TP3

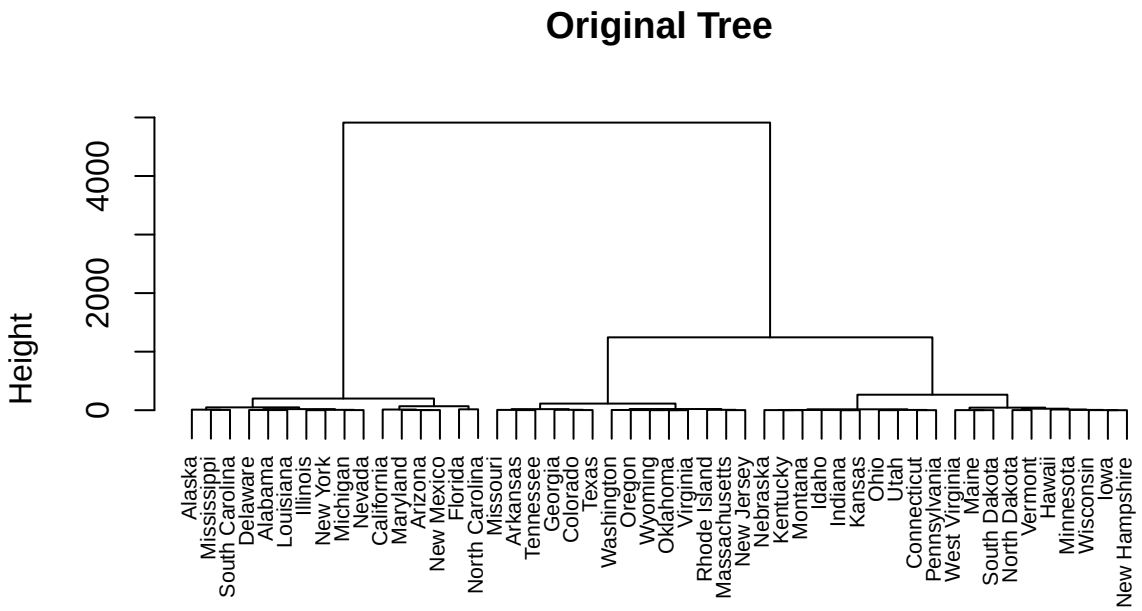
Yiye JIANG

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

4.1

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

```
n <- dim(USArrests)[1]
D <- dist(USArrests)
tree <- hclust(D^2/(2*n),method = "ward.D")
plot(tree,main="Original Tree",cex = 0.7,xlab = "",sub = "")
```



4.2

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.

```
P10 <- cutree(tree,k = 10)
cent <- matrix(NA,10,4)
w <- rep(NA,10)
```

```r
for (i in 1:10) {
  cent[i,] <- apply(USArrests[which(P10==i),],2,mean)
  w[i] <- nrow(USArrests[which(P10==i),])/n
}
names(w) <- paste0("w",1:10)
w #The weight of each cluster
```

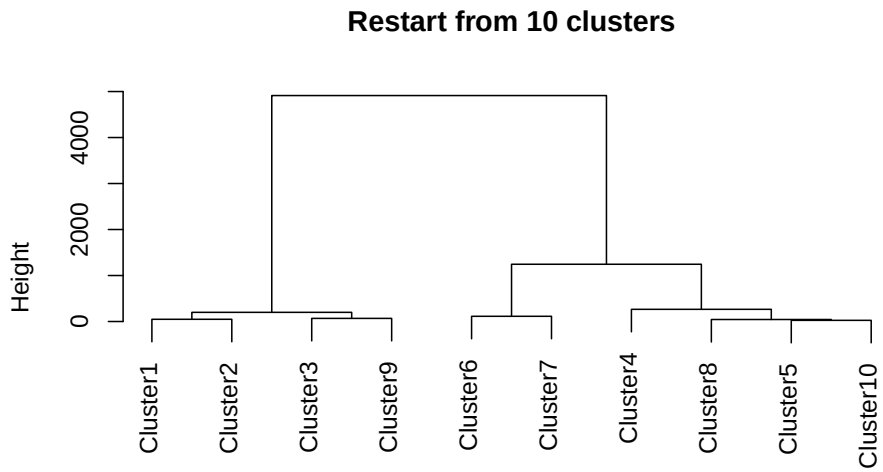
```
##   w1   w2   w3   w4   w5   w6   w7   w8   w9  w10
## 0.14 0.06 0.08 0.12 0.20 0.04 0.10 0.06 0.16 0.04
```

4.3

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.

```r
Delta <- dist(cent)
n=10
 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])}}

tree2 <- hclust(Delta,method="ward.D",members=w)
plot(tree2,main="Restart from 10 clusters",labels = paste0("Cluster",tree2$order),xlab = "",sub = "")
```



Remark: The same as the figure in Question 4.1 and it's just the zoom out.

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

5.1

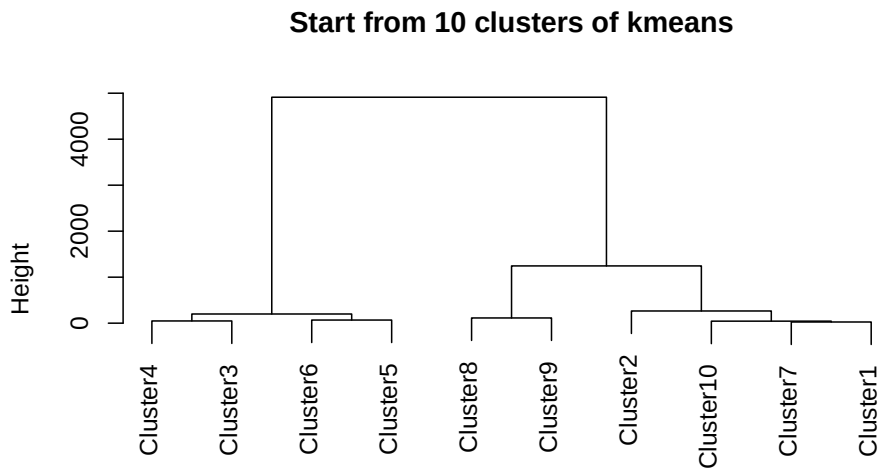
1. Build now the Ward' minimum variance dendrogram starting from the $K = 10$ clusters obtained with the k-means method (choose nstart=200). In which particular case do you think that this methodology can be helpful?

```
rm(list = ls())
mod <- kmeans(x=USArrests,centers = 10,nstart = 200)

cent <- mod$centers
w <- mod$size/dim(USArrests)[1]

Delta <- dist(cent)
n=10
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])}}

tree3 <- hclust(Delta,method="ward.D",members=w)
plot(tree3,main = "Start from 10 clusters of kmeans",labels = paste0("Cluster",tree3$order),xlab = "",su
```



Ans: From the Model Complexity's point of view, the complexity of *k-means* is *o(KpnT)*, where n is the sample size. Because it is the linear function of #observations, it is very suitable to deal with big data, and be used in the pre-processing to reduce the quantity of data involved.

Because, there is some randomness in the algorithm. So, its normal to get different outputs for different runs.

5.2

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

```
n <- dim(USArrests)[1]
D <- dist(USArrests)
tree <- hclust(D^2/(2*n),method = "ward.D")

P2 <- cutree(tree,k = 2)

cent <- matrix(NA,2,4)
for (i in 1:2) {
  cent[i,] <- apply(USArrests[which(P2==i),],2,mean)
}

mod2 <- kmeans(USArrests,centers = cent,nstart = 200)
mod2$betweenss/mod2$totss #The proportion of variance explained of kmeans is 72.9%
```

```
## [1] 0.72907
```

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

prop_inert_cutree(tree,2)
```

```
## [1] 0.6903036
```

Ans: Yes, because the criteria to be minimized for kmean is exactly within inertia, so the final within inertia will be at least not smaller than the initial within inertia, which indicates the final proportion of variance explained will not be smaller than the initial one.

Exercice 6. Clustering on the principal components of PCA.

6.1

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' dendrograms build with Z and F.*

```
rm(list = ls())

library(PCAmixdata)
data(protein)
Z <- scale(protein)

n <- dim(Z)[1]
D <- dist(Z)

tree_Z <- hclust(D^2/(2*n),method = "ward.D")

pca <- prcomp(Z)
F <- pca$x
D <- dist(F)
tree_F <- hclust(D^2/(2*n),method = "ward.D")

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

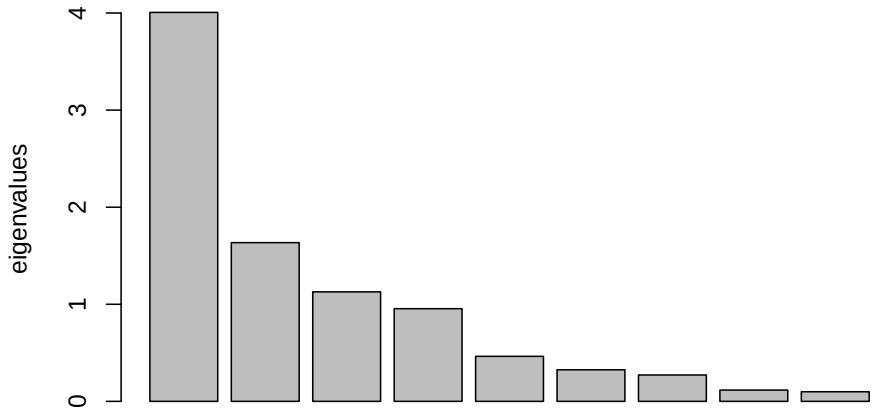
```
## [1] TRUE
```

So, the proposition has been verified.

6.2

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?

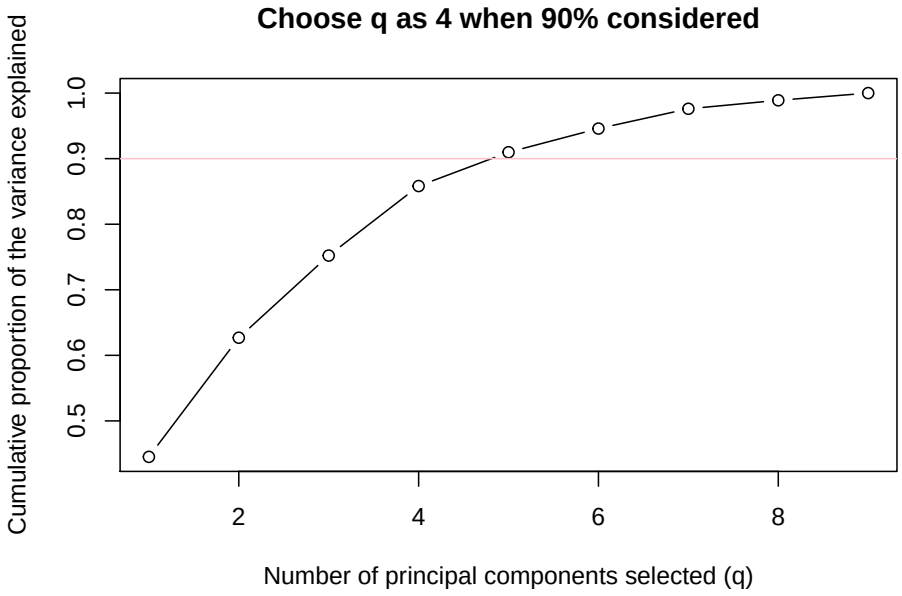
```
eigv <- pca$sdev^2
barplot(eigv,ylab = "eigenvalues")
```




```
varexp <- eigv/sum(eigv)
cvarexp <- varexp
```

```r
for (i in 2:length(eigv)) {
  cvarexp[i] <- cvarexp[i-1] + varexp[i]
}
names(cvarexp) <- paste0("q = ",1:length(eigv))

xlab <- "Number of principal components selected (q)"
ylab <- "Cumulative proportion of the variance explained"
main <- "Choose q as 4 when 90% considered"
plot(cvarexp,type = "b",xlab = xlab ,ylab = ylab, main = main)
abline(h = 0.9, col = 'pink')
```




```r
print(cvarexp , digits = 4)
```

```
##  q = 1  q = 2  q = 3  q = 4  q = 5  q = 6  q = 7  q = 8  q = 9
## 0.4452 0.6268 0.7522 0.8582 0.9098 0.9459 0.9761 0.9890 1.0000
```

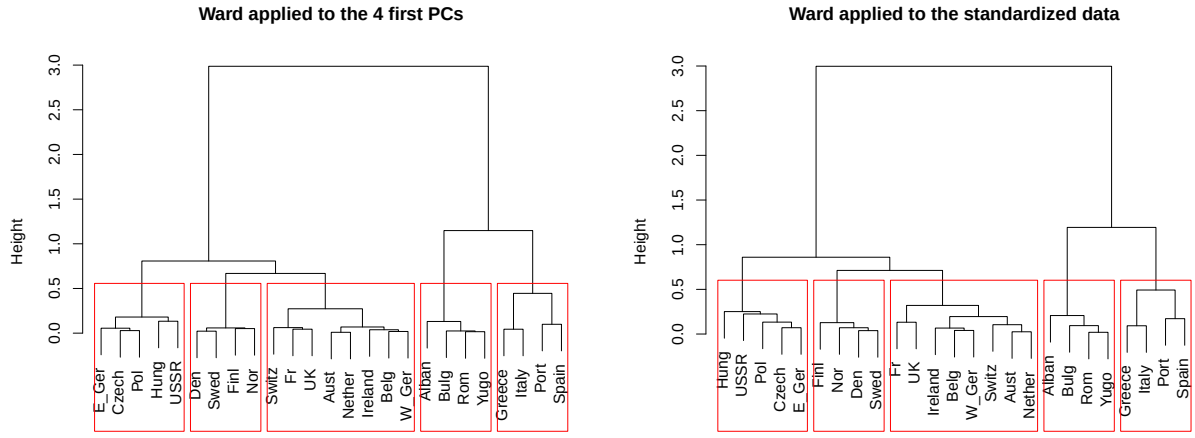
So, to "well" summarize the data, We can consider a large proportion of explained varianve, says 90%. When we want to keep 90% of the information, it leads to remain first 4 principal components. And the proportion of the variance of the data explained by these 4 principal components is 85.82%.

6.3

3. Build the Ward' 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.

```r
D <- dist(F[,1:4])
tree_pca4 <- hclust(D^2/(2*n),method = "ward.D")
plot(tree_pca4,main = "Ward applied to the 4 first PCs",xlab = "",sub = "")
rect.hclust(tree_pca4,k = 5)

plot(tree_F,main = "Ward applied to the standardized data",xlab = "",sub = "")
rect.hclust(tree_F,k = 5)
```



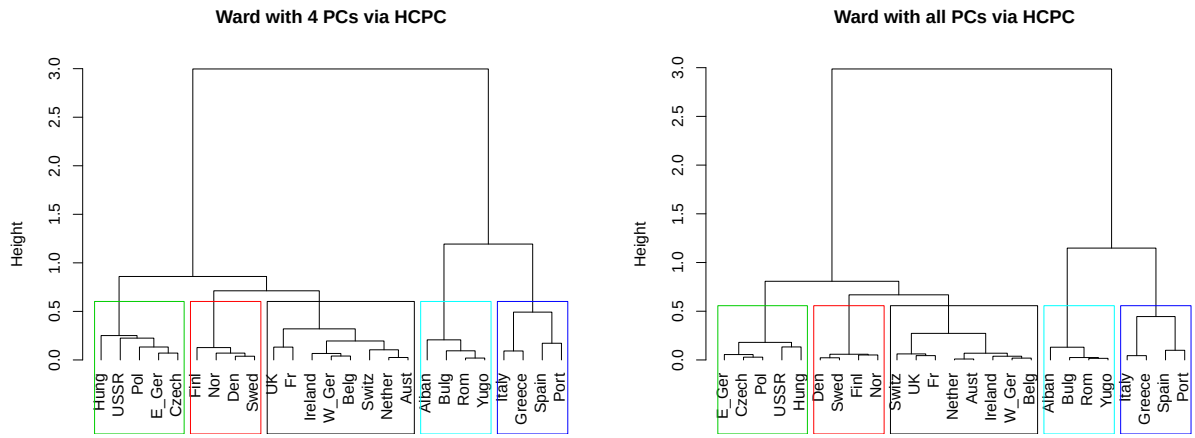
We can find that the partitions in 5 clusters obtained by two dendrograms are completely the same, only the hights are different. Because, we did't use all information contained in the data when constructing the tree_pca4. This will affect the inertia, meanwhile affect the sum of the first n-k heights corresponding to the within inertias. We got different heights, but the partitions have not been affected.

6.4

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

```
library(FactoMineR)
hcpc_all <- HCPC(as.data.frame(F),nb.clust = 5, graph = FALSE)
hcpc_4 <- HCPC(as.data.frame(F[,1:4]),nb.clust = 5, graph = FALSE)

plot(hcpc_all,choice = "tree",tree.barplot = FALSE, main = "Ward with 4 PCs via HCPC")
plot(hcpc_4,choice = "tree",tree.barplot = FALSE, main = "Ward with all PCs via HCPC")
```



The partitions are still the same.

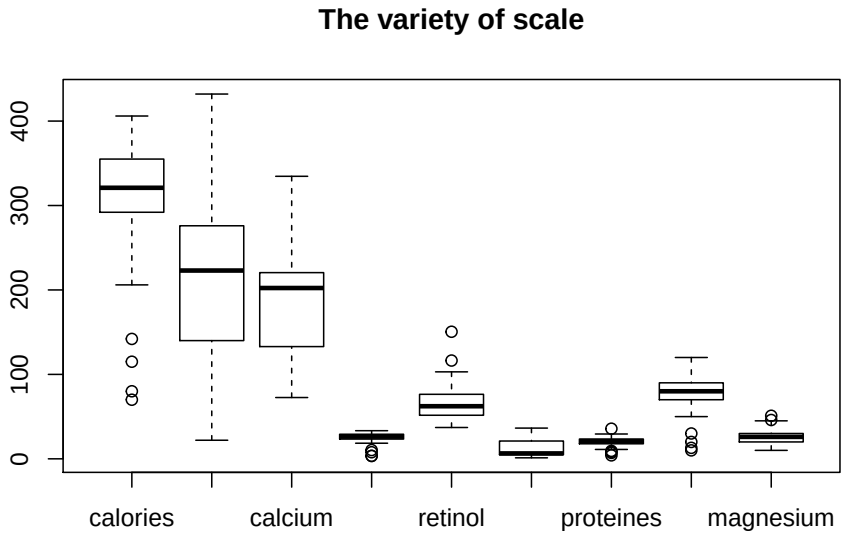
Exercice 7. Clustering numerical data: the cheeses.

7.1

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?

```
rm(list = ls())
X <- read.table("fromage.txt",header = T,row.names = 1)
boxplot(X,main="The variety of scale")
```



From the above boxplot, we can see that some boxes have been compressed to almost lines. This arises from the non-uniformed scales of variables. And if we don't scale all variables to the same levels, the one with large scale will dominate in distance. So, before we start analysis formally, all variables need to be scaled.

7.2

2. Build the matrix Z of the scaled data. Apply the Ward' 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.

```
Z <- scale(X)

n <- dim(Z)[1]
D <- dist(Z)

tree_Z <- hclust(D^2/(2*n),method = "ward.D")

#sum of the heights of the clusters
sum(tree_Z$height)
```

```
## [1] 8.689655
```

```r
#Caculate the total variance of the scaled data
gcent <- apply(Z, 2, mean)
Totv <- sum((Z-gcent)^2)/n

#Check that the sum of the heights of the clusters in the hierarchy is equal to
#the total variance of the scaled data
Totv == sum(tree_Z$height)
```

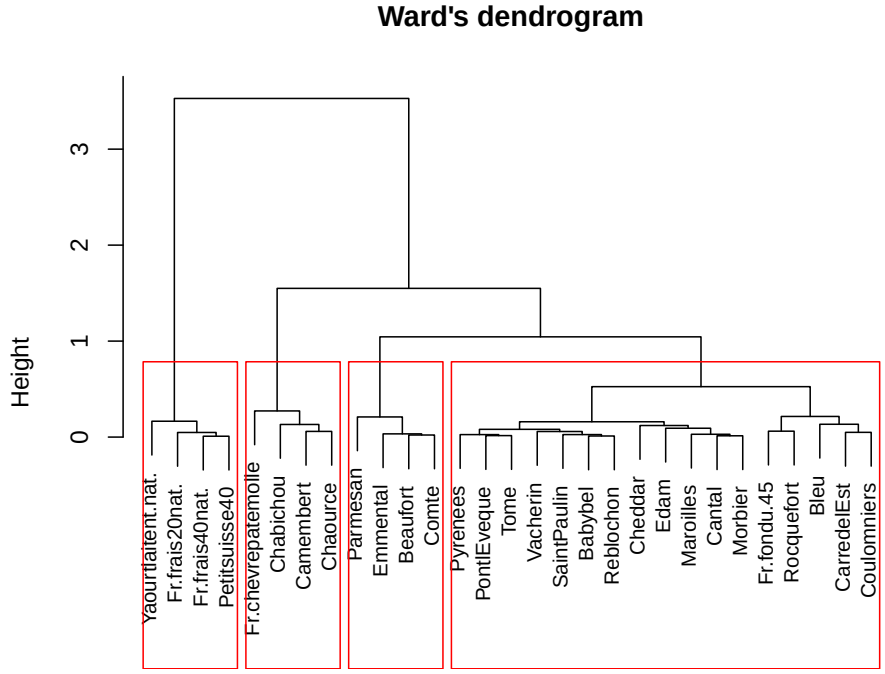
```
## [1] TRUE
```

So, the proposition has been verified.

7.3

3. Plot the dendrogram and choose the number K of clusters that seems relevant to cut the tree.

```r
plot(tree_Z,main = "Ward's dendrogram", cex = 0.8,xlab = "",sub = "")
rect.hclust(tree_Z,k = 4)
```




```r
P4 <- cutree(tree_Z,k = 4)
```

We want to facilitate the future interpretation, and there are 4 quadrants in a map, **so, its suitable to cut the tree into 4 clusters.** In this way, if the PCA is implemented with good quality, the individuals will cluster and fall into 4 quadrants, with each quadrant standing for a individual cluster.

7.4

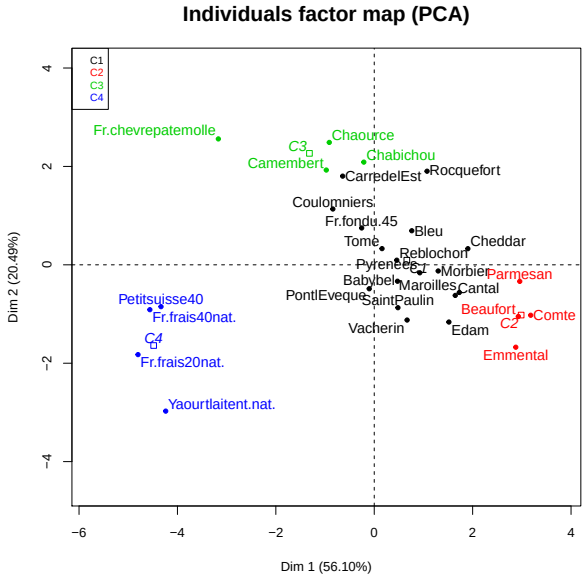
4. Cut the tree and interpret the partition in K clusters using PCA (principal component analysis) via the package FactoMineR.

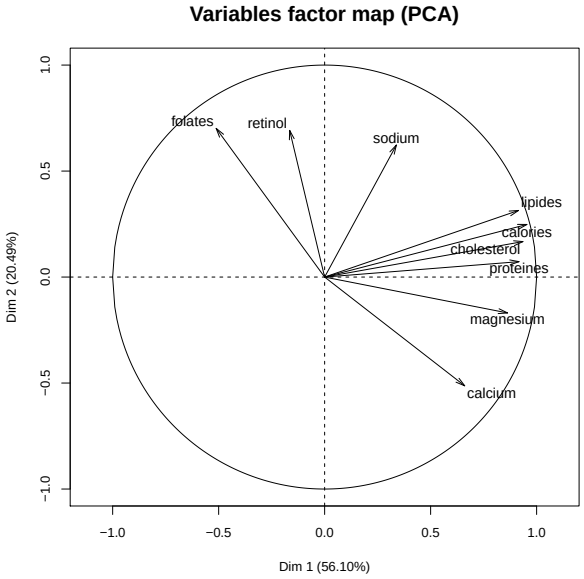
```
P4 <- as.factor(P4)
levels(P4) <- paste0("C",1:4)

res <- PCA(data.frame(P4,Z),quali.sup = 1,graph = FALSE)
print(res$eig,digits = 4)
```

```
##        eigenvalue percentage of variance cumulative percentage of variance
## comp 1   5.049114              56.10126                             56.10
## comp 2   1.844418              20.49354                             76.59
## comp 3   0.867801               9.64223                             86.24
## comp 4   0.577385               6.41539                             92.65
## comp 5   0.355210               3.94678                             96.60
## comp 6   0.175312               1.94791                             98.55
## comp 7   0.097434               1.08260                             99.63
## comp 8   0.028431               0.31590                             99.95
## comp 9   0.004895               0.05439                            100.00
```

```
par(cex.lab=0.9,cex=0.9,adj=0.5,cex.axis=0.9,cex.main=1.5)
plot(res,1:2,choix = "ind",habillage = 1)
plot(res,1:2,choix = "var")
```





Interpretation

- From the right figure(Variables map),most variables are well-projected onto plane(1,2), especially calories, calcium, lipides, folates, proteines, cholesterol, magnesium, retinol.
  - **proteines, cholesterol, magnesium, calories** contribute most to the construction of the 1st dimension.
  - **retinol, folates** contributes most to the 2nd dimension.
- The interpretation of the right figure can help to interpret the left figure. Because we had used the scaled data Z to calculate, so all variables can be compared at the same level, as well as the principal components. So, in the first figure, 0 at each axis stands for the average amount of content, while

positive axis stands for excess amount(denoted by *much* in the table below) with negative axis for deficient amount(denoted by *little* in the table below). Cheese clusters differ in the amounts of various contents, details in the table below.

		pro, cho, mag, cal	
\	dim2\dim1	much	little
retinol,	much	C3	C3
folates	little	C2	C4

As for **cluster 1**, each ingredient accounts for a balanced proportion in the cheeses in this class. Because the first two principal componants can explain 76.6% of the total information.

So, with the help of PCA, we can know how the original variables affect the clustering.

7.5

5. Confirm this interpretation using the R code below.

```
cheese <- rownames(Z)
cheese[which(P4=="C3")]
```

```
## [1] "Camembert"          "Chabichou"          "Chaource"
## [4] "Fr.chevrepatemolle"
```

```
res2 <- catdes(data.frame(P4,Z),num.var=1)

#cluster3
print(res2$quanti$'C3',digits=4)
```

```
##         v.test Mean in category Overall mean sd in category Overall sd
## retinol  4.229            1.963    1.675e-17         0.9315     0.9826
## folates  3.866            1.795   -6.221e-17         0.2014     0.9826
##           p.value
## retinol 2.348e-05
## folates 1.107e-04
```

Firstly, lets look at Cluster2.

As we alrady known from the previous figures, the cheeses in Cluster2 all have large amount of contents associated to dim2, while those to dim1 make no difference. So the contributors of dim2 should be the ones charactize this cluster.

The function has listed out the original variables which charactize(p-value is smaller than 0.05) the Cluster3, says **retinol(p-value is 2.348e-05), folates(p-value is 1.107e-04)**. They are exactly the same as what I expected. So, my interpretation has been confirmed.

And if we look at *Mean in category, Overall mean*, we can find how much difference there is between these two statistics, for each variable. For example, the mean of **retinol** in cluster3 is 2.0, much bigger than the overall one(1.7e-17$\approx$ 0). This verifies one more time that **retinol** plays a important role in clustering cluster3, and its average amount in cluster3 is greater than its overall amount. And also the average level(1.8) of **folates** is much higher than its overall one(-6.2e-17$\approx$ 0). The high levels in these 2 contributors mean that members in cluster3 also have high values of 2nd PC, which makes the could of individuals in cluster3 located at the far side of the 2nd axis.

Now, lets look at Cluster2 and Cluster4.

From the figure before, we have also known that cluster2 corrsponds to much amount of contents associated to dim1, while little to dim2. So, I suppose that the contributors of dim1 and dim2 should also charactize this cluster.

```
#cluster2
print(res2$quanti$'C2',digits=4)
```

```
##             v.test Mean in category Overall mean sd in category Overall sd
## magnesium    3.575           1.6596    7.549e-17         0.3147     0.9826
## proteines    3.112           1.4449    1.327e-16         0.4811     0.9826
## calcium      2.820           1.3090    4.570e-17         0.6065     0.9826
## cholesterol  2.510           1.1653   -1.838e-16         0.5804     0.9826
## calories     2.103           0.9761   -8.777e-17         0.1124     0.9826
##                p.value
## magnesium   0.0003503
## proteines   0.0018560
## calcium     0.0048072
## cholesterol 0.0120711
## calories    0.0355064
```

From the above results, one can find that the main variables listed are the contributors to dim1(says,**proteines, cholesterol, magnesium, calories** and also **calcium**), which is different from what I supposed before. I propose two reason which may result in this diffrence.

- First, the sample size is not large enough, so we found all individuals in cluster2 to appear in 4th quadrant. The limited samples may lead to this bias.
- Estimates from figures are always qualitative, so it can be a little different from the quantitive ones.

```
#cluster4
print(res2$quanti$'C4',digits=4)
```

```
##             v.test Mean in category Overall mean sd in category Overall sd
## magnesium   -2.991          -1.388    7.549e-17         0.1448     0.9826
## sodium      -3.277          -1.521    8.638e-17         0.2545     0.9826
## proteines   -4.014          -1.863    1.327e-16         0.2847     0.9826
## cholesterol -4.296          -1.995   -1.838e-16         0.2724     0.9826
## calories    -4.647          -2.157   -8.777e-17         0.3114     0.9826
## lipides     -4.739          -2.200   -8.207e-17         0.3655     0.9826
##                p.value
## magnesium   2.781e-03
## sodium      1.049e-03
## proteines   5.972e-05
## cholesterol 1.737e-05
## calories    3.369e-06
## lipides     2.153e-06
```

There are some contributors of dim1, together with ones of dim2 here. So, it fits in with what I expected before.

```
#cluster1
print(res2$quanti$'C1',digits=4)
```

```
##          v.test Mean in category Overall mean sd in category Overall sd
## sodium    3.109           0.4850    8.638e-17         0.7270     0.9826
## lipides   2.361           0.3683   -8.207e-17         0.3324     0.9826
## calories  2.168           0.3382   -8.777e-17         0.3212     0.9826
## retinol  -1.981          -0.3091    1.675e-17         0.5020     0.9826
```

```
##          p.value
## sodium   0.00188
## lipides  0.01824
## calories 0.03019
## retinol  0.04759
```

Cluster1 doesn't have distinctive features. But after we get other clusters, the left one is cluster1.

Exercice 8. Clustering mixed data: the wines

8.1

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?

```
rm(list = ls())
library(PCAmixdata)
data(wine)
sum_wine <- summary(wine)
sum_wine[,1:5]
```

```
##         Label           Soil   Odor.Intensity.before.shaking
##  Saumur    :11   Reference:7   Min.   :2.643
##  Bourgueuil: 6   Env1     :7   1st Qu.:2.893
##  Chinon    : 4   Env2     :5   Median :3.071
##                  Env4     :2   Mean   :3.111
##                                3rd Qu.:3.214
##                                Max.   :3.708
##  Aroma.quality.before.shaking Fruity.before.shaking
##  Min.   :2.593                Min.   :2.375
##  1st Qu.:2.926                1st Qu.:2.560
##  Median :3.107                Median :2.731
##  Mean   :3.046                Mean   :2.714
##  3rd Qu.:3.179                3rd Qu.:2.833
##  Max.   :3.429                Max.   :3.154
```

```
ncol(sum_wine) #the number of variables
```

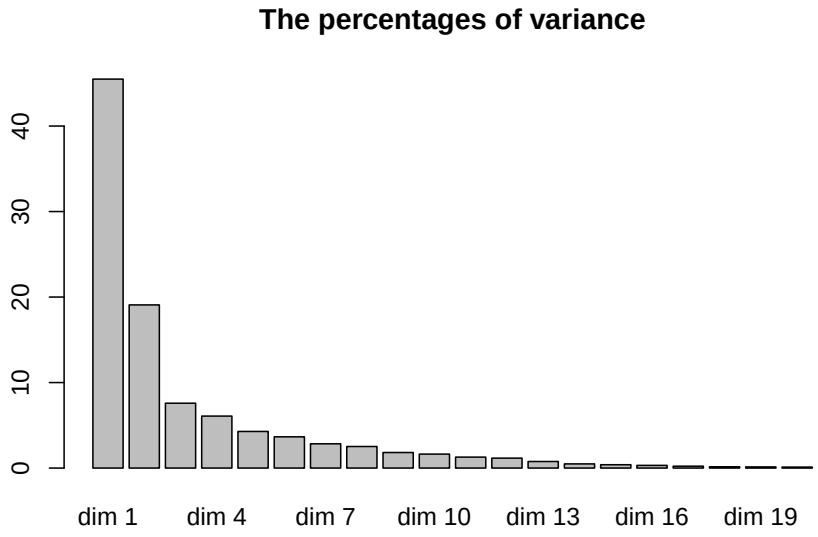
```
## [1] 31
```

From the above summary, its easy to find that there are 2 categorical variables in this dataset, while 29 numerical variables. The 2 categorical variables are **Label, Soil**, with 3, 4 levels, respectively.

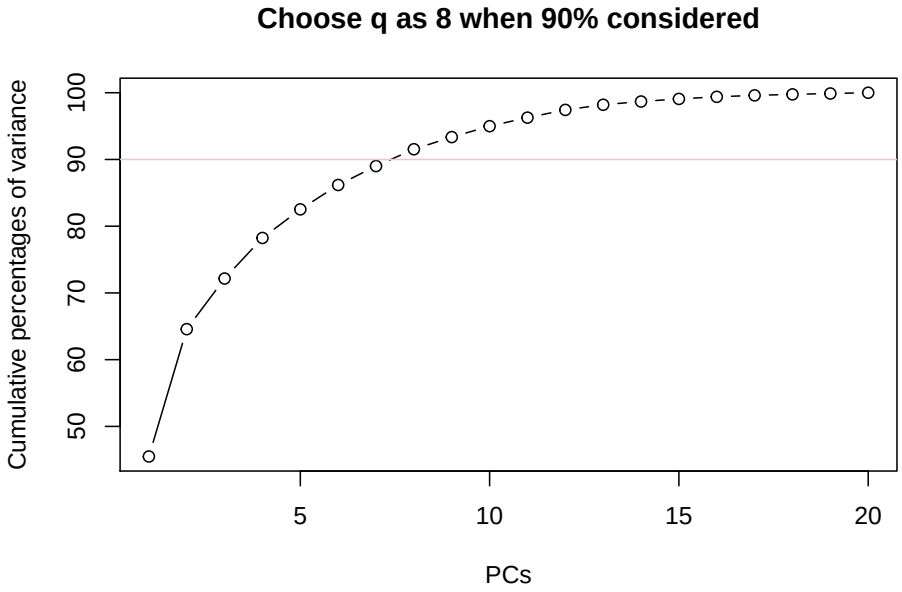
8.2

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.

```
pca_wine <- PCAmix(scale(wine[,-c(1,2)]),wine[,1:2],ndim = 36,rename.level = T,graph = FALSE)
barplot(pca_wine$eig[,2],main = "The percentages of variance")
```




```
main <- "Choose q as 8 when 90% considered"
ylab <- "Cumulative percentages of variance"
plot(pca_wine$eig[,3],main = main,type = 'b',xlab = "PCs",ylab = ylab)
abline(h = 90,col = "pink")
```




```
F <- pca_wine$ind$coord[,1:8]
```

In this case, PCA is used to recode the mixed dataset into a fully numerical dataset, rather than reduce dimension, so here we need to keep most information for following analysis. We can consider a large proportion of explained varianve, says 90%. When we want to keep 90% of the information, it leads to remain first 8 principal component.

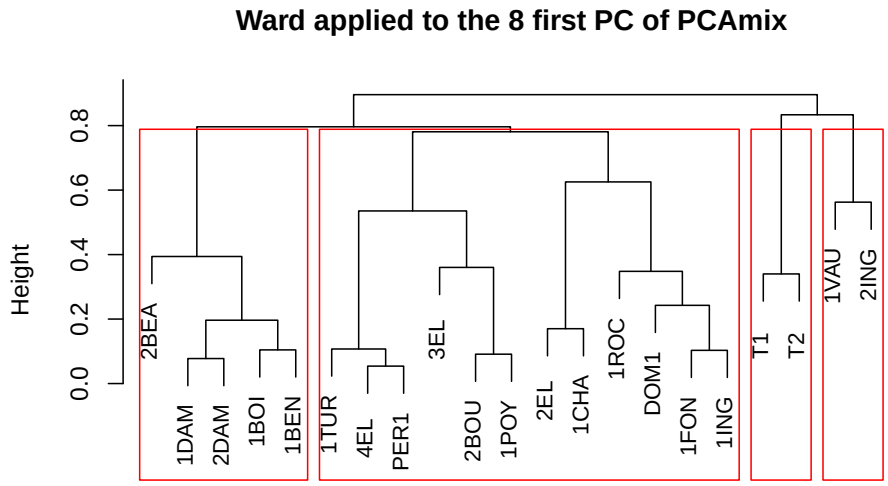
8.3

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

```
Z <- scale(F)

n <- dim(Z)[1]
D <- dist(Z)
tree_Z <- hclust(D^2/(2*n),method = "ward.D")

plot(tree_Z,main = "Ward applied to the 8 first PC of PCAmix", xlab = "", cex = 0.9,sub = "")
rect.hclust(tree_Z,k = 4)
```



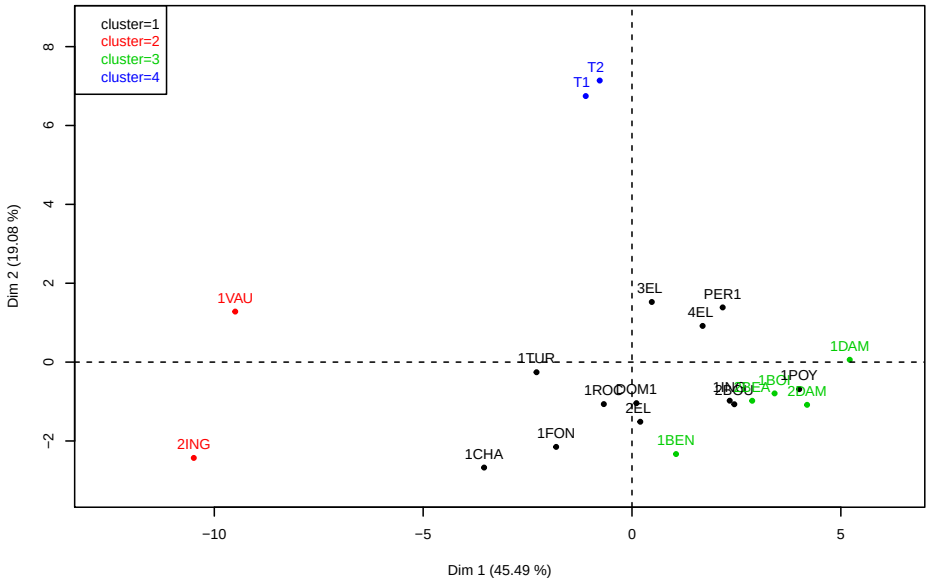

```
cluster <- cutree(tree_Z,k = 4)
```

With the same reason as the one in Exercise 7 Question 4, here we still choose k as 4 for better interpretation.

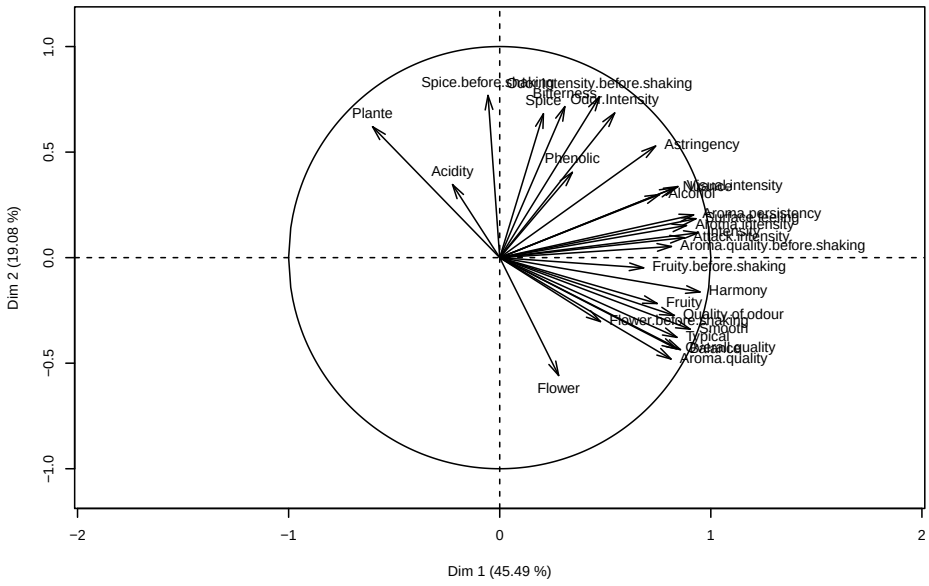
8.4

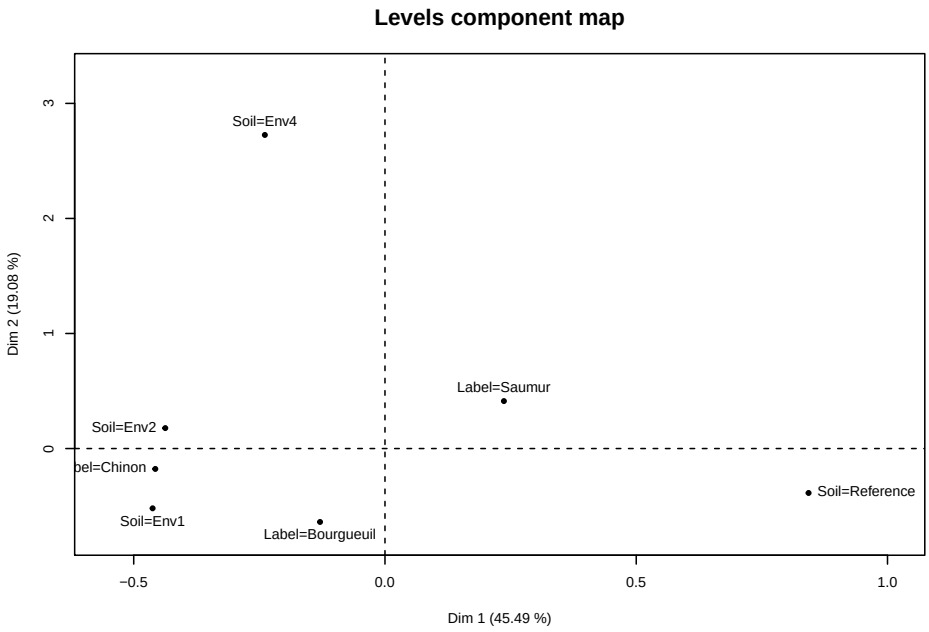
4. Interpret the partition in K clusters via the graphics of PCAmix

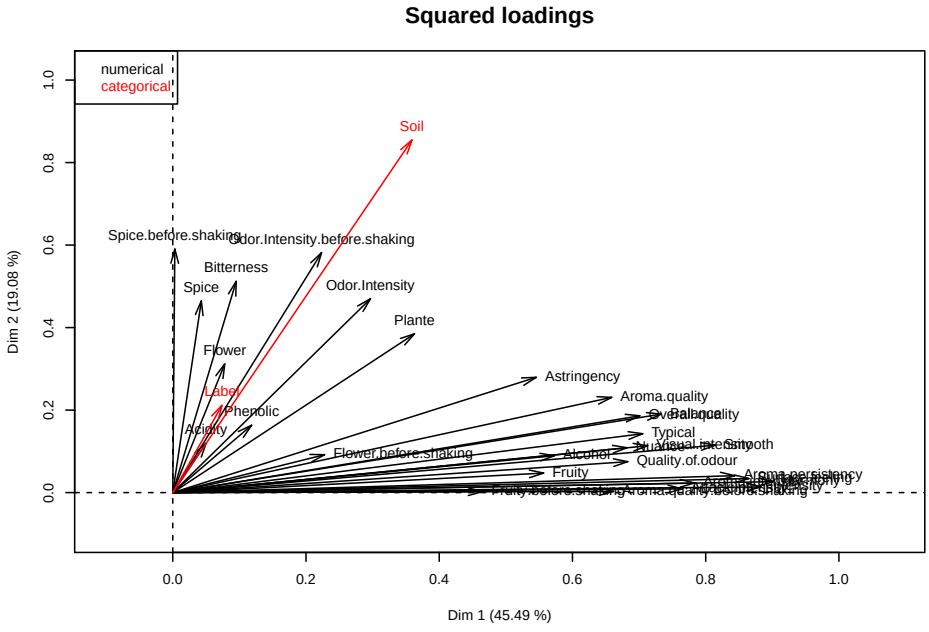
Individuals component map



Correlation circle







Interpretation

- From the aspect of contribution of dimensions:
  - For dimsension1, we can find that its main contributors are **Aroma.quality.before.shaking, Aroma.intensity, Aroma.persistency, Attack.intensity, Intensity, Surface.feeling, Harmony**(*numerical variables, judge from Correlation Circle*) and level **Soil=Reference**(*judge from Levels component map*).
  - For dimsension2, we can find that its main contributors are **Spice.before.shaking, Odor.Intensity.before.shaking, Spice, Bitterness, Odor.Intensity, Flower**(*numerical variables, judge from Correlation Circle*) and level **Soil=Env4**(*judge from Levels component map*).

- After finding out the main contributors of dim1 and 2, we can interpret the clusters' characteristics.
  - Contributors of dim1 can be used to distinguish the members in cluster3, and cluster3. Wines in cluster2 have relatively high sensory descriptors related to dim1(like,**Aroma.quality.before.shaking, Aroma.intensity**), while those in cluster2 are relatively low.
  - Members of cluster4 have high sensory descriptors associated to dim2(like,**Spice.before.shaking, Odor.Intensity.before.shaking**), while low ones to dim1.
  - cluster1 doesn't have distinctive features regarding to the first 2 dimensions.

dim2\dim1	high	low
high	C3	C2,C4
low	C3	C2

8.5

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

Firstly, lets look at cluster 3.

```
res <- catdes(data.frame(cluster,wine),num.var = 1)

#cluster3
res$category$`3`
```

```
##                  Cla/Mod Mod/Cla   Global     p.value   v.test
## Soil=Reference 71.42857     100 33.33333 0.001031992 3.281657
```

```
print(res$quanti$`3`,digits = 4)
```

```
##                               v.test Mean in category Overall mean
## Fruity.before.shaking          3.091            2.957        2.714
## Fruity                         2.833            3.092        2.847
## Aroma.quality.before.shaking   2.656            3.259        3.046
## Aroma.quality                  2.367            3.356        3.064
## Aroma.persistency              2.336            3.207        2.983
## Balance                        2.213            3.416        3.129
## Quality.of.odour               2.077            3.418        3.237
## Smooth                         2.016            3.000        2.674
## Overall.quality                1.964            3.662        3.331
##                               sd in category Overall sd  p.value
## Fruity.before.shaking                0.13289     0.1962 0.001995
## Fruity                               0.08168     0.2163 0.004612
## Aroma.quality.before.shaking         0.09359     0.2007 0.007898
## Aroma.quality                        0.10509     0.3084 0.017930
## Aroma.persistency                    0.12632     0.2396 0.019500
## Balance                              0.07427     0.3246 0.026907
## Quality.of.odour                     0.07607     0.2177 0.037822
## Smooth                               0.16467     0.4040 0.043820
## Overall.quality                      0.09038     0.4216 0.049498
```

We can find that **Soil=Reference** does show difference in cluster3, with p-value = 0.001032. From ratio *Mod/Cla* = 100, we can know that all individuals at level **Soil=Reference** fall into the cluster3. And 71.43% of the individuals in cluster3 have this level. So, this level characterizes cluster3 very much.

As for the numerical describers, they are basically the same as what I expected before. And all the *Mean in*

*category*s are higher than their *Overall mean*. Ffor example, **Aroma.quality.before.shaking** , its *Mean in category* is 3.2590, higher than *Overall Mean (3.046)*. So, it means cluster3 is the one with high value of the 1st PC, amounting to with high values of 1st PC contributors, which made it located at the positive part of axis-1.

On contrary, cluster2 is supposed to have low values in these contributors.

```
#cluster2
res$category$`2`
```

```
## NULL
```

```
print(res$quanti$`2`[1:5,],digits = 4)
```

```
##                                v.test Mean in category Overall mean
## Plante                          2.070            2.192        1.964
## Acidity                         2.040            2.715        2.385
## Bitterness                     -1.969            1.821        2.068
## Odor.Intensity.before.shaking -1.981            2.725        3.111
## Aroma.quality                  -2.572            2.517        3.064
##                                sd in category Overall sd p.value
## Plante                                 0.1120     0.1596 0.03850
## Acidity                                0.4645     0.2341 0.04131
## Bitterness                             0.1420     0.1819 0.04891
## Odor.Intensity.before.shaking          0.0825     0.2825 0.04761
## Aroma.quality                          0.0390     0.3084 0.01011
```

For **Aroma.quality.before.shaking**, its *Mean in category* is 2.689, lower than *Overall Mean* (3.046). The same for most of other axis-1 contributors. Inevitably, there are some differences in this quantitative results from the qualitative estimations we got before.

```
#cluster4
res$category$`4`
```

```
##           Cla/Mod Mod/Cla  Global     p.value   v.test
## Soil=Env4     100     100 9.52381 0.004761905 2.822714
```

```
print(res$quanti$`4`,digits = 4)
```

```
##                                v.test Mean in category Overall mean
## Spice.before.shaking            3.316            2.526        1.991
## Odor.Intensity.before.shaking   3.035            3.702        3.111
## Bitterness                      3.004            2.444        2.068
## Odor.Intensity                  2.497            3.732        3.395
## Spice                           2.494            2.524        2.178
## Plante                          2.442            2.233        1.964
## Overall.quality                -2.009            2.747        3.331
## Balance                        -2.052            2.670        3.129
## Flower                         -2.309            1.928        2.162
## Aroma.quality                  -2.311            2.572        3.064
##                                sd in category Overall sd   p.value
## Spice.before.shaking                   0.1410     0.2339 0.0009131
## Odor.Intensity.before.shaking          0.0060     0.2825 0.0024057
## Bitterness                             0.2225     0.1819 0.0026660
## Odor.Intensity                         0.0050     0.1957 0.0125125
## Spice                                  0.0845     0.2014 0.0126208
## Plante                                 0.0590     0.1596 0.0145920
## Overall.quality                        0.1045     0.4216 0.0445773
```

```
## Balance                                0.0990     0.3246 0.0401997
## Flower                                 0.1550     0.1471 0.0209267
## Aroma.quality                          0.0275     0.3084 0.0208463
```

As for cluster4, what characterize this class are mainly contributors of dim2, which is different from what I expected before. Because, there are only two members in this cluster, we don't know where other potiential individuals would go , so it's hard to get accurate estimations.If we look at the statistics, we can find for example, **Spice.before.shaking** has a higher mean(2.5260) in this category than in the entire data(1.9915), while **Aroma.quality** has a lower mean(2.5725) in this category than in the entire data(3.0636). These two variables contribute to dim1 and dim2, respectively. At this point, the estimate I made before makes some sense.

Lastly, if we also look at cluster1.

```
#cluster1
res$category$`1`
```

```
## NULL
```

```
print(res$quanti$`1`,digits = 4)
```

```
##         v.test Mean in category Overall mean sd in category Overall sd
## Acidity -2.514            2.271        2.385         0.1238     0.2341
##         p.value
## Acidity 0.01195
```

There is pretty much not describer for this cluster. So, it verifies that this cluster has nearly not distinct features.