Clustering

Marie Chavent

Master MAS, Université de Bordeaux

14 novembre 2019

Introduction

How to define groups of of observation or variables which are similar ?

Example : quantitative data where 8 mineral waters are described on 16 variables.

```
load("../data/eaux.rda")
print(data[,1:4])
```

```
##             saveur.amère saveur.sucrée saveur.acide saveur.salée
## St Yorre             3.4           3.1          2.9          6.4
## Badoit               3.8           2.6          2.7          4.7
## Vichy                2.9           2.9          2.1          6.0
## Quézac               3.9           2.6          3.8          4.7
## Arvie                3.1           3.2          3.0          5.2
## Chateauneuf          3.7           2.8          3.0          5.2
## Salvetat             4.0           2.8          3.0          4.1
## Perrier              4.4           2.2          4.0          4.9
```

- From distances between observations : which distance ?
- From links between variables : which link measure ?

Depends on the nature of the data : quantitative, qualitative or mixed.

With the Euclidean distances between observations ?

```
print(dist(data),digit=2)
```

```
##             St Yorre Badoit Vichy Quézac Arvie Chateauneuf Salvetat
## Badoit           4.1
## Vichy            7.9    4.8
## Quézac           2.9    5.3   9.7
## Arvie            3.0    1.8   5.5    4.7
## Chateauneuf      2.9    1.8   5.7    4.3   1.3
## Salvetat         4.0    1.2   5.4    4.9   1.8         1.6
## Perrier          8.2   10.6  14.7    6.2  10.1         9.9     10.3
```

With the correlations between the variables ?

```
print(cor(data[,1:4]),digit=2)
```

```
##               saveur.amère saveur.sucrée saveur.acide saveur.salée
## saveur.amère          1.00         -0.83         0.78        -0.67
## saveur.sucrée        -0.83          1.00        -0.61         0.49
## saveur.acide          0.78         -0.61         1.00        -0.44
## saveur.salée         -0.67          0.49        -0.44         1.00
```

With an automatic clustering method.

Examples of clustering.

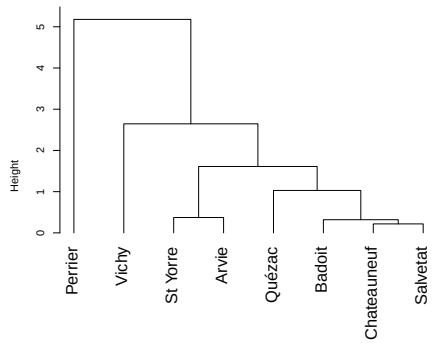
Partition in 4 clusters of the observations.

```
##             P4
## St Yorre     1
## Badoit       2
## Vichy        3
## Quézac       2
## Arvie        1
## Chateauneuf  2
## Salvetat     2
## Perrier      4
```

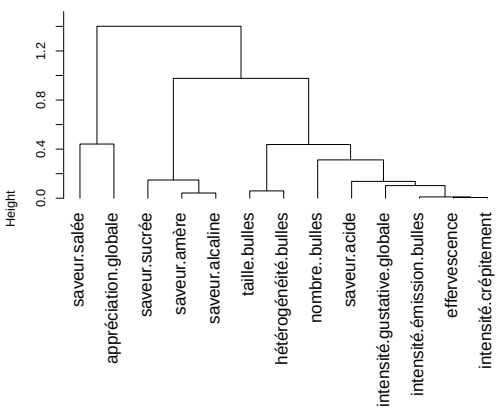
Partition in 3 clusters of the variables.

```
##                              P3
## saveur.amère                  1
## saveur.sucrée                 1
## saveur.acide                  2
## saveur.salée                  3
## saveur.alcaline               1
## appréciation.globale          3
## intensité.émission.bulles     2
## nombre..bulles                2
## taille.bulles                 2
## hétérogénéité.bulles          2
## effervescence                 2
## intensité.gustative.globale   2
## intensité.crépitement         2
```

Hierarchy of the observations



Hierarchy of the variables



There exist different clustering algorithms depending on :

- the nature of the objects to be clustered : observations or variables,
- the nature of the data : quantitative, qualitative or mixed,
- the nature of the clustering structure : parition or hierarchy,
- the nature of the clustering approach : geometric approach (distance, dissimilarity, similarity) or probabilist approach (mixture models).

In this chapter, only clustering methods for quantitative data, and geometrical approaches are studied.

Outline

1. Basic notions

Let us consider a set $\Omega = \{1, ..., i, ..., n\}$ of n observations described on p quantitative variables in a data matrix $\mathbf{X}$:

$$\mathbf{X} = \begin{array}{c} \\ 1 \\ \vdots \\ i \\ \vdots \\ n \end{array} \begin{array}{c} \begin{array}{ccc} 1 \cdots & j & \cdots p \end{array} \\ \begin{bmatrix} & \vdots & \\ \cdots & x_{ij} \in \mathbb{R} & \cdots \\ & \vdots & \end{bmatrix} \end{array}.$$

- An observation $i \in \Omega$ eis described by a vector $\mathbf{x}_i \in \mathbb{R}^p$ (row of $\mathbf{X}$).
- A weight poids w_i is associated with each observation i. The weight $w_i = \frac{1}{n}$ is usually used.

The input is then like in PCA (Principal Component Analysis), a cloud of n weighted points in $\mathbb{R}^p$.

Distance, dissimilarité ou similarité ?

A similarity is an function $s : \Omega \times \Omega \to \mathbb{R}^+$ with $\forall i, i' \in \Omega$:

$$\begin{aligned} s(i, i') &\geq 0, \\ s(i, i') &= s(i', i), \\ s(i, i) &= s(i', i') = smax \geq s(i, i') \end{aligned}$$

For binary data for instance, the contingency table between two observations i and i' is :

		individu i'	
		1	0
individu i	1	a	b
	0	c	d

Wellknown normalized ($smax = 1$) similarity measures $s(i, i')$ are in that case :

$$\text{Jaccard} : \frac{a}{a+b+c} \qquad \text{Russel et Rao} : \frac{a}{2a+b+c+d}$$

$$\text{Dice ou Czekanowski} : \frac{2a}{2a+b+c} \qquad \text{Ochiai} : \frac{a}{\sqrt{a+b}\sqrt{a+c}}$$

A dissimilarity d is a function : $\Omega \times \Omega \to \mathbb{R}^+$ with :

$$d(i, i') \geq 0, \quad d(i, i') = d(i', i), \quad d(i, i) = 0$$

Note that a similarity s is easily transformed is a dissimilarity d with :

$$d(i, i') = smax - s(i, i')$$

A distance is a dissimilarity verifying the triangular inequality :

$$d(i, j) \leq d(i, k) + d(k, j) \ \forall i, j, k \in \Omega.$$

Standard distances between two observations i and i' described by two vectors $\mathbf{x}_i$ and $\mathbf{x}_{i'}$ in $\mathbb{R}^p$:

- The simple Euclidean distance :

$$d^2(\mathbf{x}_i, \mathbf{x}_{i'}) = \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2$$

- The normalized Euclidean distance :

$$d^2(\mathbf{x}_i, \mathbf{x}_{i'}) = \sum_{j=1}^{p} \frac{1}{s_j^2} (x_{ij} - x_{i'j})^2,$$

  where ${s_j}^2 = \frac{1}{n} \sum_{i=1}^{n} (x_{ij} - \bar{x}^j)^2$ et $\bar{x}^j = \frac{1}{n} \sum_{i=1}^{n} x_{ij}$.

- Distance of city-block (or Manhattan) : $d(\mathbf{x}_i, \mathbf{x}_{i'}) = \sum_{j=1}^{p} \mid x_{ij} - x_{i'j} \mid$
- Distance of Chebychev (or max) : $d(\mathbf{x}_i, \mathbf{x}_{i'}) = \max_{j=1,\ldots,p} \mid x_{ij} - x_{i'j} \mid$

- Usually simple Euclidean distance is used when all variables are measured on the same scale. Indeed if a variable has a bigger variance, simple Euclidean distance will give more importance to the difference between the two observations on this variable than on the others.
- In case of very different sclare measures between the variables, it is better to use the normalized Euclidean distance to give all the variables the same weight. Note that this is equivalent to use simple Euclidean distance on the standardized data (centered and normalized).

Partition or hierarchy ?

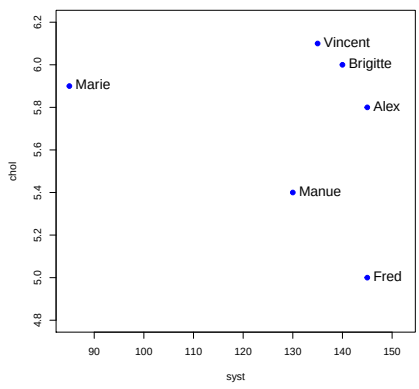
A partition P_K of Ω in K clusters is a set $(C_1, \ldots, C_k, \ldots, C_K)$ of non empty and non-overlapping clusters and which combination is Ω :

$$C_k \neq \emptyset \; \forall k \in \{1, \ldots K\},$$
$$C_k \cap C_{k'} = \emptyset \; \forall k \neq k'$$
$$C_1 \cup \ldots \cup C_K = \Omega$$

Example : if $\Omega = \{1, \ldots, 7\}$, $P_3 = (C_1, C_2, C_3)$ with $C1 = \{7\}$, $C_2 = \{5, 4, 6\}$ and $C_3 = \{1, 2, 3\}$ is a partition in 3 clusters of Ω.

Propose a partition in 3 clusters of the 6 observations below.

```
##          syst chol
## Brigitte  140  6.0
## Marie      85  5.9
## Vincent   135  6.1
## Alex      145  5.8
## Manue     130  5.4
## Fred      145  5.0
```



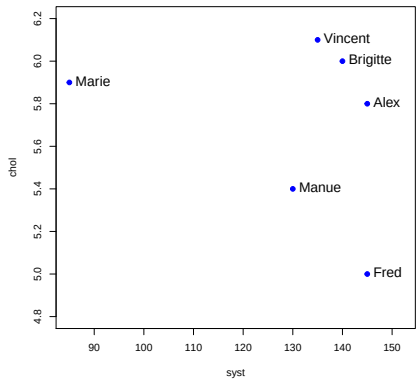
A hierarchy H of Ω is a set of non-empty clusters verifying :

- $\Omega \in H$,
- $\forall i \in \Omega,\ \{i\} \in H$ (all singletons are in the hierarchy),
- $\forall A, B \in H, A \cap B \in \{A, B, \emptyset\}$ (two clusters in the hierarchy are either disjoint or one included in the other.).

Example : $H = \{\{1\}, \ldots, \{7\}, \{4,5\}, \{2,3\}, \{4,5,6\}, \{1,2,3\}, \{4,5,6,7\}, \Omega\}$.

Give a hierarchy of the 6 observations below.

```
##          syst chol
## Brigitte  140  6.0
## Marie      85  5.9
## Vincent   135  6.1
## Alex      145  5.8
## Manue     130  5.4
## Fred      145  5.0
```



Total, within or between inertia ?

Let P_K be a partition of the set $\Omega = \{1, ..., i, ..., n\}$ of n observations described in $\mathbb{R}^p$ by $\mathbf{x}_i$ and weighted by w_i. Let use define :

$$\begin{aligned}
\mu_k &= \sum_{i \in C_k} w_i \quad \text{the weight of } C_k \\
\mathbf{g}_k &= \frac{1}{\mu_k} \sum_{i \in C_k} w_i \mathbf{x}_i \quad \text{the gravity center of } C_k \\
d^2(\mathbf{x}_i, \mathbf{x}_{i'}) &= \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 \quad \text{the squared Euclidean distance}
\end{aligned}$$

The total inertia T measures the dispersion of the cloud of n observations :

$$T = \sum_{i=1}^{n} w_i d^2(\mathbf{x}_i, \mathbf{g})$$

where $\mathbf{g}$ is the gravity center of the n observations.

The total inertia is independant of the partition P_K.

The between-cluster inertia B of the partition P_K is the inertia of the gravity centers of the clusters weighted by μ_k and measures then the separation between the clusters.

$$B = \sum_{k=1}^{K} \mu_k d^2(\mathbf{g}_k, \mathbf{g})$$

The within-cluster inertia W of the partition P_K is the sum of the inertia of the clusters and measures then the heterogeneity within the clusters.

$$W = \sum_{k=1}^{K} I(C_k)$$

$$I(C_k) = \sum_{i \in C_k} w_i d^2(\mathbf{x}_i, \mathbf{g}_k)$$

A good partition has a large between-cluster inertia and a small within-cluster inertia.

Remaks.

Total, between and within inertia is also referred to

- total, within or between variance when $w_i = \frac{1}{n}$,
- total, within or between sum of squares when $w_i = 1$.

Moreover,

- when $w_i = \frac{1}{n}$, $T = s_1^2 + \ldots + s_p^2$, where s_j^2 is the empirical variance of the jth variable.
- when $w_i = \frac{1}{n}$ and data are standardized, $T = p$.

Note that the inertia (total, between on within) calculated with the standardized data is equal to the inertia calculated with the p principal components of $\mathbf{X}$.

The wellknown relation between total, within and between inertia is :

$$T = W + B$$

Minimizing the within-cluster inertia (the heterogeneity within the clusters) is then equivalent to maximizing the between-clusters inertia (the separation between the clusters).

Note that this relation comes from the Huygens theorem (to demonstrate) :

$$\forall \mathbf{a} \in \mathbb{R}^p,\ I_{\mathbf{a}} = I_{\mathbf{g}} + (\sum_{i=1}^{n} w_i) d^2_{\mathbf{M}}(\mathbf{a}, \mathbf{g});$$

where $\mathbf{M}$ is a metric on $\mathbb{R}^p$ and $I_{\mathbf{a}}$ is the inertia about a point $\mathbf{a} \in \mathbb{R}^p$ defined as follows :

$$I_{\mathbf{a}} = \sum_{i=1}^{n} w_i d^2_{\mathbf{M}}(\mathbf{x}_i, \mathbf{a}).$$

Le proportion (in %) of the total inertia explained by the partition P_k is :

$$(1 - \frac{W}{T}) \times 100$$

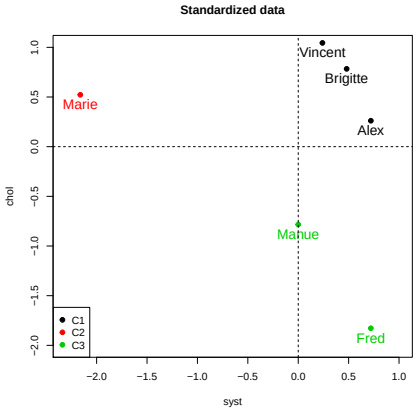
This criterion is equal to :

- 100 for the partition in n clusters (the singletons),
- 0 for the partition in one cluster (Ω).

This criterion increases when the number of clusters K increases. It can then be used only to compare partitions with the same number of clusters. If the proportion of inertia explained by one partition is greater than that explained by an other (with the same number of clusters), then we consider that the first partition is better than the second one.

Perform the total, within and between inertia (with the weights $w_i = \frac{1}{n}$) of the partition below as well as the proportion of inertia explained by this partition.

```
##           syst  chol
## Brigitte  0.48  0.78
## Marie    -2.16  0.52
## Vincent   0.24  1.04
## Alex      0.72  0.26
## Manue     0.00 -0.78
## Fred      0.72 -1.83
## g1        0.48  0.70
## g2       -2.16  0.52
## g3        0.36 -1.31
## g         0.00  0.00
```




```
## [1] "squared distances"
##         Brigitte Marie Vincent  Alex Manue  Fred    g1    g2    g3
## Marie       7.05
## Vincent     0.13  6.04
## Alex        0.33  8.38    0.84
## Manue       2.69  6.38    3.40  1.61
## Fred        6.88 13.83    8.48  4.36  1.61
## g1          0.01  7.01    0.18  0.25  2.42  6.43
## g2          7.05  0.00    6.04  8.38  6.38 13.83  7.01
## g3          4.38  9.70    5.54  2.58  0.40  0.40  4.02  9.70
## g           0.84  4.95    1.15  0.59  0.61  3.86  0.72  4.95  1.83
```


Outline

A good partition of Ω has

- homogeneous cluster : observations within a cluster are similar,
- separated cluster : observation of two different clusters are not similar.

Let

$$\mathcal{H} : \mathcal{P}_K(\Omega) \to \mathbb{R}^+$$

be a criterion measuring the heterogeneity of a partition P_K.

For instance the diameter of a partition is the largest diameter of its clusters :

$$\mathcal{H}(P_K) = \max_{k=1,\dots,K} \max_{i,i' \in C_k} \underbrace{d(\mathbf{x}_i, \mathbf{x}_{i'})}_{\text{diam}(C_k)}.$$

This criterion can be performed with any dissimilarity matrix.

An other very popular heterogeneity criterion is the intra-cluster inertia of a partition i.e. the sum of the inertia of its clusters :

$$\mathcal{H}(P_K) = \sum_{k=1}^{K} I(C_k) = W.$$

This criterion is performed only with quantitative data.

This criterion is very popular because here homogeneous clusters (small within-cluster inertia) corresponds to well separated clusters (large between-cluster inertia) thanks to the central relation $T = W + B$.

The aim of partitionning methods is then to find the partition P_K of Ω which minimizes $\mathcal{H}(P_K)$.

Because Ω is a finite set, $\mathcal{P}_K(\Omega)$ is also finite. The partition P_K which globaly minimizes $\mathcal{H}(P_K)$ can then be found by complete enumeration. But in practice, this is not possible because :

$$card(\mathcal{P}_K(\Omega)) \sim \frac{n^K}{K!}.$$

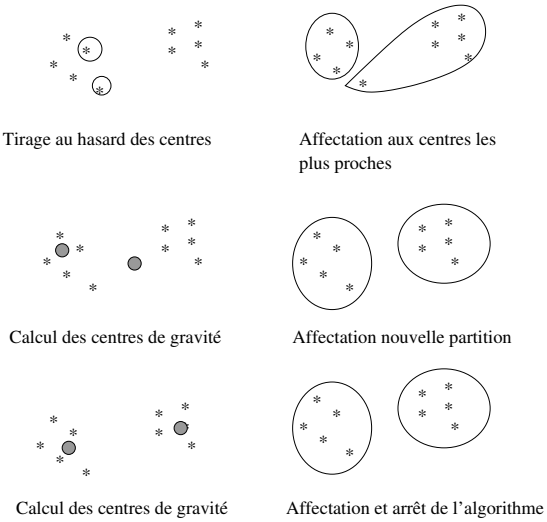
For instance, perform approximatively the number of partitions in $K = 3$ clusters of a set of $n = 100$ observations.

Partitionning methods are then usually heuristics with the following general scheme :

- Start from a possible solution i.e. a partition P_K^0.
- At setp m+1, find the partition $P_K^{m+1} = g(P_K^m)$ such that $\mathcal{H}(P_K^{m+1}) \leq \mathcal{H}(P_K^m)$.

When $\mathcal{H}(P_K)$ is the within-cluster inertia, this heuristic refers to the wellknown k-means partionning method. The k-means algorithm starts from an initial partition and repeats :

- a representation step where gravity centers of each clusters is performed,
- an affectation step where each observation is assigned to the cluster with nearest gravity center (smallest Euclidean distance).



The k-means algorithm :

(a) Initialization
Choose a partition $P_K = (C_1, \ldots, C_K)$ and perform $\mathbf{g}_1, \ldots, \mathbf{g}_K$

(b) affectation step
$test \leftarrow 0$
for all i in 1 to n do

find the cluster $k*$ such that

$$k* = \arg \min_{k=1,\ldots,K} d(\mathbf{x}_i, \mathbf{g}_k)$$

find the cluster ℓ of i

If $k* \neq \ell$

$test \leftarrow 1$
$C_{k*} \leftarrow C_{k*} \cup \{i\}$
$C_\ell \leftarrow C_\ell \backslash \{i\}$

(c) Representation step
For all k in 1 to K perform the gravity center of the new clusster C_k

(d) If $test = 0$ END, otherwise go in (b)

Some properties of the k-means algorithm

- The algorithm converges toward a local minimum of the within-cluster inertia (to demonstrate).
- The solution (the final partition) depends on the initial partition. In practice :
  - The algorithm is repeated N times with different initial partitions chosen at random.
  - The best partition (with the smallest within-cluster inertia) among the N final partitions is chosen.
- The complexity of the algorithm is in $o(KpnT)$ where T is the number of iterations. This complexity is relatively small and the k-means algorithm applies then to big data sets (where n is large).

Let us now study a small example of clustering with the k-means algorithm. This example also highlight two practical point of views :

- Why is it sometimes important to standardize the data ?
- How to interpret the clusters with PCA ?

The dataset gives the quantity of proteins consumed in 9 types of foods by 25 european countries : 25 observations and 9 quantitative variables.

```
##       Red.Meat White.Meat Eggs Milk Fish Cereals Starchy.Foods Nuts Fruite.veg.
## Alban     10.1        1.4  0.5  8.9  0.2      42           0.6  5.5         1.7
## Aust       8.9       14.0  4.3 19.9  2.1      28           3.6  1.3         4.3
## Belg      13.5        9.3  4.1 17.5  4.5      27           5.7  2.1         4.0
## Bulg       7.8        6.0  1.6  8.3  1.2      57           1.1  3.7         4.2
## Czech      9.7       11.4  2.8 12.5  2.0      34           5.0  1.1         4.0
```


The k-means algorithm is applied to this dataset with :

- $K = 4$ clusters,
- $N = 5$ repetitions of the algorithm.

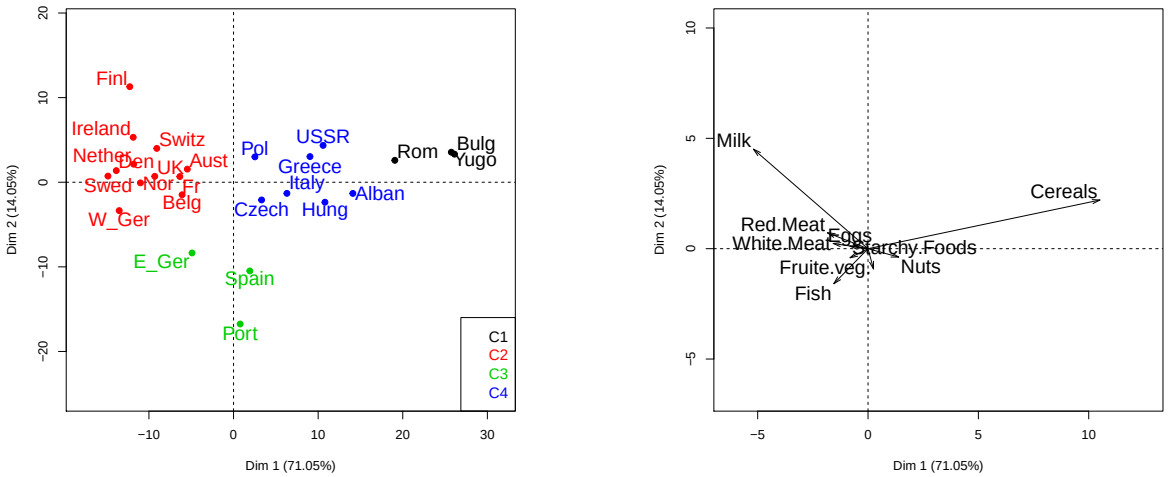
Partition in 4 clusters :

```
##   Alban    Aust    Belg    Bulg   Czech     Den   E_Ger    Finl      Fr  Greece    Hung Ireland   Italy  Nether     Nor
##       4       2       2       1       4       2       3       2       2       4       4       2       4       2       2
##    Port     Rom   Spain    Swed   Switz      UK    USSR   W_Ger    Yugo
##       3       1       3       2       2       2       4       2       1
```

Proportion of inertia explained by this partition

```
## [1] 0.758
```

A non normalized PCA is performed to vizualize and interpret the partition.



Give an interpretation of this partition in 4 clusters of the 25 countries.

The *k*-means algorithm is now applied to the standardized dataset.

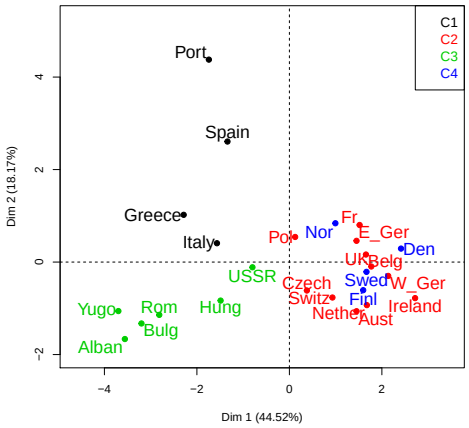
Partition in 4 clusters :

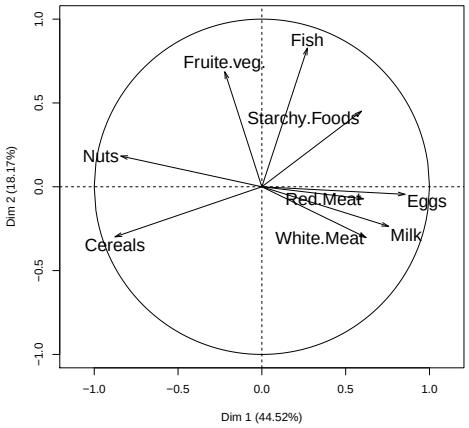
```
##    Alban     Aust     Belg     Bulg    Czech      Den    E_Ger     Finl       Fr   Greece     Hung Ireland   Italy  Nether      Nor
##        3        2        2        3        2        4        2        4        2        1        3        2        1        2        4
##    Port      Rom    Spain     Swed    Switz       UK     USSR   W_Ger    Yugo
##       1        3        1        4        2        2        3        2        3
```

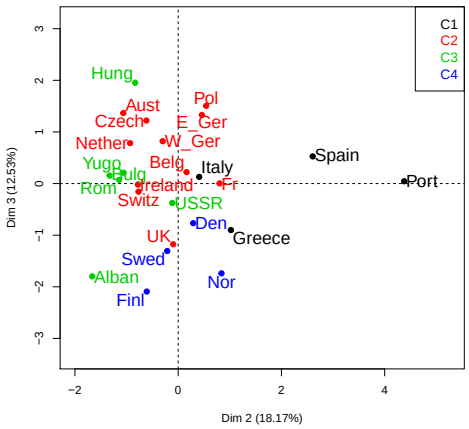
Proportion of inertia explained by this partition

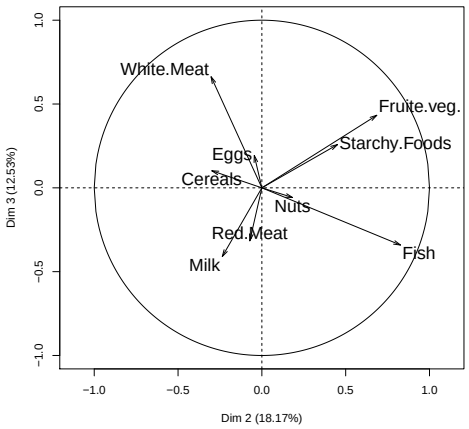
```
## [1] 59.8
```

Give an interpretation of this partition in 4 clusters using the plots of the normalized PCA hereafter.









Outline

The clustering structure is now the hierarchy (as defined in the section 1).

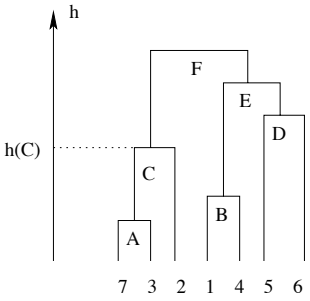
A binary hierarchy is a hierarchy H of Ω whose clusters are the combination of two clusters. The number of clusters (except the singletons) of a binary hierarchy is $n-1$.

An indexed hierarchy is a couple (H, h) where H is a binary hierarchy and h is a function from H in $\mathbb{R}^+$ such that :

$$\forall A \in H, h(A) = 0 \Leftrightarrow A \text{ is a singleton}$$
$$\forall A, B \in H, A \neq B, A \subset B \Rightarrow h(A) \leq h(B) \text{ (h increasing function)}$$

The graphical representation of an indexed hierarchy is called a dendrogram (or hierarchical tree) and the function h measures the height of the clusters in the dendrogram.

For instance $H = \{\{1\}, \ldots, \{7\}, \{7,3\}, \{2,7,3\}, \{1,4\}, \{5,6\}, \{1,4,5,6\}, \Omega\}$ can be indexed to get the following dendrogram :



An indexed hierarchy defines a sequence of nested partitions from 2 to n clusters. These partitions are obtained by cutting the dendrogram according to a sequence of horizontal lines.

For instance cut the dendrogram above to get the partition in 2 clusters and the partition in 4 clusters.

Because the function h is increasing, the dendrogram has no inversion : if $C = A \cup B$ the cluster C is higher than the clusters A and B in the dendrogram.

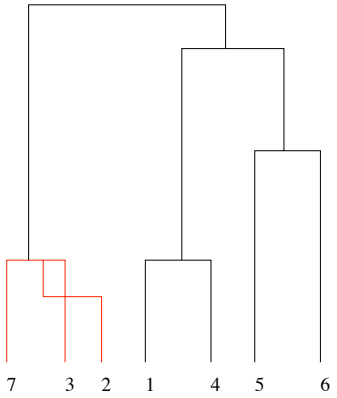


FIGURE – Example of inversion in the dendrogram of a hierarchy

Note that an indexed hierarchy has several equivalent representation. Indeed the order of the representation of the n observations at the bottom the hierarchy can be modified and the number of possible representations is 2^{n-1}.

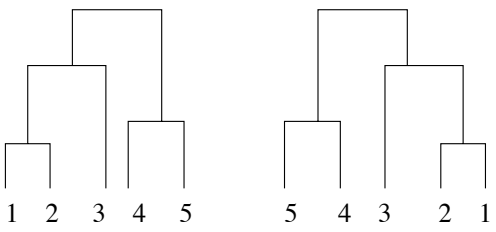


FIGURE – Two representations of the same indexed hierarchy

The ascendant hierarchical clustering algorithm :

(a) Initialization
The initial partition is the partition of the singletons $P_n = (C_1, \ldots, C_n)$ with $C_k = \{k\}$, $k = 1 \ldots n$.

(b) Aggregative step
Aggregate the two clusters C_k and $C_{k'}$ of the partition $P_K = (C_1, \ldots, C_K)$ in K clusters built in the previous step, which minimize an aggregation measure $D(C_k, C_{k'})$. A new partition P_{K-1} in $K-1$ clusters is then obtained.

(c) Repeat set (b) until the partition in one cluster $P_1 = (\Omega)$ is obtained.

This algorithm depends on the choice of the aggregation measure

$$D : \mathcal{P}(\Omega) \times \mathcal{P}(\Omega) \to \mathbb{R}^+$$

used to measure the dissimilarity between two clusters.

For instance three dissimilarity measures between two clusters A and B are represented herafter. Note that they are based on the choice of the the dissimilarity d between two observations.

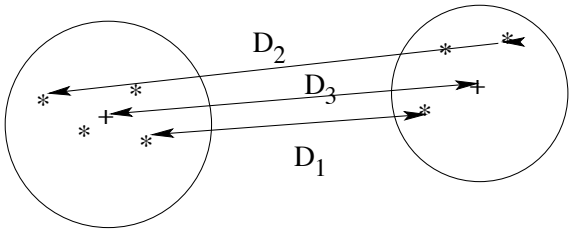


FIGURE – Three dissimilarity measures between clusters

The single link measure is :

$$D_1(A, B) = \min_{i \in A, i' \in B} d(\mathbf{x}_i, \mathbf{x}_{i'})$$

The complete link measure is :

$$D_2(A, B) = \max_{i \in A, i' \in B} d(\mathbf{x}_i, \mathbf{x}_{i'})$$

The Ward's minimum variance measure is :

$$D_3(A, B) = \frac{\mu_A \mu_B}{\mu_A + \mu_B} d^2(\mathbf{g}_A, \mathbf{g}_B)$$

where

$$\mu_A = \sum_{i \in A} w_i \text{ is the weight of the cluster,}$$

$$\mathbf{g}_A = \frac{1}{\mu_A} \sum_{i \in A} w_i \mathbf{x}_i \text{ is the gravity center of the cluster.}$$

The function h of the indexed hierarchy (H, h) is usually defined as :

$$h(A \cup B) = D(A, B)$$

where $h(A \cup B)$ is the height of the cluster $A \cup B$ in the dendrogram of H.

However the property $A \subset B \Rightarrow h(A) \leq h(B)$ is not satisfied for every aggregation measures D. It is satisfied for the three aggregation measures introduced previously (single link, complete link and Ward) but it is not satisfied for instance for the average link. In that case

$$h(A \cup B) = max(D(A, B), h(A), h(B))$$

can be used to avoid inversions in the dendrogram of the hierarchy.

Property of the Ward's minimum variance measure.

$$D_3(A,B) = \frac{\mu_A \mu_B}{\mu_A + \mu_B} d^2(\mathbf{g}_A, \mathbf{g}_B) = I(A \cup B) - I(A) - I(B) \quad (1)$$

It is then easy to show that if A and B are two clusters of a partition P_K in K clusters aggregated to build a partition P_{K-1} in $K-1$ clusters, we have :

$$D_3(A,B) = I(A \cup B) - I(A) - I(B) = \mathcal{H}(P_{K-1}) - \mathcal{H}(P_K) \quad (2)$$

where $\mathcal{H}$ is the intra-cluster inertia heterogeneity criterion.

We deduce from (1) and (2) that :

- the Ward's aggregation measure $D(A,B)$ interprets also as the increase of the within-cluster inertia when a A et de B are aggregated in a new partition.
- The sum of all the heights of the Ward's dendrogram is equal to the total inertia T.
- The sum of the $n-K$ heights of the Ward's dendrogram is equal to the within-cluster W of the partition P_K of this dendrogram.

The Ward's ascendant hierarchical clustering algorithm aggregates at each step the two clusters that give the partition of smallest within-cluster inertia. The criterion optimized is then the same as in the k-means method.

Implementation of the Ward's ascendant hierarchical clustering algorithm.

(a) Initialization : build the matrix $\Delta = (\delta_{ij})_{n\times n}$ of the Ward measures between the singletons :

$$\delta_{ij} = D_3(\{i\},\{j\}) = \frac{w_i w_j}{w_i + w_j} d^2(\mathbf{x}_i, \mathbf{x}_j).$$

(b) Aggregation step

- Aggregate the two clusters A and B of P_K which minimize $D_3(A,B)$ to built a new partition P_{K-1}.
- Perform the Ward measure between $A \cup B$ and the other clusters C of P_{K-1} with the Lance & Williams formula :

$$D_3(A\cup B, C) = \frac{\mu_A+\mu_C}{\mu_A+\mu_B+\mu_C} D_3(A,C) + \frac{\mu_B+\mu_C}{\mu_A+\mu_B+\mu_C} D_3(B,C) - \frac{\mu_C}{\mu_A+\mu_B+\mu_C} D_3(A,B)$$

(c) Repeat step (b) until the partition in one cluster is obtained

This Ward algorithm takes in input :

- the vector $\mathbf{w} = (w_i)_{i=1,\dots,n}$ of the weights of the observations,
- the matrix $\mathbf{D} = (d_{ij})_{n\times n}$ of the Euclidean distances between the observations.

With the funcion `hclust` of R, the arguments must be the following :

- `hclust(d=`Δ`, method="ward.D")` when the weights are uniform,
- `hclust(d=`Δ`, method="ward.D", members=`$\mathbf{w}$`)` otherwise.

When the weights are $w_i = \frac{1}{n}$, the matrix Δ of the Ward measures between the singletons is :

$$\Delta = \frac{\mathbf{D}^2}{2n}$$

where $\mathbf{D}^2 = (d_{ij}^2)_{n\times n}$.

The Ward algorithm above works with any dissimilarity matrix **D**. But when the dissimilarities are not Euclidean distances, the heterogeneity criterion minimized at each step is :

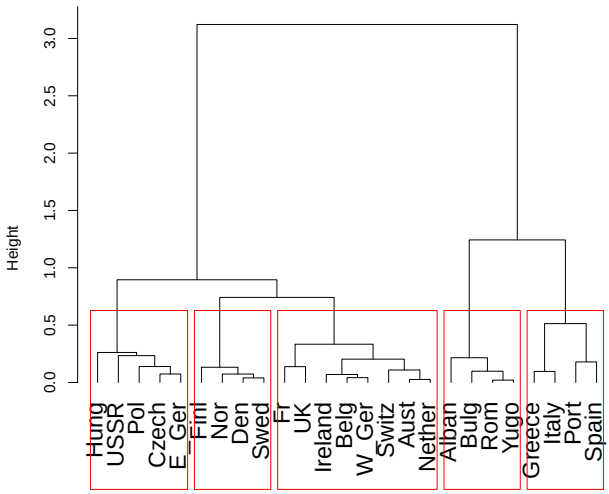
$$\mathcal{H}(P_K) = \sum_{k=1}^{K} \sum_{i \in C_k} \sum_{j \in C_k} \frac{w_i w_j}{2\mu_k} d_{ij}^2,$$

which interprets as a pseudo within-cluster inertia criterion.

Indeed for Euclidean distances, it can be shown that :

$$\sum_{i \in C_k} \sum_{j \in C_k} \frac{w_i w_j}{2\mu_k} d^2(\mathbf{x}_i, \mathbf{x}_j) = I(C_k).$$

Example of the standardized protein dataset.



Sum of the heights :

```
## [1] 9
```

The dendrogram suggests to cut the tree in 3 or 5 clusters.

The clusters can be interpreted, via the maps of PCA or via appropriate descriptive statistics.

The tree is cut to get a partition in 5 clusters of the 25 countries.

```
P5 <- cutree(tree,k=5)
P5
```

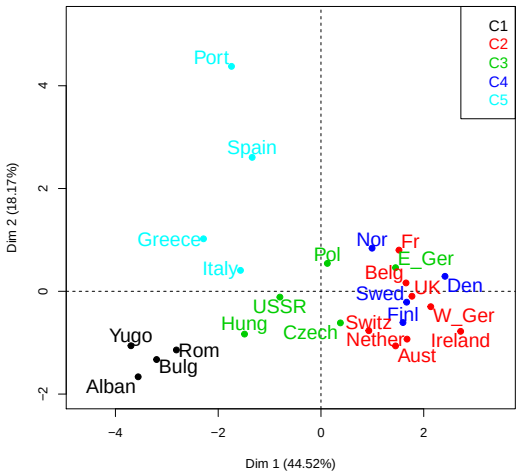
```
##   Alban    Aust    Belg    Bulg   Czech     Den   E_Ger    Finl      Fr  Greece    Hung Ireland   Italy  Nether     Nor     P
##       1       2       2       1       3       4       3       4       2       5       3       2       5       2       4
##    Port     Rom   Spain    Swed   Switz      UK    USSR   W_Ger    Yugo
##       5       1       5       4       2       2       3       2       1
```

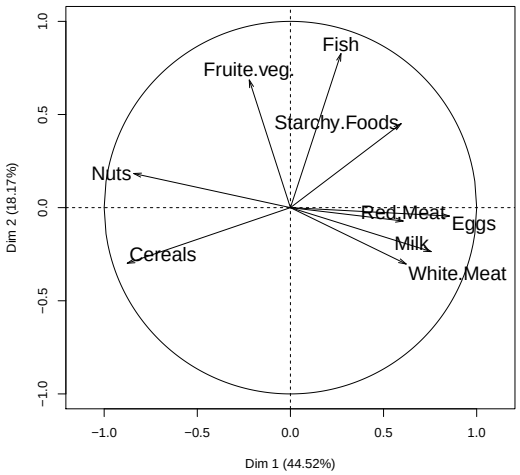
```
K <- 5
T <- sum(tree$height)
W <- sum(tree$height[1:(n-K)])

#Proportion of inertia explained by this partition
(1-W/T)
```

```
## [1] 0.67
```

Can this proportion of explained inertia be compared with the 0.58 of the k-means 4-clusters partition ?

Interpretation of the 5 clusters via the PCA maps.





Outline

Clusters interpretation

The clusters can be interpreted via adequate descriptive statistics calculated for :

- active variables : used in the clustering process,
- illustrative variables : used only for description purpose.

These variables can be numerical or categorial and the descriptive statistics will evaluate for instance :

- which levels explain the clusters i.e. are more frequent in a cluster, are mostly observed in a cluster.
- which numerical variables explain the clusters i.e. have mean values in the cluster different from the mean value in the complete dataset.

The function `catdesc` of the R package FactoMineR.

```
## [1] "Alban" "Bulg"  "Rom"   "Yugo"
##               v.test Mean in category Overall mean sd in category Overall sd p.value
## Cereals          3.8              1.8      2.6e-16           0.54          1 0.00017
## Nuts             2.2              1.0     -1.1e-17           0.41          1 0.02972
## Fish            -2.3             -1.1      8.3e-17           0.12          1 0.02342
## Milk            -2.4             -1.1     -2.1e-16           0.15          1 0.01862
## Starchy.Foods   -3.1             -1.5      1.8e-16           0.70          1 0.00190
## Eggs            -3.4             -1.6      5.9e-17           0.39          1 0.00070
```

The column `v.test` gives the so called test values of the numerical variables in the cluster.
The last column gives a `p.value` and by default the variables with a `p.value` superior to 0.05 are not plotted here.

The test-value $t_k(X^j)$ of a numerical variable X^j in a cluster C_k measures the difference between the mean of X^j in C_k and the mean of X^j in all the dataset, divided by the standard deviation of the mean of n_k observations drawn randomly without replacement (obtained from the empirical variance of X^j denoted $\sigma^2(X^j)$ hereafter) :

$$t_k(X^j) = \frac{\bar{X}_k^j - \bar{X}^j}{\sqrt{\frac{(n-n_k)}{n-1}\frac{\sigma^2(X^j)}{n_k}}} \tag{3}$$

If the test-value of a variable in a cluster is large (in absolute value), this variable charactizes this cluster.

Moreover, under the null hypothesis that the n_k observations of C_k are drawn randomly without replacement, the statistic $t_k(X^j)$ is approximalty $\mathbb{N}(0,1)$. If the `p.value` of this test is small (smaller than 0.05 for instance), this variable characterizes this cluster.

```
## $C2
##            v.test Mean in category Overall mean sd in category Overall sd p.value
## Red.Meat      3.5             1.03      1.7e-16           0.94          1 0.00052
## Eggs          3.2             0.96      5.9e-17           0.52          1 0.00125
## White.Meat    2.5             0.76      2.0e-17           0.70          1 0.01091
## Cereals      -2.4            -0.70      2.6e-16           0.28          1 0.01832
##
## $C3
##               v.test Mean in category Overall mean sd in category Overall sd p.value
## Starchy.Foods      2              0.8      1.8e-16           0.59          1   0.049
##
## $C4
##             v.test Mean in category Overall mean sd in category Overall sd p.value
## Milk           2.9             1.37     -2.1e-16           0.59          1  0.0033
## Fish           2.5             1.18      8.3e-17           0.51          1  0.0115
## Nuts          -2.1            -0.98     -1.1e-17           0.18          1  0.0371
## Fruite.veg.   -2.4            -1.14     -7.7e-17           0.28          1  0.0150
##
## $C5
##             v.test Mean in category Overall mean sd in category Overall sd p.value
## Fruite.veg.    3.6              1.7     -7.7e-17           0.31          1 0.00038
## Nuts           2.9              1.3     -1.1e-17           0.70          1 0.00423
## Fish           2.1              1.0      8.3e-17           1.20          1 0.03214
## White.Meat    -2.4             -1.1      2.0e-17           0.22          1 0.01554
```


The function `catdesc` describes also the clusters with the levels of the categorical variables.

```
##           Cla/Mod Mod/Cla Global p.value v.test
## zone=east      44     100     36    0.01    2.6
```

Cla/Mod = proportion of the level s in the cluster k
$= \frac{n_{ks}}{n_s}$

Mod/Cla = proportion of the cluster k in the level s
$= \frac{n_{ks}}{n_k}$

Global = proportion of the level s in all the dataset
$= \frac{n_s}{n}$

The test-value $t_k(X^s)$ of a level s in a cluster C_k is the formular (3) applied to the indicator vector X^s and then

$$t_k(X^s) = \frac{\bar{X}_k^s - \bar{X}^s}{\sqrt{\frac{(n-n_k)}{n-1}\frac{\sigma^2(X^s)}{n_k}}} \tag{4}$$

where :

- $\bar{X}_k^s = \frac{n_{ks}}{n}$ and $\bar{X}^s = \frac{n_s}{n}$
- $\sigma^2(X^s) = \frac{n_s}{n}(1 - \frac{n_s}{n})$.

If the test-value of a level in a cluster is large (in absolute value), this frequency of the level in the cluster is different (higher or smaller) from the frequency on all the observations i.e. the level is over or under represented in the cluster.

Again under the null hypothesis that the n_k observations of C_k are drawn randomly without replacement, the statistic $t_k(X^s)$ is approximalty $\mathbb{N}(0,1)$. If the `p.value` of this test is small (smaller than 0.05 for instance), this level characterizes this cluster.


```
## $C1
##           Cla/Mod Mod/Cla Global p.value v.test
## zone=east      44     100     36    0.01    2.6
##
## $C2
##           Cla/Mod Mod/Cla Global p.value v.test
## zone=west     100     100     32 9.2e-07    4.9
## zone=east       0       0     36 1.2e-02   -2.5
##
## $C3
##           Cla/Mod Mod/Cla Global p.value v.test
## zone=east      56     100     36  0.0024      3
##
## $C4
##            Cla/Mod Mod/Cla Global p.value v.test
## zone=north     100     100     16 7.9e-05    3.9
##
## $C5
##            Cla/Mod Mod/Cla Global p.value v.test
## zone=south     100     100     16 7.9e-05    3.9
```