Segmentação

Adilson dos Anjos

Segmentação

Objetivo

- O objetivo dessa aula é apresentar alguns métodos de Segmentação.
- As análises serão realizadas com uso do R;

Pacotes utilizados nessa aula

- FactoMineR
- readxl
- bpca
- graphics
- cluster
- fpc
- ape

Conjuntos de dados utilizados nessa aula

- cafe.xls

Introdução

- Sinônimos de Segmentação:
  - Análise de Agrupamentos;
  - Cluster Analysis.

Aplicações em Análise Sensorial

- Marketing:
  - definição de grupos focais;
  - estratégias de propaganda para cada segmento;
- Economia:
  - análise de grupos de consumidores de um produto;
  - análise de perfis de consumo/preferência;
- Estudos demográficos: agrupamento por regiões;

- Objetivo: formar grupos considerando características que permitam medir a similaridade ou dissimilaridade;
- Os grupos devem ser homogêneos internamente e heterogêneos externamente;

- O número de grupos pode ser definido por algum critério subjetivo;
- "Use sua experiência para decidir sobre o número de grupos!!"

- Com poucas variáveis, pode-se definir grupos por uma simples inspeção gráfica (gráfico de dispersão com duas variáveis)
- Existem várias medidas que podem ser utilizadas para realizar o agrupamento: coeficiente de correlação, alguma medida de associação, distância euclidiana entre outras;

- Distância Euclidiana: considere o vetor x de coordenadas reais $(x_1, x_2, \ldots, x_p)$ descrevendo os objetos que serão agrupados quanto a semelhança.
- A distância euclidiana que indica a proximidade entre dois objetos A e B é definida por:

$$d(A, B) = \sqrt{\sum_{i=1}^{p} (x_i(A) - x_i(B))^2}$$

Existem vários algoritmos para formação de grupos:

- **Técnicas Hierárquicas**:
  - os elementos são classificados em grupos em diferentes etapas, de maneira hierárquica (árvore de classificação);
  - Os agrupamentos são formados a partir de uma matriz de parecença, que é atualizada a cada união de um par de objetos.
  - Neste prodedimento, os objetos individuais vão se juntando sucessivamente.

- **Técnicas de Partição**:
  - Os elementos são agrupados formando uma partição do conjunto como um todo;
  - Fazem o caminho oposto aos métodos hierárquicos aglomerativos.
  - Neste método, um único grupo de objetos é subdividido em dois com a maior distância. Estes subgrupos são então particionados sucessivamente até se obter os objetos individuais.

Método Hierárquico

- Método da Centróide: formar grupos com elementos com a menor distância entre si;
- Método das médias das distâncias:
- Método do Vizinho mais Próximo/Perto (Ligação Simples)
- Método do Vizinho mais Distante/Longe (Ligação Completa)
- Método de Ward (**mais empregado**)

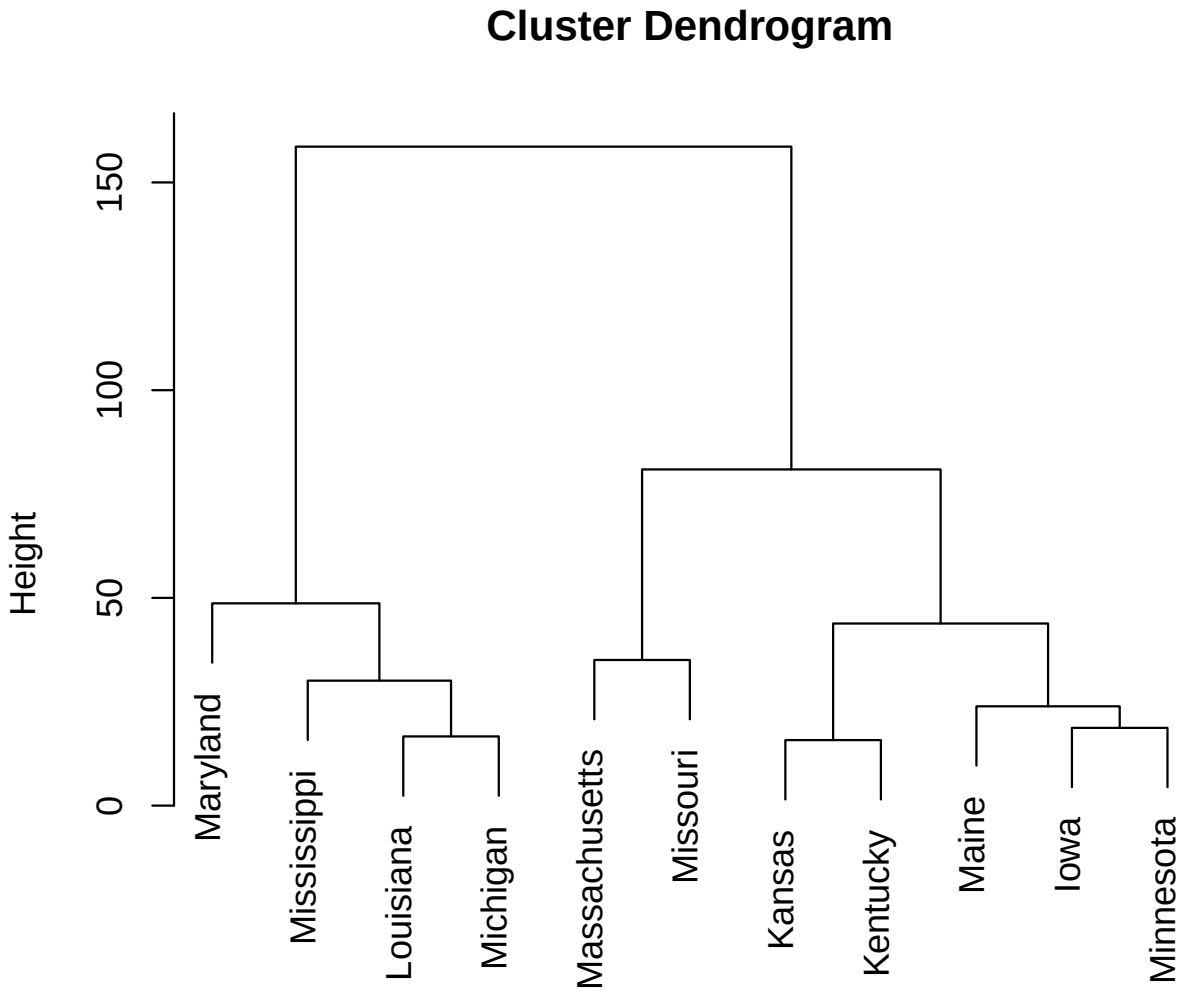
Método Hierárquico: Método de Ward - minimiza a variância dentro do cluster.

- Minimizar a Soma de Quadrados dentro do grupo e maximizar a Soma de quadrados entre Grupos: ANOVA
  - Passo 1: Calcular SQDP para os possíveis (n-1) grupos distintos e selecionar o agrupamento com a menor SQDP;
  - Passo 2: Calcular SQDP para os possíveis (n-2) grupos distintos (fixada a união obtida no Passo 1) e selecionar o agrupamento com a menor SQDP;
  - Os próximos passos consistem na formação de (n-3), (n-4),. . . ,1 grupos, selecionando-se sempre o agrupamento com menor SQDP;
  - O número de grupos é definido em função dos saltos em cada passo.

Métodos de partição (Não Hierárquico) Método das k-médias

- Passo 1: Formação de uma partição inicial. Em geral, adota-se k observações como sementes do algoritmo para formação de k grupos.
- Passo 2: Percorrer a lista de observações e calcular as distâncias de cada uma delas ao CENTRÓIDE (médias) do grupo. Fazer a realocação da observação ao grupo em que ela apresentar menor distância. Recalcular os centróides dos grupos que ganharam e perderam observações.
- Passo 3: Repetir o passo dois até que nenhuma alteração seja feita.
- Passo 4: Em cada passo, pode-se calcular SQDP como função objetivo para avaliação da partição e então procurar identificar novas mudanças que possam levar a uma melhora na partição.

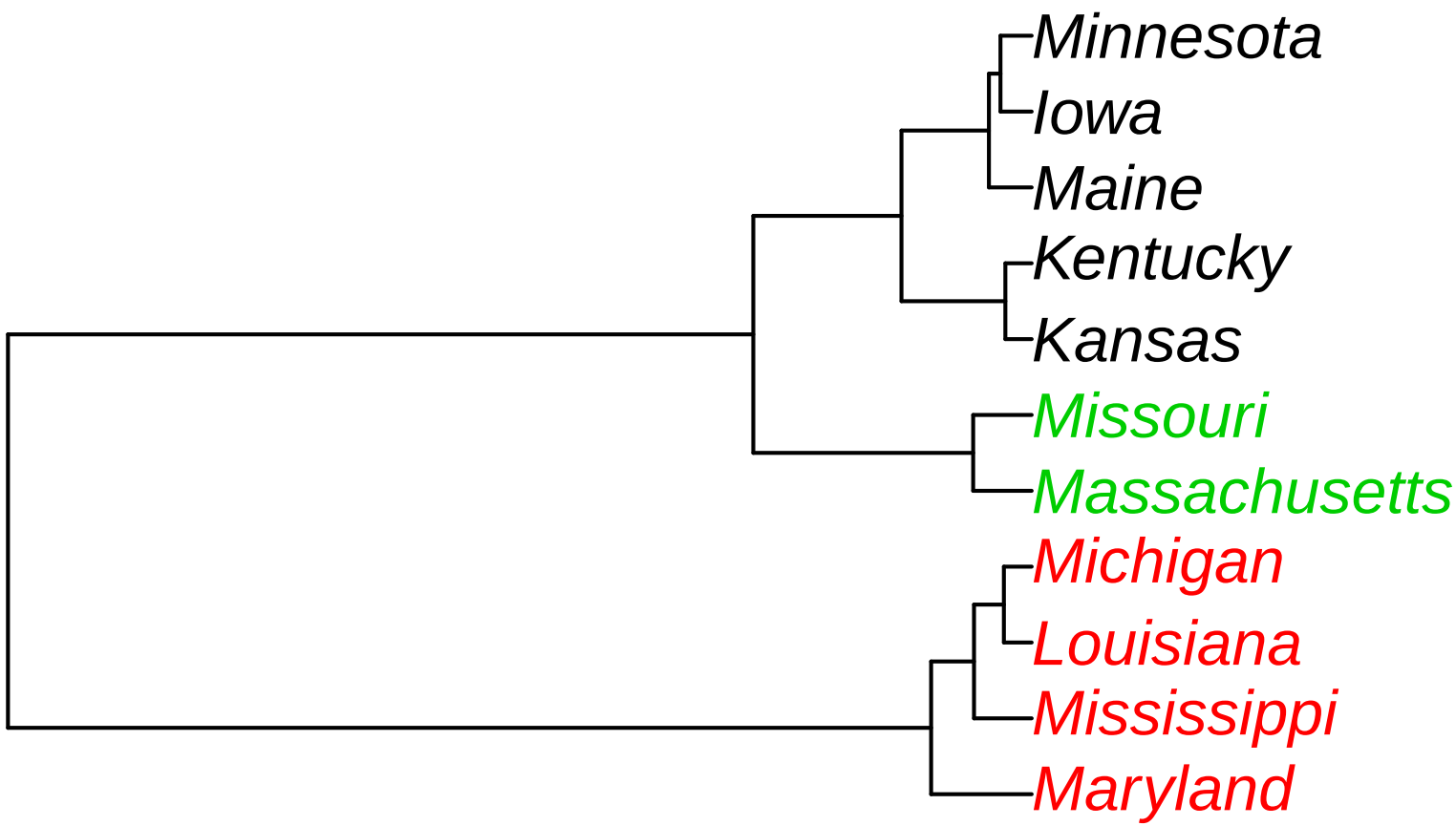
Representação gráfica: Dendrograma.



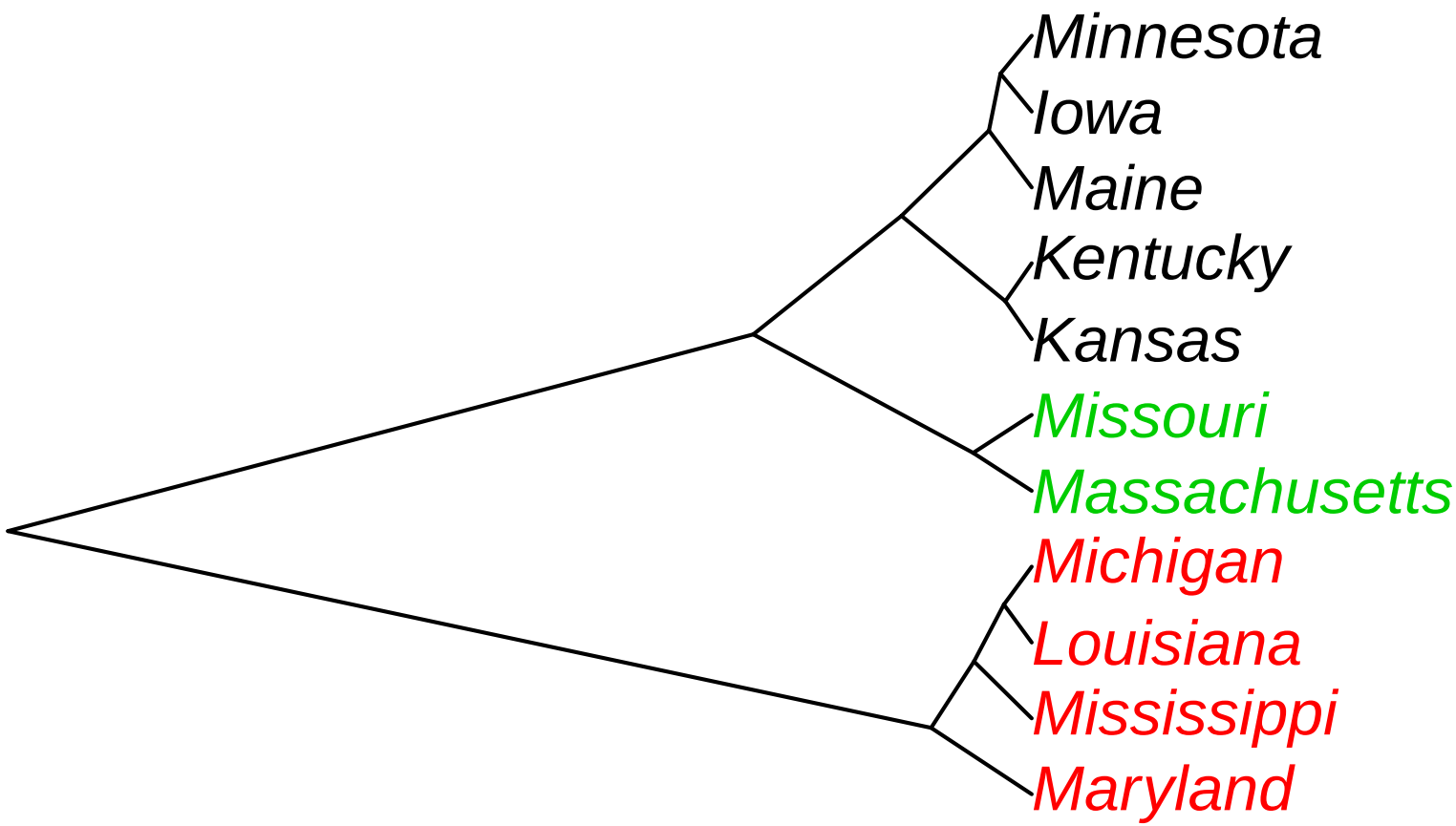
No pacote *ape* do R existem outras formas de representação:

```
library(ape)
hc <- hclust(dist(USArrests[15:25,]), "ward.D")
grupos<-cutree(hc, k=3)
hc.phylo<-as.phylo(hc)
```

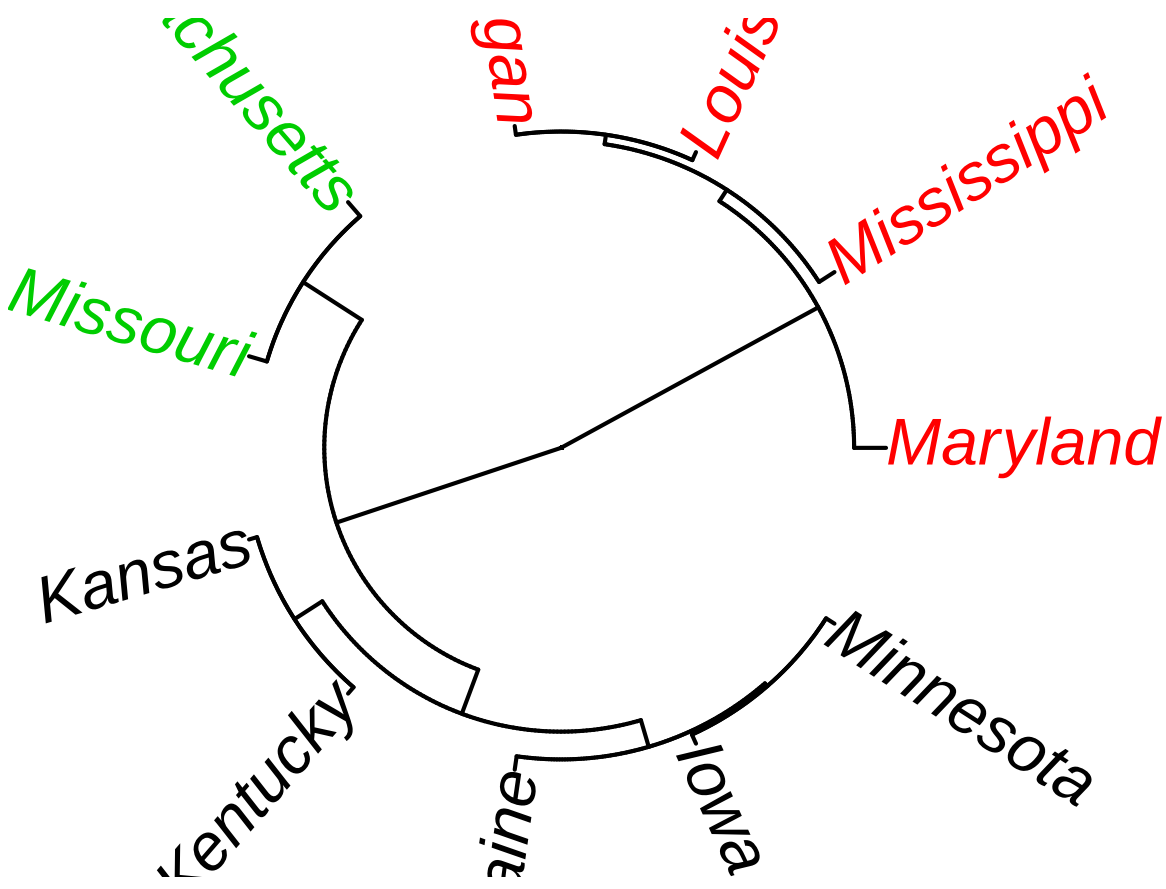
```
plot(hc.phylo, type='phylogram', tip.color=grupos)
```



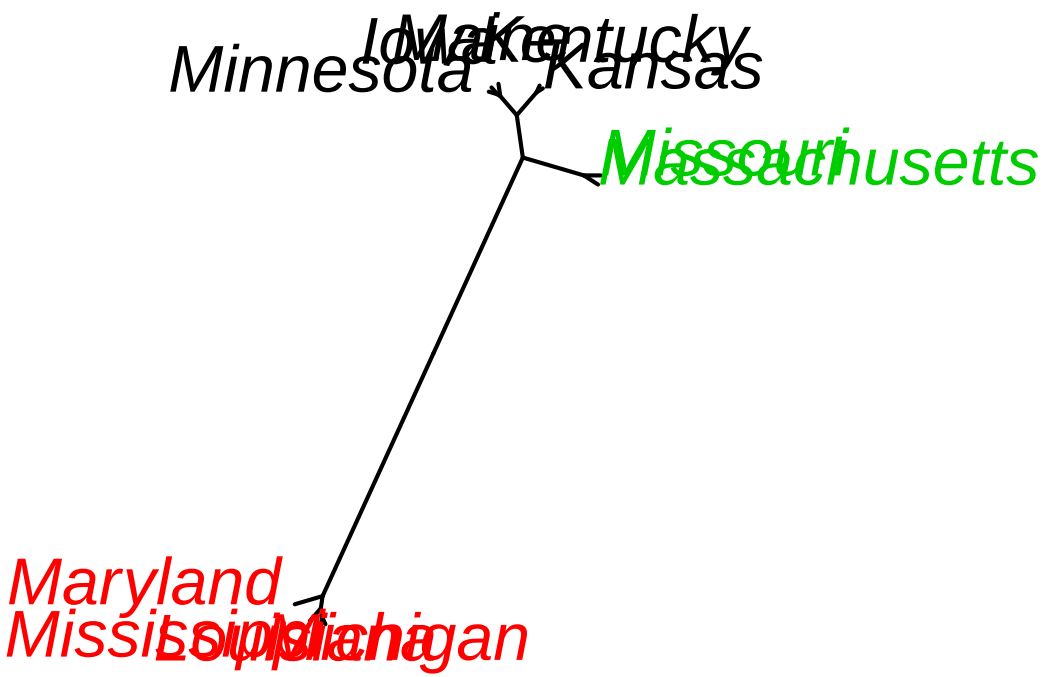
```
plot(hc.phylo, type='cladogram', tip.color=grupos)
```



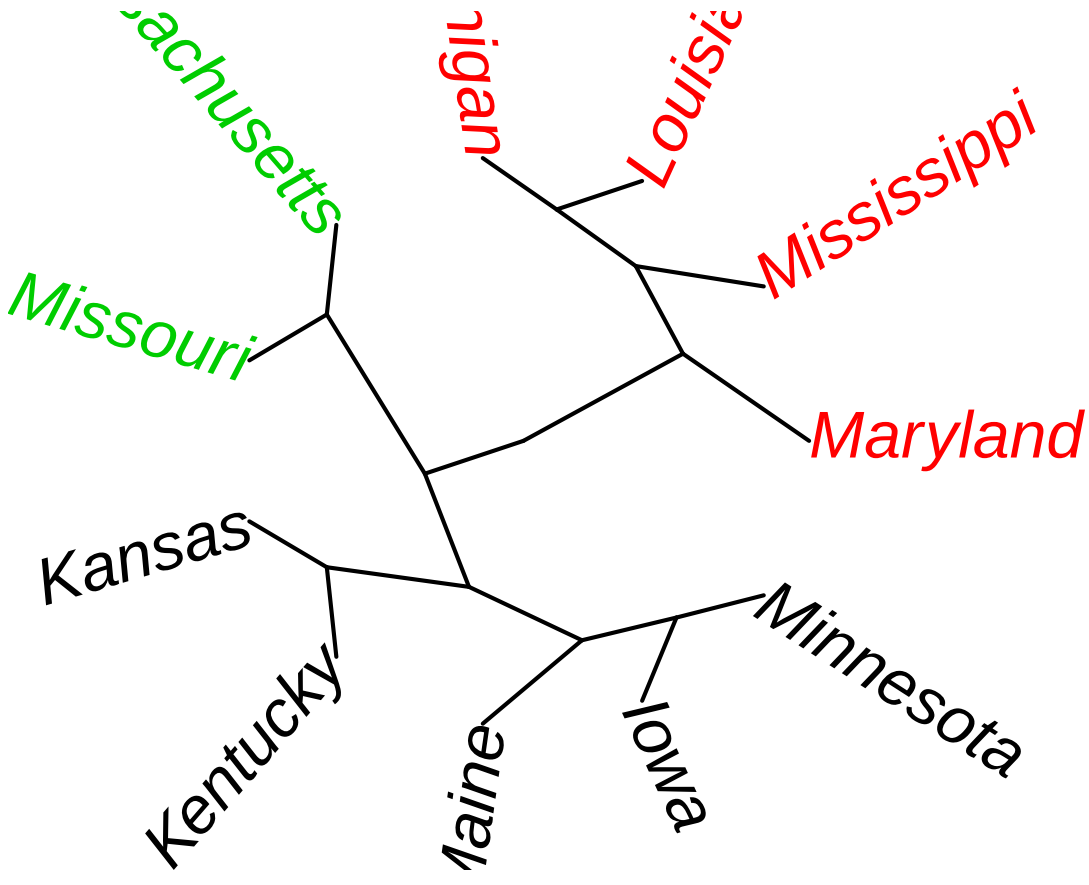
```
plot(hc.phylo, type='fan', tip.color=grupos)
```



```
plot(hc.phylo, type='unrooted', tip.color=grupos)
```



```
plot(hc.phylo, type='radial', tip.color=grupos)
```



Após a formação dos grupos:

- Avaliar os grupos formados: comparar médias entre grupos, por exemplo;
- Obter estatísticas descritivas de cada grupo.

Uso de Componentes Principais

- Agrupamento Hierárquico baseado em Componentes Principais;
- Componentes principais são representações Euclidianas;
- Combina-se a ideia de agrupamento e Componentes Principais;
- no R: função HCPC combina a informação da posição do elemento no espaço das Componentes Principais com o método hierárquico de classificação;

Exemplos

Carregar os pacotes:

```
library(FactoMineR)
library(readxl)
library(bpca)
library(graphics)
library(cluster)
library(fpc)
```

Cluster hierárquico: Dados sobre café

```
cafe<-read_excel('cafe.xls')
```

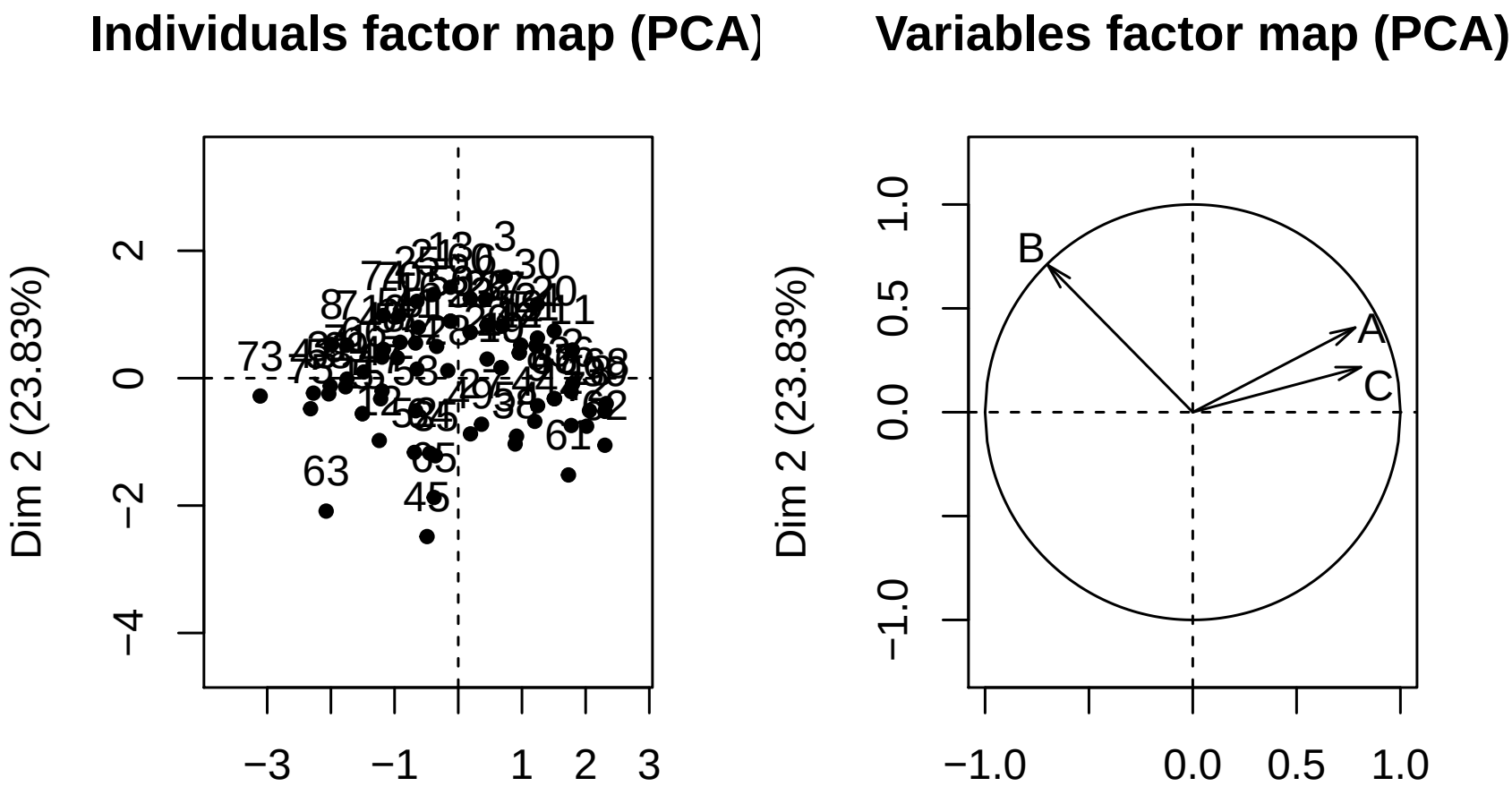
```
head(cafe)
```

```
# A tibble: 6 x 4
  Consumidor     A     B     C
       <chr> <dbl> <dbl> <dbl>
1         C1     8     7     3
2         C2     9     4     7
3         C3     9     8     7
4         C4     8     3     7
5         C5     3     7     4
6         C6     7     8     8
```

Obtendo as componentes principais

Outra maneira de se obter os gráficos de consumidores e produtos:

```
par(mfrow=c(1,2))
plot(cafe.pca,choix=c('ind')) # linhas: consumidores
plot(cafe.pca,choix=c('var')) # colunas: produtos
```



Qual a contribuição de cada um nas dimensões?

```
head(round(cafe.pca$ind$contrib[,1:2],2), n=10) # consumido
```

```
   Dim.1 Dim.2
1   0.09  0.46
2   2.45  0.01
3   0.41  4.75
4   2.39  1.03
5   1.72  0.58
6   0.14  2.87
7   2.32  0.50
8   3.00  0.52
9   1.18  0.35
10  1.05  0.38
```

```r
round(cafe.pca$var$contrib[,1:2],2) # produtos
```

```
  Dim.1 Dim.2
A 34.87 23.22
B 27.75 70.20
C 37.38  6.58
```

Qual a correlação entre produtos e componentes e a variação explicada por cada componente?

```
round(cafe.pca$var$coord[,1:2],2)
```

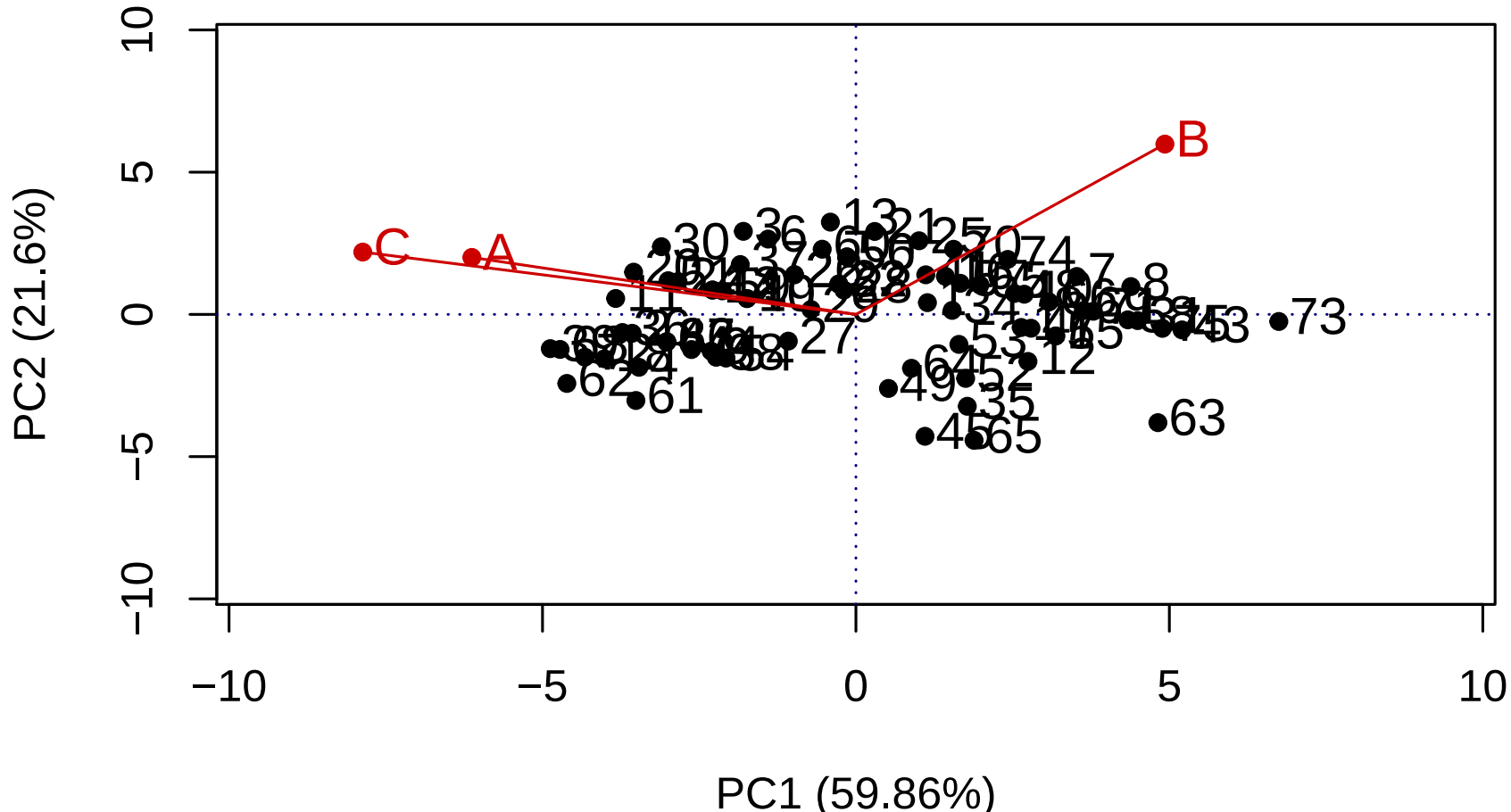
```
  Dim.1 Dim.2
A  0.78  0.41
B -0.70  0.71
C  0.81  0.22
```

```
round(cafe.pca$eig,2)
```

```
       eigenvalue percentage of variance cumulative percent
comp 1       1.76                 58.53
comp 2       0.71                 23.83
comp 3       0.53                 17.64
```

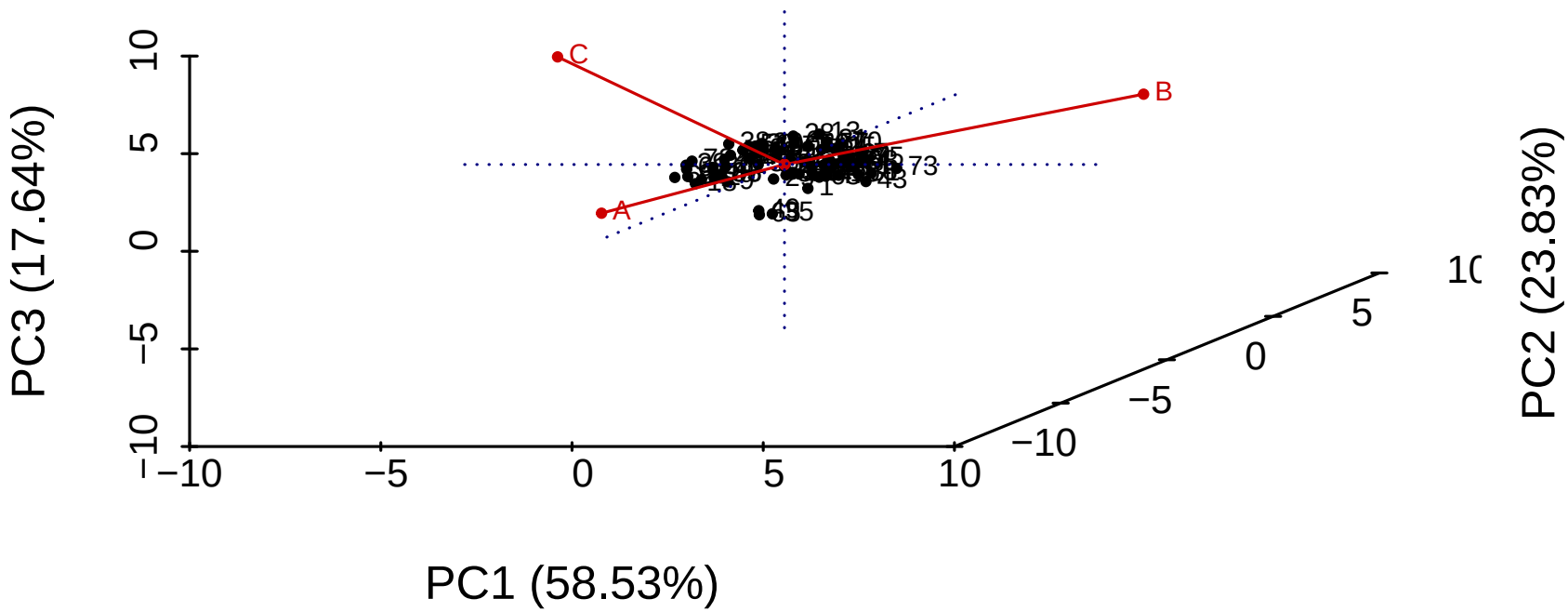
Utilizando o pacote bpca

```
bp2 <- bpca(cafe[,2:4],d=1:2,scale=F)
plot.bpca.2d(bp2,var.cex=1.2,obj.cex=1.2,var.offset=.2,
    var.factor=.45)
```



Utilizando 3 dimensões

```
bp3 <- bpca(cafe[,2:4],d=1:3)
plot.bpca.3d(bp3)
```



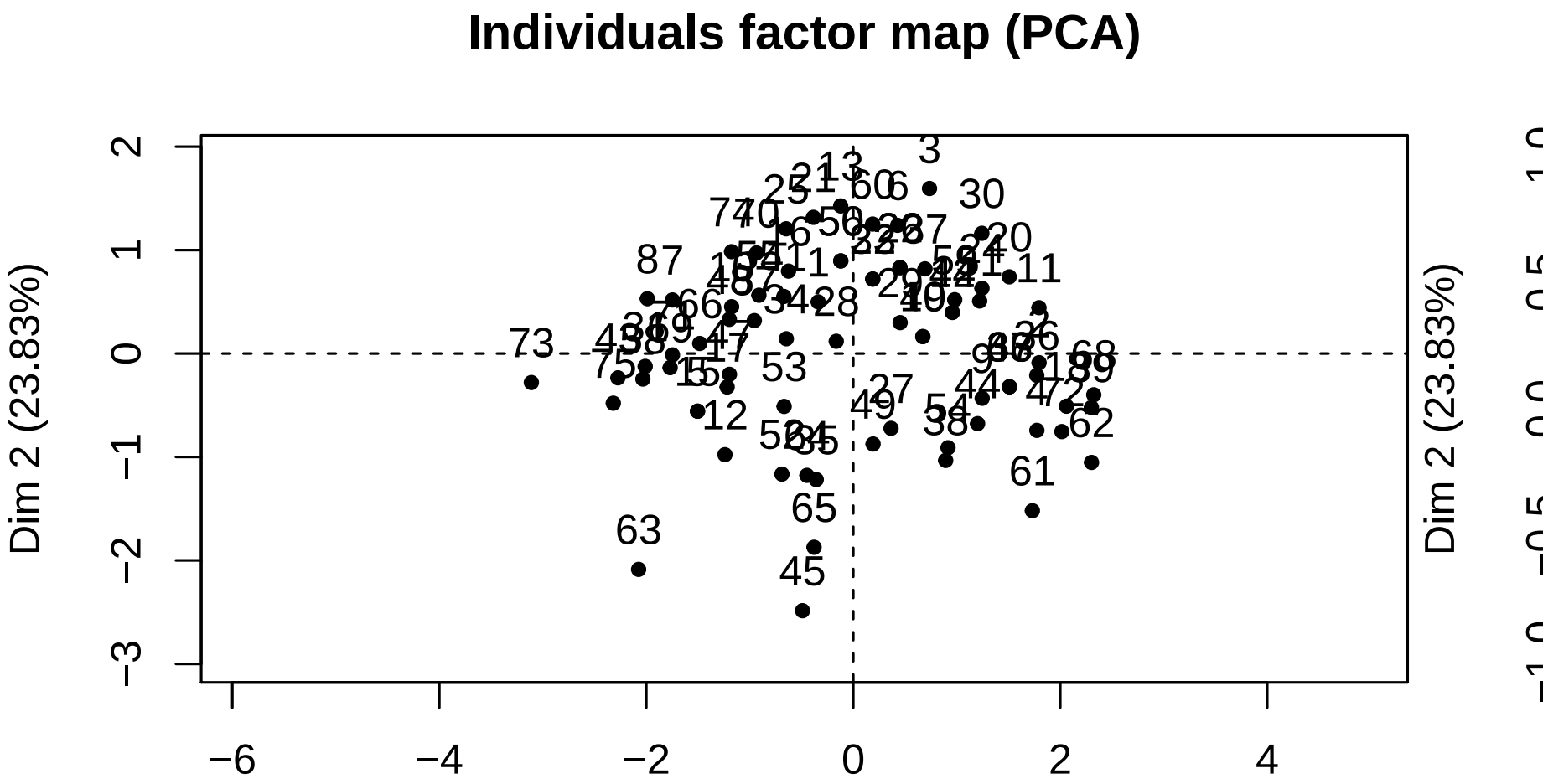
Experimente rotacionar o gráfico

```
plot.bpca.3d(bp3,rgl=T)
```

Utilizando a função HCPC

Quantos grupos foram sugeridos?

```
cafe.hpc<-HCPC(PCA(cafe[,2:4]),graph=FALSE)
```

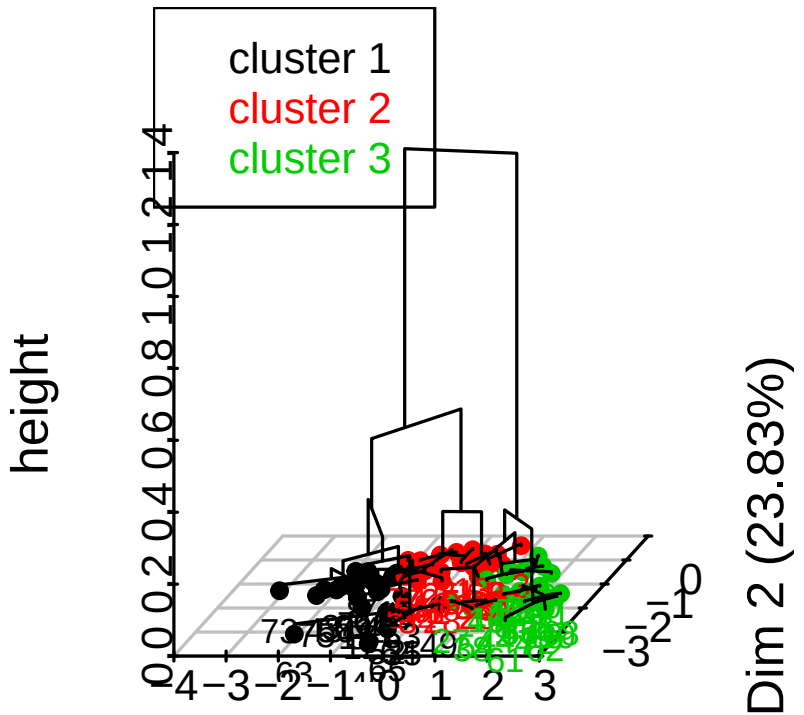


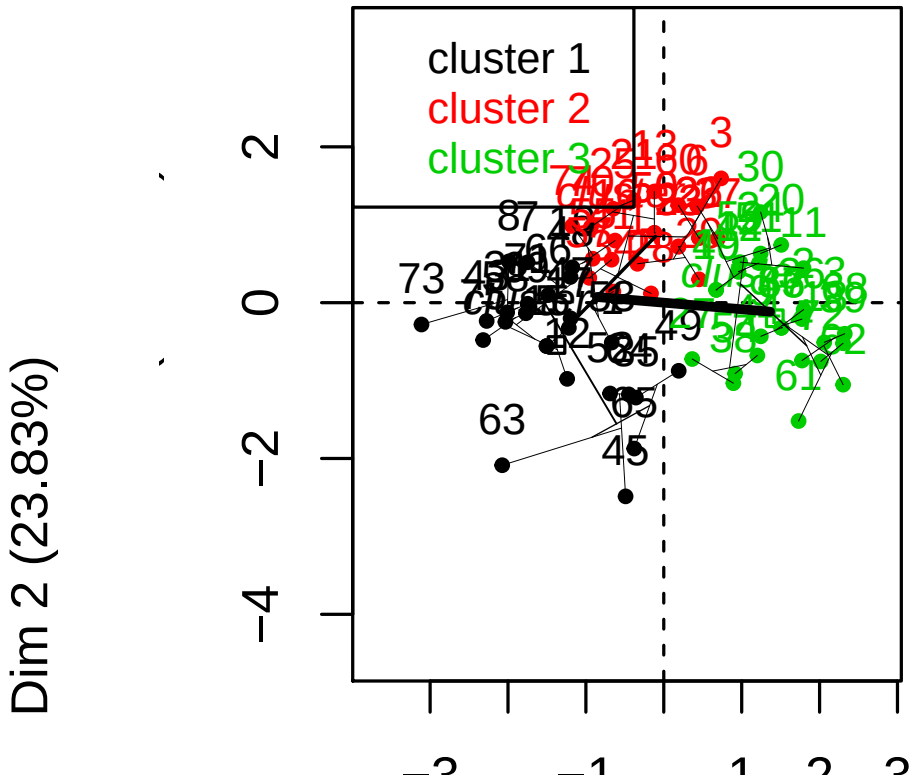
Representação gráfica

```
par(mfrow=c(1,2))
plot(cafe.hpc)
plot(cafe.hpc,choice = 'map')
```

archical clustering on the factor

Factor map





Observe as notas de cada consumidor para cada um dos produtos:

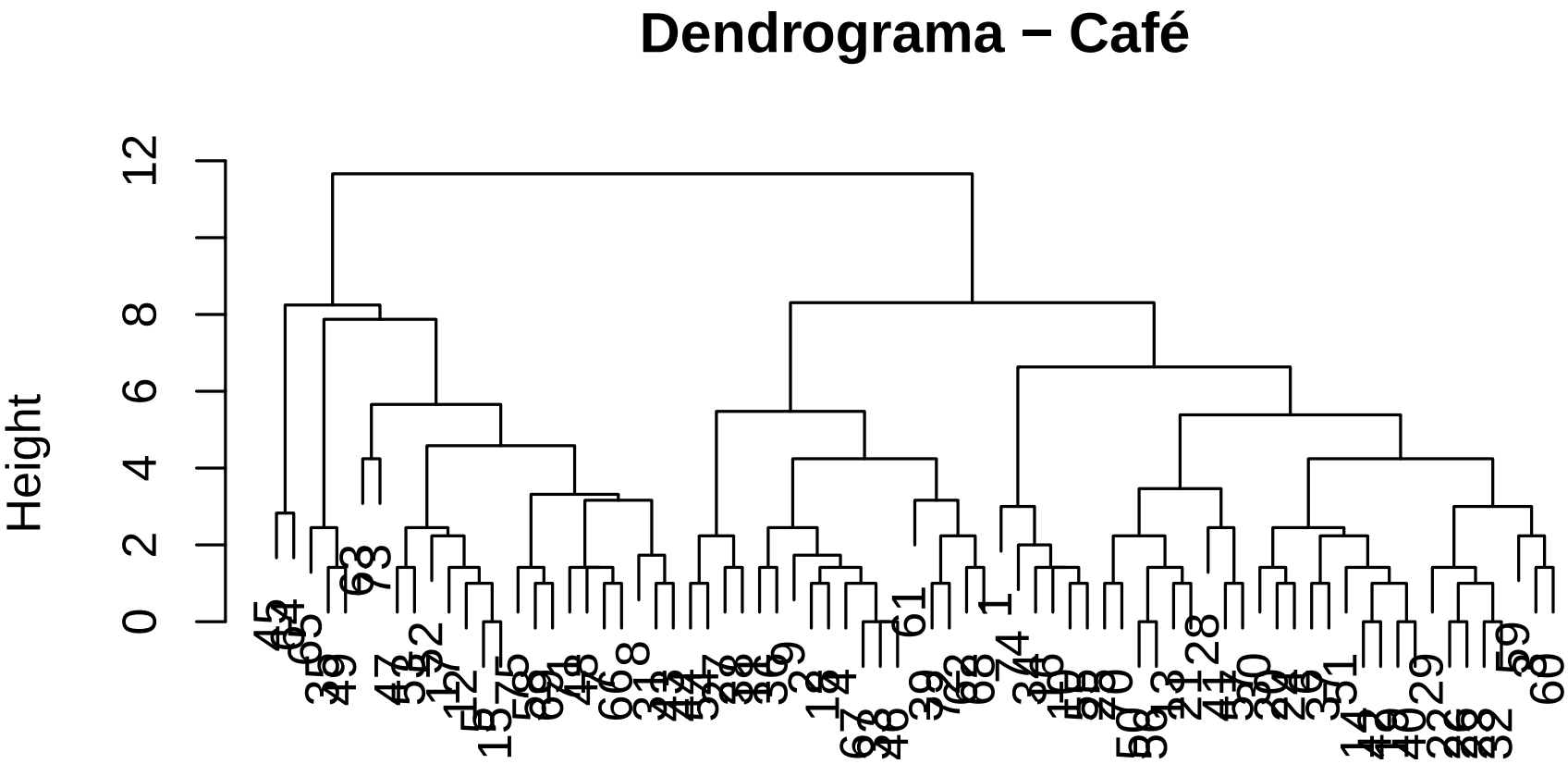
```
cafe.hpc$data.clust
```

```
   A B C clust
1  8 7 3     2
2  9 4 7     3
3  9 8 7     2
4  8 3 7     3
5  3 7 4     1
6  7 8 8     2
7  4 9 4     1
8  5 9 2     1
9  8 4 6     3
10 6 8 3     1
11 9 5 8     3
12 3 6 4     1
```

```
cafe.d<-dist(cafe[,2:4]) # matriz de distâncias
hc1<-hclust(cafe.d) # ligação completa
```

Dendrograma

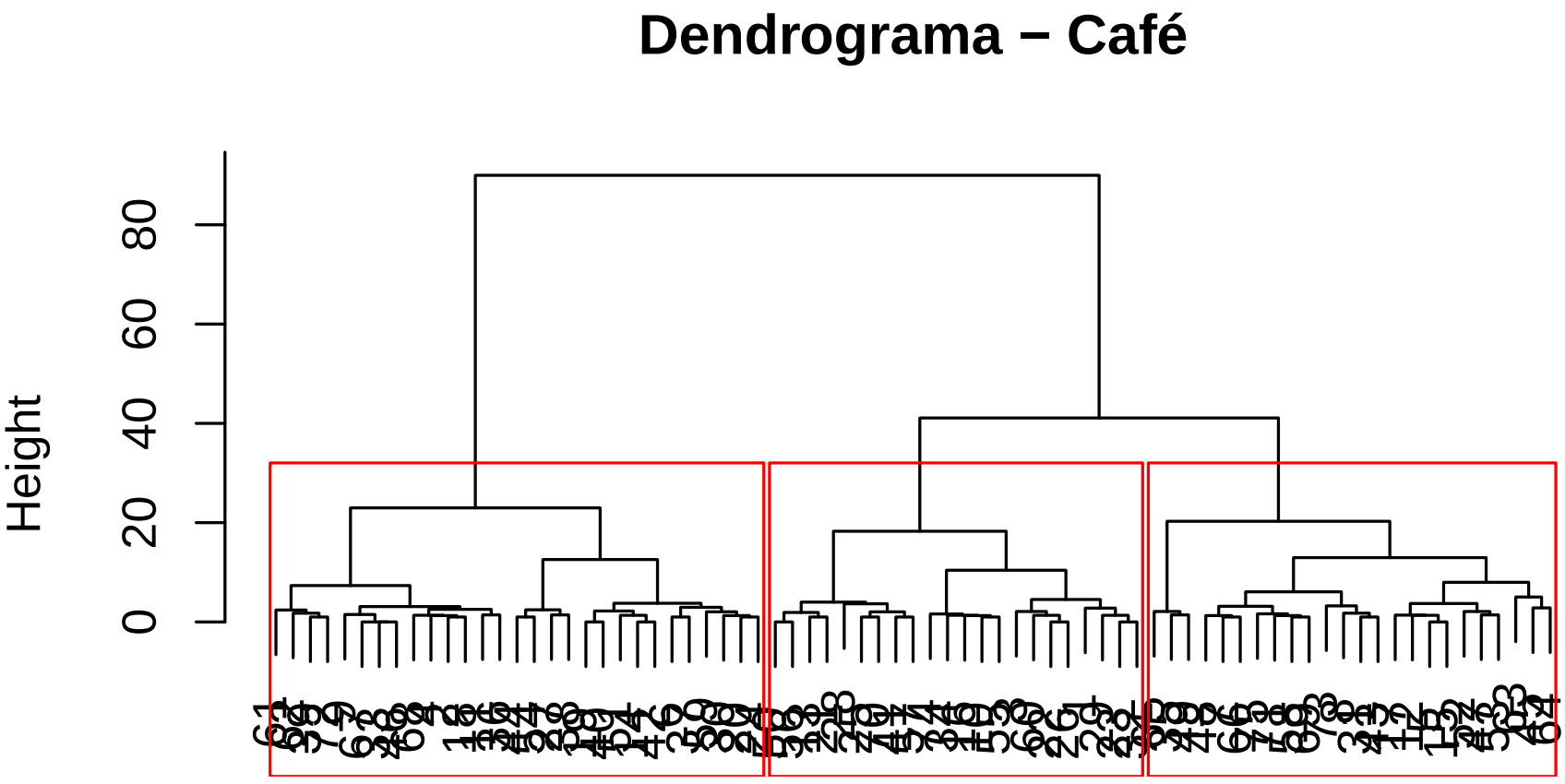
```
plot(hc1,main= "Dendrograma - Café", xlab="Café", sub="Liga
```



```r
hc2 <- hclust(cafe.d,method="ward.D") # Ward
```

Criando o dendrograma e identificando os grupos:

```
plot(hc2, main= "Dendrograma - Café", xlab="Café", sub="Wa
rect.hclust(hc2, k = 3, border = 'red') # k=3 grupos
```



Identificando os consumidores

```
cafe.g <- cutree(hc2, k=3) # identificando consumidores
cafe.g
```

```
 [1] 1 2 1 2 3 2 3 3 2 1 2 3 1 2 3 1 3 2 2 2 1 1 1 2 1 1 2
[36] 2 2 2 2 2 1 2 3 2 3 2 3 3 3 1 2 3 3 2 1 1 1 3 2 1 2 2
[71] 3 2 3 1 3
```

```
names(cafe)
```

```
[1] "Consumidor" "A"          "B"          "C"
```

```
table(cafe.g, cafe$Consumidor)
```

```
cafe.g C1 C10 C11 C12 C13 C14 C15 C16 C17 C18 C19 C2 C20 C2
     1  1   1   0   0   1   0   0   1   0   0   0  0   0
     2  0   0   1   0   0   1   0   0   0   1   1  1   1
     3  0   0   0   1   0   0   1   0   1   0   0  0   0

cafe.g C25 C26 C27 C28 C29 C3 C30 C31 C32 C33 C34 C35 C36 C
     1   1   1   0   1   1  1   0   0   1   0   1   0   0
     2   0   0   1   0   0  0   1   0   0   1   0   0   1
     3   0   0   0   0   0  0   0   1   0   0   0   1   0

cafe.g C40 C41 C42 C43 C44 C45 C46 C47 C48 C49 C5 C50 C51 C
     1  0   1   0   0   0   0   0   0   0   0   0  1   0
```

Número de observações por grupo

```
table(cafe.hpc$data.clust$clus)
```

```
 1  2  3
25 23 27
```

Outras descrições

- **para** - indivíduos típicos do cluster
- **dist** - indivíduos mais distantes do cluster

```
cafe.hpc$desc.ind$para
```

```
Cluster: 1
       17         47          5         15         71
0.3889737 0.5273408 0.6057734 0.6057734 0.6470441
------------------------------------------------------------
Cluster: 2
       50         56         25         60         22
0.5642345 0.5642345 0.6799459 0.6834437 0.7808691
------------------------------------------------------------
Cluster: 3
       36         33         46         67         44
0.4101621 0.5592715 0.5592715 0.5592715 0.6678750
```

```
cafe.hpc$desc.ind$dist
```

```
Cluster: 1
      63        45        73        65        35
3.489146 3.237696 3.136740 3.120972 2.851907
------------------------------------------------------------
Cluster: 2
      13        21        28         3        60
2.373666 2.242346 2.048836 2.004520 1.992492
------------------------------------------------------------
Cluster: 3
      62        61        39        68        72
3.114985 3.055524 2.828082 2.814811 2.753213
```

Método não hierárquico: k-means

Dados Café

```
head(cafe)
```

```
# A tibble: 6 x 4
  Consumidor     A     B     C
       <chr> <dbl> <dbl> <dbl>
1         C1     8     7     3
2         C2     9     4     7
3         C3     9     8     7
4         C4     8     3     7
5         C5     3     7     4
6         C6     7     8     8
```

Utilizando a função kmeans:

```
cafe.k<-kmeans(cafe[,2:4],3)

table(cafe.k$cluster)
```

```
 1  2  3
27 22 26
```

Identificando os grupos

```
cafe.k$cluster
```

```
 [1] 2 1 2 1 3 2 3 3 1 3 1 3 2 1 3 2 3 1 1 1 2 2 2 1 2 2 1
[36] 1 2 1 1 1 2 1 3 1 3 1 3 3 3 2 1 3 3 1 2 2 2 3 1 2 1 1
[71] 3 1 3 2 3
```