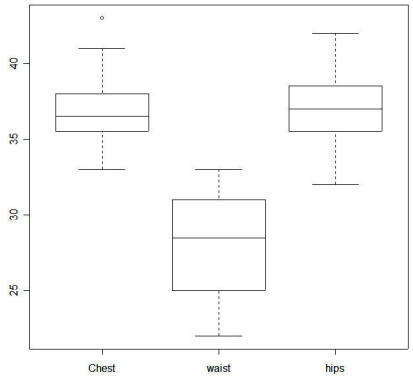
TP4 : Classification

1. EXPLORATION DES DONNEES
2. PARTITIONNEMENT : KMEANS
3. CLASSIFICATION HIERARCHIQUE

1. EXPLORATION DES DONNEES

Téléchargez le jeu de données « measure.txt ». Ce jeu de données donne les mensurations du tour de poitrine, taille et tour des hanche de 20 individus composés de 10 femmes et 10 hommes. L'objectif de cet exercice est de reconnaître à l'aide d'une classification qui est qui.

```
> mensuration<-read.table("measure.txt")
> boxplot(mensuration)
```



Si les échelles sont différentes d'une variable à l'autre, il faut normaliser d'abord les variables (fonction scale) avant d'appliquer kmeans.

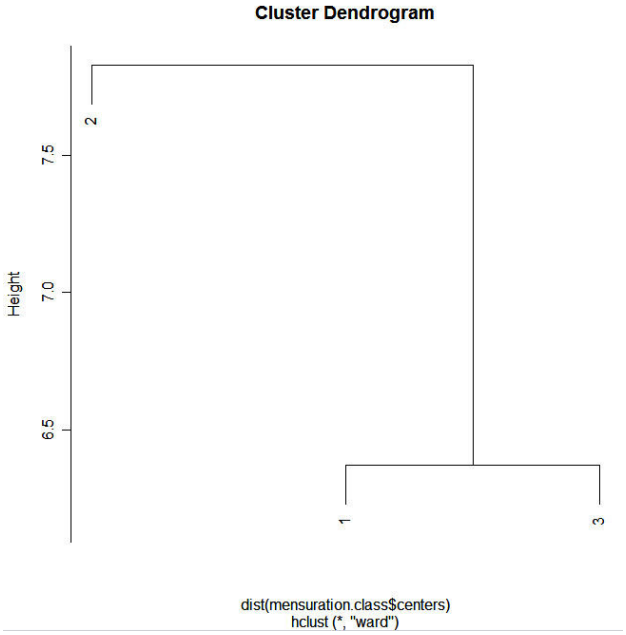
2. PARTITIONNEMENT : KMEANS

Choix du nombre de classe

Le choix du nombre de classe peut se poser quand on ignore totalement le nombre de classe.
Alors on choisit aléatoirement un nombre quelconque puis on analyse le dendogramme des barycentres des classes obtenues pour déduire le nombre de classe.
On peut également procéder à une classification hiérarchique sur l'ensemble des individus pour deviner le nombre de classes sous-jacent.

```
> mensuration.class<-kmeans(mensuration,3)
> mensuration.class
$cluster
 [1] 1 2 2 2 1 2 2 2 2 2 3 3 3 3 3 3 3 3 2 3
$centers
     Chest    waist     hips
1 36.00000 29.50000 32.50000
2 39.00000 31.11111 38.66667
3 35.22222 24.55556 36.44444
```

```
$withinss
[1]  9.00000 88.88889 46.00000
$size
[1] 2 9 9

>  mensuration.hclust<-hclust(dist(mensuration.class$centers),method="ward")
>  mensuration.hclust
Call:
hclust(d = dist(mensuration.class$centers), method = "ward")

Cluster method   : ward
Distance         : euclidean
Number of objects: 3
> plot(mensuration.hclust)
```



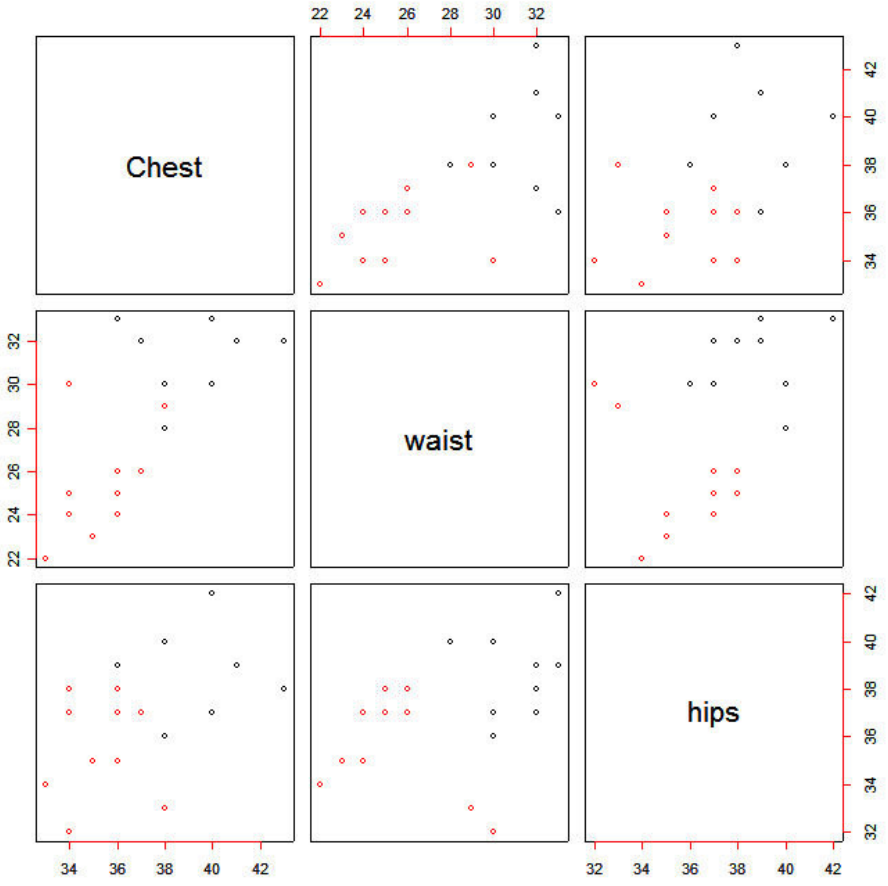
Le dendogramme montre l'existence de deux classes (gles classe 1 et 3 sont très proche et un grand saut les séparent de la classe 2). Nous réitérons donc la procédure avec un nb de classe de 2 puis avec les barycentres des partition obtenues

```
> mensuration.class<-kmeans(mensuration,centers=2)
> mensuration.class
$cluster
 [1] 2 1 1 1 2 1 1 1 1 1 2 2 2 2 2 2 2 1 2

$centers
    Chest    waist     hips
1 39.00000 31.11111 38.66667
2 35.36364 25.45455 35.72727
$withinss
[1]  88.88889 121.45455
$size
[1]  9 11

> mensuration.class1<-kmeans(mensuration,centers=mensuration.class$centers)
```

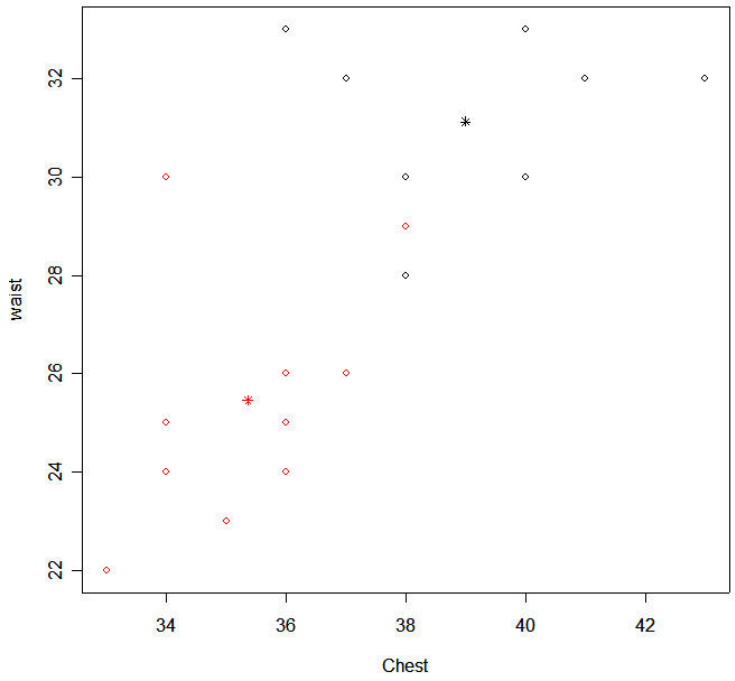
```
> mensuration.class1
$cluster
 [1] 2 1 1 1 2 1 1 1 1 1 2 2 2 2 2 2 2 1 2
$centers
     Chest    waist     hips
1 39.00000 31.11111 38.66667
2 35.36364 25.45455 35.72727
$withinss
[1]  88.88889 121.45455
$size
[1]  9 11
```

Visualisation des résultats de la partition

```
> plot(mensuration,col=mensuration.class1$cluster)
>  points(mensuration.class1$centers, col = 1:2, pch = 8)
```



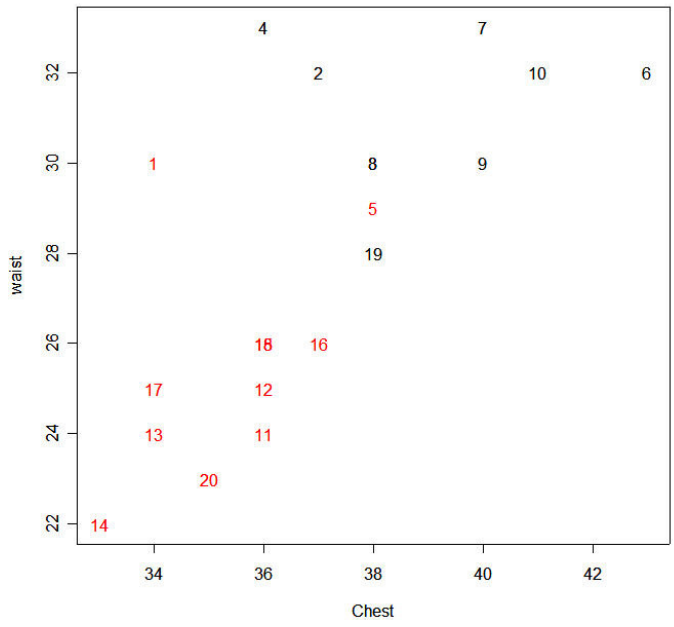
```
> plot(mensuration[1:2],col=mensuration.class1$cluster)
> points(mensuration.class1$centers, col = 1:2, pch = 8)
```



affichage plus lisible

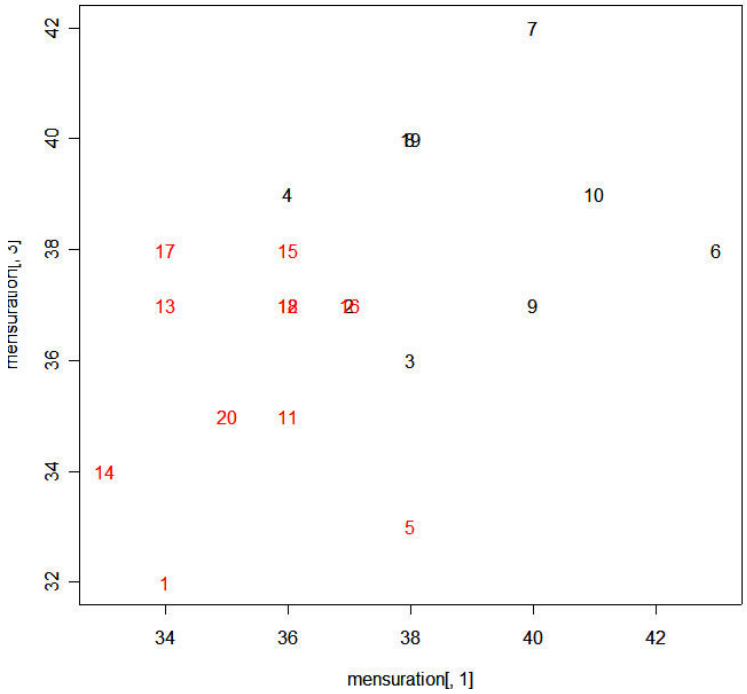
chest & waist

```
> plot(mensuration[1:2],type="n")
> text(mensuration[1:2],labels,col=mensuration.class1$cluster)
```



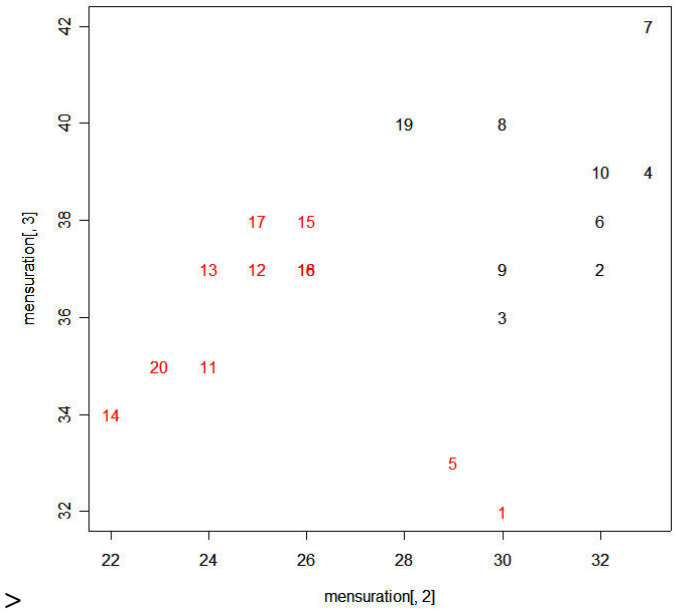
chest &hips

```
> plot(mensuration[,1],mensuration[,3],type="n")
> text(mensuration[,1], mensuration[,3],labels,col=mensuration.class1$cluster)
```



Waist & Hips

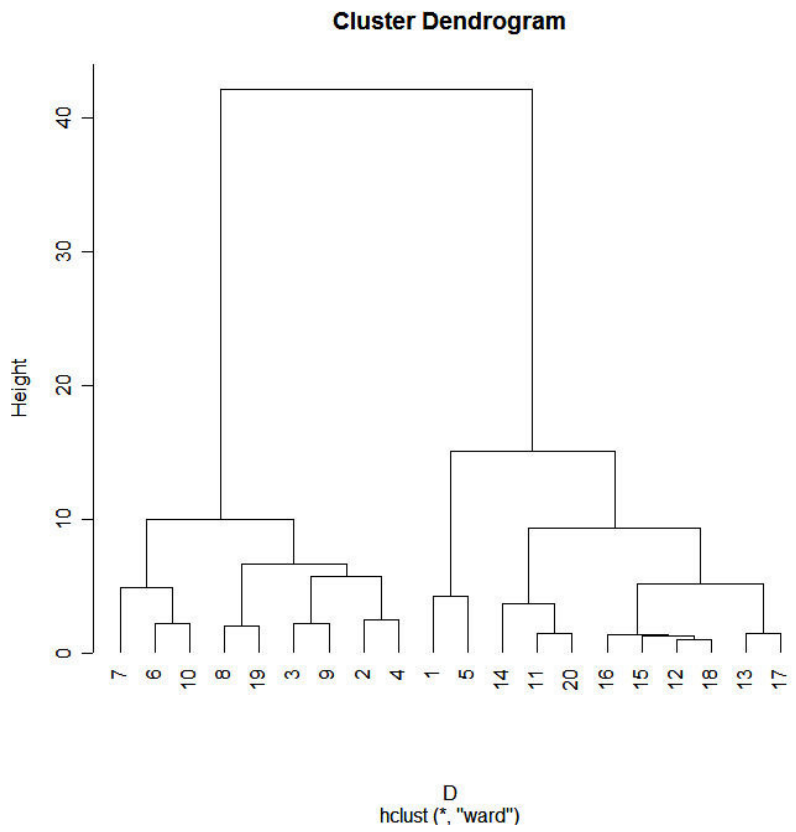
```
> plot(mensuration[,2],mensuration[,3],type="n")
> text(mensuration[,2], mensuration[,3],labels,col=mensuration.class1$cluster)
```



>

3. CLASSIFICATION HIERARCHIQUE

```
> D<-dist(mensuration)
> resuhist<-hclust(D,method="ward")
> plot(resuhist)
> plot(hc, hang = -1)
```



Couper le dendogramme afin d'extraire une partition à 5 classes : fonction cutree

```
> memb <- cutree(resuhist, k = 8)
> memb
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20
 1  2  3  2  1  4  4  5  3  4  6  7  8  6  7  7  8  7  5  6
```

Interprétez ces résulats.

```
 >cent <- NULL
 >    for(k in 1:8){
 >      for(k in 1:8){cent <- rbind(cent, colMeans(mensuration[memb == k, , drop = FALSE]))}
> cent
       Chest    waist     hips
[1,] 36.00000 29.50000 32.50000
[2,] 36.50000 32.50000 38.00000
[3,] 39.00000 30.00000 36.50000
[4,] 41.33333 32.33333 39.66667
[5,] 38.00000 29.00000 40.00000
[6,] 34.66667 23.00000 34.66667
```

```
[7,] 36.25000 25.75000 37.25000
[8,] 34.00000 24.50000 37.50000
>
> resuhist1 <- hclust(dist(cent)^2, method = "cen", members = table(memb))
> resuhist1

Call:
hclust(d = dist(cent)^2, method = "cen", members = table(memb))

Cluster method   : centroid
Distance         : euclidean
Number of objects: 8

>  opar <- par(mfrow = c(1, 2))
>  plot(resuhist,  labels = FALSE, hang = -1, main = "Original Tree")
>  plot(resuhist1, labels = FALSE, hang = -1, main = "Re-start from 8 clusters")
>  par(opar)
```