TP 3 : Classification non supervisée (Clustering)

Zaineb Smida

On reprend ici le jeu de données sur les 25 pays de l'Union Européenne (source : Eurostat 2002) qu'on importe de la façon suivante :

```
pays <- read.table("pays_eu.txt", header = T)
row.names(pays) <- pays$PAYS
```

En revanche, on va considérer uniquement les variables suivantes :

- espérance de vie à la naissance pour un homme (en années)
- espérance de vie à la naissance pour une femme (en années)
- taux d'activité (en pourcentage) : population active/population en âge de travailler
- taux d'inflation (en pourcentage)
- taux d'emploi (en pourcentage) : actifs occupés/population en âge de travailler
- taux de chômage (en pourcentage): chômeurs/population active

On extrait donc les colonnes qui nous intéressent :

```
paysred <- pays[, c("EVH", "EVF", "TEMP",  "TINF", "TACT", "TCHOM")]
```

Packages nécessaires :

```
Warning: le package 'cluster' a été compilé avec la version R 4.4.2
```

0.1 Question

Pourquoi faut-il centrer et réduire les variables ?

Conseils

Pour regarder les différences de magnitude entre les variables, on peut utiliser des indicateurs statistiques de base.

Solution

On constate que les moyennes sont différentes (voir par par exemple `TINF` et `EVH`)

```
summary(paysred) # les moyennes sont très différentes
```

```
      EVH             EVF             TEMP            TINF             TACT
 Min.   :64.80   Min.   :76.00   Min.   :48.70   Min.   :0.400   Min.   :51.5
 1st Qu.:72.10   1st Qu.:78.70   1st Qu.:54.80   1st Qu.:1.900   1st Qu.:58.4
 Median :75.10   Median :80.70   Median :58.30   Median :2.400   Median :63.4
 Mean   :73.41   Mean   :80.23   Mean   :57.49   Mean   :2.784   Mean   :63.5
 3rd Qu.:75.80   3rd Qu.:81.50   3rd Qu.:61.90   3rd Qu.:3.600   3rd Qu.:68.2
 Max.   :77.70   Max.   :83.50   Max.   :65.60   Max.   :7.500   Max.   :75.9
     TCHOM
 Min.   : 2.700
 1st Qu.: 4.900
 Median : 7.300
 Mean   : 8.096
 3rd Qu.: 9.500
 Max.   :19.800
```

0.2 Question

Centrer et réduire les variables

Conseils

On pourra utiliser la fonction `scale()`

Solution

On créé un nouvel objet qui contient les données centrées et réduites

```
paysred2 <- scale(paysred, center = TRUE, scale = TRUE)
```

0.3 Méthode AMM

On décide tout d'abord de faire une classification par Agrégation autour des Moyennes Mobiles (AMM ou k-means en anglais). On utilisera la fonction `kmeans()`.

0.3.1 Question

Représenter le R^2 global en fonction du nombre de groupes et justifier ainsi le choix du nombre de groupes.

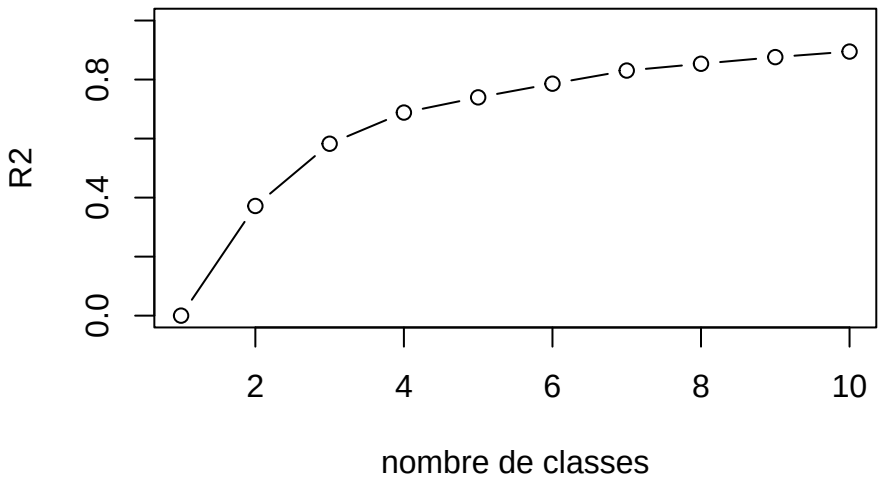
Conseils

On utilisera la fonction `kmeans()` en faisant varier le paramètre `centers` pour modifier le nombre de classe. On pourra fixer `nstart` à 100. L'objet retourné contient un certain nombre d'informations sur la classification : par exemple l'inertie inter-classes est donnée dans `$betweenss` et l'inertie totale par `$totss` ce qui permet de calculer le R^2.

Solution

On va stocker les valeurs de R^2 dans un vecteur pour chaque nombre de classes et on le représente. Ici, on décide de garder $k = 4$ classes, ce qui correspond à une rupture de pente à environ $R^2 = 0.69$.

```
R2 <- numeric(10)
for(k in 1:10) {
  my_clus <- kmeans(paysred2, k, nstart = 100)
  R2[k] <- my_clus$betweenss/my_clus$totss
}
plot(R2, type = "b", ylim = c(0, 1), xlab = "nombre de classes")
```



0.3.2 Question

Confirmer votre choix avec le calcul du coefficient silhouette moyen que vous représenterez graphiquement en fonction du nombre de groupes.

Conseils

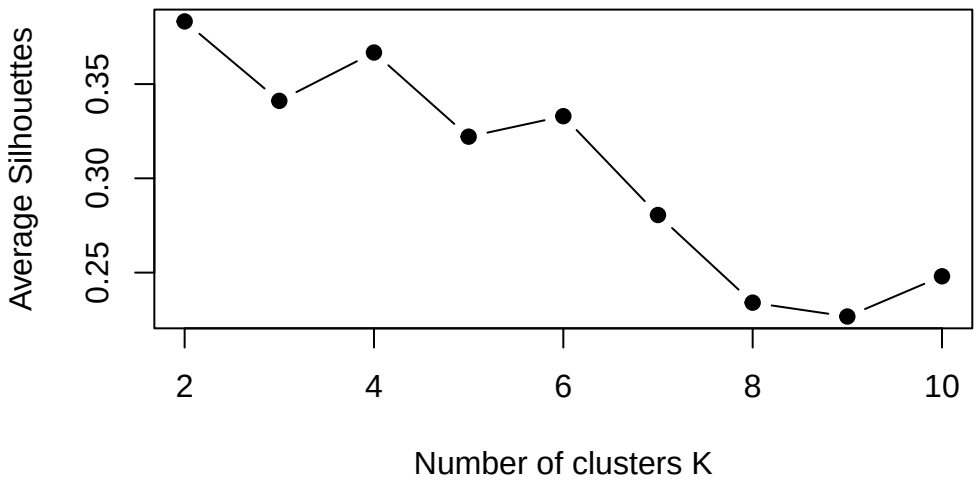
On utilisera la fonction `avg_sil()` suivante qui fait appel au package **cluster**

```
library(cluster) #pour avoir la fonction silhouette
avg_sil <- function(k) {
  km.res = kmeans(paysred2, centers = k, nstart = 100)
  ss = silhouette(km.res$cluster, dist(paysred2))
  mean(ss[, 3])
}
```

Solution

Le choix de 4 classes est justifié

```r
avg_sil_values = sapply(2:10, avg_sil)
plot(2:10, avg_sil_values,
     type = "b", pch = 19,
     xlab = "Number of clusters K",
     ylab = "Average Silhouettes")
```




```r
#coroborre le choix k=4
```

0.3.3 Question

Créer un vecteur contenant les numéros des classes obtenues avec la fonction `kmeans()` et donner les effectifs des classes obtenues.

> Conseils
>
> On peut récupérer les numéros des classes à partir de la syntaxe `$cluster`; on peut créer un `factor` avec la fonction `as.factor()`.

Solution

```
set.seed(4)
resuclassif <- kmeans(paysred2, 4, nstart = 100)
my_cluster <- as.factor(resuclassif$cluster)
table(my_cluster)
```

```
my_cluster
 1  2  3  4
11  7  5  2
```

- Le 1er cluster est le plus gros avec 11 observations (Rep-tche, Danemark, Allemagne, Irlande, Chypre, Pays-Bas, Autriche, Portugal, Finlande, Suede, Royaume-Uni),
- le 2eme groupe contient 7 observations (Belgique, Grece, Espagne, France, Italie, Luxembourg, Malte),
- le 3eme groupe contient 5 observations (Estonie, Lettonie, Lituanie, Pologne, Slovaquie),
- le plus petit groupe contient uniquement 2 observations (Hongrie et Slovénie).

0.3.4 Question

Réaliser les boîtes à moustaches des variables pour chaque groupe, puis calculer les moyennes des variables à l'intérieur de chaque groupe.

Conseils

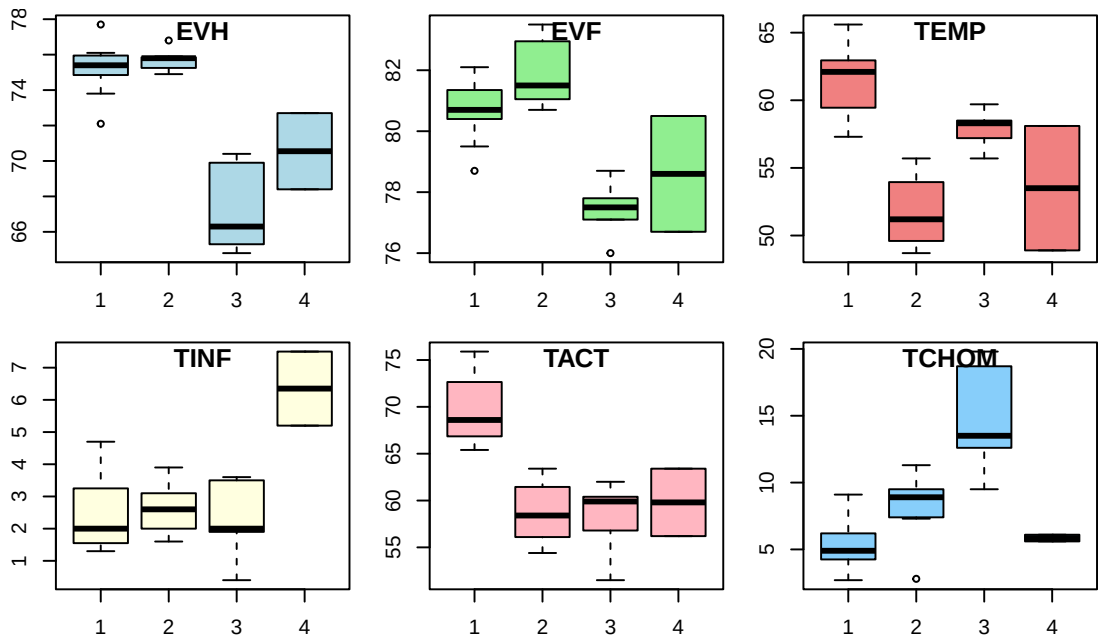
Pour calculer les moyennes par classe, on pourra utiliser la fonction `tapply()` (voir notes de cours).

Solution

On représente les boîtes à moustaches :

```r
# Définir une palette de couleurs
colors <- c("lightblue", "lightgreen", "lightcoral", "lightyellow",
            "lightpink", "lightskyblue")

# Créer les boîtes à moustaches avec des couleurs différentes
par(mfrow = c(2, 3), mar = c(3, 3, 0.25, 0.25))
boxplot(EVH ~ my_cluster, data = paysred, col = colors[1])
title(main = "EVH", line = -1)
boxplot(EVF ~ my_cluster, data = paysred, col = colors[2])
title(main = "EVF", line = -1)
boxplot(TEMP ~ my_cluster, data = paysred, col = colors[3])
title(main = "TEMP", line = -1)
boxplot(TINF ~ my_cluster, data = paysred, col = colors[4])
title(main = "TINF", line = -1)
boxplot(TACT ~ my_cluster, data = paysred, col = colors[5])
title(main = "TACT", line = -1)
boxplot(TCHOM ~ my_cluster, data = paysred, col = colors[6])
title(main = "TCHOM", line = -1)
```



On calcule les moyennes par groupe :

```r
my_cluster <- factor(paste0("G", resuclassif$cluster))
my_mean <- data.frame(
  EVH = tapply(paysred$EVH, my_cluster, mean),
  EVF = tapply(paysred$EVF, my_cluster, mean),
  TEMP = tapply(paysred$TEMP, my_cluster, mean),
  TINF = tapply(paysred$TINF, my_cluster, mean),
  TACT = tapply(paysred$TACT, my_cluster, mean),
  TCHOM = tapply(paysred$TCHOM, my_cluster, mean))
```

0.3.5 Question

Calculer le rapport de corrélation pour chaque variable et les classer par ordre d'importance.

Conseils

Pour calculer le rapport de coorélation R^2 pour chaque variable en fonction de votre cluster, vous pouvez utiliser une régression linéaire avec la fonction `lm()` et extraire la valeur de R^2 à partir de `$r.squared`. N'oubliez pas de répéter cette opération pour chaque variable de manière adéquate.

Solution

On calcule les rapports de corrélation. On constate que toutes les variables diffèrent suffisamment selon les groupes. Les deux plus importantes sont `EVH` et `TEMP`.

```r
# Calcul des rapports par variable
summary(lm(EVH ~ my_cluster, data = paysred))$r.squared #eta2=0.82
```

```
[1] 0.821046
```

```r
summary(lm(TEMP ~ my_cluster, data = paysred))$r.squared #eta2=0.73
```

```
[1] 0.7322316
```

```r
summary(lm(TACT ~ my_cluster, data = paysred))$r.squared #eta2=0.71
```

```
[1] 0.7065003
```

```r
summary(lm(EVF ~ my_cluster, data = paysred))$r.squared #eta2=0.70
```

```
[1] 0.7046557
```

```
summary(lm(TCHOM ~ my_cluster, data = paysred))$r.squared #eta2=0.66
```

```
[1] 0.6612389
```

```
summary(lm(TINF ~ my_cluster, data = paysred))$r.squared #eta2=0.50
```

```
[1] 0.5031193
```

0.3.6 Question

Représenter le nuage de points des deux variables les plus importantes, en distinguant les observations par des couleurs différentes selon leur groupe d'appartenance. Vous ajouterez également les centres des classes sur le graphique.

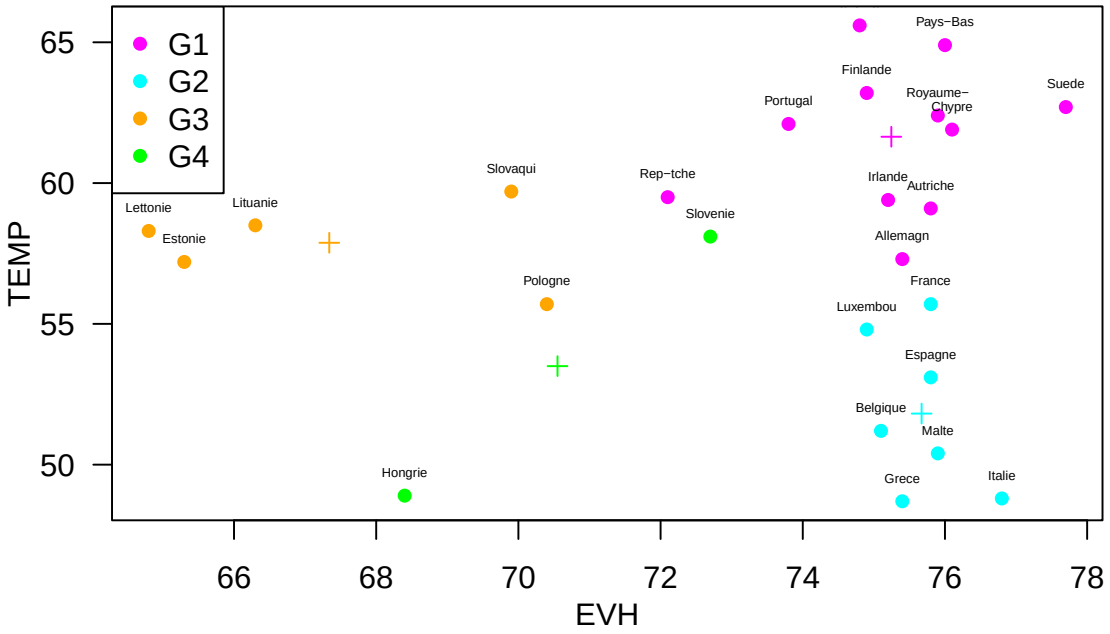
Conseils

Pour ajouter les centres des classes, vous pouvez utiliser la fonction `points()`. Pour inclure une légende et identifier chaque groupe, vous pouvez utiliser la fonction `legend()`.

Solution

On représente le nuage de points des deux variables `EVH` et `TEMP` avec les différents clusters :

```
my_col <- c("magenta", "cyan", "orange", "green")
par(las = 1,  mar = c(3, 3, 0.75, 0.5), mgp = c(2., 1, 0))
plot(paysred$EVH, paysred$TEMP, main = "", xlab = "EVH", ylab = "TEMP",
     col = my_col[my_cluster], pch = 16)
text(paysred$EVH, paysred$TEMP, row.names(paysred), cex = 0.4, pos = 3)
# Ajouter les moyennes des groupes
points(my_mean$EVH, my_mean$TEMP, col = my_col, pch = 3)
# Ajouter une légende en haut à gauche pour identifier chaque groupe
legend("topleft", legend = paste0("G", 1:4), pch = 16,
       col = c("magenta", "cyan", "orange", "green"))
```



0.3.7 Question

Définir une typologie des pays en prenant les précautions nécessaires. Ensuite, représentez les observations sur les premières composantes principales de l'ACP, que vous choisirez après avoir interprété les résultats de l'ACP.

Conseils

Pour définir une typologie des pays, commencez par utiliser les informations obtenues précédemment. Pour effectuer l'ACP, vous pouvez utiliser la fonction `PCA()`. Sélectionnez ensuite les axes principaux en justifiant votre choix (précisez le critère utilisé). Interprétez les résultats en examinant les corrélations des variables avec les axes principaux pour en comprendre la signification. Enfin, représentez les observations sur les premières composantes principales en utilisant `$ind$coord`.

Solution

Description des groupes à partir de la question précédente : nous pouvons examiner les moyennes des groupes à partir de :

```
my_mean
```

```
        EVH      EVF     TEMP     TINF     TACT     TCHOM
G1 75.24545 80.70000 61.64545 2.472727 69.64545  5.436364
G2 75.67143 81.95714 51.81429 2.614286 58.75714  8.114286
G3 67.34000 77.42000 57.88000 2.280000 58.12000 14.820000
G4 70.55000 78.60000 53.50000 6.350000 59.80000  5.850000
```

- G1 (11 pays) : `EVH` élevé, `EVF` élevé, `TEMP` élevé, `TINF` faible, `TACT` élevé, `TCHOM` faible
- G2 (7 pays) : `EVH` élevé, `EVF` élevé, `TEMP` faible, `TINF` moyen, `TACT` faible, `TCHOM` moyen
- G3 (5 pays) : `EVH` faible, `EVF` moyen, `TEMP` moyen, `TINF` faible, `TACT` faible, `TCHOM` élevé
- G4 (2 pays) : `EVH` faible, `EVF` moyen, `TEMP` faible, `TINF` élevé, `TACT` faible, `TCHOM` faible.

Nous pouvons également consulter la représentation graphique des groupes réalisée lors de la question précédente. Attention : certaines informations sont incomplètes.

Représentation graphique sur les CP de l'ACP :

L'ACP permet de prendre en compte la structure de corrélation entre les variables. Nous procédons à la réalisation de l'ACP :

```
library(FactoMineR)
resuacp <- PCA(paysred, graph = F)
resuacp$eig
```

```
       eigenvalue percentage of variance cumulative percentage of variance
comp 1 2.70896986             45.1494977                           45.14950
comp 2 1.61698646             26.9497744                           72.09927
comp 3 1.09813286             18.3022143                           90.40149
comp 4 0.46803851              7.8006419                           98.20213
comp 5 0.07771596              1.2952659                           99.49739
comp 6 0.03015635              0.5026059                          100.00000
```

```
resuacp$var$cor
```

```
            Dim.1       Dim.2       Dim.3       Dim.4       Dim.5
EVH    0.80832491  0.52245064 -0.12072744  0.13079462  0.20486498
EVF    0.68532175  0.64794497 -0.19030570  0.19907692 -0.18569994
TEMP   0.51398269 -0.78497787 -0.07610402  0.32409499  0.00290965
```

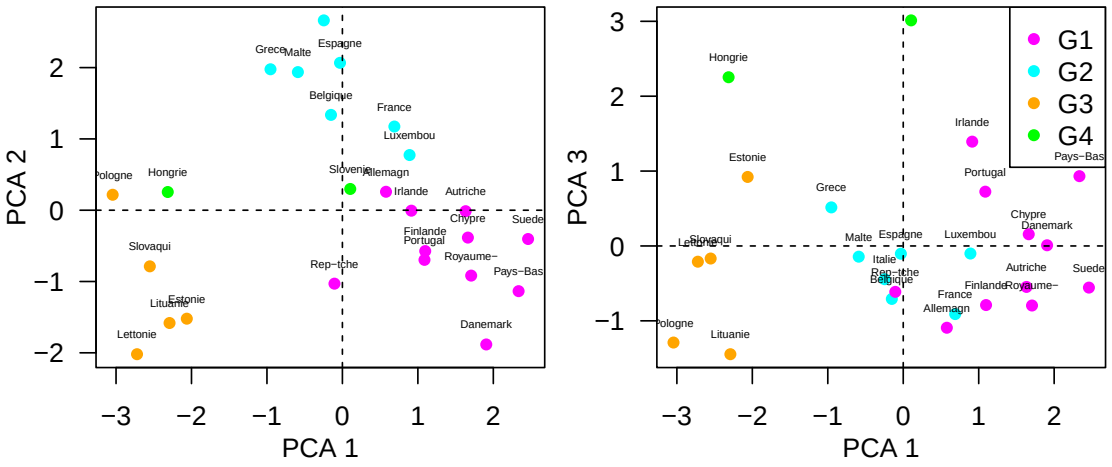
```
TINF  -0.05627699  0.15314097  0.94744226  0.27489332 -0.00235498
TACT   0.83484951 -0.53331533  0.02646053 -0.01777997 -0.02946236
TCHOM -0.78841347  0.01146512 -0.37842127  0.47997878  0.01948807
```

On garde 3 axes (selon le critère de Kaiser). Le critère de la part d'inertie expliquée nous donne le même nombre d'axes également.

- l'axe 1 est corrélé linéairement >0 avec `EVH`, `TACT` et négativement à `TCHOM`.
- l'axe 2 corrélé linéairement >0 avec `EVF` et négativement avec `TEMP`
- l'axe 3 est corrélée linéairement >0 avec `TINF`

On représente les groupes sur les premières composantes principales de l'ACP :

```r
my_col <- c("magenta", "cyan", "orange", "green")
par(las = 1,  mar = c(3, 3, 0.75, 0.5), mgp = c(2., 1, 0),
    mfrow = c(1, 2))
plot(resuacp$ind$coord[, 1], resuacp$ind$coord[, 2], main = "",
     xlab = "PCA 1", ylab = "PCA 2",
     col = my_col[my_cluster], pch = 16)
text(resuacp$ind$coord[, 1], resuacp$ind$coord[, 2],
     row.names(paysred), cex = 0.4, pos = 3)
abline(h = 0, lty = 2)
abline(v = 0, lty = 2)
plot(resuacp$ind$coord[, 1], resuacp$ind$coord[, 3], main = "",
     xlab = "PCA 1", ylab = "PCA 3",
     col = my_col[my_cluster], pch = 16)
text(resuacp$ind$coord[, 1], resuacp$ind$coord[, 3],
     row.names(paysred), cex = 0.4, pos = 3)
abline(h = 0, lty = 2)
abline(v = 0, lty = 2)
legend("topright", legend = paste0("G", 1:4), pch = 16,
       col = c("magenta", "cyan", "orange", "green"))
```



- Le groupe 1 contient des pays qui sont fortement corrélés à l'axe 1, i.e. qui ont des valeurs fortes de `EVH`, `TACT` et faibles de `TCHOM`. Il est opposé au Groupe 3 sur l'axe 1 qui contient lui des pays avec des valeurs faibles de `EVH`, `TACT` et fortes de `TCHOM`
- Le groupe 2 a des valeurs fortes positives sur l'axe 2, i.e. il contient des pays avec des valeurs fortes de `EVF` et faibles de `TEMP`
- Le groupe 4 est influencé par l'axe 3, i.e. qu'il contient des pays avec des valeurs fortes de `TINF`

0.4 Méthode CAH

On réalise ensuite une Classification Ascendante Hiérarchique (CAH) sur ces mêmes données. On utilisera la fonction `hclust()` du package **cluster**.

0.4.1 Question

En effectuant une CAH, commencer par calculer la matrice des distances entre individus à partir des données centrées et réduites. Ensuite, utiliser la fonction `hclust()` pour réaliser la CAH. Déterminer le nombre de groupes à retenir et justifier votre choix en vous appuyant sur des critères appropriés.

Conseils

Pour centrer et réduire les données, utilisez la fonction `scale()`. Ensuite, pour calculer la matrice des distances, utilisez la fonction `dist()`.
Pour déterminer le nombre de groupes, vous pouvez examiner le graphique des hauteurs

et le dendrogramme (référez-vous aux notes de cours pour plus de détails).

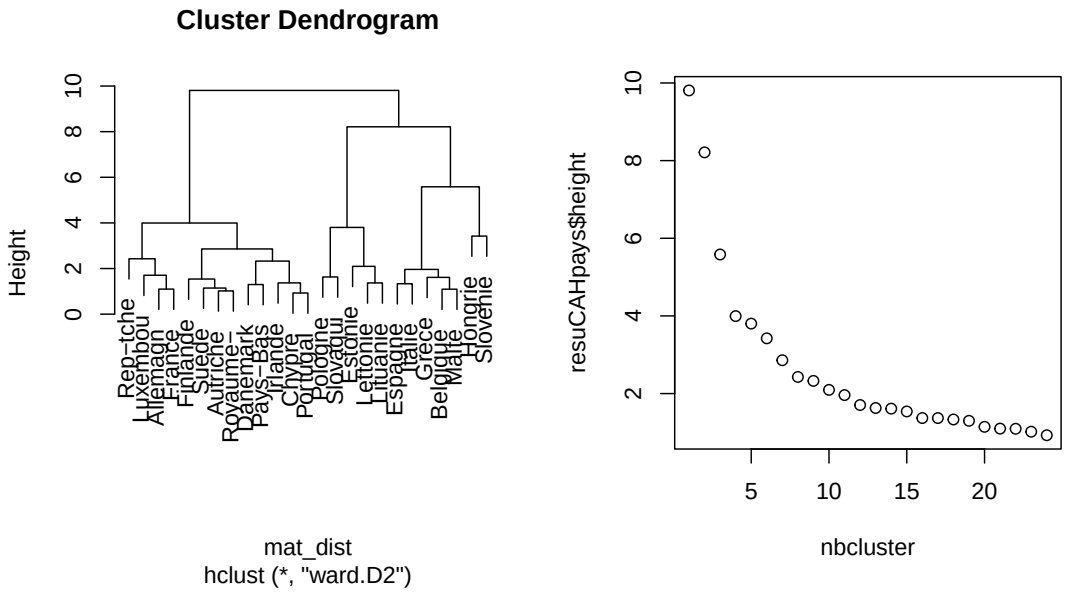
Solution

On calcule la matrice de distance et on fait une CAH :

```
mat_dist <- dist(scale(paysred))
resuCAHpays <- hclust(mat_dist, method="ward.D2")
```

On représente le graphique des hauteurs ainsi que le dendrogramme.

```
nbcluster <- 24:1
par(mfrow = c(1, 2))
plot(resuCAHpays)
plot(nbcluster, resuCAHpays$height)
```



Donc, on choisit 4 classes, bien que 3 puissent suffire.

0.4.2 Question

Comparer le résultat de cette classification avec les classes obtenues par la méthode des k-means.

Conseils

On pourra utiliser la fonction `cutree()` pour récupérer les classes de la CAH et la fonction `table()` pour comparer les résultats des deux classifications.

Solution

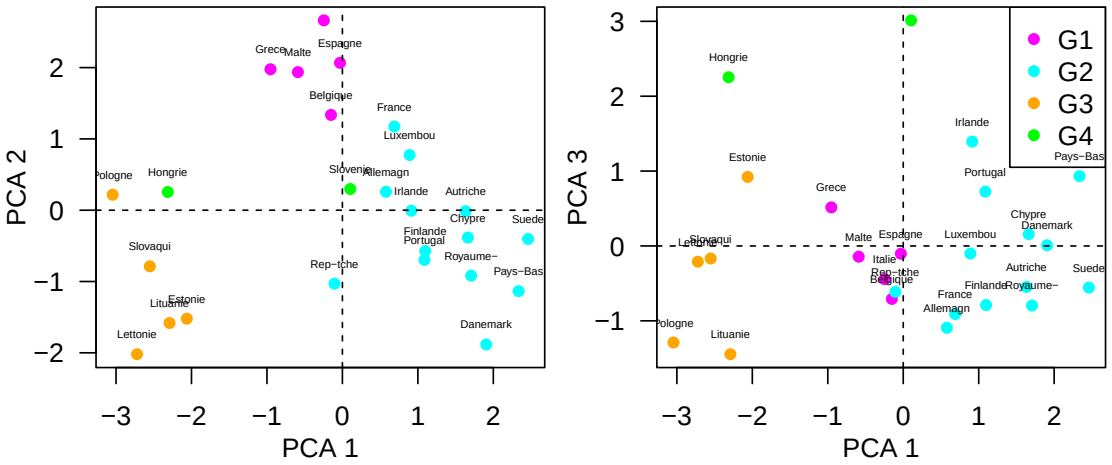
On récupère les classes de la CAH et on regarde les effectifs

```
my_cluster_2 <- factor(cutree(resuCAHpays, k = 4))
table(my_cluster_2)
```

```
my_cluster_2
 1  2  3  4
 5 13  5  2
```

On constate que le plus gros cluster a 13 observations, il y a deux groupes avec 5 observations et 1 groupe avec 2 observations. On représente les groupes sur les premières composantes principales de l'ACP :

```
my_col <- c("magenta", "cyan", "orange", "green")
par(las = 1,  mar = c(3, 3, 0.75, 0.5), mgp = c(2., 1, 0),
    mfrow = c(1, 2))
plot(resuacp$ind$coord[, 1], resuacp$ind$coord[, 2], main = "",
     xlab = "PCA 1", ylab = "PCA 2",
     col = my_col[my_cluster_2], pch = 16)
text(resuacp$ind$coord[, 1], resuacp$ind$coord[, 2],
     row.names(paysred), cex = 0.4, pos = 3)
abline(h = 0, lty = 2)
abline(v = 0, lty = 2)
plot(resuacp$ind$coord[, 1], resuacp$ind$coord[, 3], main = "",
     xlab = "PCA 1", ylab = "PCA 3",
     col = my_col[my_cluster_2], pch = 16)
text(resuacp$ind$coord[, 1], resuacp$ind$coord[, 3],
     row.names(paysred), cex = 0.4, pos = 3)
abline(h = 0, lty = 2)
abline(v = 0, lty = 2)
legend("topright", legend = paste0("G", 1:4), pch = 16,
       col = c("magenta", "cyan", "orange", "green"))
```



Si on compare les clusters des k-means et de la CAH, on constate que les clusters correspondent (à l'exception du numéro qui diffère), sauf pour deux observations qui n'ont pas été attribuées aux mêmes classes. Il s'agit de la France et du Luxembourg. En effet, les coordonnées de ces deux pays dans les trois premières composantes principales se situent entre les classes 1 et 2, ce qui explique pourquoi une méthode les affecte à une classe tandis qu'une autre les place dans une classe différente. L'interprétation des classes reste identique, puisque les groupes sont composés quasiment des mêmes observations.

```
table(my_cluster, my_cluster_2)
```

```
          my_cluster_2
my_cluster  1  2  3  4
        G1  0 11  0  0
        G2  5  2  0  0
        G3  0  0  5  0
        G4  0  0  0  2
```

0.4.3 Question

Réaliser la CAH sur les composantes principales données par l'ACP. Commenter.

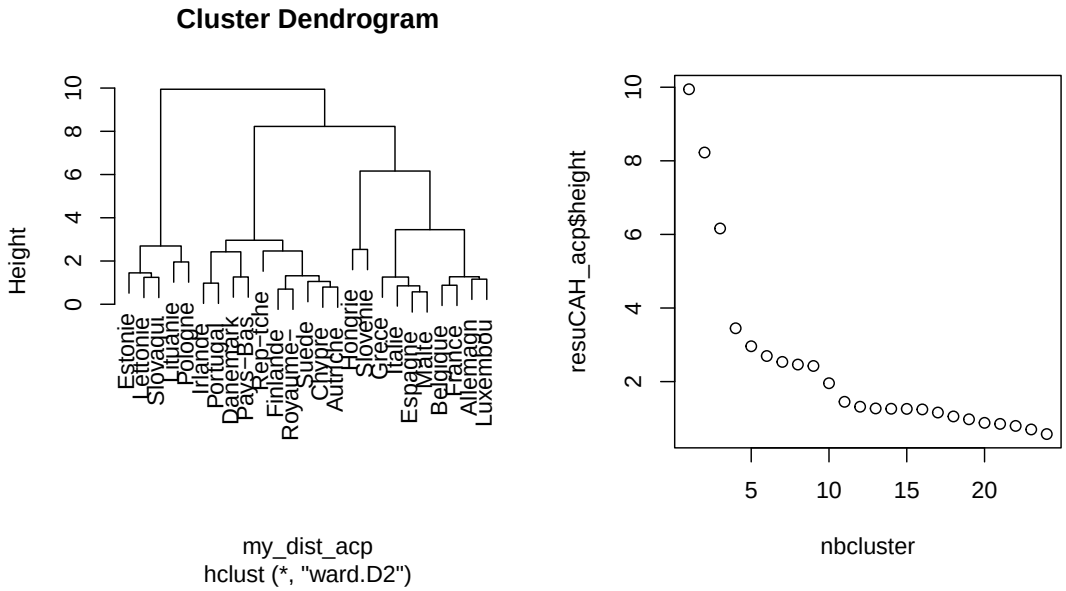
Solution

On calcule la matrice de distance sur les composantes principales et on fait la CAH :

```
my_dist_acp <- dist(resuacp$ind$coord[, 1:3])
resuCAH_acp <- hclust(my_dist_acp, method="ward.D2")
```

On représente le graphique des hauteurs ainsi que le dendrogramme. On choisit 4 classes.

```
nbcluster <- 24:1
par(mfrow = c(1, 2))
plot(resuCAH_acp)
plot(nbcluster, resuCAH_acp$height)
```



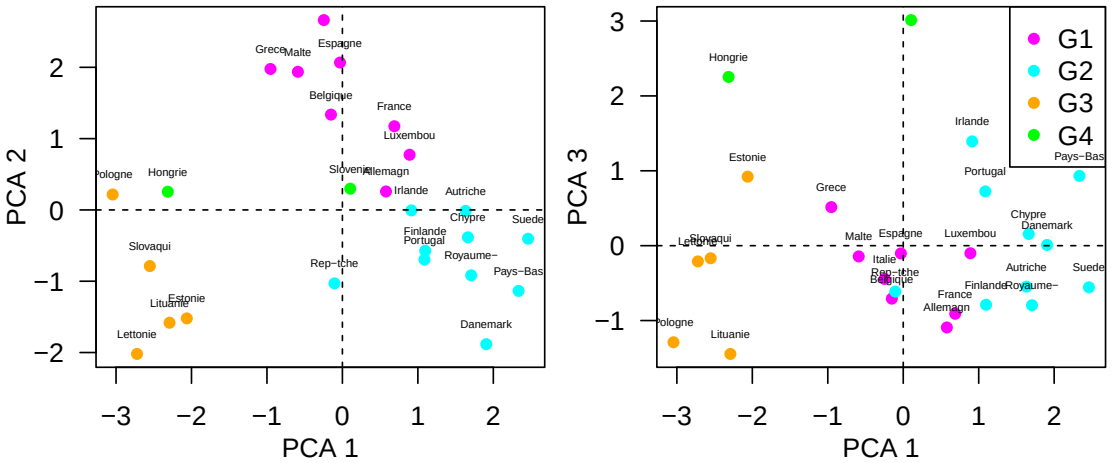
On récupère les classes de la CAH et on regarde les effectifs

```
my_cluster_3 <- factor(cutree(resuCAH_acp, k = 4))
table(my_cluster_3)
```

```
my_cluster_3
 1  2  3  4
 8 10  5  2
```

On constate que le plus grand cluster comporte 10 observations, le second en compte 8, le troisième en a 5 et le dernier 2. Nous représentons les groupes sur les premières composantes principales de l'ACP :

```r
my_col <- c("magenta", "cyan", "orange", "green")
par(las = 1,  mar = c(3, 3, 0.75, 0.5), mgp = c(2., 1, 0),
    mfrow = c(1, 2))
plot(resuacp$ind$coord[, 1], resuacp$ind$coord[, 2], main = "",
     xlab = "PCA 1", ylab = "PCA 2",
     col = my_col[my_cluster_3], pch = 16)
text(resuacp$ind$coord[, 1], resuacp$ind$coord[, 2],
     row.names(paysred), cex = 0.4, pos = 3)
abline(h = 0, lty = 2)
abline(v = 0, lty = 2)
plot(resuacp$ind$coord[, 1], resuacp$ind$coord[, 3], main = "",
     xlab = "PCA 1", ylab = "PCA 3",
     col = my_col[my_cluster_3], pch = 16)
text(resuacp$ind$coord[, 1], resuacp$ind$coord[, 3],
     row.names(paysred), cex = 0.4, pos = 3)
abline(h = 0, lty = 2)
abline(v = 0, lty = 2)
legend("topright", legend = paste0("G", 1:4), pch = 16,
       col = c("magenta", "cyan", "orange", "green"))
```



Si on compare les clusters avec ceux des k-means, on remarque que les clusters correspondent (à l'exception du numéro qui change). Par ailleurs, on constate qu'une seule observation est affectée à deux groupes différents. Il s'agit de l'Allemagne. En effet, les coordonnées de ce pays dans les trois premières composantes principales se situent entre la classe 1 et la classe 2, ce qui explique pourquoi une méthode l'affecte à l'une des classes tandis qu'une autre méthode l'affecte à l'autre classe. L'interprétation des classes reste identique, car les groupes sont composés quasiment des mêmes observations.

```
table(my_cluster, my_cluster_3)
```

```
          my_cluster_3
my_cluster  1  2  3  4
        G1  1 10  0  0
        G2  7  0  0  0
        G3  0  0  5  0
        G4  0  0  0  2
```