


UNIVERSITAT DE
BARCELONA


Cluster analysis



**Píndoles d'estadística avançada
STeL (Març 2021)
Sessió 2**

PROF. S. CIVIT

Distances



- Distance functions are available in at least four packages of the R language.
- 1. Package **stats**, function **dist** : 6 distances (documentation: **?dist**)
 - The choice of coefficient is done by typing its name in quotes. Example:
 - `dist(data, method="binary")` or `dist(data, "binary")`
- 2. Package **vegan**, function **vegdist** : 10 distances (documentation: **?vegdist**)
 - The choice of coefficient is done by typing its name in quotes.
 - Example: `vegdist(data, method="bray")` or `vegdist(data, "bray")`
- 3. Package **ade4**, function **dist.binary** : 10 binary distances (documentation: **?dist.binary**)
 - These similarities (S) are converted to distances through the transformation $D = \sqrt{1 - S}$
 - The choice of coefficient is done by typing its number in the list above. Example:
 - `dist.binary(data, method=1)` ou `dist.binary(data, 1)` ou `dist.binary(data, "1")`

Hierarchical Cluster I (1/7)

```
# Example: analysis of the file fangataufa.txt .
data<-read.table("fangataufa.txt",sep="\t",header=TRUE)

# Standardize the variables
data2<-scale(data[,2:14])

# Compute Euclidean distance
data.D1 = dist(data2, method="eucl")

# Agglomerative clustering, UPGMA method:
hclust(d, method = "average", members=NULL)
```

Hierarchical Cluster I (2/7)

```
# methods: "ward", "single", "complete", "average" (=UPGMA),
"mcquitty" (=WPGMA), "centroid" (=UPGMC) or "median"
(=WPGMC)".
clusterAV = hclust(data.D1, method="average")
clusterW<-hclust(data.D1^2, method="ward.D2")#ward

# Plot the dendrogram
plot(clusterAV)
plot(clusterAV, hang=-1, labels=data[,1])
```

Hierarchical Cluster I (3/7)

Compute cophenetic correlation ?cophenetic

Cophenetic distances of the dendrogram

```
Ours.coph = cophenetic(clusterAV)
```

Cophenetic correlation

```
cor(data.D1, Ours.coph)
```

Examine the following functions:

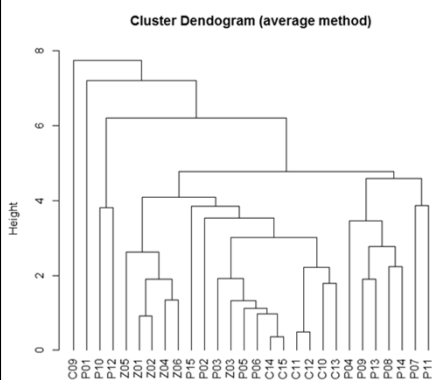
?identify,

?rect.hclust,

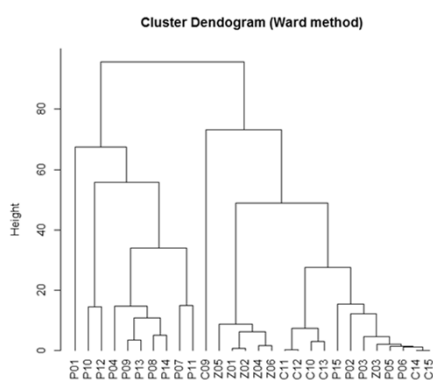
?cutree

Repeat the analysis plotting the dendrogram and cophenetic correlation for Agglomerative clustering, Ward method

Hierarchical Cluster I (4/7)



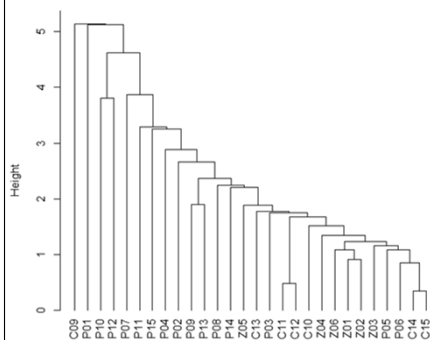
data D1
hclust("average")
cor(data.D1, Ours.coph)=0.85



data D1*2
hclust("ward.D2")
cor(data.D1, Ours.coph)= 0.62

Hierarchical Cluster (5/7)

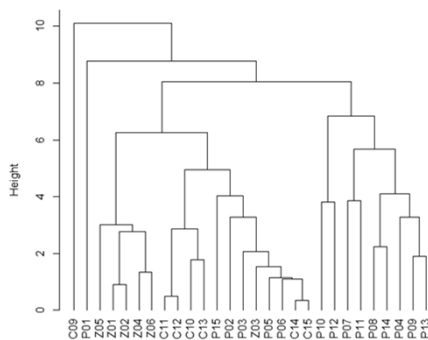
Cluster Dendrogram (Single method)



data.D1
hclust("single")

$\text{cor}(\text{data.D1}, \text{Ours.coph}) = 0.81$

Cluster Dendrogram (Complete method)



data.D1
hclust("complete")

$\text{cor}(\text{data.D1}, \text{Ours.coph}) = 0.77$

Hierarchical Cluster I (6/7)

#Suppose k= 7 groups

```
groups.7 <- cutree(clusterAV,7) #7 groups?
```

#identify members

```
plot(clusterAV, hang=-1, labels=data[,1], main="Cluster  
Dendrogram (UPGMA method), k=7")
```

```
rect.hclust(clusterAV,k=7, border="blue", )
```

number of elements and elements within groups

```
table(groups.7)
```

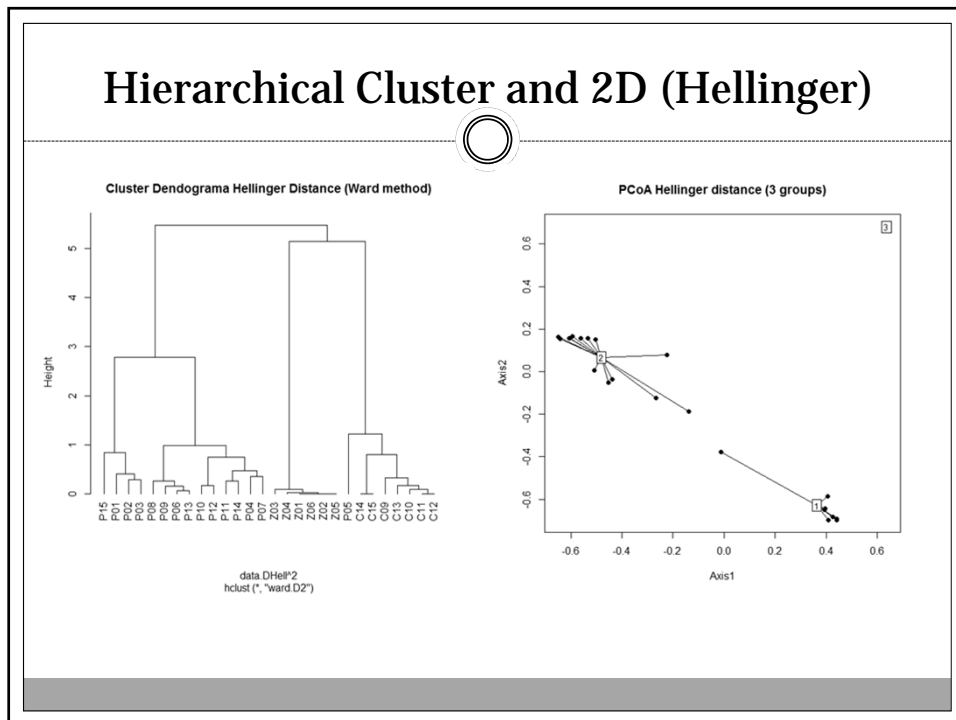
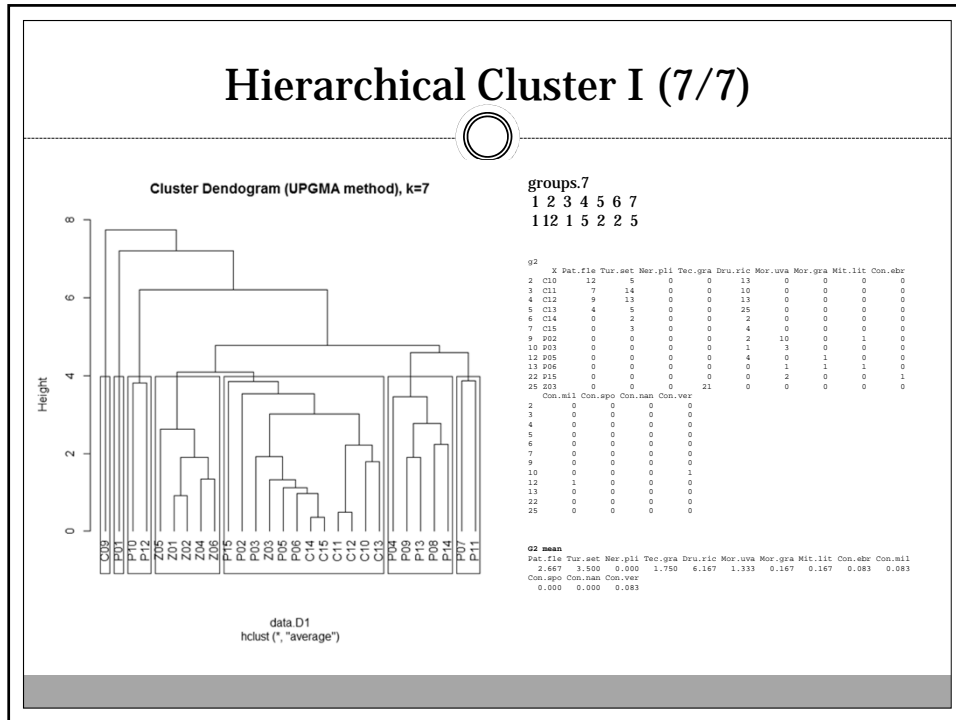
```
g1=data[groups.7==1, ]
```

```
g2=data[groups.7==2, ]...
```

```
g7=data[groups.7==7, ]
```

R output object, showing the mean of each variables by group

```
G2mean<-round(apply(g2[-1], 2, mean), 3)
```



UNIVERSITAT DE BARCELONA

Cluster analysis

NO HIERARCHICAL CLUSTERING

PROF. S. CIVIT

Anàlisi Cluster: Com deiem...

- Preten establir agrupacions d'objectes de dades de manera **que els objectes d'un grup siguin similars** (o estiguin relacionats) entre si i **siguin diferents** (o no estiguin relacionats) **amb els objectes d'altres grups**

Intra-cluster distances are minimized

Inter-cluster distances are maximized

Partitioning algorithms: Conceptes bàsics

- Construeix una partició d'un conjunt de n objectes en un conjunt de k clusters
 - Cada objecte pertany **exactament a un clúster**
 - El nombre de clústers **k es dona per endavant**
- **Partitioning method**: Construeix una partició d'una base de dades D d'objectes n en un conjunt de k clústers (C_1, C_2, \dots, C_k) per exemple minimitzar la suma de les distàncies al quadrat

$$TESS = \sum_{i=1}^k \sum_{j=1}^{|C_i|} d^2(\omega_{ij} - c(i))$$

entre $c(i)$ centroid o medoid (representative object) del grup i

Heuristic methods

k-means (MacQueen 1967): Cada clúster està representat pel **centre del clúster**

k-medoids or PAM (Partition around medoids) (Kaufman and Rousseeuw 1987): Cada clúster està representat per **un dels objectes del cluster**

Possible combinations:

$$C(n, k) = \frac{1}{k!} \sum_{j=1}^k (-1)^{k-j} \binom{k}{j} j^n \approx \frac{k^n}{k!}$$

$n=25, k=8: C(25, 8) \sim 6.9 \cdot 10^{17}$

"Time efficiency" problem: Computational resources used by the algorithm

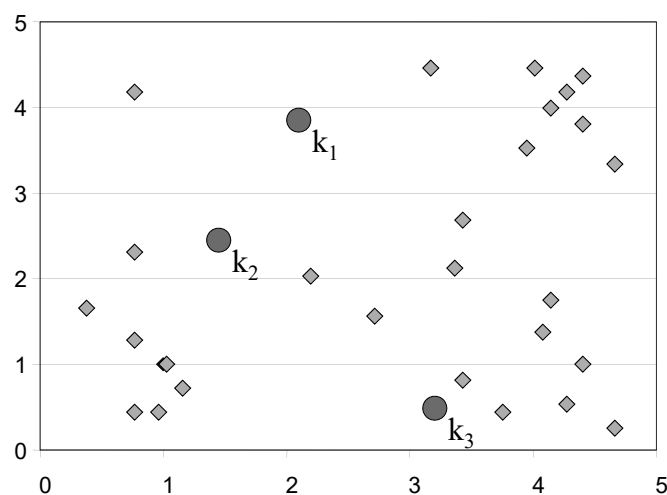
The *K-Means* Clustering Algorithm

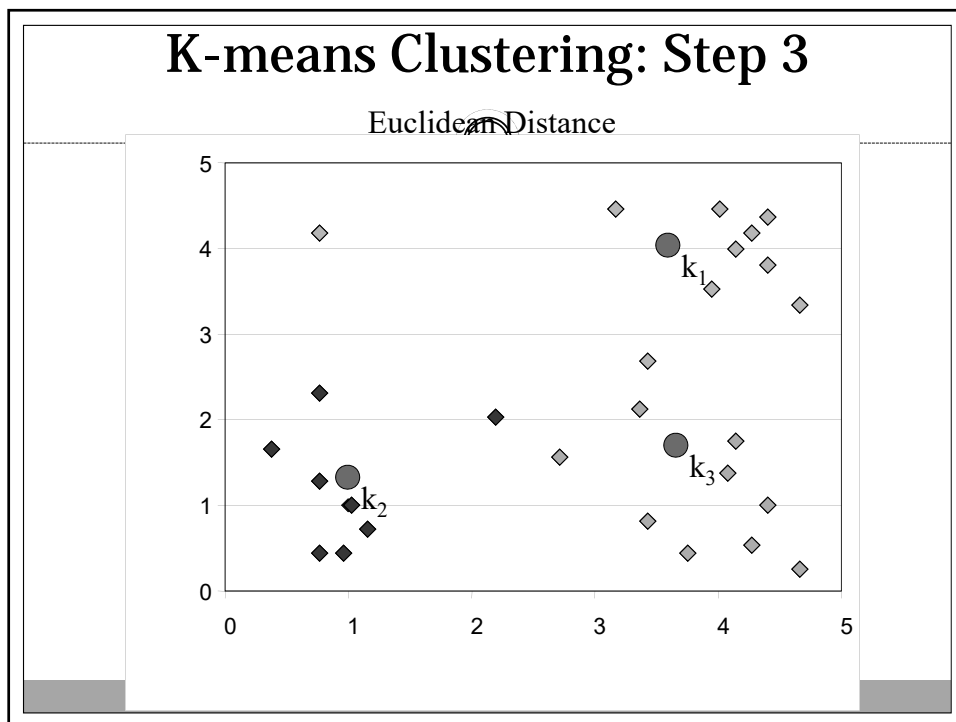
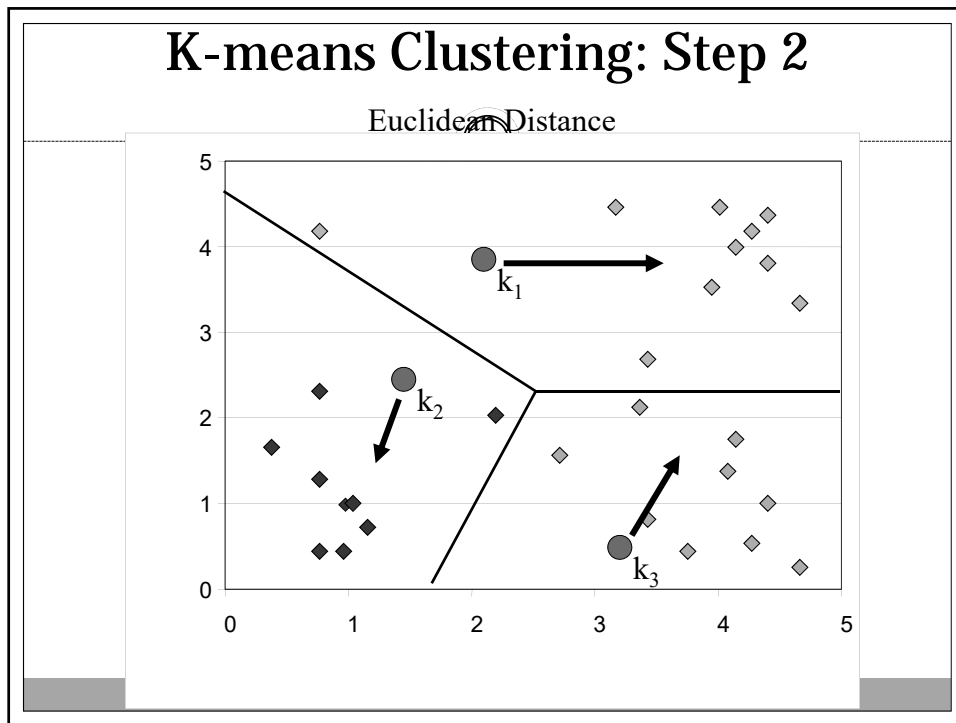
Algorisme pas a pas:

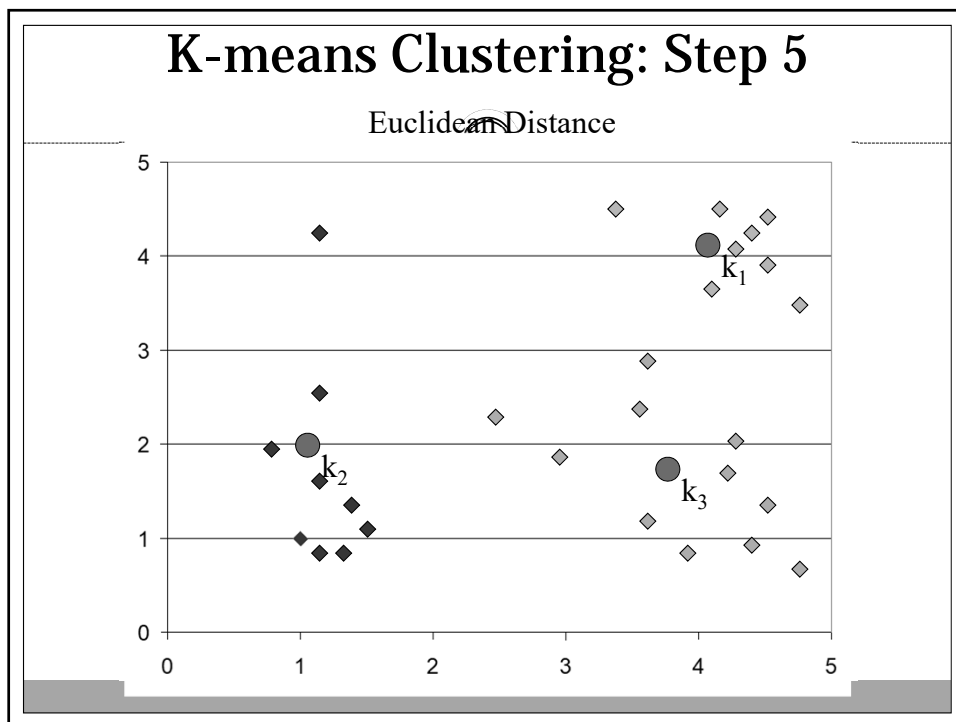
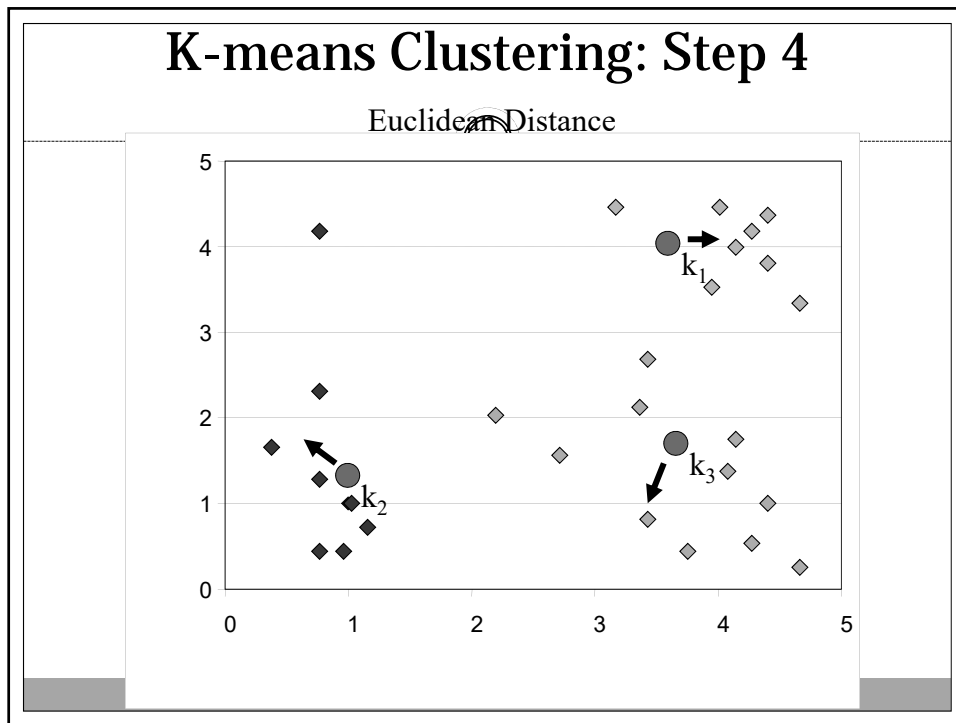
1. Decidiu un valor per a **k**.
2. Inicialitzeu els **k** centres de clúster (aleatòriament, si cal).
3. Decidiu la pertinença al grup dels **N** objectes assignant-los al **centre de clúster** més proper (**mitjana aritmètica**).
4. Torneu a estimar els **k** **centres de clúster**, assumint que les pertinença trobades anteriorment són correctes..
5. Si cap dels **N** **objectes no ha canviat de pertinença** a l'última iteració, **sortiu**. En cas contrari, aneu al 3.

K-means Clustering: Step 1

Euclidean Distance







Dataset: Countries

`Países.txt` is a ASCII file containing, socioeconomic variables from different countries. **Countries** are in rows, **socioeconomic variables** are in columns.

Country: Identifier of the country
 Pob : Population (in millions)
 PIB : Gross Domestic Product per inhabitant (GDP)
 Urb : Rate of population in urban areas
 Analf : illiterate rate
 Estud: Tax students
 Vida : Life Expectancy
 Nutric : Index nutritional needs met
 ContInd Indicator industrial weighted measure the impact of the greenhouse effect
 ContVeh : Weighted indicator (weighted combinations of CO, NOx..) associated with the air pollution from mobile.
 SecPrim : Percentage of GDP working into the primary sector

Non Hierarchical Cluster (Kmeans) (1/7)

1. Remembering...

Example: analysis of the file `países.txt`

```
countries<-read.table("Países.txt", sep="\t", header=TRUE)
```

Standardize the variables

```
countries2<-scale(countries[, -1])
```

Compute Euclidean distance

```
countries.D1 = dist(countries2, method="eucl")
```

Agglomerative clustering, UPGMA method:

```
library(cluster)
```

```
clusterW<-hclust(countries.D1^2, method="ward.D2")#ward
```

Plot the dendrogram

```
plot(clusterW, hang=-1, labels=countries[,1])
```

Non Hierarchical Cluster (Kmeans) (2/7)



2. **#Compute partitioning in k groups** (parameter 'centers') using 'kmeans' of the 'stats' package. This is a more interesting function for K-means because the analysis can be automatically repeated a large number of times (parameter 'nstart'). The function finds the best solution smallest value of sum of within-groups sums-of-squares after repeating the analysis 'nstart' times.

Suppose 2 groups (k=2)??

```
result.km.2 = kmeans(countries2, centers=2, nstart=1000)
```

```
names(result.km.2)
```

```
"cluster" "centers" "totss" "withinss" "tot.withinss" "betweenss"
"size" "iter" "ifault"
```

Non Hierarchical Cluster (Kmeans) (3/7)



```
result.km.2$size
```

```
K-means clustering with 2 clusters of sizes 16, 25
```

```
result.km.2$centers
```

```
Cluster means:
```

	Pob	PIB	Urb	Analf	Estud	Vida	Nutric
1	0.3127299	-0.8225494	-0.6566295	0.9695506	-0.6911512	-1.0473118	-0.910400
2	-0.2001471	0.5264316	0.4202429	-0.6205124	0.4423367	0.6702796	0.582656
	ContInd	ContVeh	SecPrim				
1	-0.8298794	-0.9100805	1.042405				
2	0.5311228	0.5824516	-0.667139				

```
result.km.2$cluster
```

```
[1] 1 1 2 2 1 2 2 2 2 1 2 1 1 2 2 2 1 2 2 2 1 2 2 2 1 1 2 2 2 1 1 1 2 1 2 2 2 1 2
[39] 2 1 2
```

Non Hierarchical Cluster (Kmeans) (4/7)

```
# The total error sum of squares (TESS) (totss)
result.km.2$totss
# The total error sum of squares within groups (tot.withinss)
result.km.2$tot.withinss
# Compute different cluster validity index using the 'clustIndex'
function of 'cclust' (Pseudo-F statistics (Calinski-Harabasz) ):
library(cclust)
PseudoF.km.2<-clustIndex(result.km.2, countries2,
index="calinski")
# Compute overall mean silhouette:
library(cluster)
Silh.km.2<-silhouette(result.km.2$cluster,dist(countries2))
Overall.Silh.km.2<-mean(Silh.km.2[,3])
```

Non Hierarchical Cluster (Kmeans) (5/7)

```
# Repeat the K-means analysis for K = 3, K = 4, K = 5 and compare the
values of the indices.
# Do the indices reach a maximum for some intermediate value of K?
# Which partition is the best?
# Which index seems the most useful?
```

```
result.km.2 = kmeans(countries2, centers=2, nstart=100)
K=2
result.km.2$totss      result.km.2$tot.withinss      PseudoF.km.2      Overall.Silh.km.2
400                   212.3181                     34.47467          0.398

K=3
...
```

Non Hierarchical Cluster (Kmeans) (6/7)

3. **Compute K-means** for a range of values of K using `'cascadeKM'` of the `'vegan'` package. This function is a *wrapper* for the `'kmeans'` function of the `'stats'` package, that is, a function that uses a basic function, adding new properties to it. It creates several partitions forming a cascade from small (parameter `'inf.gr'`) to large values of K (parameter `'sup.gr'`).

```
result.cascadeKM = cascadeKM(countries2, inf.gr=2, sup.gr=5,
iter = 1000, criterion = "calinski")
```

Look at the structure of the results file:

```
summary(result.cascadeKM)
```

The element `'partition'` contains a table showing the group attributed to each object:

```
result.cascadeKM$partition
```

Non Hierarchical Cluster (Kmeans) (7/7)

The result can be plotted using the function `clusplot()` in `cluster` package.

```
library(cluster)
```

```
clusplot(countries2, result.km.2$cluster, main = "Kmeans plot, k =
2", color = TRUE, labels=2)
```

"Clustering variable mean values by group" to be added by

```
aggregate(countries2, by=list(cluster=result.km.2$cluster), mean)
```

