UNIVERSITAT DE BARCELONA

Cluster analysis

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

PROF. S. CIVIT

2021

1

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")

2021

2

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)
```

2021

3

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])
```

2021

4

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,

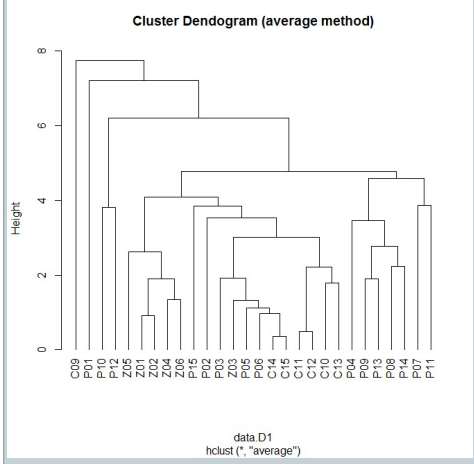
?cutree

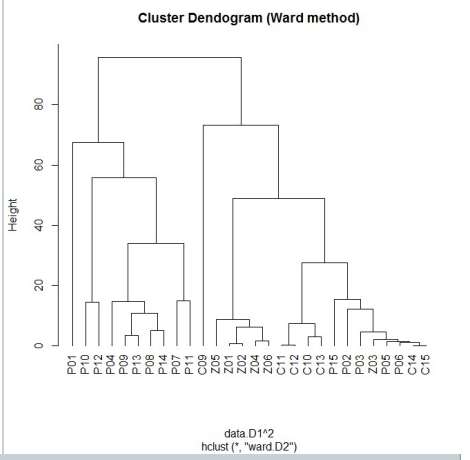
Repeat the analysis plotting he dendogram and coefenetic correlation for Agglomerative clustering, Ward method

2021

5

Hierarchical Cluster I (4/7)





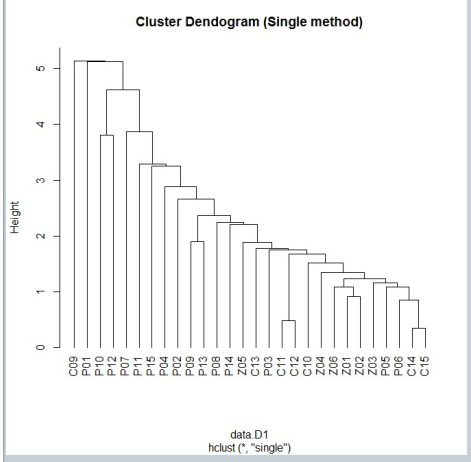
cor(data.D1, Ours.coph)=0.85

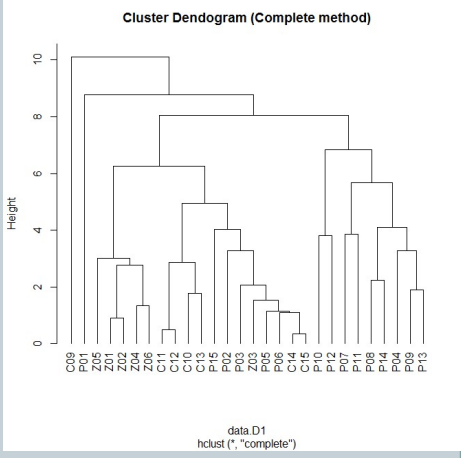
cor(data.D1, Ours.coph)= 0.62

2021

6

Hierarchical Cluster (5/7)





cor(data.D1, Ours.coph)= 0.81

cor(data.D1, Ours.coph)=0.77

2021

7

Hierarchical Cluster I (6/7)

#Suppose k= 7 groups

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

#identify members

```
plot(clusterAV, hang=-1, labels=data[,1], main="Cluster
  Dendogram (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,]
```

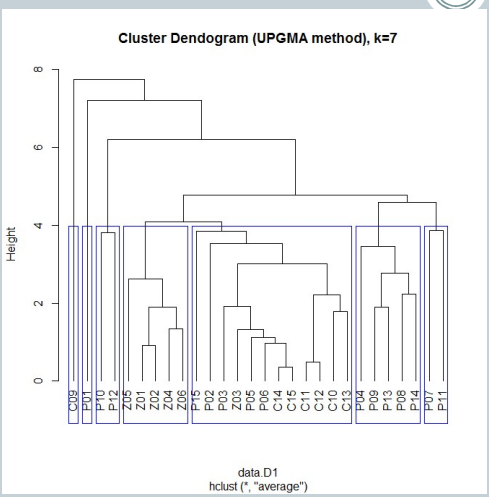
R output object, showing the mean of each variables by group

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

2021

8

Hierarchical Cluster I (7/7)



groups.7
1 2 3 4 5 6 7
1 12 1 5 2 2 5

g2

	X	Pat.fle	Tur.set	Ner.pli	Tec.gra	Dru.ric	Mor.uva	Mor.gra	Mit.lit	Con.ebr
2	C10	12	5	0	0	13	0	0	0	0
3	C11	7	14	0	0	10	0	0	0	0
4	C12	9	13	0	0	13	0	0	0	0
5	C13	4	5	0	0	25	0	0	0	0
6	C14	0	2	0	0	2	0	0	0	0
7	C15	0	3	0	0	4	0	0	0	0
9	P02	0	0	0	0	2	10	0	1	0
10	P03	0	0	0	0	1	3	0	0	0
12	P05	0	0	0	0	4	0	1	0	0
13	P06	0	0	0	0	0	1	1	1	0
22	P15	0	0	0	0	0	2	0	0	1
25	Z03	0	0	0	21	0	0	0	0	0

	Con.mil	Con.spo	Con.nan	Con.ver
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
9	0	0	0	0
10	0	0	0	1
12	1	0	0	0
13	0	0	0	0
22	0	0	0	0
25	0	0	0	0

G2 mean

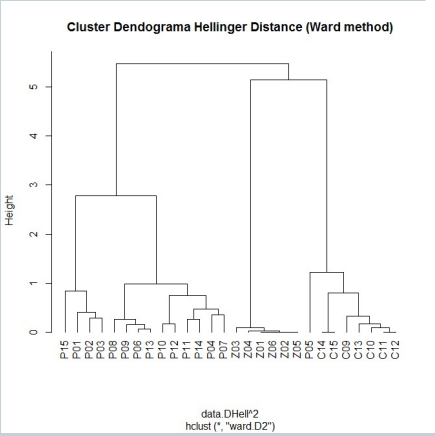
Pat.fle	Tur.set	Ner.pli	Tec.gra	Dru.ric	Mor.uva	Mor.gra	Mit.lit	Con.ebr	Con.mil
2.667	3.500	0.000	1.750	6.167	1.333	0.167	0.167	0.083	0.083

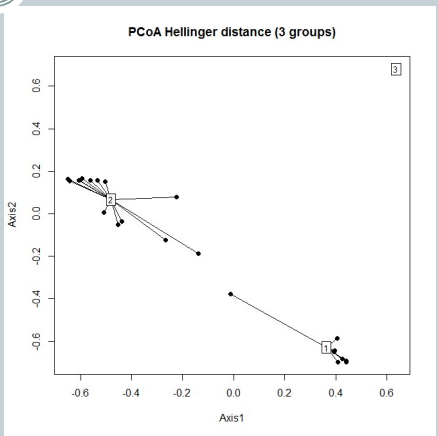
Con.spo	Con.nan	Con.ver
0.000	0.000	0.083

2021

9

Hierarchical Cluster and 2D (Hellinger)





2021

10

Dataset: Countries

`Paises.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
```

2021

11

Non Hierarchical Cluster (Kmeans) (1/7)

1. **Remembering...**

Example: analysis of the file **`paises.txt`**

```
countries<-read.table("Paises.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])
```

2021

12

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"
```

2021

13

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

2021

14

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])
```

2021

15

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

….

2021

16

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
```

2021

17

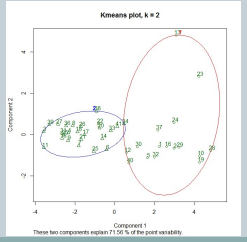
Non Hierarchical Cluster (Kmeans) (7/7)

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)
```




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

2021

18