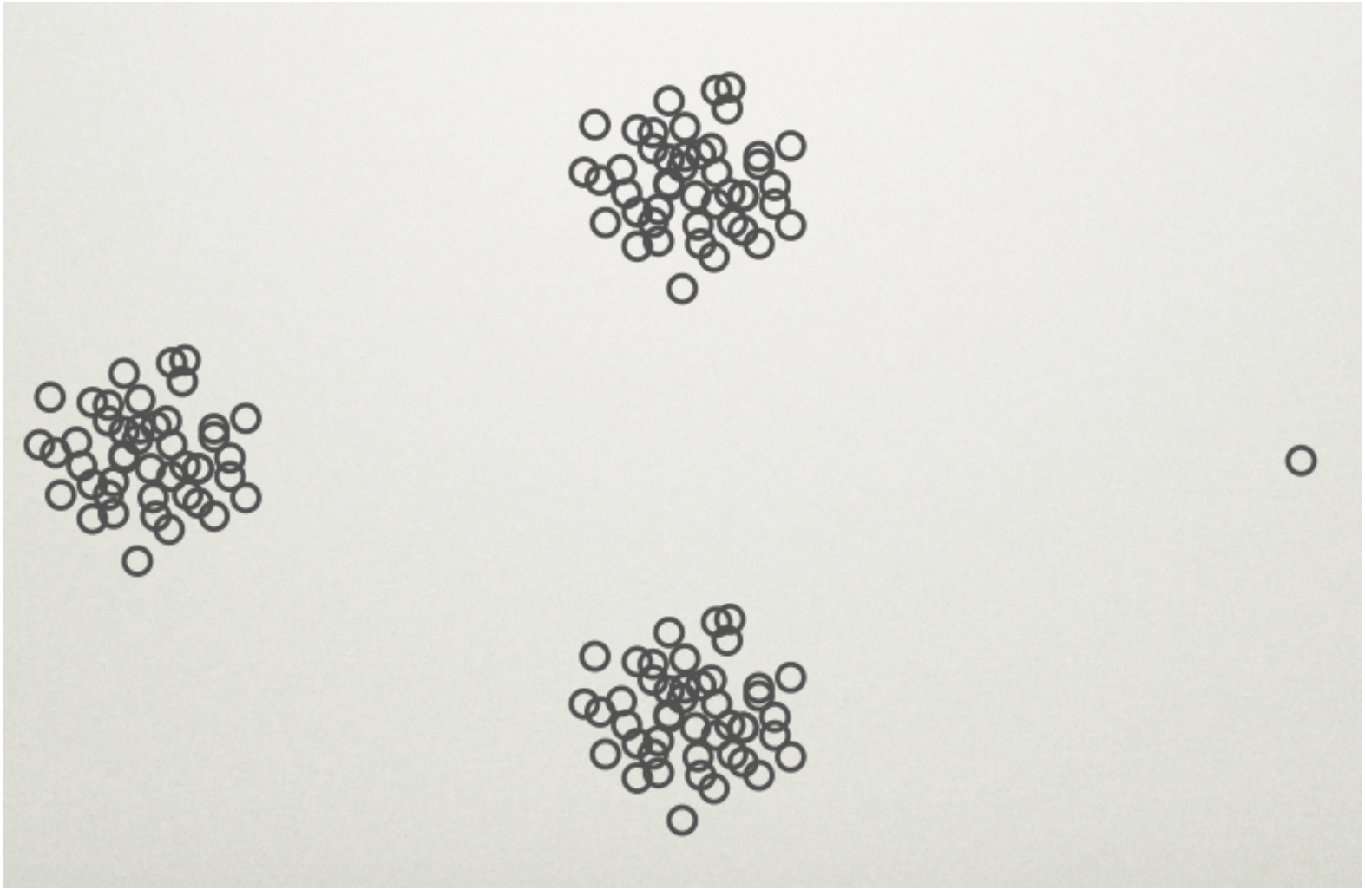


Farthest point initialization



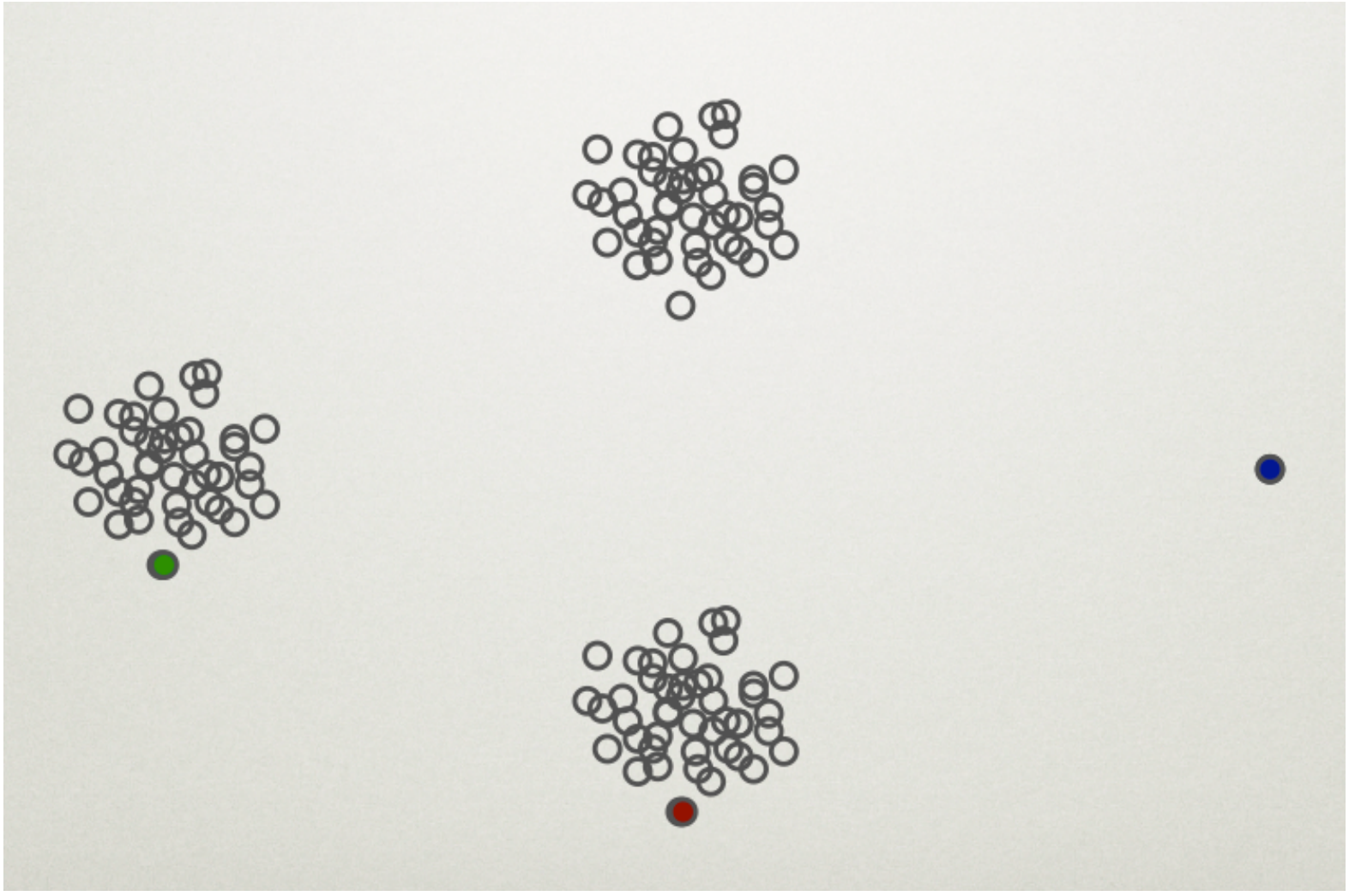
Farthest point initialization

But, very sensitive to outliers



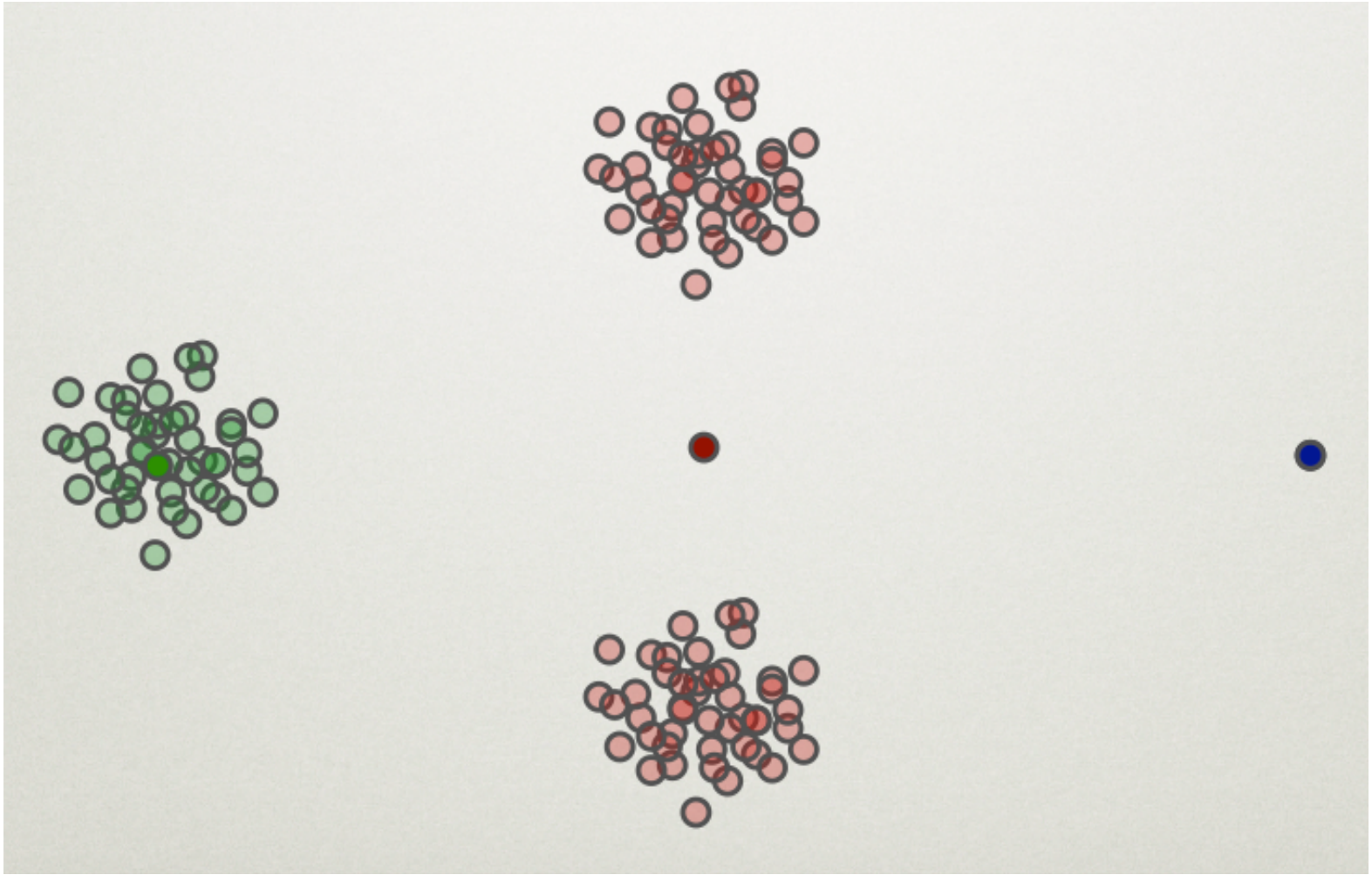
Farthest point initialization

But, very sensitive to outliers



Farthest point initialization

But, very sensitive to outliers



A solution: K-means++

Pick the initial centroids as follows

1. Pick uniformly at random $i \in \{x_{1\cdot}, \ldots, x_{I\cdot}\}$, put $m_1 \leftarrow x_{i\cdot}$
2. $q \leftarrow q + 1$
3. Sample $i \in \{x_{1\cdot}, \ldots, x_{I\cdot}\}$ with probability

$$\frac{\min_{q'=1,\ldots,q-1} \|x_{i\cdot} - m_{q\cdot'}\|_2^2}{\sum_{i'=1}^{n} I \min_{q'=1,\ldots,q-1} \|x_{i\cdot'} - m_{q\cdot'}\|_2^2}$$

4. Put $m_{q\cdot} \leftarrow x_i$
5. If $q < Q$ go back to step 2.

Then use K-means based on these initial clusters

This is between random initialization and furthest point initialization

Conclusion

In order to perform clustering, some decisions must be made:

- Should the features first be standardized?
- In case of hierarchical clustering:
  - What dissimilarity measure should be used?
  - What type of linkage should be used?
  - Where should we cut the dendogram in order to obtain clusters?
- In case of K-means clustering:
  - How many clusters should we look for the data?
- In practice, we try several different choices, and look for the one with the most useful or interpretable solution.

There is no single right answer!

Conclusion - Comparison

- AHC do not require numbers of clusters (shape of the tree)
- Users usually like hierarchical clusters
- Assumption of hierarchical structure might be unrealistic?
- AHC slow. Complexity varies : $O(n3)$ to $O(n^2)$ for single and complete linkage.

- K-means requires the choice of Q
- Fast. Complexity : $O(nQ)$
- Dependent on the initialisation

Be aware...

$\Rightarrow$ Any time clustering is performed on a data set we will find clusters!!

- Represent true groups or clustering noise?
- Is my clustering stable? : performing clustering on subsamples with different choices of the parameters $\rightarrow$ what patterns consistently emerge

$\Rightarrow$ Be careful about how the results of a clustering analysis are reported.

Susan Holmes: Reproducible research workflow in R for the analysis of personalized human microbiome data

NIPS Talk

$\Rightarrow$ These results should not be taken as the absolute truth about a data set, rather, they should constitute a starting point for the developments of a scientific hypothesis and further study.

- Underlying assumption of K-means? (Exercice Lab)
- High dimensionality

Clustering in practice

- Function HCPC in FactoMineR
- Function kmeans in R
- hc.complete =hclust (dist(x), method =”complete ”)

Outline

K-means after hierarchical clustering

The partition obtained by hierarchical clustering is not optimal and can be improved using K-means

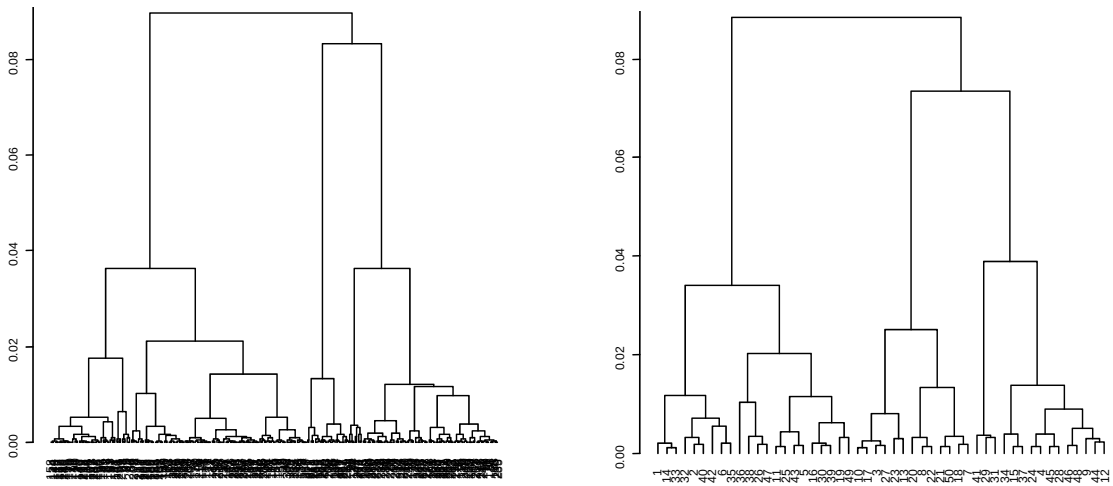
Algorithm:

- use the obtained hierarchical partition to initialize K-means
- run a few iterations of kmeans

$\Longrightarrow$ potentially improved partition
Advantage: more "robust" partition
Disadvantage: loss of hierarchical structure

Hierarchical clustering in high dimension

- If many individuals, hierarchical algorithm is too long
  - Use K-means to partition into around 100 classes
  - Build tree using these classes (weighted by the number of individuals in each class)
  - Idea: gives us the "top" of the tree



Tree from original data Tree using classes

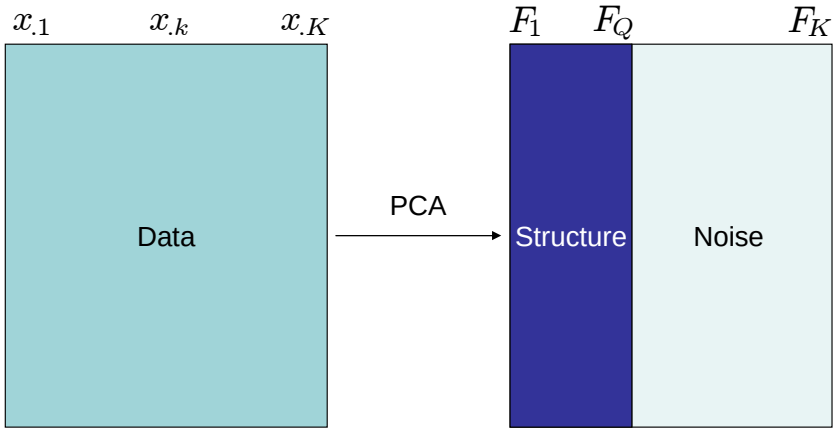
- If many variables: do PCA and keep only first dimensions $\Longrightarrow$ takes us to classical case

PCA before clustering (pre-processing)

$\Rightarrow$ AHC or K-means on the data

$\Rightarrow$ AHC or K-means on the principal components

PCA transforms the raw variables into orthogonal principal components $F_{.1}, ..., F_{.Q}$ with decreasing variance $\lambda_1 \geq \lambda_2 \geq ...\lambda_Q$

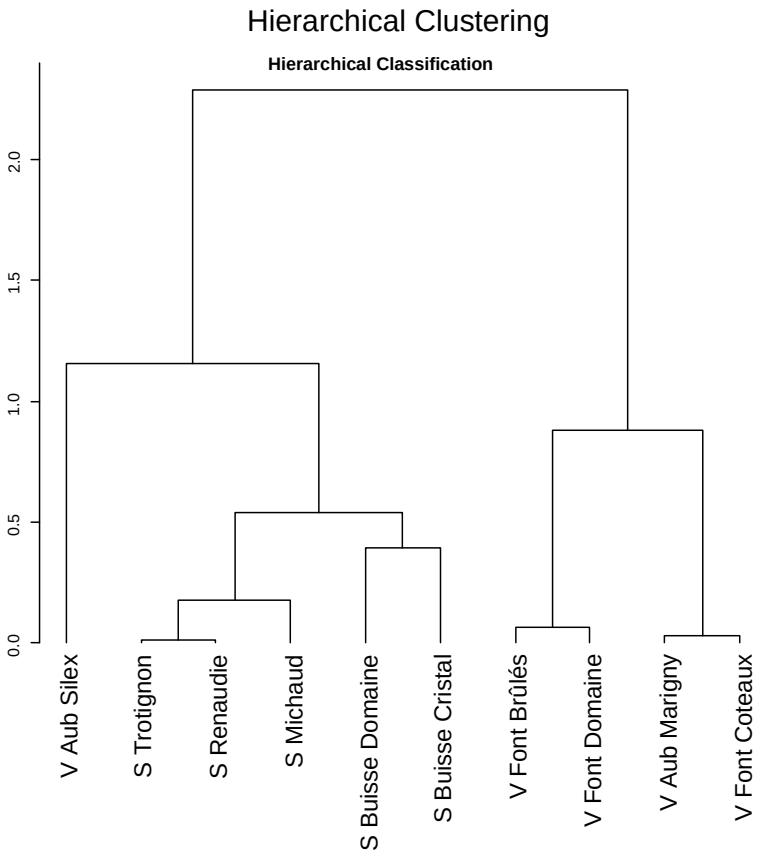


$\Rightarrow$ Keeping the first components makes the clustering more stable (denoise)

$\Rightarrow$ But how many components do we keep?

Representation of Hierarchical clustering using PCA

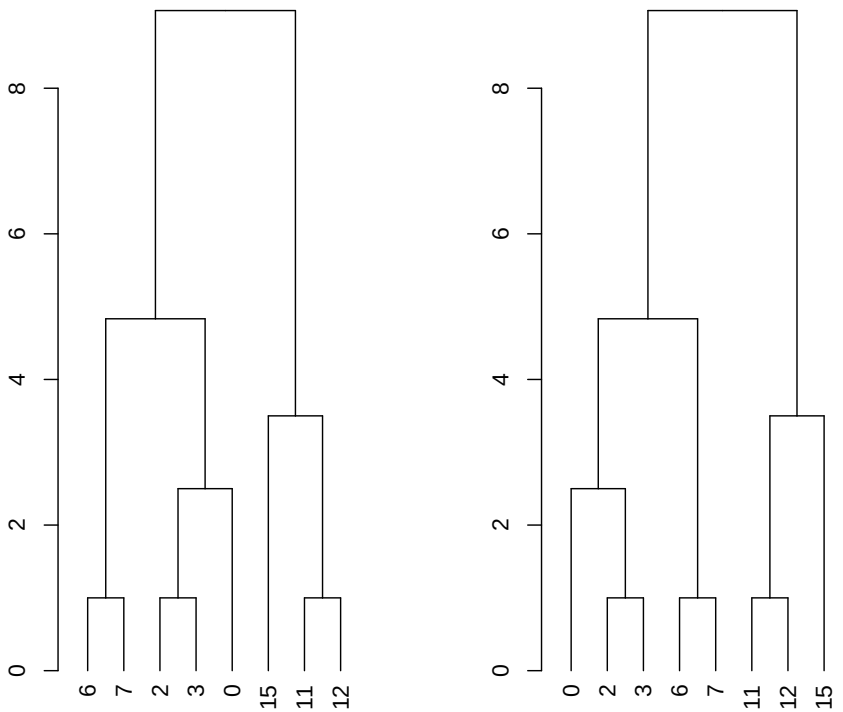
AHC onto the first 5 principal components



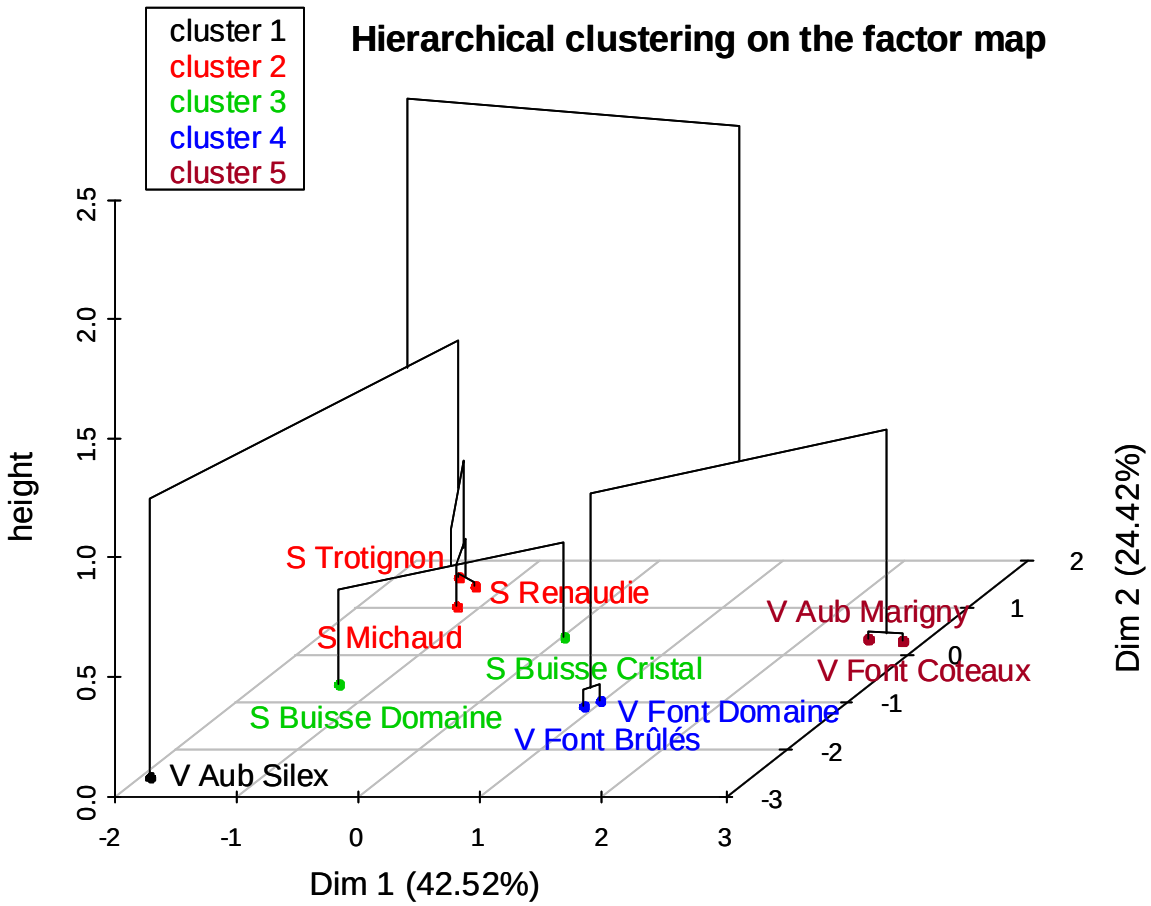
observations are sorted according to their coordinate on $F_{.1}$

Why sorting the tree?

```
X <- c(6,7,2,0,3,15,11,12)
names(X) <- X
library(cluster)
par(mfrow=c(1,2))
plot(as.dendrogram(agnes(X)))
plot(as.dendrogram(agnes(sort(X))))
```

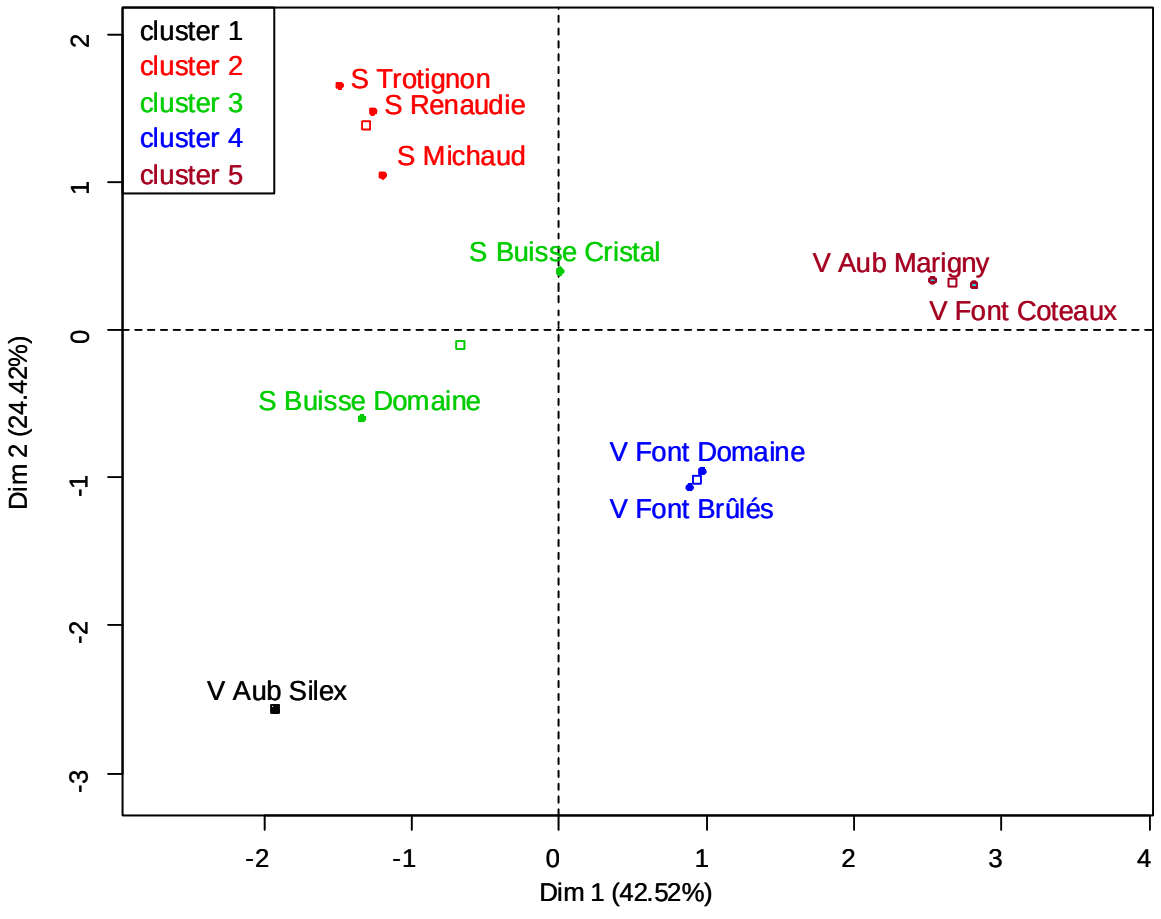


Hierarchical tree on the principal components map



Hierarchical tree gives an idea of the other dimensions

Partition on the principal components map



Continuous vision (principal components) and discontinuous (clusters)

What to do with categorical data???

Which distance to use?

- two categories: Jaccard index, Dice's coefficient simple match, etc. Indices well- fitted for presence/absence data
- with more than 2 categories: use for example the χ^2 distance

Counterpart of PCA for categorical data: Multiple Correspondence Analysis

MCA as a preprocessing

- transform categorical variables in continuous ones
- delete the last dimensions to denoise (more stable clustering)
- clustering on all MCA dimensions: clustering with χ^2 distances

Multiple Correspondence Analysis (MCA)

$X_{I\times m}$ m categorical variables coded with indicator matrix A

$$X = \begin{array}{|c|c|c|}\hline y & \dots & attack \\ y & \dots & attack \\ y & \dots & attack \\ n & \dots & suicide \\ & & \\ & & \\ & & \\ n & \dots & accident \\ n & \dots & suicide \\ \hline \end{array} \quad A = \begin{array}{|cc|c|ccc|}\hline 1 & 0 & \dots & 1 & 0 & 0 \\ 1 & 0 & \dots & 1 & 0 & 0 \\ 1 & 0 & \dots & 1 & 0 & 0 \\ 0 & 1 & \dots & 0 & 1 & 0 \\ & & & & & \\ & & & & & \\ & & & & & \\ 0 & 1 & \dots & 0 & 0 & 1 \\ 0 & 1 & \dots & 0 & 1 & 0 \\ \hline \end{array} \quad D_p = \begin{pmatrix} p_1 & & 0 \\ & \ddots & \\ 0 & & p_J \end{pmatrix}$$

For a category c, the frequency of the category: $p_c = I_c/I$.
A SVD on weighted matrix: $Z = \frac{1}{\sqrt{mI}}(A - 1p^T)D_p^{-1/2} = U\Lambda V'$

The PC ($F = U\Lambda$) satisfies: $F_l = \arg\max_{F_l \in \mathbb{R}^I} \quad \frac{1}{m}\sum_{j=1}^{m} \eta^2(F_l, X_j)$

$\eta^2(F_l, X_j) = \frac{\sum_{c=1}^{C_j}(F_{.c} - F_{..})^2}{\sum_i \sum_c (F_{ic})^2}$

Benzecri, 1973 :*"All in all, doing a data analysis, in good mathematics, reduces to computing eigenvectors; all the science (or the art) of it is in finding the right matrix to diagonalize"*