TP3: Clustering with R

Me rendre un compte-rendu de ce TP (sauf l'exercice 1) pour le **vendredi 21/12** au plus tard.

Le fichier pdf est à déposer **directement sur moodle** (pas d'envoie par mail) à l'adresse suivante :

https://moodle-miashs.uf-mi.u-bordeaux.fr/

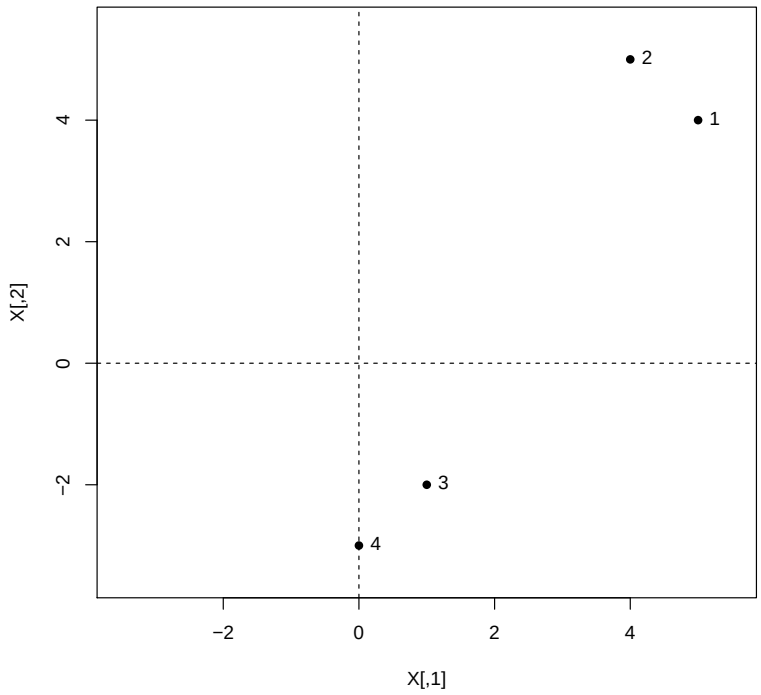
Ce travail peut être réalisé en **binôme**.

Exercice 1. The k-means algorithm

1. Let X be a data matrix where a set $\Omega = \{1, 2, 3, 4\}$ of $n = 4$ individuals are described by $p = 2$ variables. The individuals are weighted by $w_i = 1$. Apply *by hand* the k-means algorithm to Ω with $K = 2$ and with the two first rows of X chosen as initial centers. Perform the within-cluster sum of squares of the final partition.

```
X <- matrix(c(5,4,4,5,1,-2,0,-3),4,2,byrow=TRUE)
X
```

```
##      [,1] [,2]
## [1,]    5    4
## [2,]    4    5
## [3,]    1   -2
## [4,]    0   -3
```

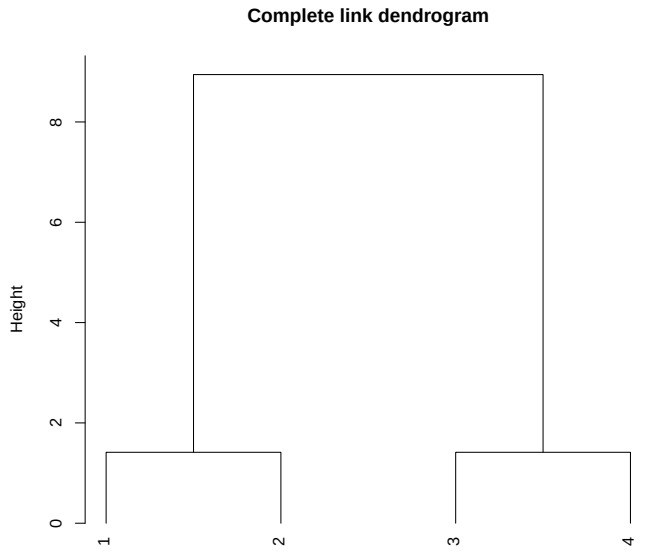


2. Use now the R function **kmeans()** to repeat the previous question. Check that you find the same results.

3. Perform the total sum of squares T of the data. Check that $T = B + W$ where B is the between-clusters sum of squares and W is the within-clusters sum of squares of the final partition.
4. Perform the proportion of variance explained by the final partition.

Exercice 2. The complete link ascendant hierarchical clustering algorithm.

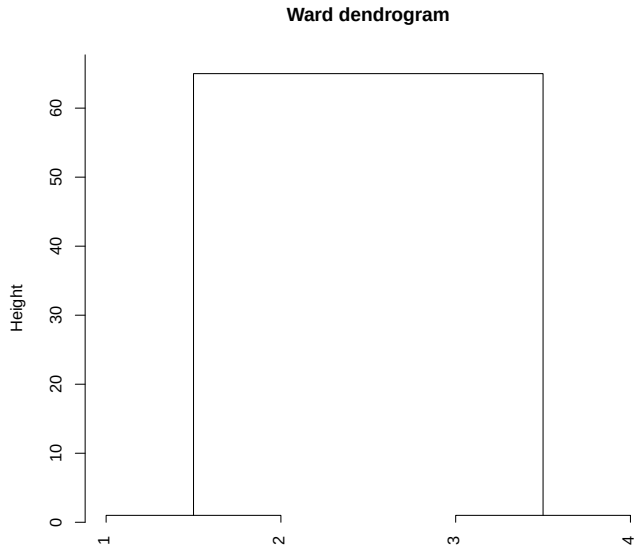
1. Apply now *by hand* the complete link hierarchical clustering algorithm to $\Omega = \{1, 2, 3, 4\}$ using the Euclidean distance to compare two individuals. Give the hierarchy H and plot the dendrogram obtained in that way. What partition in two clusters is obtained by cutting this dendrogram ?
2. Use now the R function **hclust() plot()** and **cutree()** to repeat the previous question. Check that you find the same results and then the following dendrogram.



3. Built the complete link dendrogram using the Manhattan distance instead of the Euclidean distance.

Exercice 3. The Ward's minimum variance hierarchical clustering algorithm.

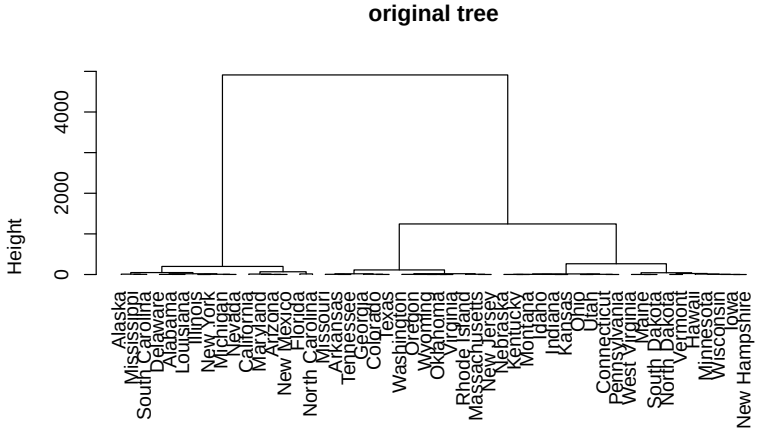
1. Apply now *by hand* the Ward's minimum variance method to $\Omega = \{1, 2, 3, 4\}$ where the individuals are still weighted by $w_i = 1$. Plot the dendrogram obtained in that way.
2. Use now the R function **hclust()** with the indications given in appendix to repeat the previous question. Check that you find the same results and then the following dendrogram.



Exercice 4. Reconstruct the upper part of the Ward dendrogram.

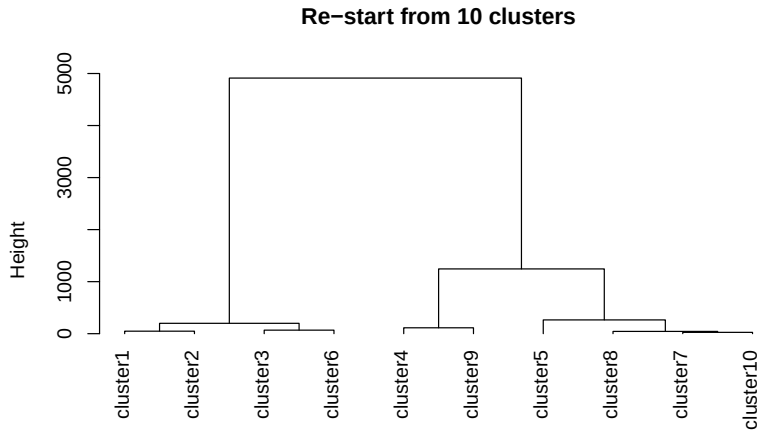
1. Use now the R function **hclust()** to apply the Ward's minimum hierarchical clustering method to the $n = 50$ american states described in the data **USArrests**. Here the individuals (states) are weighted by $\frac{1}{n}$.

```
#Violent crime rates by US state
help(USArrests)
```



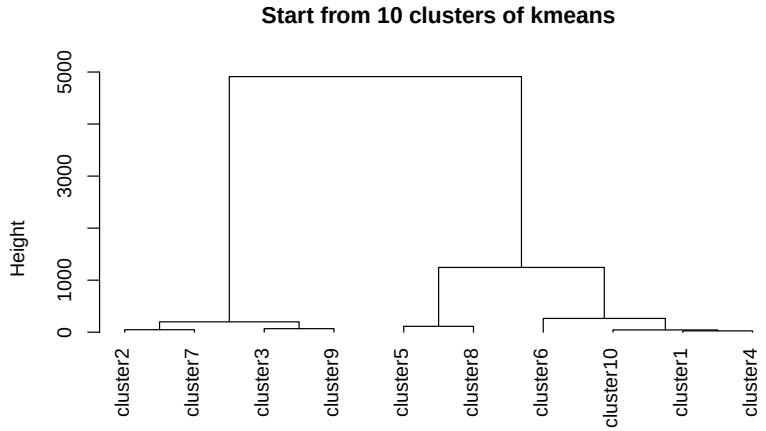
2. Cut the tree into ten clusters. What is the weight μ_k of each cluster ? Perform a new data matrix with 10 rows (the 10 centers of the clusters) and the vector $(\mu_1, ..., \mu_{10})$ of the weights of the 10 centers.

3. Reconstruct the upper part of the tree from the cluster centers using the recommandation given in apprendix to deal with non uniform weights and the R function below.



Exercice 5. Combine k-means and Ward's minimum variance clustering.

1. Build now the Ward's minimum variance dendrogram starting from the $K = 10$ clusters obtained with the k-means method (choose **nstart=200**). In which particular case do you think that this methodology can be helpfull ?



2. Build now a partition in $K = 2$ clusters with the k-means method starting from the partition in two clusters of the Ward's dendrogram. Compare the proportion of variance explained by this partition with that of the partition by cutting the Ward's dendrogram. Was this result expected ?

```
prop_inert_cutree <- function(tree,K)
{
  #tree= Ward's minimum variance tree
  P <- cutree(tree,k=K)
  W <- sum(tree$height[1:(n-K)])
  Tot <- sum(tree$height)
  return(1-W/Tot)
}
```

Exercice 6. Clustering on the principal components of PCA.

Let X be a numerical data matrix of dimension $n \times p$. The clustering methods give usually the exact same results when applied

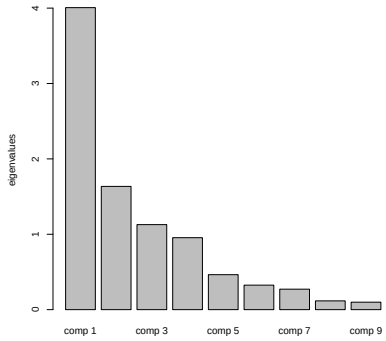
- to the standardized data matrix Z of dimension $n \times p$,
- to the matrix of all the principal components F of dimension $n \times r$ where r is the rank of the original

1. Check this result with the Ward method and the $n = 25$ european countries described in the data **protein** (weighted by $\frac{1}{n}$). More precisely compare the heights of the clusters in the Ward's dendrograms build with Z and F.

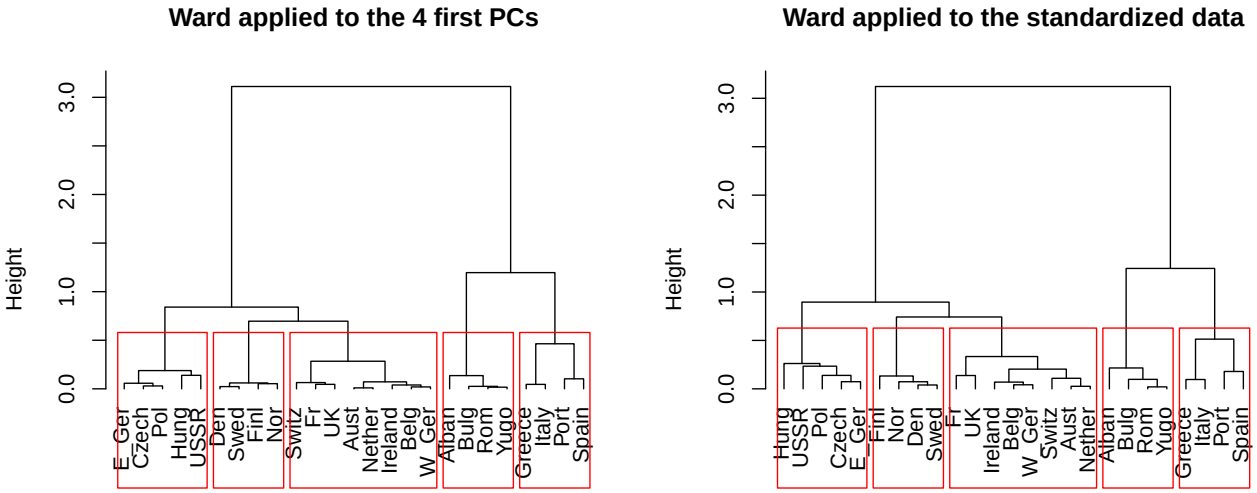
```
library(PCAmixdata)
data(protein)
```

```
all.equal(tree_F$height,tree_Z$height)
```

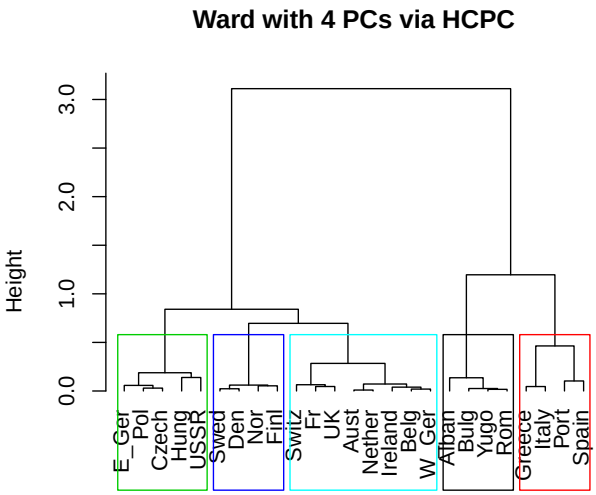
2. Choose now the number q of principal components that summarizes "well" the data. What is the proportion of the variance of the data explained with these q principal components ?

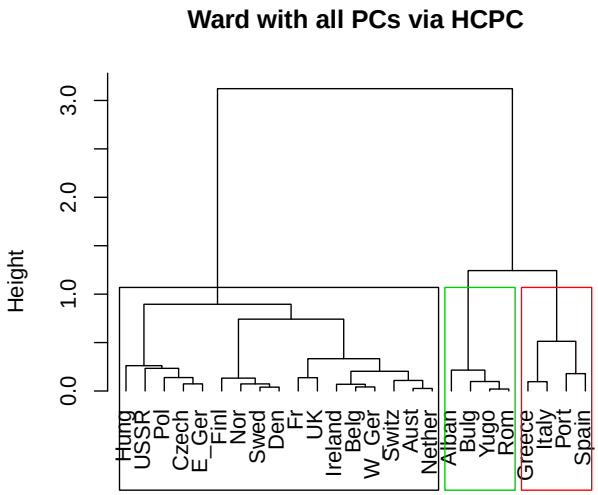


3. Build the Ward's dendrogram the q first principal components. Compare using the function **rect.hclust** the partition in 5 clusters obtained with this dendrogram and with the dendrogram built on all the PCs.



4. Same question but using the function **HCPC()** of the package **FactoMineR**.



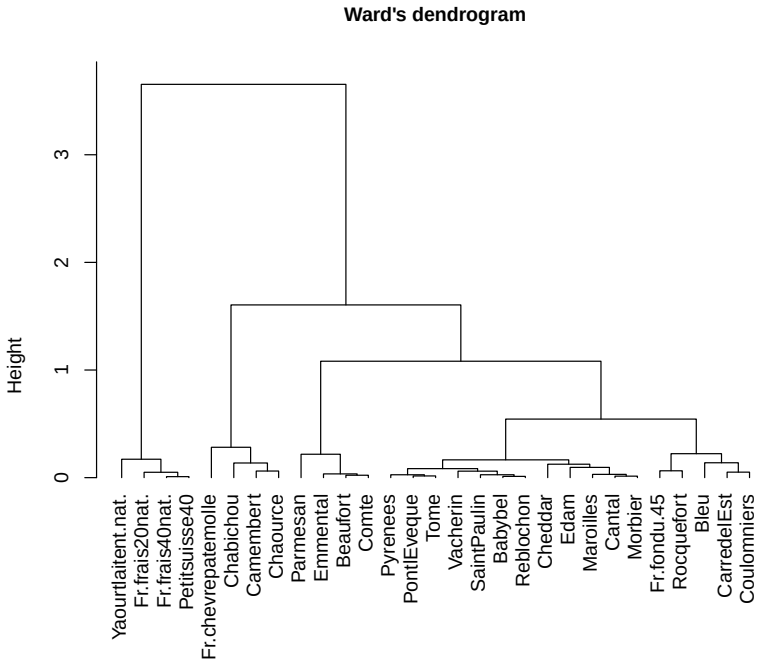



```
## [1] "HCPC"
```

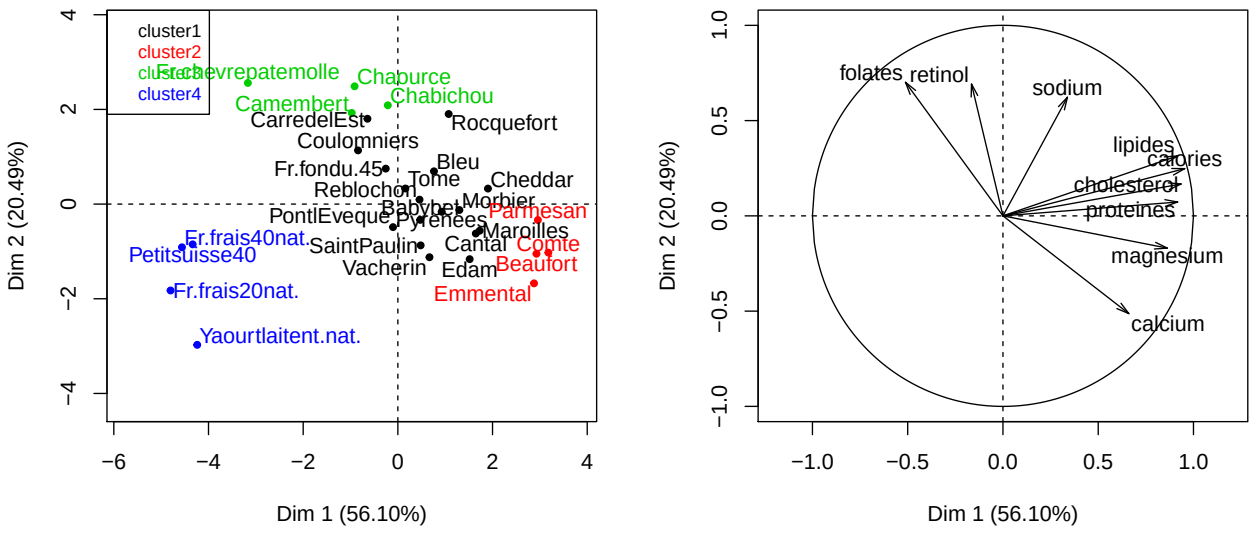
Exercice 7. Clustering numerical data: the cheeses.

The file "fromages.txt" describes $n = 29$ cheeses on $p = 9$ numerical variables.

1. Import this dataset in a data matrix X. Do you think these data shoud be scaled before clustering ?
2. Build the matrix Z of the scaled data. Apply the Ward's minimum variance method to the $n = 29$ cheeses described in Z and weighted by $\frac{1}{n}$. Check that the sum of the heights of the clusters in the hierarchy is equal to the total variance of the scaled data.
3. Plot the dendrogram and choose the number K of clusters that seems relevant to cut the tree.



5. Cut the tree and interpret the partition in K clusters using PCA (principal component analysis) via the package FactoMineR.



6. Confirm this interpretation using the R code below.

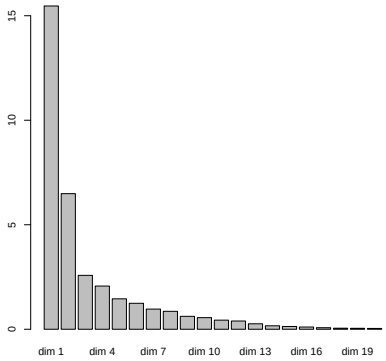
```
?catdes
res <- catdes(data.frame(part,X),num.var=1)
#cluster2
print(res$quanti$'cluster2'[,1:5],digits=2)
```

```
##             v.test Mean in category Overall mean sd in category Overall sd
## magnesium      3.6               46           27            3.6       11.1
## proteines      3.1               30           20            3.3        6.8
## calcium        2.8              281          186           44.0       71.3
## cholesterol    2.5              108           75           16.4       27.8
## calories       2.1              390          300           10.3       90.3
```

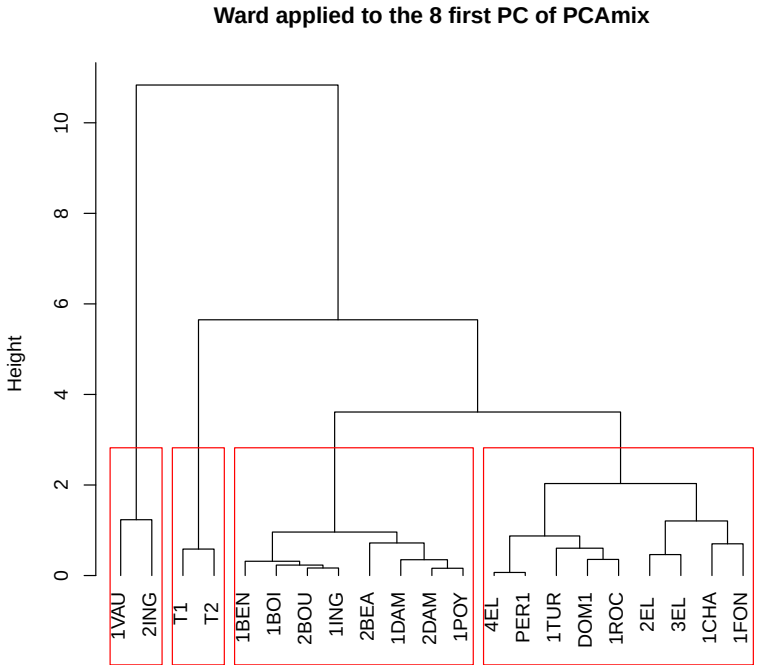
Exercice 8. Clustering mixed data: the wines.

1. The wines dataset describes $n = 21$ wines on a mixture of $p = 31$ numerical and categorical variables. How many variables are categorical and how many are numerical ? How many levels for each categorical variable ?

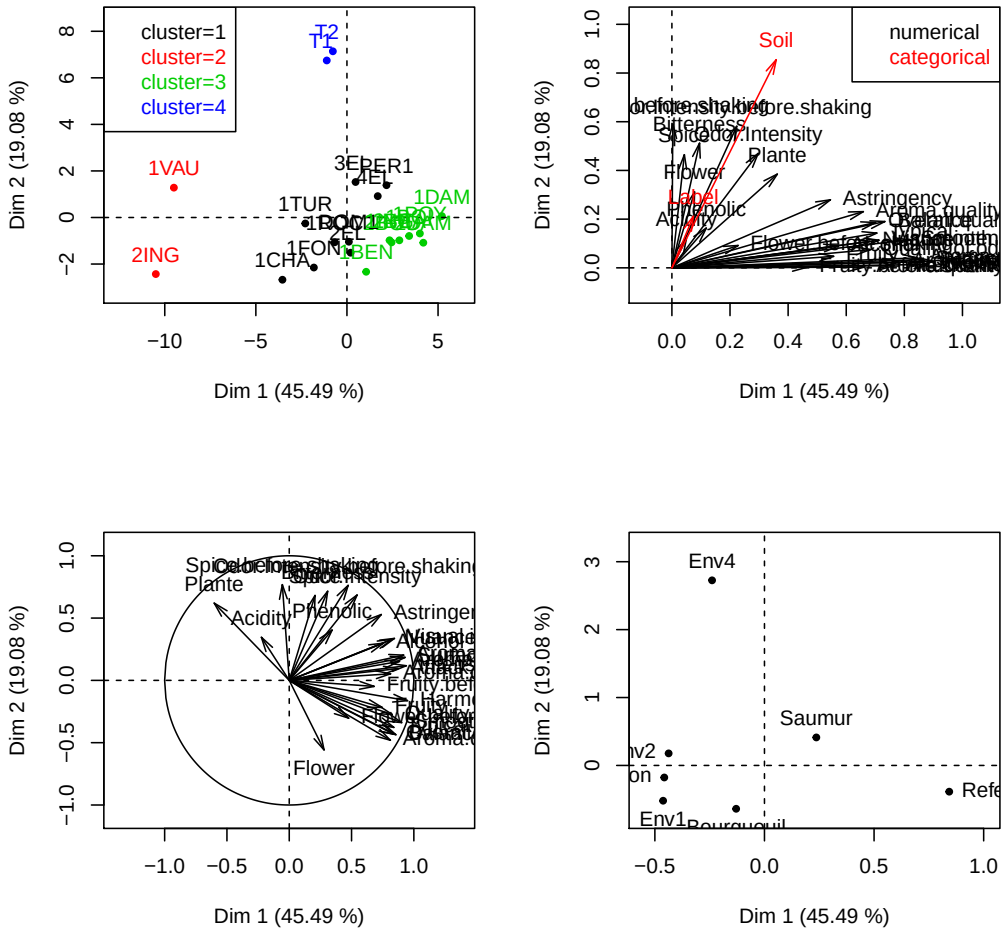
2. This dataset is first recoded into a numerical dataset using the funcion **PCAmix()** of the R package **PCAmixdata**. Choose the number q of principal components kept to built the matrix F of the q first PCs.



2. Build the Ward's minimum variance dendrogram on F and choose the number K of clusters that seems relevant to cut the tree.



3. Interpret the partition in K clusters via the graphics of **PCAmix**



4. Interpret now the cluster with the descriptive statistics provided by the function **catdes**.

```
res <- catdes( data.frame(cluster,wine),1)
#cluster1
res$category$`1`
res$quanti$`1`
```

Appendix

The R function **hclust()** implements the ascendant hierarchichal clustering algorithm using the Lance & Williams formula. The Ward's agregation measure $D(A,B) = \frac{\mu_A \mu_B}{\mu_A + \mu_B} d^2(g_A, g_B)$ is then used only in the initialisation step where the aggregation measures between the singletons of the partition P_n are performed and stored in the $n \times n$ matrix $\Delta = [\delta_{ij}]$ knowing that:

$$\delta_{ij} := D(\{i\},\{j\}) = \frac{w_i w_j}{w_i + w_j} d_{ij}^2.$$

When all the weights w_i are uniform (all equal to 1 or all equal to $\frac{1}{n}$ for instance) the function **hclust** implements the Ward's minimum variance algorithm with the following arguments:

- `method = "ward.D"`,
- `d` = Δ,

- `members = NULL`.

The argument **members=NULL** (by default) means that the weights of the individuals are considered as uniform. The argument **d** must be the matrix Δ of the *agregation measures* between the singletons. If all the individuals are weighted by $1/n$, the argument **d** must then be the matrix $\Delta = \frac{\mathbf{D}^2}{2n}$ where $\mathbf{D} = [d_{ij}]$ is the matrix of the Euclidean distance between the individuals. The R code is then:

```
> D <- dist(X)
> tree <- hclust(D^2/(2*n),method="ward.D")
```

If all the individuals are weighted by 1, the argument **d** must be the matrix $\Delta = \frac{\mathbf{D}^2}{2}$.

When the weights w_i are non uniform the function **hclust** implements the Ward's minimum variance algorithm with the following arguments:

- `method = "ward.D"`,
- `d` = Δ,
- `members = w`.

The argument **members=w** with **w**! =NULL means that the weights w_i of the individuals are non uniform. The argument $d = \Delta$ is then more complicated to perform. For instance the following R code can be used:

```
> Delta <-  D
> for (i in 1:(n-1)) {
    for (j in (i+1):n) {
      Delta[n*(i-1) - i*(i-1)/2 + j-i] <-
        Delta[n*(i-1) - i*(i-1)/2 + j-i]^2*w[i]*w[j]/(w[i]+w[j])}}
> tree <- hclust(Delta,method="ward.D",members=w)
```