Feuille de Travaux Dirigés n° 24
Classifications

Exercice XXIV.1. Étude des caractéristiques d'un ensemble d'hôtels

Partie I : Classification hiérarchique ascendante

1. Récupérer les données dans **R** en exécutant les instructions suivantes. Penser à remplacer `"C :\\..."` par le répertoire dans lequel vous avez enregistré le fichier que vous souhaitez ouvrir.

```
> Chemin <- "C:\\..."
> hotels <- read.csv(paste(Chemin, "MASTER2TD24_EX1.CSV",
+     sep = ""), row.names = 1)
```

2. Quelles sont les différentes variables reproduites dans le tableau au verso ? Quelle est leur nature ? Qui sont les individus et les variables sur qui on va faire porter la classification hiérarchique ascendante ? Obtenir les statistiques descriptives, les covariances et les corrélations entre les variables quantitatives du jeu de données. Créer ensuite le graphique en étoile des hôtels.

	PAYS	ETOILE	CONFORT	CHAMBRE
1	Grèce : 8	Min. :0.000	Min. :2.00	Min. : 50.0
2	Maroc :12	1st Qu. :2.000	1st Qu. :4.00	1st Qu. :148.0
3	Portugal : 5	Median :3.000	Median :5.00	Median :250.0
4	Tunisie :10	Mean :2.974	Mean :5.18	Mean :261.2
5	Turquie : 4	3rd Qu. :4.000	3rd Qu. :6.00	3rd Qu. :317.0
6		Max. :5.000	Max. :9.00	Max. :800.0

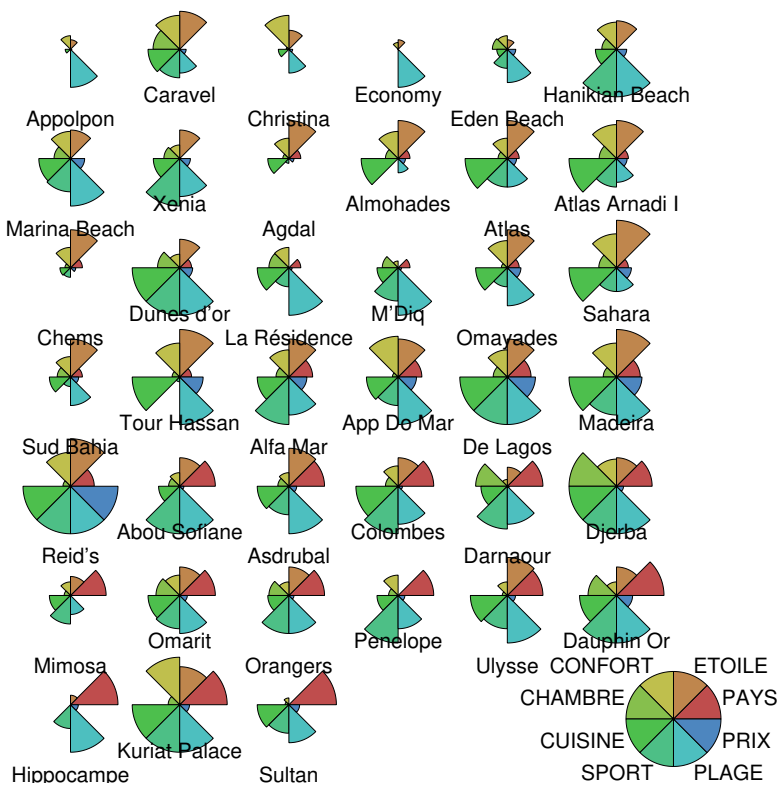
	CUISINE	SPORT	PLAGE	PRIX
1	Min. : 1.000	Min. : 0.000	Min. : 0.00	Min. : 369.0
2	1st Qu. : 5.000	1st Qu. : 4.000	1st Qu. : 6.50	1st Qu. : 447.0
3	Median : 7.000	Median : 6.000	Median : 8.00	Median : 495.0
4	Mean : 6.667	Mean : 6.231	Mean : 7.77	Mean : 529.9
5	3rd Qu. : 9.000	3rd Qu. :10.000	3rd Qu. :10.00	3rd Qu. : 574.0
6	Max. :10.000	Max. :10.000	Max. :10.00	Max. :1101.0

```
> palette(rainbow(12, s = 0.6, v = 0.75))
> stars(hotels, key.loc = c(14.5, 2), draw.segments = T,
+     main = "Diagramme en étoile des hôtels")
> palette("default")
```

	ETOILE	CONFORT	CHAMBRE	CUISINE	SPORT	PLAGE	PRIX
ETOILE	2.24	1.48	18.03	2.36	0.43	-0.51	111.63
CONFORT	1.48	2.47	17.01	2.32	0.19	-0.22	102.57
CHAMBRE	18.03	17.01	22449.75	167.10	246.90	74.76	-721.16
CUISINE	2.36	2.32	167.10	7.02	4.18	1.84	207.25
SPORT	0.43	0.19	246.90	4.18	11.87	4.98	147.87
PLAGE	-0.51	-0.22	74.76	1.84	4.98	7.39	126.37
PRIX	111.63	102.57	-721.16	207.25	147.87	126.37	19006.99

	ETOILE	CONFORT	CHAMBRE	CUISINE	SPORT	PLAGE	PRIX
ETOILE	1.00	0.63	0.08	0.60	0.08	-0.12	0.54
CONFORT	0.63	1.00	0.07	0.56	0.04	-0.05	0.47
CHAMBRE	0.08	0.07	1.00	0.42	0.48	0.18	-0.03
CUISINE	0.60	0.56	0.42	1.00	0.46	0.26	0.57
SPORT	0.08	0.04	0.48	0.46	1.00	0.53	0.31
PLAGE	-0.12	-0.05	0.18	0.26	0.53	1.00	0.34
PRIX	0.54	0.47	-0.03	0.57	0.31	0.34	1.00

Diagramme en étoile des hôtels

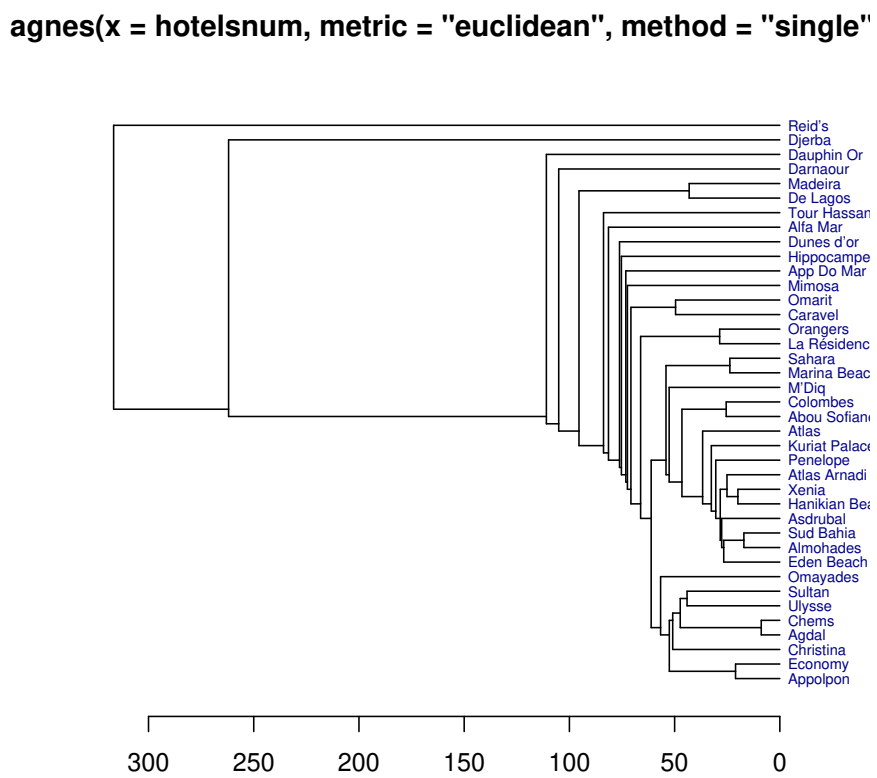


2. Faire la classification hiérarchique ascendante des observations en utilisant les distances euclidienne et Manhattan et les liaisons simple, complète et de Ward.

```
> library(cluster)
> hotelsnum <- hotels[, -1]
> res.cash <- agnes(hotelsnum, metric = "euclidean", method = "single")
> split(rownames(hotelsnum), cutree(res.cash, k = 3))
```

```
$`1`
 [1] "Appolpon"       "Caravel"        "Christina"
 [4] "Economy"        "Eden Beach"     "Hanikian Beach"
 [7] "Marina Beach"   "Xenia"          "Agdal"
[10] "Almohades"      "Atlas"          "Atlas Arnadi I"
[13] "Chems"          "Dunes d'or"     "La Résidence"
[16] "M'Diq"          "Omayades"       "Sahara"
[19] "Sud Bahia"      "Tour Hassan"    "Alfa Mar"
[22] "App Do Mar"     "De Lagos"       "Madeira"
[25] "Abou Sofiane"   "Asdrubal"       "Colombes"
[28] "Darnaour"       "Mimosa"         "Omarit"
[31] "Orangers"       "Penelope"       "Ulysse"
[34] "Dauphin Or"     "Hippocampe"     "Kuriat Palace"
[37] "Sultan"

$`2`
[1] "Reid's"

$`3`
[1] "Djerba"
```

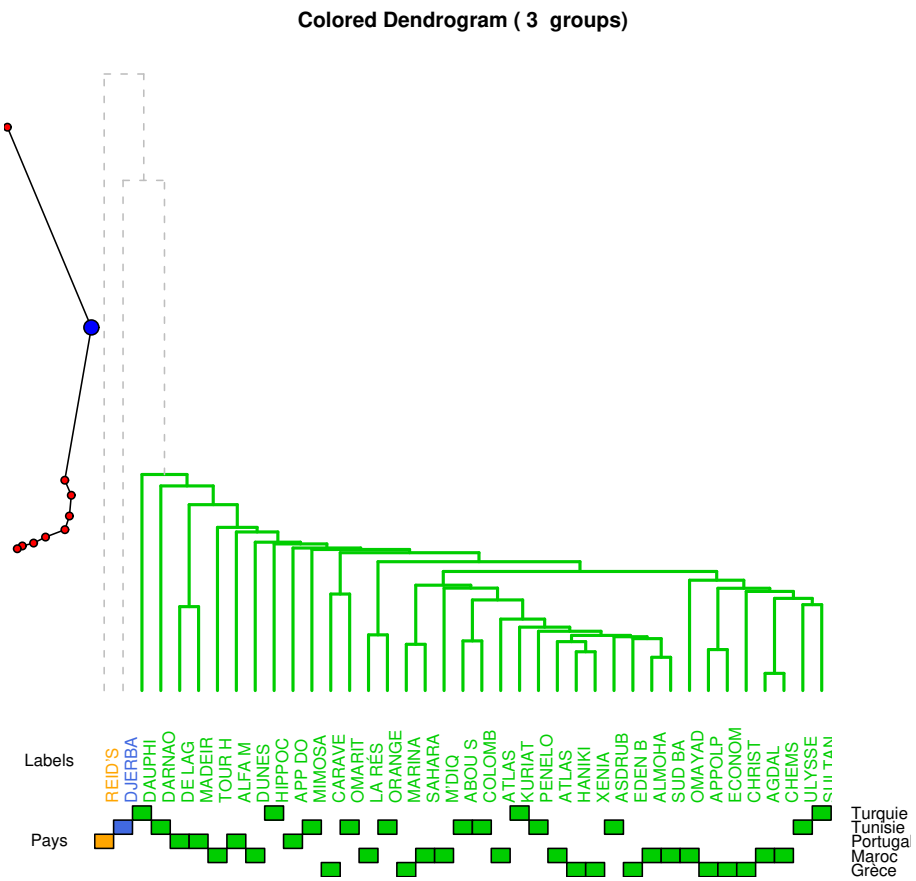
```
> res.dendro <- as.dendrogram(as.hclust(res.cash))
> plot(res.dendro, horiz = TRUE, center = TRUE, nodePar = list(lab.cex = 0.6,
+     lab.col = "darkblue", pch = NA), main = deparse(res.cash$call))
```



Un exemple de script de graphique pour représenter de manière plus complète les résultats d'une classification ascendante hiérarchique. Ce script provient d'un site présentant des graphiques réalisés avec **R** : R Graph Gallery[1]. Nous allons dorénavant présenter ces graphiques tout en proposant le code permettant d'obtenir les deux types de représentation. Il est également possible d'utiliser la fonction `silhouette` du package `cluster`.

```
> install.packages("fpc")
> library(fpc)
> source("http://addictedtor.free.fr/packages/A2R/lastVersion/R/A2R")
> d.hotels <- dist(hotelsnum, "euclidean")
> h.hotels <- hclust(d.hotels, method = "single")
> Pays <- hotels[, 1]
> hubertgamma <- rep(0, 10)
> for (i in 1:10) {
+     hubertgamma[i] <- cluster.stats(d.hotels, cutree(h.hotels,
+         k = i + 1), G2 = FALSE, G3 = FALSE, silhouette = FALSE)$hubertgamma
+ }
> A2Rplot(h.hotels, k = 3, fact.sup = Pays, criteria = hubertgamma,
+     boxes = FALSE, col.up = "gray", col.down = c("orange",
+         "royalblue", "green3", "red2", "purple"))
```

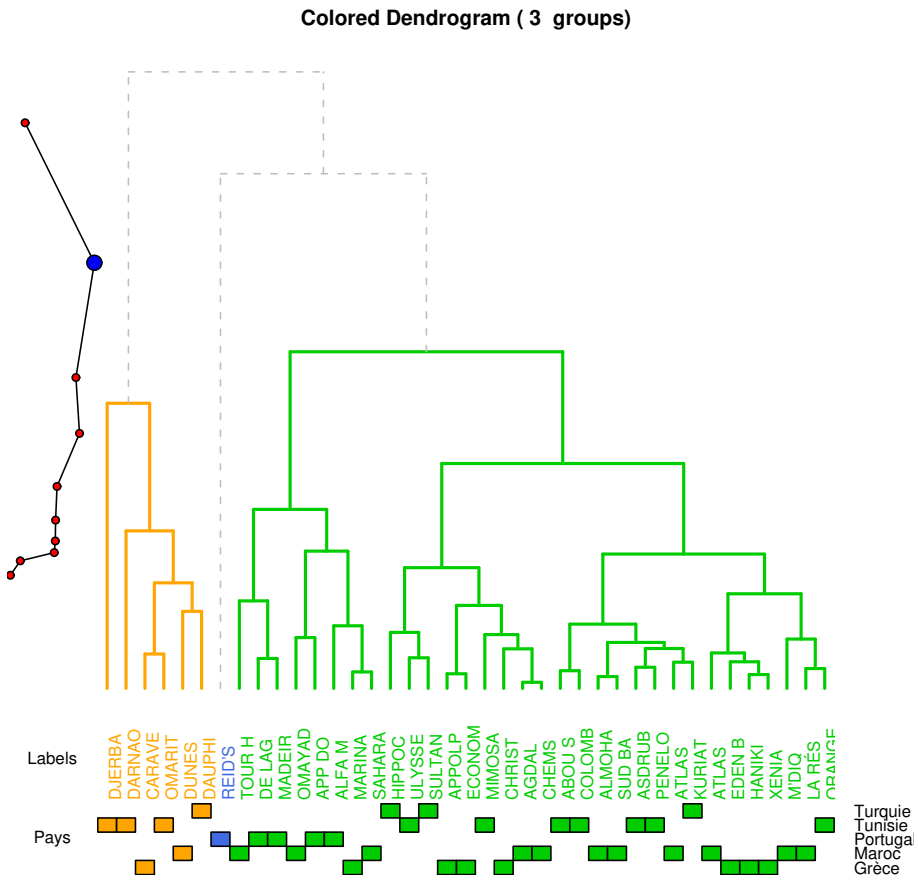
[1] `http ://addictedtor.free.fr/graphiques/`.




```
> res.cash <- agnes(hotelsnum, metric = "euclidean", method = "complete")
> split(rownames(hotelsnum), cutree(res.cash, k = 3))
```

```
$`1`
 [1] "Appolpon"       "Christina"      "Economy"
 [4] "Eden Beach"     "Hanikian Beach" "Marina Beach"
 [7] "Xenia"          "Agdal"          "Almohades"
[10] "Atlas"          "Atlas Arnadi I" "Chems"
[13] "La Résidence"   "M'Diq"          "Omayades"
[16] "Sahara"         "Sud Bahia"      "Tour Hassan"
[19] "Alfa Mar"       "App Do Mar"     "De Lagos"
[22] "Madeira"        "Abou Sofiane"   "Asdrubal"
[25] "Colombes"       "Mimosa"         "Orangers"
[28] "Penelope"       "Ulysse"         "Hippocampe"
[31] "Kuriat Palace"  "Sultan"

$`2`
[1] "Caravel"    "Dunes d'or" "Darnaour"   "Djerba"     "Omarit"
[6] "Dauphin Or"

$`3`
[1] "Reid's"
```

```
> res.dendro <- as.dendrogram(as.hclust(res.cash))
> plot(res.dendro, horiz = TRUE, center = TRUE, nodePar = list(lab.cex = 0.6,
+     lab.col = "darkblue", pch = NA), main = deparse(res.cash$call))
> d.hotels <- dist(hotelsnum, "euclidean")
> h.hotels <- hclust(d.hotels, method = "complete")
> Pays <- hotels[, 1]
> hubertgamma <- rep(0, 10)
> for (i in 1:10) {
+     hubertgamma[i] <- cluster.stats(d.hotels, cutree(h.hotels,
+         k = i + 1), G2 = FALSE, G3 = FALSE, silhouette = FALSE)$hubertgamma
+ }
> A2Rplot(h.hotels, k = 3, fact.sup = Pays, criteria = hubertgamma,
+     boxes = FALSE, col.up = "gray", col.down = c("orange",
+         "royalblue", "green3", "red2", "purple"))
```

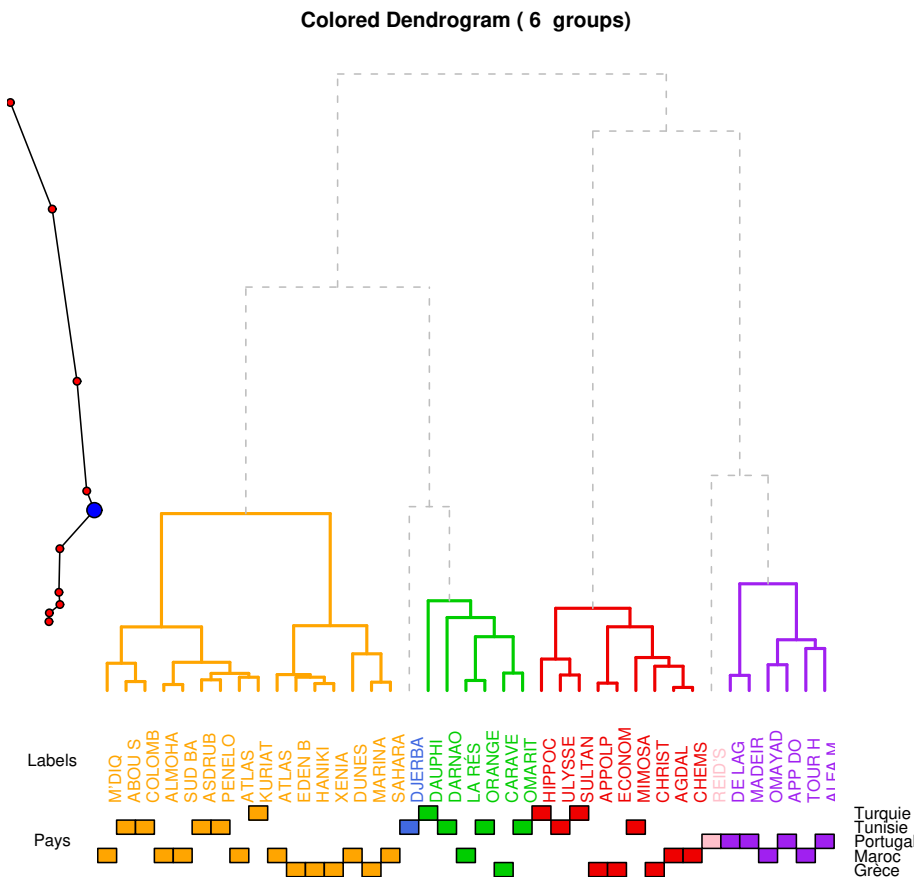



```
> res.cash <- agnes(hotelsnum, metric = "euclidean", method = "ward")
> split(rownames(hotelsnum), cutree(res.cash, k = 6))
$`1`
[1] "Appolpon"   "Christina"  "Economy"    "Agdal"      "Chems"
[6] "Mimosa"     "Ulysse"     "Hippocampe" "Sultan"

$`2`
[1] "Caravel"      "La Résidence" "Darnaour"     "Omarit"
[5] "Orangers"     "Dauphin Or"
```

```
$`3`
 [1] "Eden Beach"     "Hanikian Beach" "Marina Beach"
 [4] "Xenia"          "Almohades"      "Atlas"
 [7] "Atlas Arnadi I" "Dunes d'or"     "M'Diq"
[10] "Sahara"         "Sud Bahia"      "Abou Sofiane"
[13] "Asdrubal"       "Colombes"       "Penelope"
[16] "Kuriat Palace"

$`4`
[1] "Omayades"    "Tour Hassan" "Alfa Mar"    "App Do Mar"
[5] "De Lagos"    "Madeira"

$`5`
[1] "Reid's"

$`6`
[1] "Djerba"
```

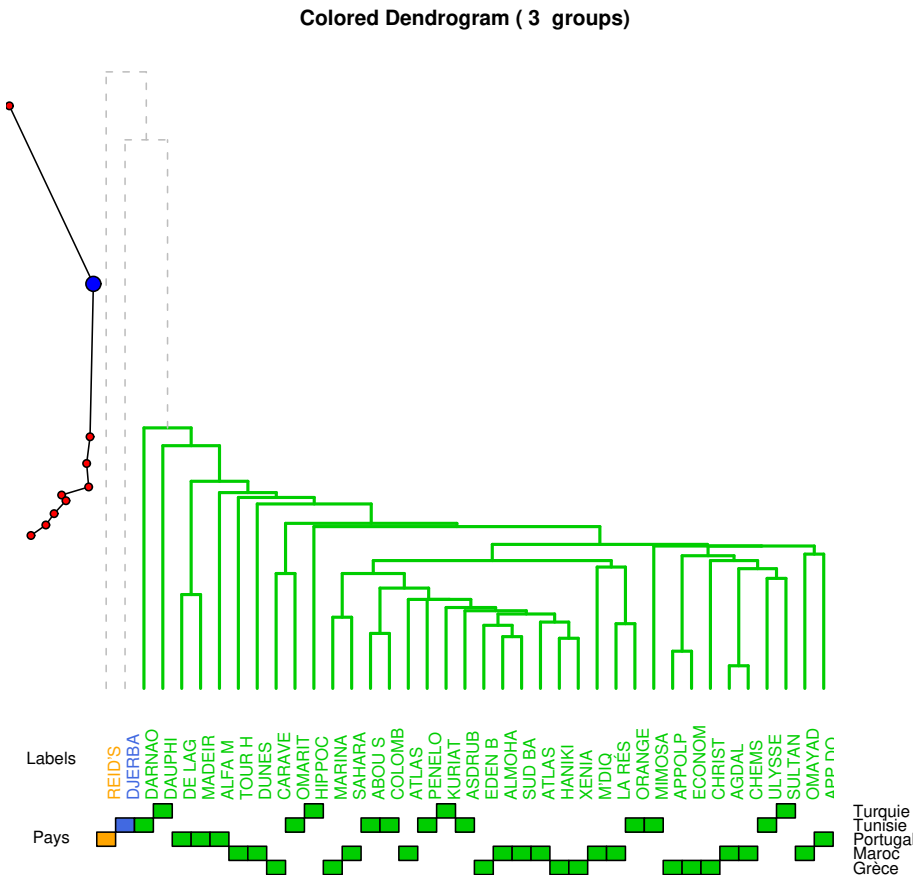
```
> res.dendro <- as.dendrogram(as.hclust(res.cash))
> plot(res.dendro, horiz = TRUE, center = TRUE, nodePar = list(lab.cex = 0.6,
+     lab.col = "darkblue", pch = NA), main = deparse(res.cash$call))
```

```
> d.hotels <- dist(hotelsnum, "euclidean")
> h.hotels <- hclust(d.hotels, method = "ward")
> Pays <- hotels[, 1]
> hubertgamma <- rep(0, 10)
> for (i in 1:10) {
+     hubertgamma[i] <- cluster.stats(d.hotels, cutree(h.hotels,
+         k = i + 1), G2 = FALSE, G3 = FALSE, silhouette = FALSE)$hubertgamma
+ }
> A2Rplot(h.hotels, k = 6, fact.sup = Pays, criteria = hubertgamma,
+     boxes = FALSE, col.up = "gray", col.down = c("orange",
+         "royalblue", "green3", "red2", "pink", "purple"))
```




```
> res.cash <- agnes(hotelsnum, metric = "manhattan", method = "single")
> split(rownames(hotelsnum), cutree(res.cash, k = 3))

$`1`
 [1] "Appolpon"       "Caravel"        "Christina"
 [4] "Economy"        "Eden Beach"     "Hanikian Beach"
 [7] "Marina Beach"   "Xenia"          "Agdal"
[10] "Almohades"      "Atlas"          "Atlas Arnadi I"
[13] "Chems"          "Dunes d'or"     "La Résidence"
[16] "M'Diq"          "Omayades"       "Sahara"
[19] "Sud Bahia"      "Tour Hassan"    "Alfa Mar"
[22] "App Do Mar"     "De Lagos"       "Madeira"
[25] "Abou Sofiane"   "Asdrubal"       "Colombes"
[28] "Darnaour"       "Mimosa"         "Omarit"
[31] "Orangers"       "Penelope"       "Ulysse"
[34] "Dauphin Or"     "Hippocampe"     "Kuriat Palace"
[37] "Sultan"

$`2`
[1] "Reid's"

$`3`
[1] "Djerba"
```

```
> res.dendro <- as.dendrogram(as.hclust(res.cash))
> plot(res.dendro, horiz = TRUE, center = TRUE, nodePar = list(lab.cex = 0.6,
+     lab.col = "darkblue", pch = NA), main = deparse(res.cash$call))
> d.hotels <- dist(hotelsnum, "manhattan")
> h.hotels <- hclust(d.hotels, method = "single")
> Pays <- hotels[, 1]
> hubertgamma <- rep(0, 10)
> for (i in 1:10) {
+     hubertgamma[i] <- cluster.stats(d.hotels, cutree(h.hotels,
+         k = i + 1), G2 = FALSE, G3 = FALSE, silhouette = FALSE)$hubertgamma
+ }
> A2Rplot(h.hotels, k = 3, fact.sup = Pays, criteria = hubertgamma,
+     boxes = FALSE, col.up = "gray", col.down = c("orange",
+         "royalblue", "green3", "red2", "purple"))
```

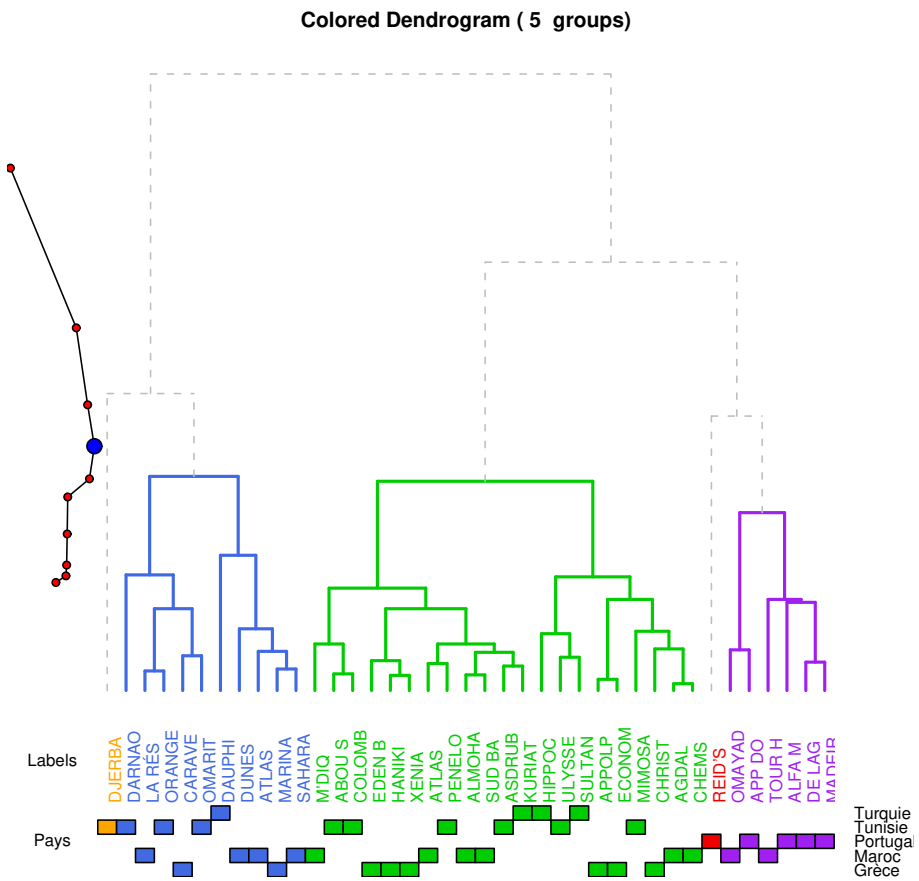



```
> res.cash <- agnes(hotelsnum, metric = "manhattan", method = "complete")
> split(rownames(hotelsnum), cutree(res.cash, k = 5))
$`1`
 [1] "Appolpon"       "Christina"      "Economy"
 [4] "Eden Beach"     "Hanikian Beach" "Xenia"
 [7] "Agdal"          "Almohades"      "Atlas"
[10] "Chems"          "M'Diq"          "Sud Bahia"
[13] "Abou Sofiane"   "Asdrubal"       "Colombes"
[16] "Mimosa"         "Penelope"       "Ulysse"
```

```
[19] "Hippocampe"     "Kuriat Palace"  "Sultan"

$`2`
 [1] "Caravel"        "Marina Beach"   "Atlas Arnadi I"
 [4] "Dunes d'or"     "La Résidence"   "Sahara"
 [7] "Darnaour"       "Omarit"         "Orangers"
[10] "Dauphin Or"

$`3`
[1] "Omayades"    "Tour Hassan" "Alfa Mar"    "App Do Mar"
[5] "De Lagos"    "Madeira"

$`4`
[1] "Reid's"

$`5`
[1] "Djerba"
```

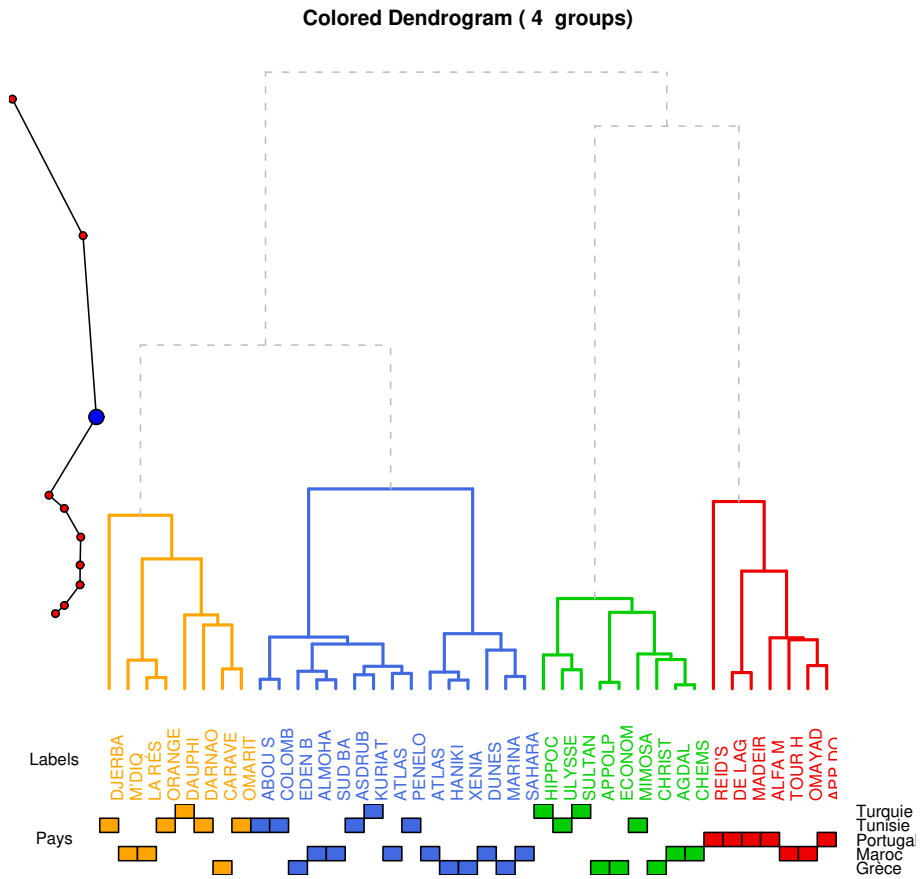
```
> res.dendro <- as.dendrogram(as.hclust(res.cash))
> plot(res.dendro, horiz = TRUE, center = TRUE, nodePar = list(lab.cex = 0.6,
+     lab.col = "darkblue", pch = NA), main = deparse(res.cash$call))
```

```
> d.hotels <- dist(hotelsnum, "manhattan")
> h.hotels <- hclust(d.hotels, method = "complete")
> Pays <- hotels[, 1]
> hubertgamma <- rep(0, 10)
> for (i in 1:10) {
+     hubertgamma[i] <- cluster.stats(d.hotels, cutree(h.hotels,
+         k = i + 1), G2 = FALSE, G3 = FALSE, silhouette = FALSE)$hubertgamma
+ }
> A2Rplot(h.hotels, k = 5, fact.sup = Pays, criteria = hubertgamma,
+     boxes = FALSE, col.up = "gray", col.down = c("orange",
+         "royalblue", "green3", "red2", "purple"))
```




```
> res.cash <- agnes(hotelsnum, metric = "manhattan", method = "ward")
> split(rownames(hotelsnum), cutree(res.cash, k = 4))
$`1`
[1] "Appolpon"   "Christina"  "Economy"    "Agdal"      "Chems"
[6] "Mimosa"     "Ulysse"     "Hippocampe" "Sultan"

$`2`
[1] "Caravel"    "Darnaour"   "Djerba"     "Omarit"     "Dauphin Or"

$`3`
 [1] "Eden Beach"     "Hanikian Beach" "Marina Beach"
 [4] "Xenia"          "Almohades"      "Atlas"
 [7] "Atlas Arnadi I" "Dunes d'or"     "La Résidence"
[10] "M'Diq"          "Sahara"         "Sud Bahia"
[13] "Abou Sofiane"   "Asdrubal"       "Colombes"
[16] "Orangers"       "Penelope"       "Kuriat Palace"

$`4`
[1] "Omayades"    "Tour Hassan" "Alfa Mar"    "App Do Mar"
[5] "De Lagos"    "Madeira"     "Reid's"
> res.dendro <- as.dendrogram(as.hclust(res.cash))
> plot(res.dendro, horiz = TRUE, center = TRUE, nodePar = list(lab.cex = 0.6,
+     lab.col = "darkblue", pch = NA), main = deparse(res.cash$call))
```

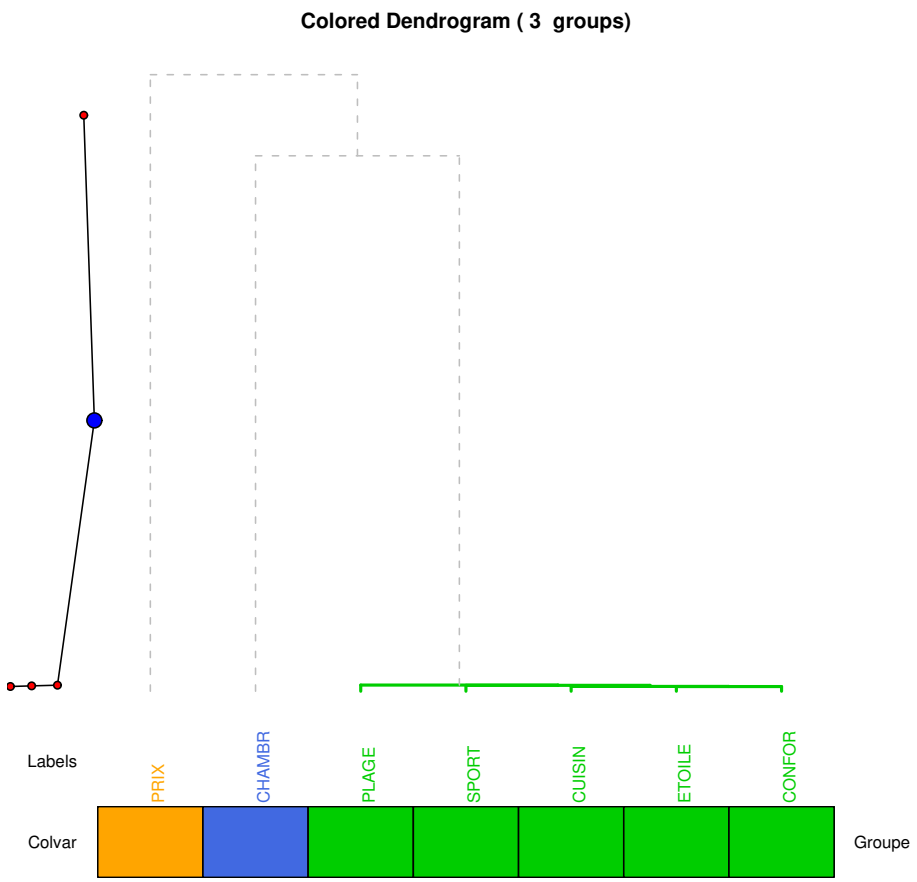
```
> d.hotels <- dist(hotelsnum, "manhattan")
> h.hotels <- hclust(d.hotels, method = "ward")
> Pays <- hotels[, 1]
> hubertgamma <- rep(0, 10)
> for (i in 1:10) {
+     hubertgamma[i] <- cluster.stats(d.hotels, cutree(h.hotels,
+         k = i + 1), G2 = FALSE, G3 = FALSE, silhouette = FALSE)$hubertgamma
+ }
> A2Rplot(h.hotels, k = 4, fact.sup = Pays, criteria = hubertgamma,
+     boxes = FALSE, col.up = "gray", col.down = c("orange",
+         "royalblue", "green3", "red2", "purple"))
```



3. Faire la classification hiérarchique ascendante des variables en utilisant les distances euclidienne et Manhattan et les liaisons simple, complète et de Ward.

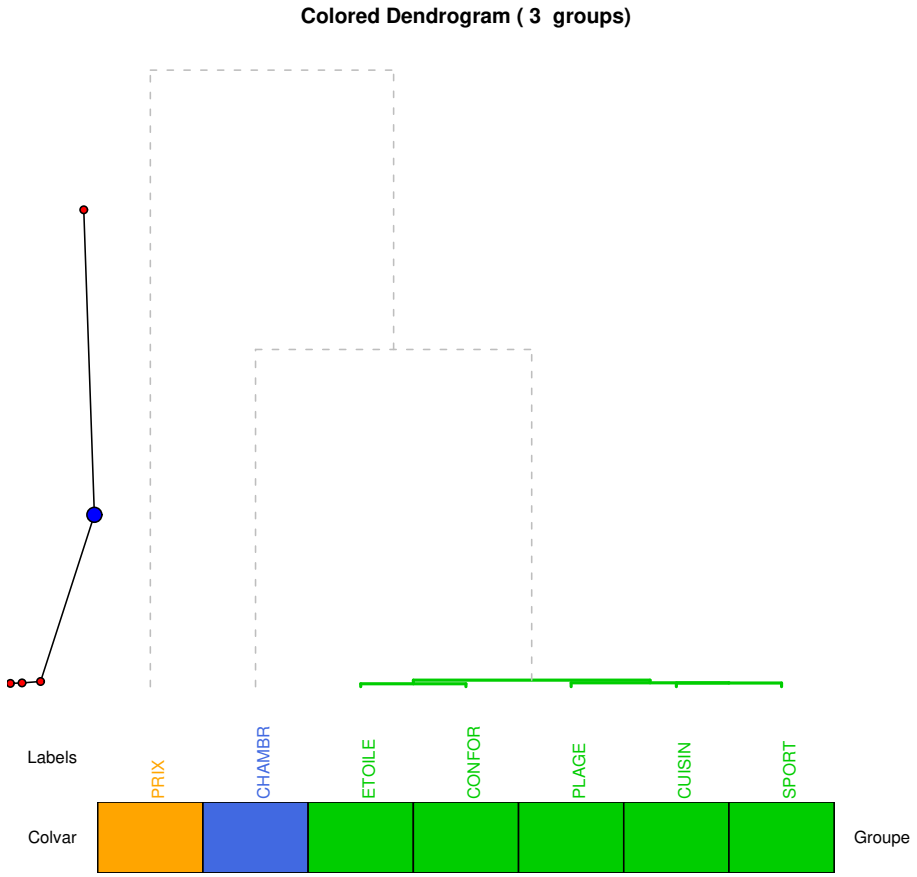
```
> thotelsnum <- t(hotelsnum)
> res.cash <- agnes(thotelsnum, metric = "euclidean",
+     method = "single")
> split(rownames(thotelsnum), cutree(res.cash, k = 3))
$`1`
[1] "ETOILE"  "CONFORT" "CUISINE" "SPORT"   "PLAGE"

$`2`
```

```
[1] "CHAMBRE"

$`3`
[1] "PRIX"
> res.dendro <- as.dendrogram(as.hclust(res.cash))
> plot(res.dendro, horiz = TRUE, center = TRUE, nodePar = list(lab.cex = 0.6,
+     lab.col = "darkblue", pch = NA), main = deparse(res.cash$call))
> d.thotels <- dist(t(hotelsnum), "euclidean")
> h.thotels <- hclust(d.thotels, method = "single")
> Colvar <- factor(rep("Groupe", 7))
> hubertgamma <- rep(0, 5)
> for (i in 1:5) {
+     hubertgamma[i] <- cluster.stats(d.thotels, cutree(h.thotels,
+         k = i + 1), G2 = FALSE, G3 = FALSE, silhouette = FALSE)$hubertgamma
+ }
> A2Rplot(h.thotels, k = 3, fact.sup = Colvar, criteria = hubertgamma,
+     boxes = FALSE, col.up = "grey", col.down = c("orange",
+         "royalblue", "green3", "red2", "purple"))
```




```
> res.cash <- agnes(thotelsnum, metric = "euclidean",
+     method = "complete")
> split(rownames(thotelsnum), cutree(res.cash, k = 3))
$`1`
[1] "ETOILE"  "CONFORT" "CUISINE" "SPORT"   "PLAGE"
```

```
$`2`
[1] "CHAMBRE"

$`3`
[1] "PRIX"
> res.dendro <- as.dendrogram(as.hclust(res.cash))
> plot(res.dendro, horiz = TRUE, center = TRUE, nodePar = list(lab.cex = 0.6,
+     lab.col = "darkblue", pch = NA), main = deparse(res.cash$call))
> d.thotels <- dist(t(hotelsnum), "euclidean")
> h.thotels <- hclust(d.thotels, method = "complete")
> Colvar <- factor(rep("Groupe", 7))
> hubertgamma <- rep(0, 5)
> for (i in 1:5) {
+     hubertgamma[i] <- cluster.stats(d.thotels, cutree(h.thotels,
+         k = i + 1), G2 = FALSE, G3 = FALSE, silhouette = FALSE)$hubertgamma
+ }
> A2Rplot(h.thotels, k = 3, fact.sup = Colvar, criteria = hubertgamma,
+     boxes = FALSE, col.up = "grey", col.down = c("orange",
+         "royalblue", "green3", "red2", "purple"))
```

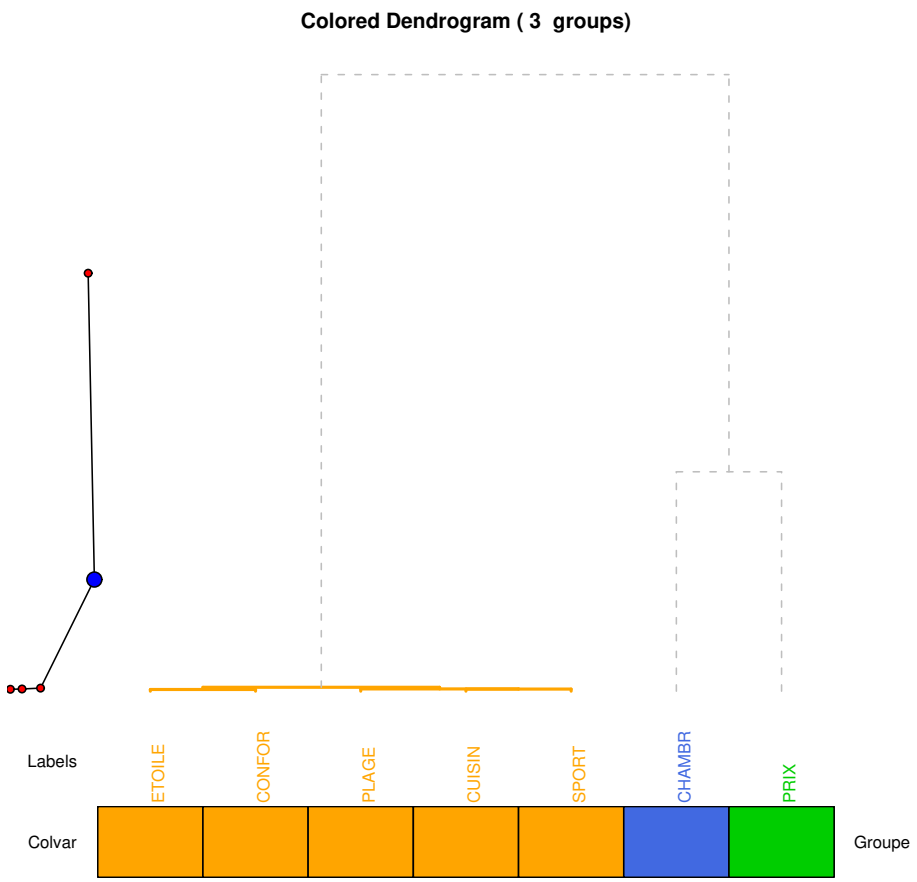



```
> res.cash <- agnes(thotelsnum, metric = "euclidean",
+     method = "ward")
> split(rownames(thotelsnum), cutree(res.cash, k = 3))
```

```
$`1`
[1] "ETOILE"  "CONFORT" "CUISINE" "SPORT"   "PLAGE"

$`2`
[1] "CHAMBRE"

$`3`
[1] "PRIX"
```

```
> res.dendro <- as.dendrogram(as.hclust(res.cash))
> plot(res.dendro, horiz = TRUE, center = TRUE, nodePar = list(lab.cex = 0.6,
+     lab.col = "darkblue", pch = NA), main = deparse(res.cash$call))
```

```
> d.thotels <- dist(t(hotelsnum), "euclidean")
> h.thotels <- hclust(d.thotels, method = "ward")
> Colvar <- factor(rep("Groupe", 7))
> hubertgamma <- rep(0, 5)
> for (i in 1:5) {
+     hubertgamma[i] <- cluster.stats(d.thotels, cutree(h.thotels,
+         k = i + 1), G2 = FALSE, G3 = FALSE, silhouette = FALSE)$hubertgamma
+ }
> A2Rplot(h.thotels, k = 3, fact.sup = Colvar, criteria = hubertgamma,
+     boxes = FALSE, col.up = "grey", col.down = c("orange",
+         "royalblue", "green3", "red2", "purple"))
```

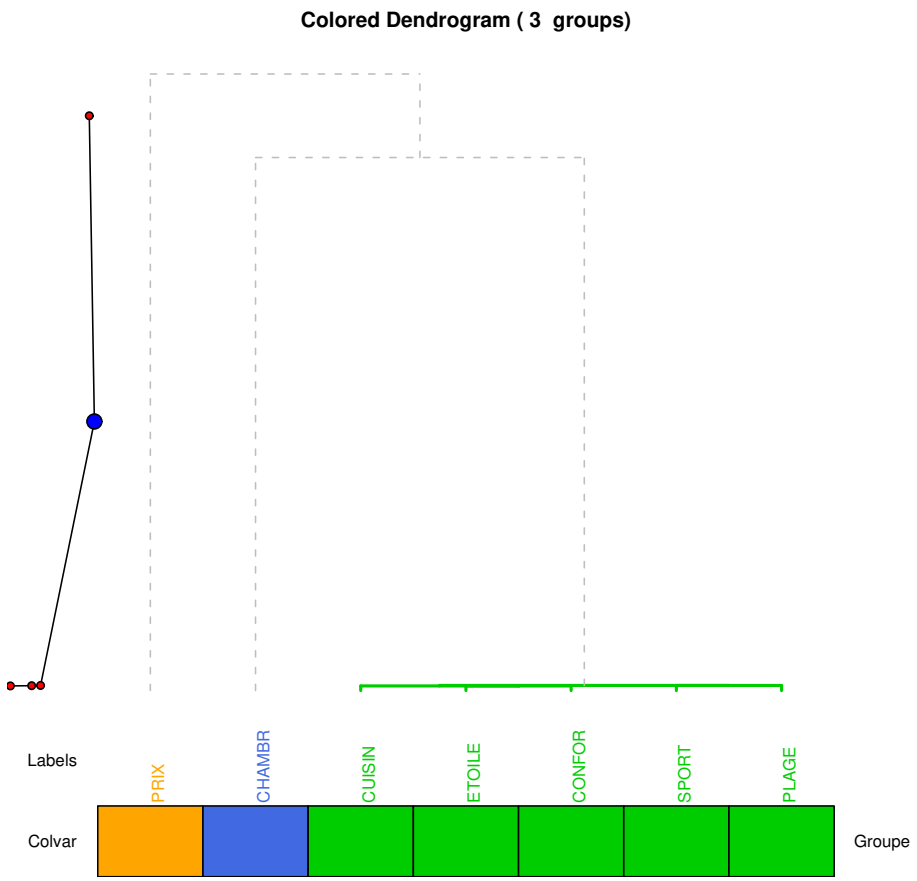


```
> res.cash <- agnes(thotelsnum, metric = "manhattan",
+     method = "single")
> split(rownames(thotelsnum), cutree(res.cash, k = 3))
```

```
$`1`
[1] "ETOILE"  "CONFORT" "CUISINE" "SPORT"   "PLAGE"

$`2`
[1] "CHAMBRE"

$`3`
[1] "PRIX"
```

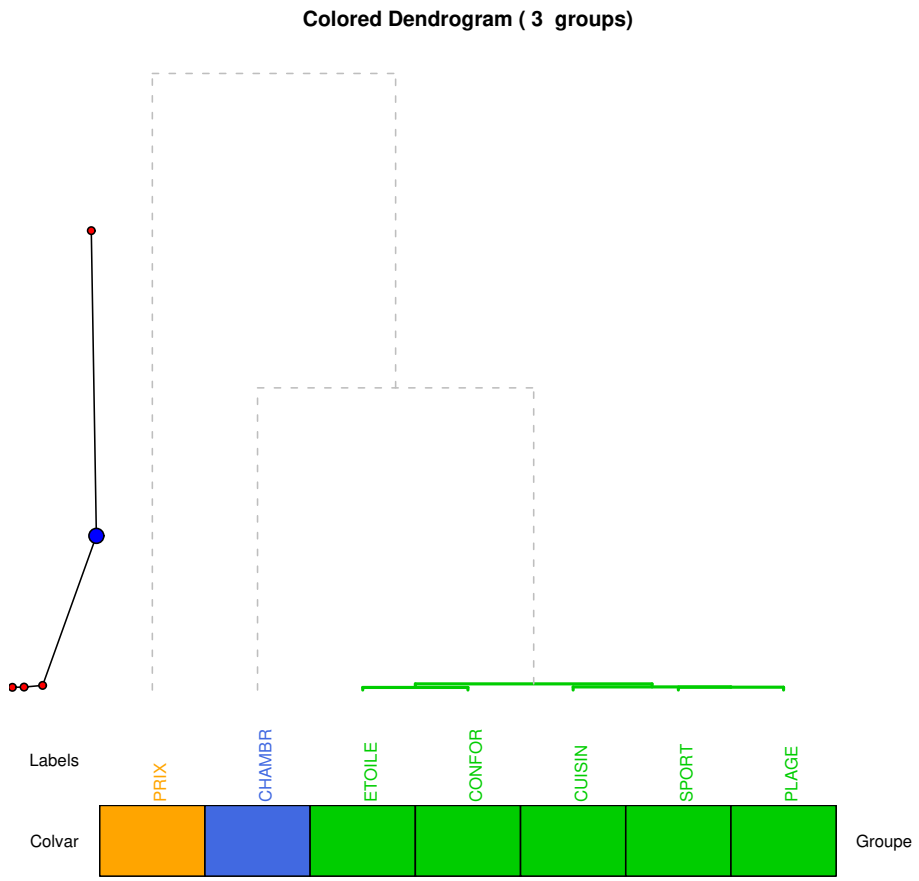
```
> res.dendro <- as.dendrogram(as.hclust(res.cash))
> plot(res.dendro, horiz = TRUE, center = TRUE, nodePar = list(lab.cex = 0.6,
+     lab.col = "darkblue", pch = NA), main = deparse(res.cash$call))
```

```
> d.thotels <- dist(t(hotelsnum), "manhattan")
> h.thotels <- hclust(d.thotels, method = "single")
> Colvar <- factor(rep("Groupe", 7))
> hubertgamma <- rep(0, 5)
> for (i in 1:5) {
+     hubertgamma[i] <- cluster.stats(d.thotels, cutree(h.thotels,
+         k = i + 1), G2 = FALSE, G3 = FALSE, silhouette = FALSE)$hubertgamma
+ }
> A2Rplot(h.thotels, k = 3, fact.sup = Colvar, criteria = hubertgamma,
+     boxes = FALSE, col.up = "grey", col.down = c("orange",
+         "royalblue", "green3", "red2", "purple"))
```



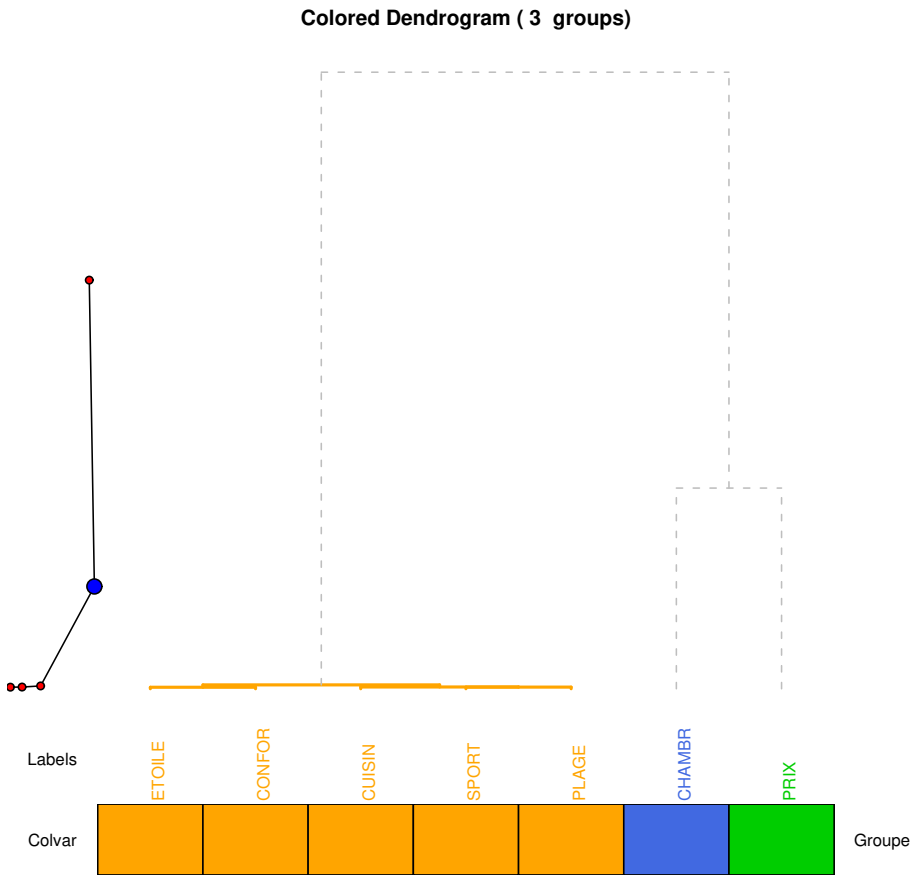

```
> res.cash <- agnes(thotelsnum, metric = "manhattan",
+     method = "complete")
> split(rownames(thotelsnum), cutree(res.cash, k = 3))
$`1`
[1] "ETOILE"  "CONFORT" "CUISINE" "SPORT"   "PLAGE"

$`2`
[1] "CHAMBRE"

$`3`
[1] "PRIX"
> res.dendro <- as.dendrogram(as.hclust(res.cash))
> plot(res.dendro, horiz = TRUE, center = TRUE, nodePar = list(lab.cex = 0.6,
+     lab.col = "darkblue", pch = NA), main = deparse(res.cash$call))
> d.thotels <- dist(t(hotelsnum), "manhattan")
> h.thotels <- hclust(d.thotels, method = "complete")
> Colvar <- factor(rep("Groupe", 7))
> hubertgamma <- rep(0, 5)
> for (i in 1:5) {
+     hubertgamma[i] <- cluster.stats(d.thotels, cutree(h.thotels,
+         k = i + 1), G2 = FALSE, G3 = FALSE, silhouette = FALSE)$hubertgamma
+ }
> A2Rplot(h.thotels, k = 3, fact.sup = Colvar, criteria = hubertgamma,
```

```
+     boxes = FALSE, col.up = "grey", col.down = c("orange",
+         "royalblue", "green3", "red2", "purple"))
```




```
> res.cash <- agnes(thotelsnum, metric = "manhattan",
+     method = "ward")
> split(rownames(thotelsnum), cutree(res.cash, k = 3))
$`1`
[1] "ETOILE"  "CONFORT" "CUISINE" "SPORT"   "PLAGE"

$`2`
[1] "CHAMBRE"

$`3`
[1] "PRIX"
> res.dendro <- as.dendrogram(as.hclust(res.cash))
> plot(res.dendro, horiz = TRUE, center = TRUE, nodePar = list(lab.cex = 0.6,
+     lab.col = "darkblue", pch = NA), main = deparse(res.cash$call))
> d.thotels <- dist(t(hotelsnum), "manhattan")
> h.thotels <- hclust(d.thotels, method = "ward")
> Colvar <- factor(rep("Groupe", 7))
> hubertgamma <- rep(0, 5)
> for (i in 1:5) {
+     hubertgamma[i] <- cluster.stats(d.thotels, cutree(h.thotels,
+         k = i + 1), G2 = FALSE, G3 = FALSE, silhouette = FALSE)$hubertgamma
```

```
+ }
> A2Rplot(h.thotels, k = 3, fact.sup = Colvar, criteria = hubertgamma,
+     boxes = FALSE, col.up = "grey", col.down = c("orange",
+         "royalblue", "green3", "red2", "purple"))
```

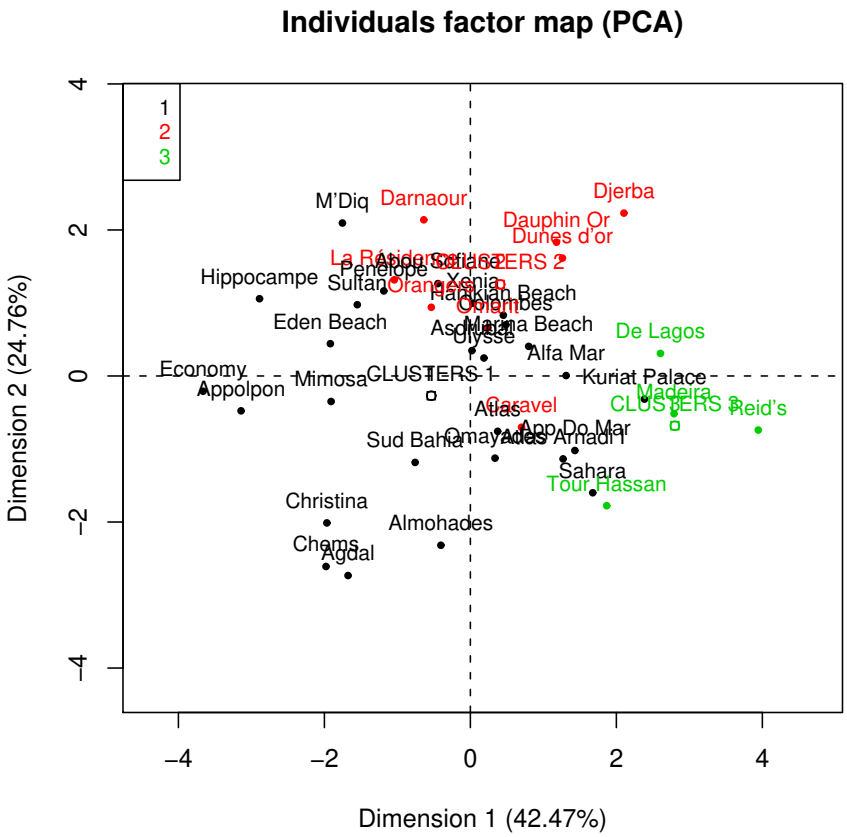


Partie II : K-moyennes

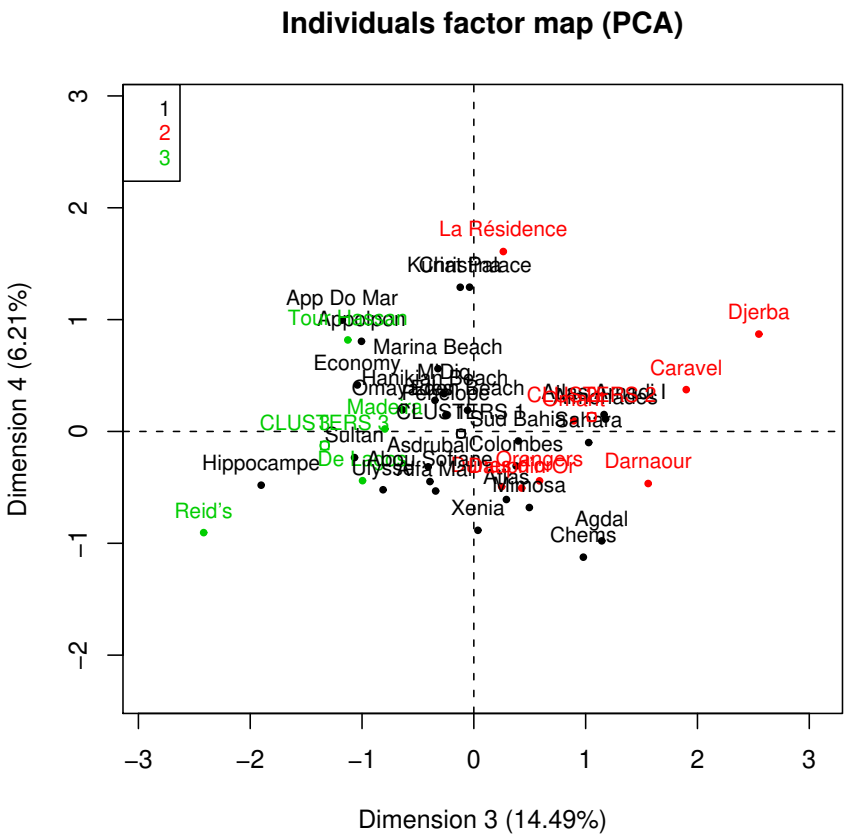
1. Obtenir la classification des hôtels en trois groupes à l'aide de la méthode des $K-$moyennes qui portera sur toutes les variables du tableau. Représenter graphiquement les trois groupes sur les premier et second plans factoriels qui ont été déterminés au TD 5. Qu'observe-t-on ? Comment se répartissent les groupes ?

```
> clhotel <- kmeans(hotelsnum, 3, nstart = 50)
> colhotnum <- cbind(factor(clhotel$cluster), hotelsnum)
> colcennum <- cbind(factor(1:3), clhotel$centers)
> colnames(colhotnum) <- c("CLUSTERS", colnames(hotelsnum))
> colnames(colcennum) <- c("CLUSTERS", colnames(hotelsnum))
> datas <- rbind(colcennum, colhotnum)
> datas$CLUSTERS <- factor(datas$CLUSTERS)
> plot(hotelsnum, col = colhotnum$CLUSTERS)
> library(FactoMineR)
```

```
> res.pca <- PCA(datas, graph = FALSE, quali.sup = 1,
+     ind.sup = 1:3)
> plot(res.pca, habillage = 1, new.plot = FALSE, cex = 0.8)
```

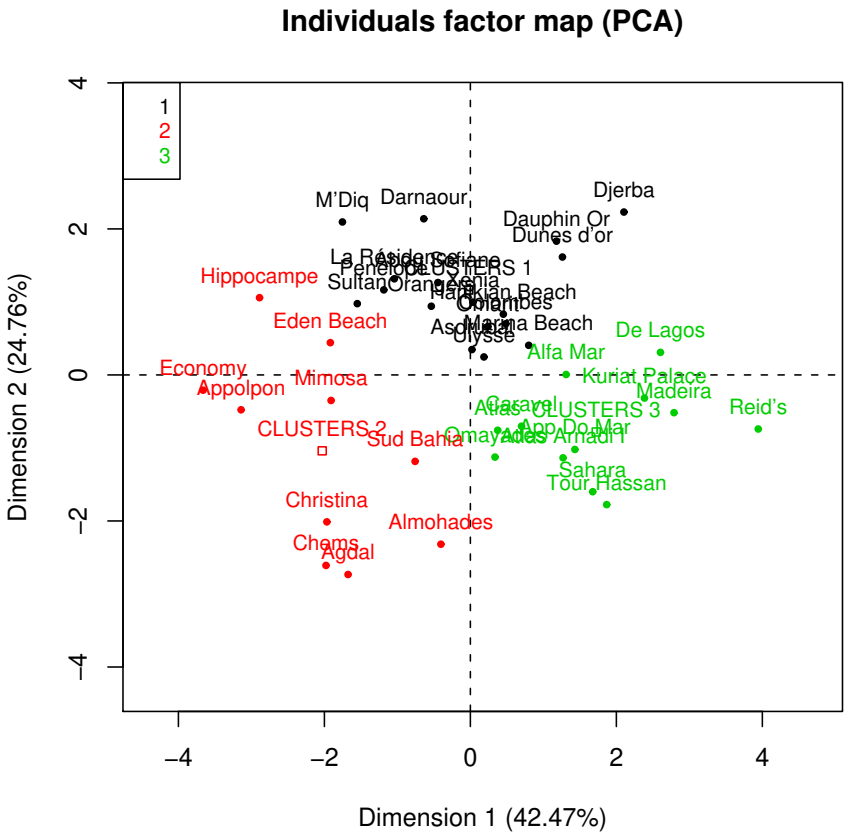



```
> plot(res.pca, axes = c(3, 4), habillage = 1, new.plot = FALSE,
+     cex = 0.8)
```

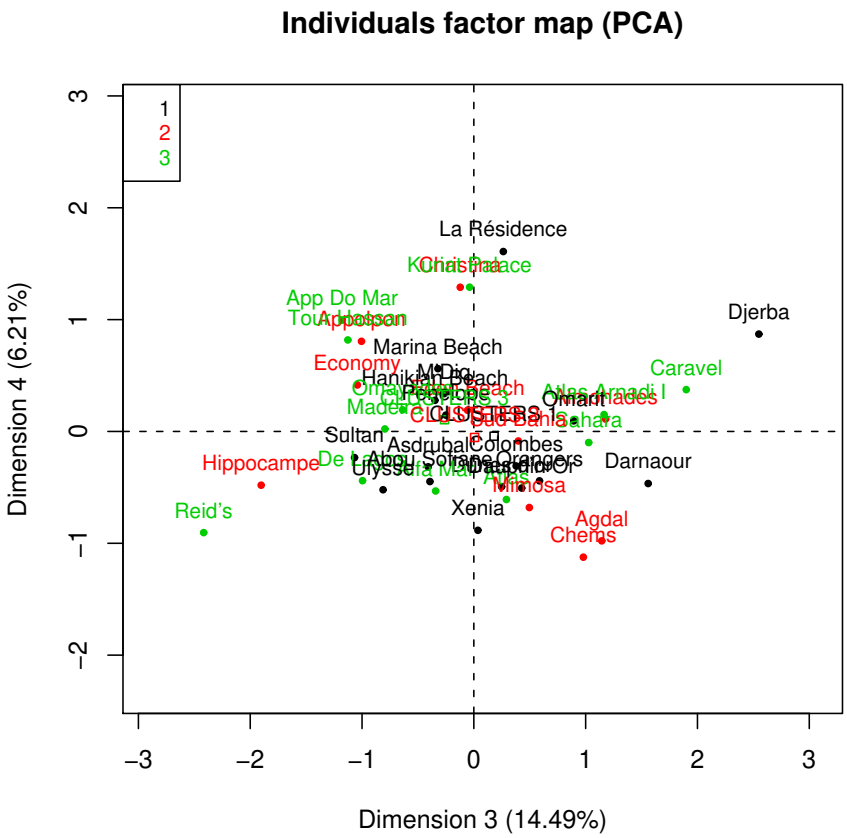


2. Obtenir la classification des hôtels en trois groupes à l'aide de la méthode des $K-$moyennes qui portera sur les coordonnées des hôtels dans le premier plan factoriel. Représenter graphiquement ces trois nouveaux groupes sur le premier plan factoriel. Qu'observe-t-on ? Comment se répartissent les groupes ?

```
> clhotel2 <- kmeans(res.pca$ind$coord[, 1:4], 3, nstart = 50)
> colhotnum2 <- data.frame(cbind(factor(clhotel2$cluster),
+     hotelsnum))
> colnames(colhotnum2) <- c("CLUSTERS", colnames(hotelsnum))
> datas2 <- colhotnum2
> datas2$CLUSTERS <- factor(datas2$CLUSTERS)
> plot(hotelsnum, col = colhotnum2$CLUSTERS)

> res.pca2 <- PCA(datas2, graph = FALSE, quali.sup = 1)
> plot(res.pca2, habillage = 1, new.plot = FALSE, cex = 0.8)
```



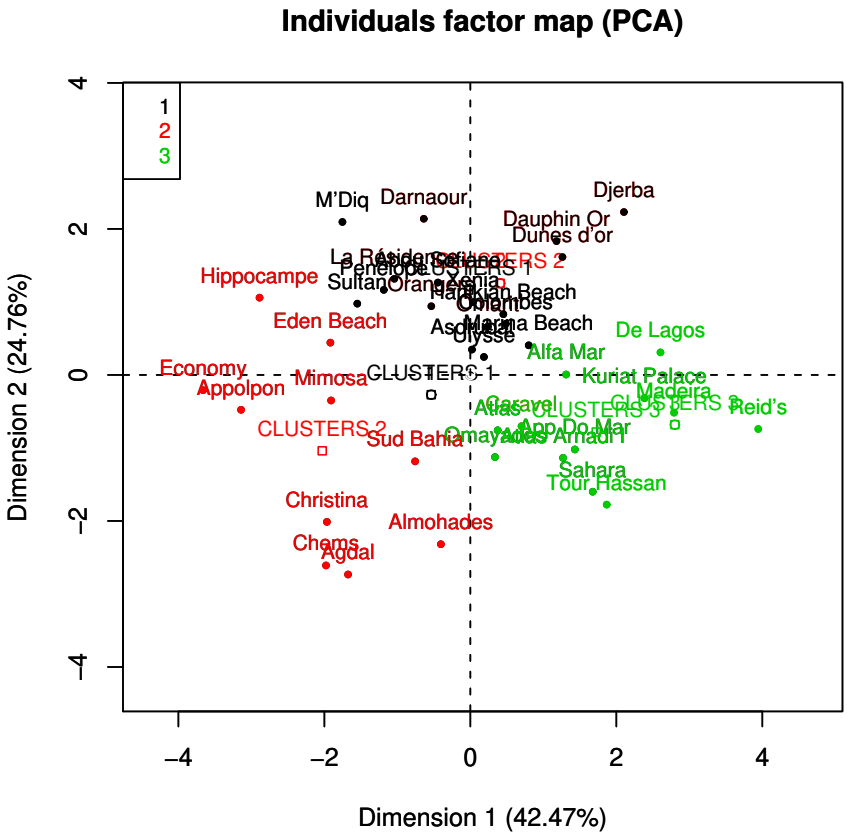

```
> plot(res.pca2, axes = c(3, 4), habillage = 1, new.plot = FALSE,
+     cex = 0.8)
```

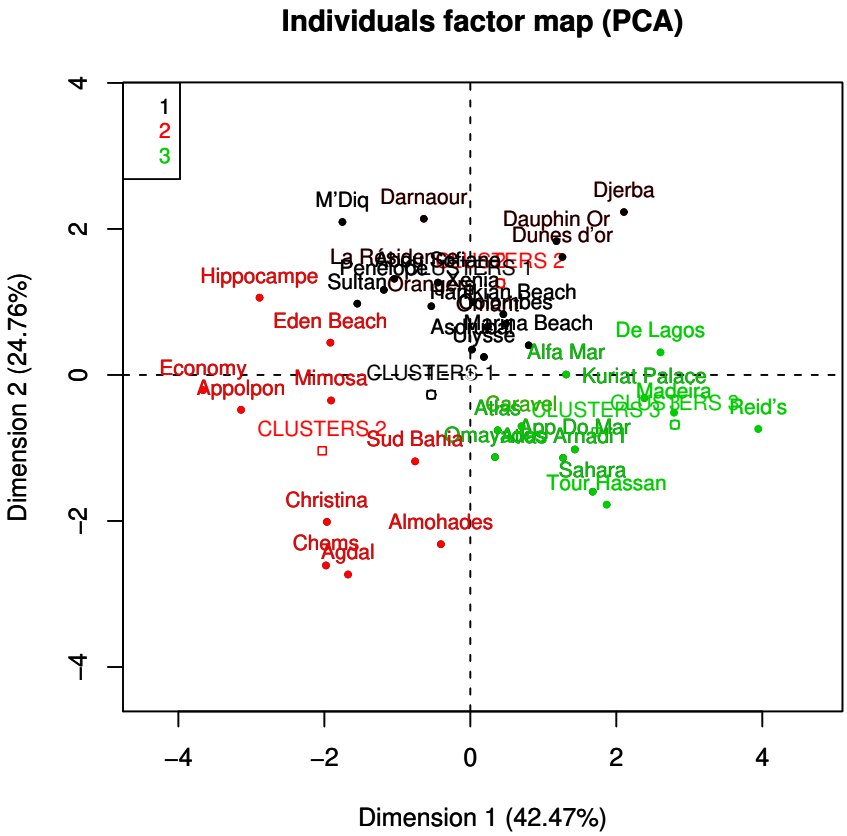


3. Quelles sont les différences de classement entre les deux classements de la question 1. et de la question 2. ? Le premier plan factoriel traduit-il fidèlement l'ensemble des données ? On pourra se réferrer au diagramme des valeurs propres obtenu au TD 5.

```
> layout(1:2)
> plot(PCA(datas, graph = FALSE, quali.sup = 1, ind.sup = 1:3),
+     habillage = 1, new.plot = FALSE, cex = 0.8)
> plot(PCA(datas2, graph = FALSE, quali.sup = 1), habillage = 1,
+     new.plot = FALSE, cex = 0.8)
> layout(1)
```


```
> layout(1:2)
> plot(PCA(datas, graph = FALSE, quali.sup = 1, ind.sup = 1:3),
+     habillage = 1, new.plot = FALSE, cex = 0.8)
> plot(PCA(datas2, graph = FALSE, quali.sup = 1), habillage = 1,
+     new.plot = FALSE, cex = 0.8)
> layout(1)
```


Individuals factor map (PCA)
1
2
3
M'Diq
Darnaour
Djerba
Dauphin Or
Dunes d'or
Hippocampe
Eden Beach
Economy
Appolpon
Mimosa
CLUSTERS 1
CLUSTERS 2
Sud Bahia
Christina
Almohades
Chems
Agdal
Alfa Mar
De Lagos
Madeira
Reid's
App Do Mar
Sahara
Tour Hassan
Dimension 1 (42.47%)
Dimension 2 (24.76%)
−4
−2
0
2
4

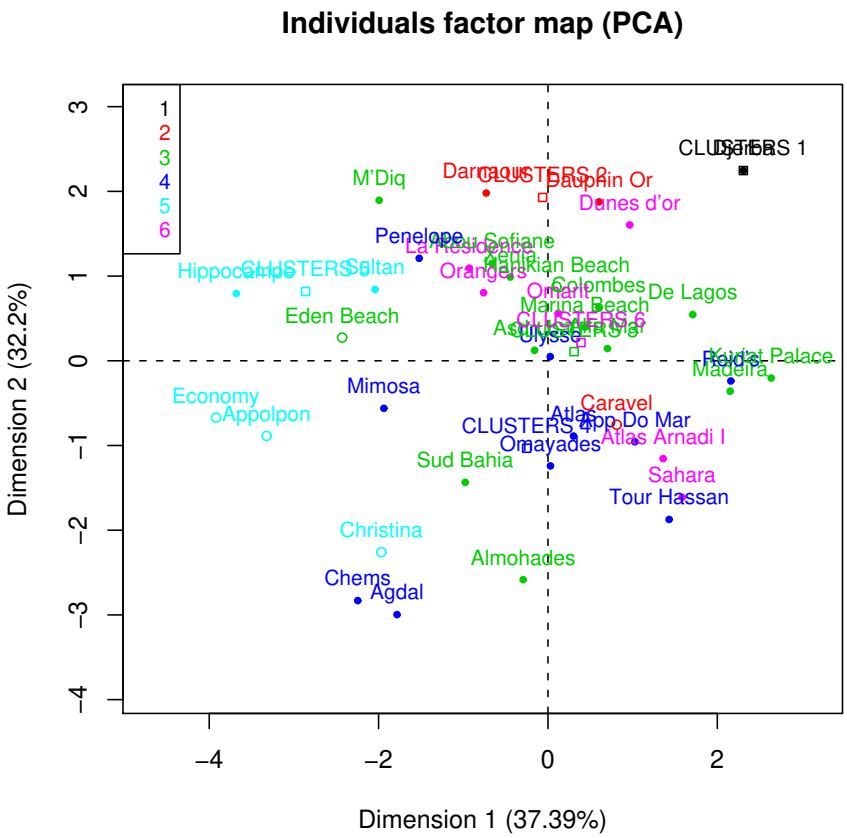


4. On décide de vérifier si l'attribution des étoiles est conforme aux critères de constitution des groupes par la méthode des $K-$moyennes. Puisqu'il existe 6 catégories d'étoiles, de 0 à 5, classer les hôtels en 6 groupes à l'aide de la méthode des $K-$moyennes portant cette fois-ci sur toutes les variables à l'exclusion de la variable prix. Attention les groupes obtenus ne sont pas nécessairement numérotés par ordre croissant des étoiles.

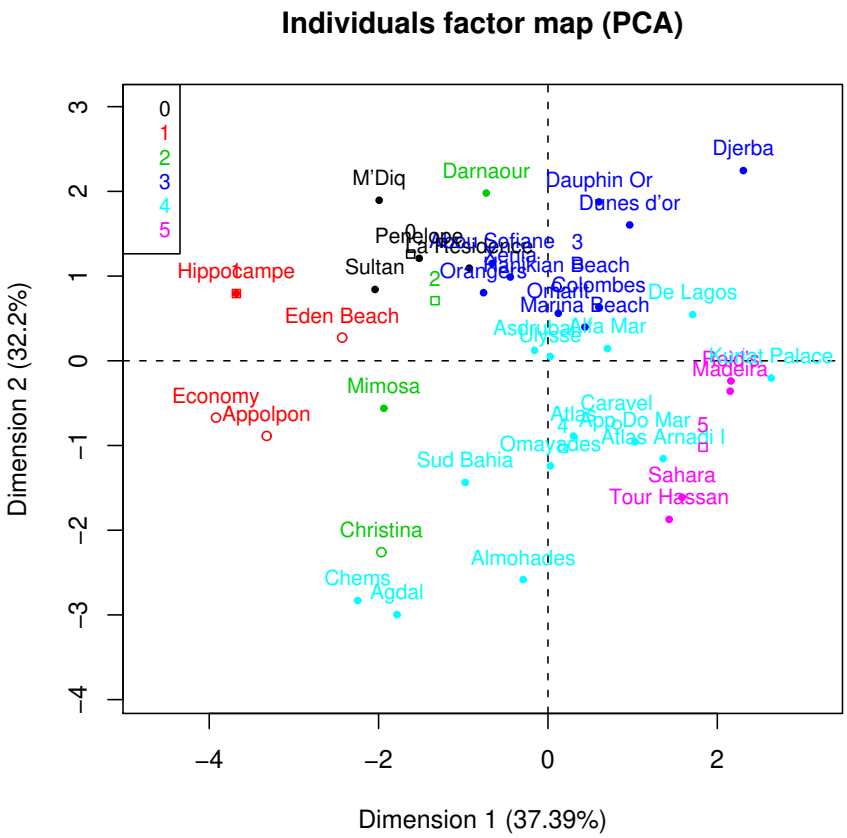
```
> hotelsnum2 <- hotelsnum[, -7]
> clhotel3 <- kmeans(hotelsnum2, 6, nstart = 50)
> colhotnum3 <- cbind(factor(clhotel3$cluster), factor(hotels$ETOILE),
+     hotelsnum2)
> colnames(colhotnum3) <- c("CLUSTERS", "ETOILE_Q", colnames(hotelsnum2))
> datas3 <- colhotnum3
> datas3$CLUSTERS <- factor(datas3$CLUSTERS)
> plot(hotelsnum2, col = colhotnum3$CLUSTERS)
> cbind(clhotel3$cluster, hotels$ETOILE)
               [,1] [,2]
Appolpon          5    1
Caravel           2    4
Christina         5    2
Economy           5    1
```

```
Eden Beach        3    1
Hanikian Beach    3    3
Marina Beach      3    3
Xenia             3    3
Agdal             4    4
Almohades         3    4
Atlas             4    4
Atlas Arnadi I    6    4
Chems             4    4
Dunes d'or        6    3
La Résidence      6    0
M'Diq             3    0
Omayades          4    4
Sahara            6    5
Sud Bahia         3    4
Tour Hassan       4    5
Alfa Mar          3    4
App Do Mar        4    4
De Lagos          3    4
Madeira           3    5
Reid's            4    5
Abou Sofiane      3    3
Asdrubal          3    4
Colombes          3    3
Darnaour          2    2
Djerba            1    3
Mimosa            4    2
Omarit            6    3
Orangers          6    3
Penelope          4    0
Ulysse            4    4
Dauphin Or        2    3
Hippocampe        5    1
Kuriat Palace     3    4
Sultan            5    0
```

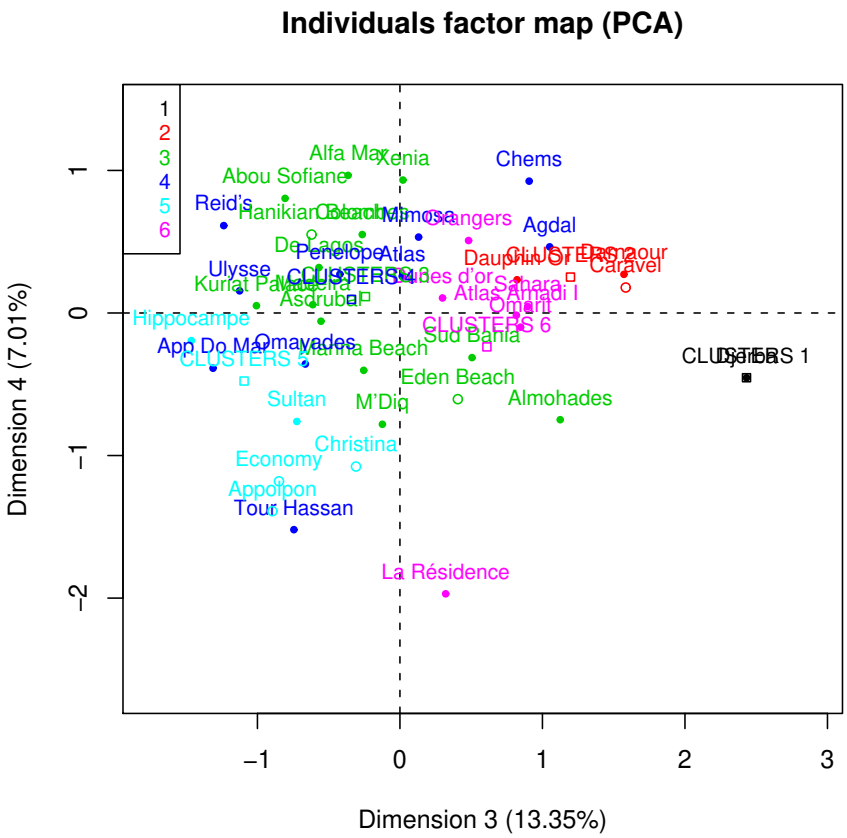
```
> res.pca3 <- PCA(datas3, graph = FALSE, quali.sup = c(1,
+     2), ind.sup = 1:6)
> plot(res.pca3, habillage = 1, new.plot = FALSE, cex = 0.8)
```



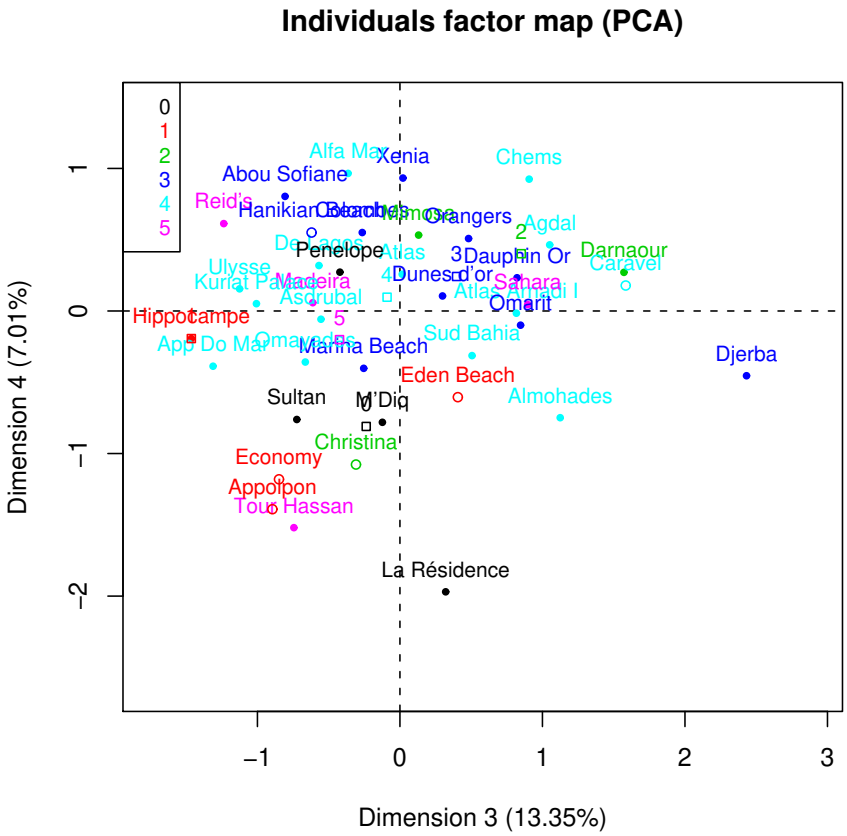

```
> plot(res.pca3, habillage = 2, new.plot = FALSE, cex = 0.8)
```




```
> plot(res.pca3, axes = c(3, 4), habillage = 1, new.plot = FALSE,
+     cex = 0.8)
```




```
> plot(res.pca3, axes = c(3, 4), habillage = 2, new.plot = FALSE,
+     cex = 0.8)
```



5. La fonction `cascadeKM` du package `vegan` s'avère particulièrement utile lorsque nous n'avons pas de connaissance a priori sur le nombre de groupes à constituer en utilisant la méthode des $K-$moyennes. Elle permet en effet de créer des partitions pour un nombre de groupes compris entre deux valeurs fixés par l'utilisateur et de calculer des critères de sélection du nombre de groupes comme les deux suivant :

   (i) Le critère de Calinski et Harabasz[2] est défini par :

$$\text{Critère Calinksi-Harabasz} = \frac{\frac{SSB}{K-1}}{\frac{SSW}{n-K}}, \tag{1}$$

   où n est le nombre de valeurs du jeu de données et K le nombre de groupes. SSW est la somme des carrés intra-groupe tandis que SSB est la somme des carrés inter-groupe. Cette indice n'est autre qu'une statistique F d'analyse de la variance.

   (ii) Le critère SSI[3] (Simple Structure Index) est le résultat de la multiplication de plusieurs éléments mesurant l'interprétabilité de la partition

[2]Calinski, T. and J. Harabasz. 1974. A dendrite method for cluster analysis. *Commun. Stat.* **3** : 1-27.

[3]Dolnicar, S., K. Grabler and J. A. Mazanec. 1999. A tale of three cities : perceptual charting for analyzing destination images. Pp. 39-62 in : Woodside, A. et al. [eds.] *Consumer psychology of tourism, hospitality and leisure.* CAB International, New York.

calculée. La meilleure des partition est celle pour laquelle la valeur du critère SSI est la plus forte.

Il y a donc le choix entre trois options : `criterion = "calinski"`, `criterion = "ssi"` et `criterion = "all"`.
Les options `inf.gr` et `sup.gr` permettent de spécifier les valeurs limites inférieure et supérieur des nombres de groupes pour lesquels des partitions seront calculées.
L'option `iter` permet de choisir le nombre de configurations initiales choisies aléatoirement pour chaque nombre de groupe pour lequel une partition a été obtenue.
La fonction `plot.cascadeKM` permet de représenter synthétiquement les objets de classe `cascadeKM`, c'est-à-dire les résultats de la fonction `cascadeKM`. `min.g` et `max.g` indiquent les valeurs limites du nombre de groupes qui seront représentés. `grpmts.plot = TRUE` spécifie que le graphique doit être affiché. Ceci est la valeur par défaut. `sortg = TRUE` indique que les groupes doivent être réordonnés. Ils le sont de la manière suivante :

(i) Une matrice de distance `simple matching`[4] est calculée entre les objets en fonction de leur appartenance à aux groupes obtenus par $K-$moyennes, pour K allant de `min.g` à `max.g`.

(ii) La première coordonnée principale (PCoA[5]) est déterminée à l'aide de la matrice de distance centrée.

(iii) La première coordonnée principale est utilisée pour ordonner les objets sur le graphique.

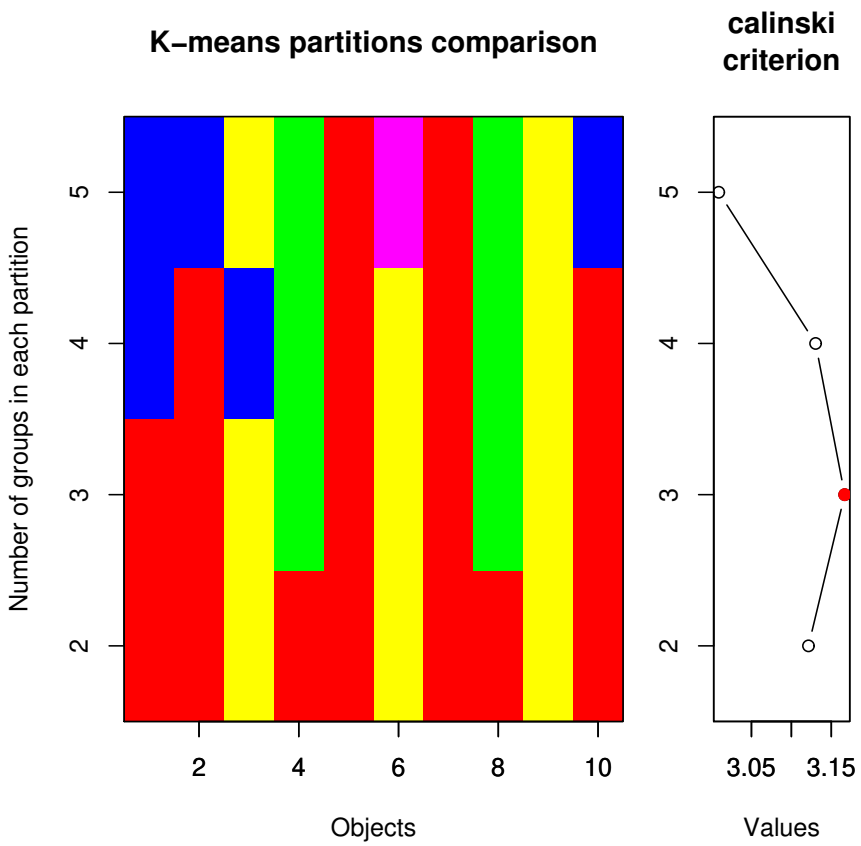
Le comportement par défaut est `sortg = FALSE`. `gridcol = NA`

```
> library(vegan)
> mat1 <- matrix(runif(100), 10, 10)
> res1 <- cascadeKM(mat1, 2, 5, iter = 25, criterion = "calinski")
> plot1 <- plot(res1)
```

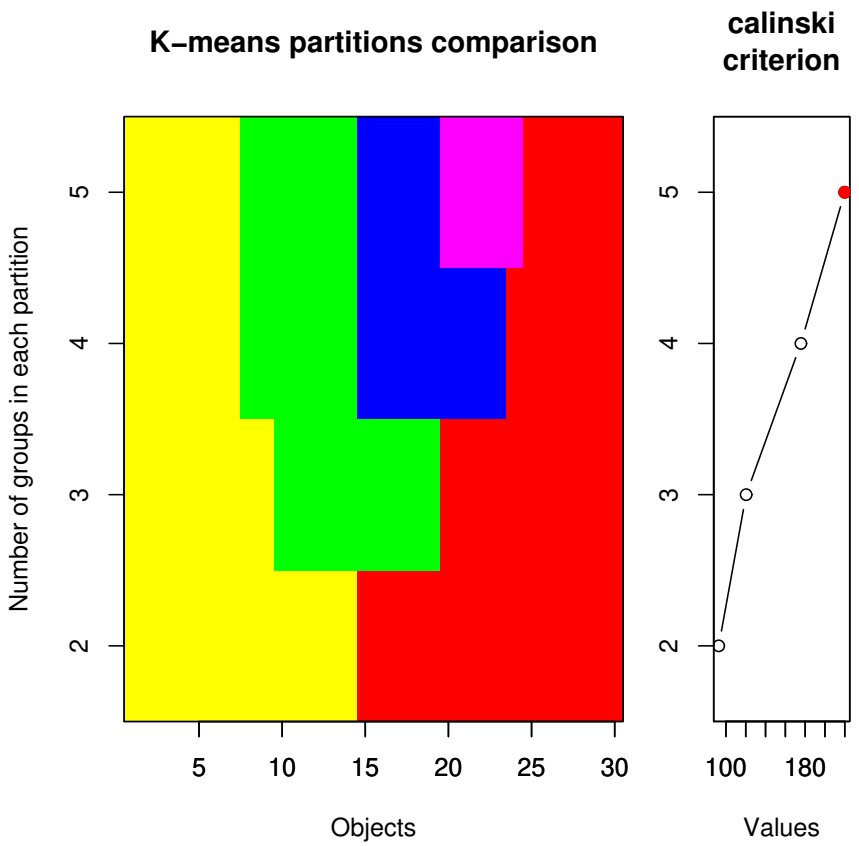
[4]La distance `simple matching` entre deux objets $x_i = (x_{i,1}, \ldots, x_{i,p})$ et $x_j = (x_{j,1}, \ldots, x_{j,p})$ est :

$$d(x_i, x_j) = \frac{\sum_{k=1}^{p} \mathbf{1}(x_{i,k} = x_{j,k} = 1) + \sum_{k=1}^{p} \mathbf{1}(x_{i,k} = x_{j,k} = 0)}{p}. \tag{2}$$

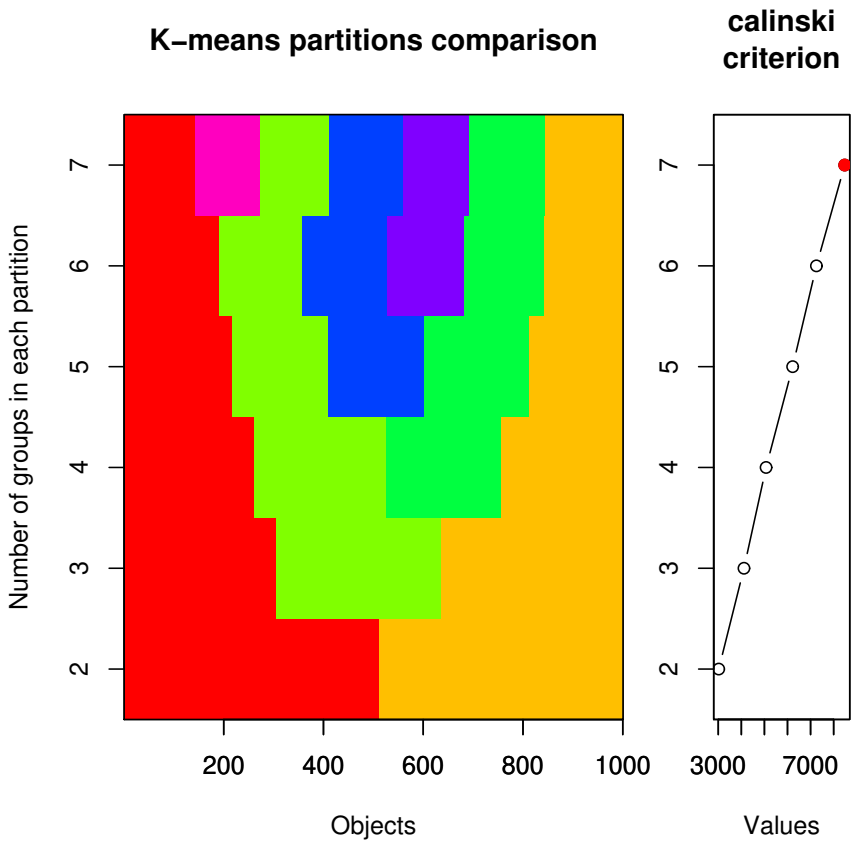
[5]Gower, J. C. 1966. Some distance properties of latent root and vector methods used in multivariate analysis. *Biometrika* **53** : 325-338.




```
> vec2 <- sort(matrix(runif(30), 30, 1))
> res2 <- cascadeKM(vec2, 2, 5, iter = 25, criterion = "calinski")
> plot2 <- plot(res2)
```




```
> vec3 <- sort(matrix(runif(1000), 1000, 1))
> res3 <- cascadeKM(vec3, 2, 7, iter = 25, criterion = "calinski")
> plot3 <- plot(res3, gridcol = NA)
```


K−means partitions comparison
calinski criterion
Number of groups in each partition
2
3
4
5
6
7
200
400
600
800
1000
Objects
3000
7000
Values

NUMERO	NOM	PAYS	ETOILE	CONFORT	CHAMBRE	CUISINE	SPORT	PLAGE	PRIX
1	Appolpon	Grèce	1	4	56	2	0	8	390
2	Caravel	Grèce	4	7	471	7	6	5	468
3	Christina	Grèce	2	7	93	3	0	5	427
4	Economy	Grèce	1	3	56	1	0	8	369
5	Eden Beach	Grèce	1	4	286	3	4	7	499
6	Hanikian Beach	Grèce	3	6	282	5	10	10	526
7	Marina Beach	Grèce	3	6	310	7	7	10	587
8	Xenia	Grèce	3	4	300	6	10	8	534
9	Agdal	Maroc	4	5	146	5	1	0	447
10	Almohades	Maroc	4	6	250	8	0	3	482
11	Atlas	Maroc	4	5	196	9	6	6	511
12	Atlas Arnadi I	Maroc	4	7	324	10	6	5	532
13	Chems	Maroc	4	5	138	3	2	0	450
14	Dunes d'or	Maroc	3	4	400	10	10	10	569
15	La Résidence	Maroc	0	5	366	7	4	10	419
16	M'Diq	Maroc	0	3	300	5	7	10	421
17	Omayades	Maroc	4	6	144	7	4	8	579
18	Sahara	Maroc	5	7	330	10	5	5	598
19	Sud Bahia	Maroc	4	5	260	5	2	6	495
20	Tour Hassan	Maroc	5	7	170	10	1	10	730
21	Alfa Mar	Portugal	4	6	254	7	10	8	646
22	App,Do Mar	Portugal	4	8	140	7	6	10	652
23	De Lagos	Portugal	4	6	273	10	10	10	802
24	Madeira	Portugal	5	7	260	10	8	10	761
25	Reid's	Portugal	5	7	169	10	10	10	1101
26	Abou Sofiane	Tunisie	3	4	225	5	10	10	434
27	Asdrubal	Tunisie	4	4	225	7	6	10	489
28	Colombes	Tunisie	3	5	250	9	10	8	436

NUMERO	NOM	PAYS	ETOILE	CONFORT	CHAMBRE	CUISINE	SPORT	PLAGE	PRIX
29	Darnaour	Tunisie	2	3	550	6	9	8	399
30	Djerba	Tunisie	3	6	800	10	10	10	477
31	Mimosa	Tunisie	2	4	150	5	6	4	375
32	Omarit	Tunisie	3	5	425	7	7	8	486
33	Orangers	Tunisie	3	4	366	5	8	8	447
34	Penelope	Tunisie	0	5	200	5	10	7	473
35	Ulysse	Tunisie	4	4	130	8	7	10	495
36	Dauphin Or	Turquie	3	4	500	8	10	10	617
37	Hippocampe	Turquie	1	2	50	1	5	10	489
38	Kuriat Palace	Turquie	4	9	232	10	10	10	520
39	Sultan	Turquie	0	3	110	7	6	8	534

. . . . . .