Agrupaments (*Clustering*)

Introducció

Problema: Donat un conjunt d'objectes, classificar-los en grups (clusters) basant-nos en les seves semblances i diferències

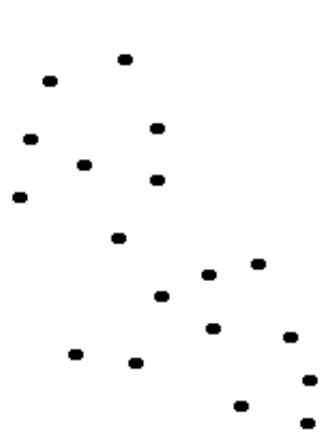
Algunes aplicacions en biologia:

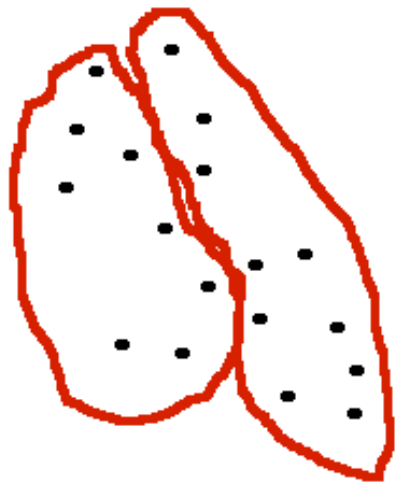
- Classificació jeràrquica d'organismes (relacionada amb una filogènia)
- Agrupament de gens amb pautes d'expressió similars
- Agrupament de gens per semblança seqüencial
- Agrupament de proteïnes per semblança estructural

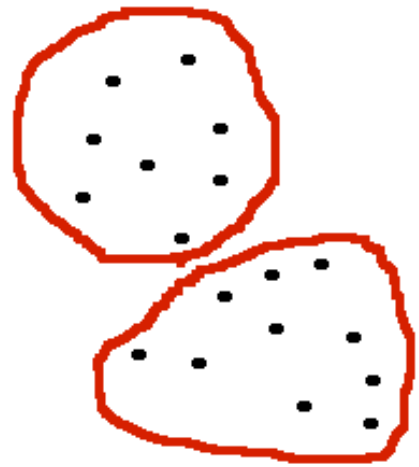
Principis bàsics

Homogeneïtat: Objectes dins el mateix cluster han de ser propers (semblants)

Separació: Objectes dins clusters diferents han de ser llunyans





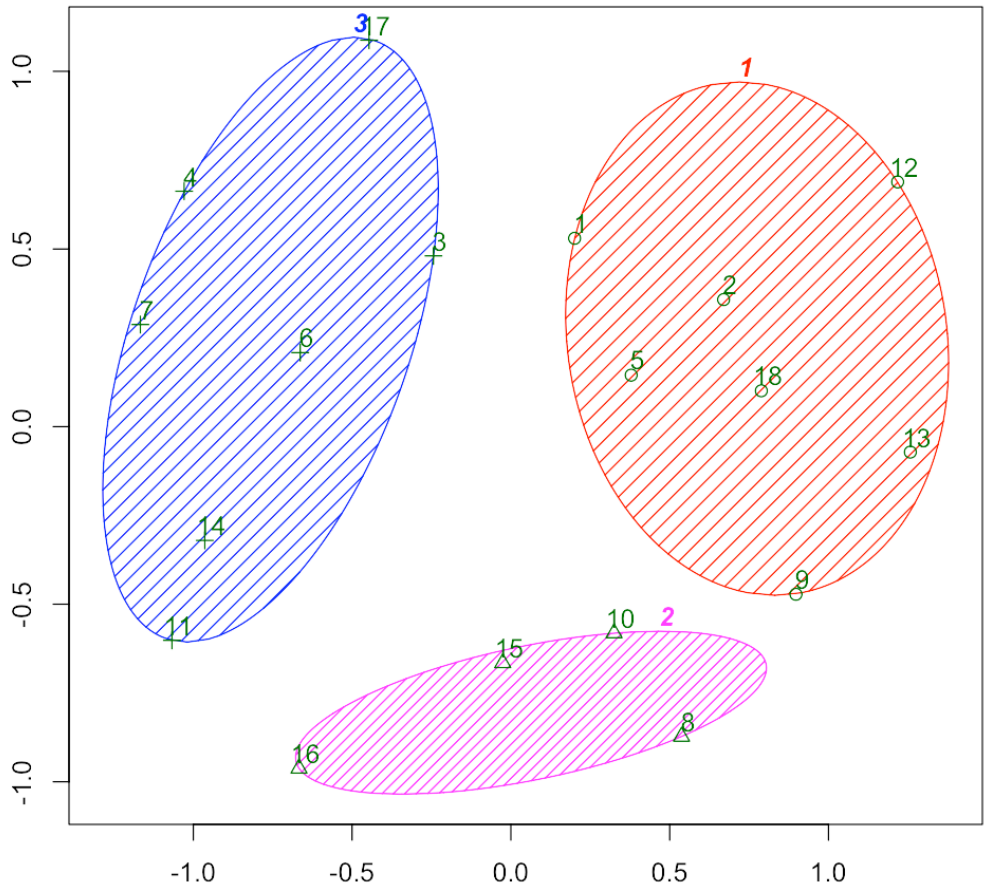


dolent bo

Com formalitzar i calcular aquests principis intuïtius?

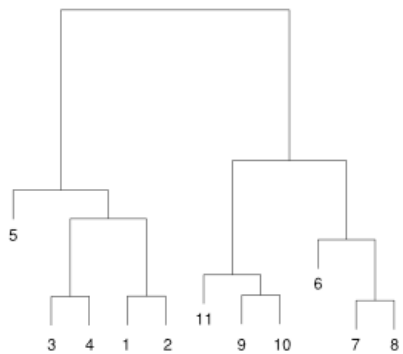
Tipus de clustering

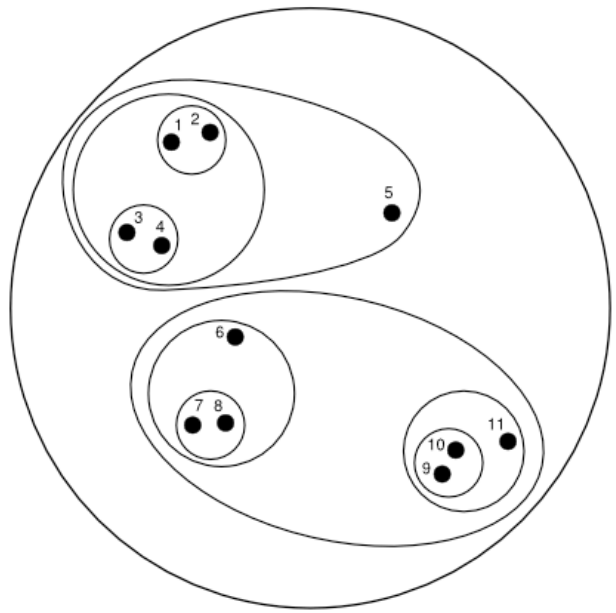
- De partició: Dividim els objectes en un nombre prefixat de clusters; possiblement provam diversos nombres de clusters i ens quedam amb el millor



Tipus de clustering

- Jeràrquic: Successivament agrupam (aglomeratius) o dividim (divisius) objectes o grups d'objectes. Produeix un arbre de classificació on els objectes pertanyen a clusters inclosos dins clusters inclosos dins clusters ...





k-means

L'algoritme de les k-mitjanes (k-means) cerca una partició del conjunt d'objectes, representats com a elements d'un espai $\mathbb{R}^n$, en un nombre fixat k de clusters

Aquests clusters s'identifiquen per mitjà dels seus punts mitjans (*means*)

Recordau que donat $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n$,

$$\|\mathbf{x}\| = \sqrt{\sum_{i=1}^{n} x_i^2} \in \mathbb{R}$$

i que donats $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, $\|\mathbf{x} - \mathbf{y}\|$ és la distància euclidiana entre $\mathbf{x}$ i $\mathbf{y}$.

k-means

Fixem el nombre de clusters k

Donats punts $\mathbf{x}_1, \ldots, \mathbf{x}_p \in \mathbb{R}^n$, l'objectiu és trobar k punts $\mathbf{c}_1, \ldots, \mathbf{c}_k \in \mathbb{R}^n$ que minimitzin

$$SS_C(\mathbf{x}_1, \ldots, \mathbf{x}_p; k) = \sum_{i=1}^{p} \min_{j=1,\ldots,k} \|\mathbf{x}_i - \mathbf{c}_j\|^2$$

Aleshores cada $\mathbf{c}_j$ definirà el cluster format pels $\mathbf{x}_i$ que estan més a prop d'ell que de cap altre $\mathbf{c}_l$:

$$C_j = \{\mathbf{x}_i \mid \|\mathbf{x}_i - \mathbf{c}_j\| < \|\mathbf{x}_i - \mathbf{c}_l\| \text{ per a tot } l \neq j\}$$

i

$$SS_C(\mathbf{x}_1, \ldots, \mathbf{x}_p; k) = \sum_{j=1}^{k} \sum_{\mathbf{x}_i \in C_j} \|\mathbf{x}_i - \mathbf{c}_j\|^2$$

k-means: Algoritme de Lloyd

1. Escollim $\mathbf{c}_1, \ldots, \mathbf{c}_k$ (com vulguem)
2. Assignam cada punt $\mathbf{x}_i$ al cluster C_j definit pel centre $\mathbf{c}_j$ més proper
3. Substituïm cada centre $\mathbf{c}_j$ pel punt mitjà del seu cluster C_j:

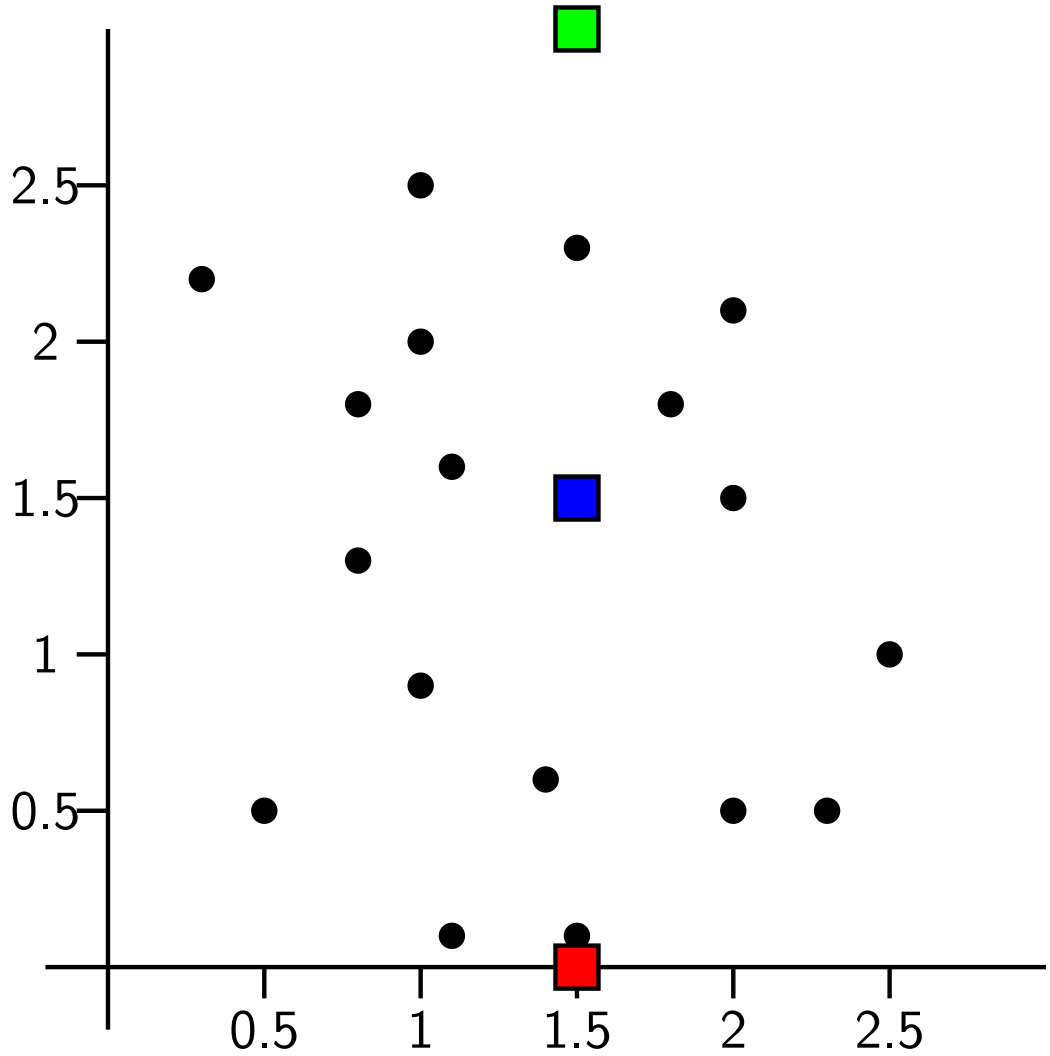
$$\mathbf{c}_j = \Big(\sum_{\mathbf{x}_i \in C_j} \mathbf{x}_i \Big) / |C_j|$$

4. Es repeteixen (2)–(3) fins que els clusters estabilitzen, o un nombre prefixat d'iteracions

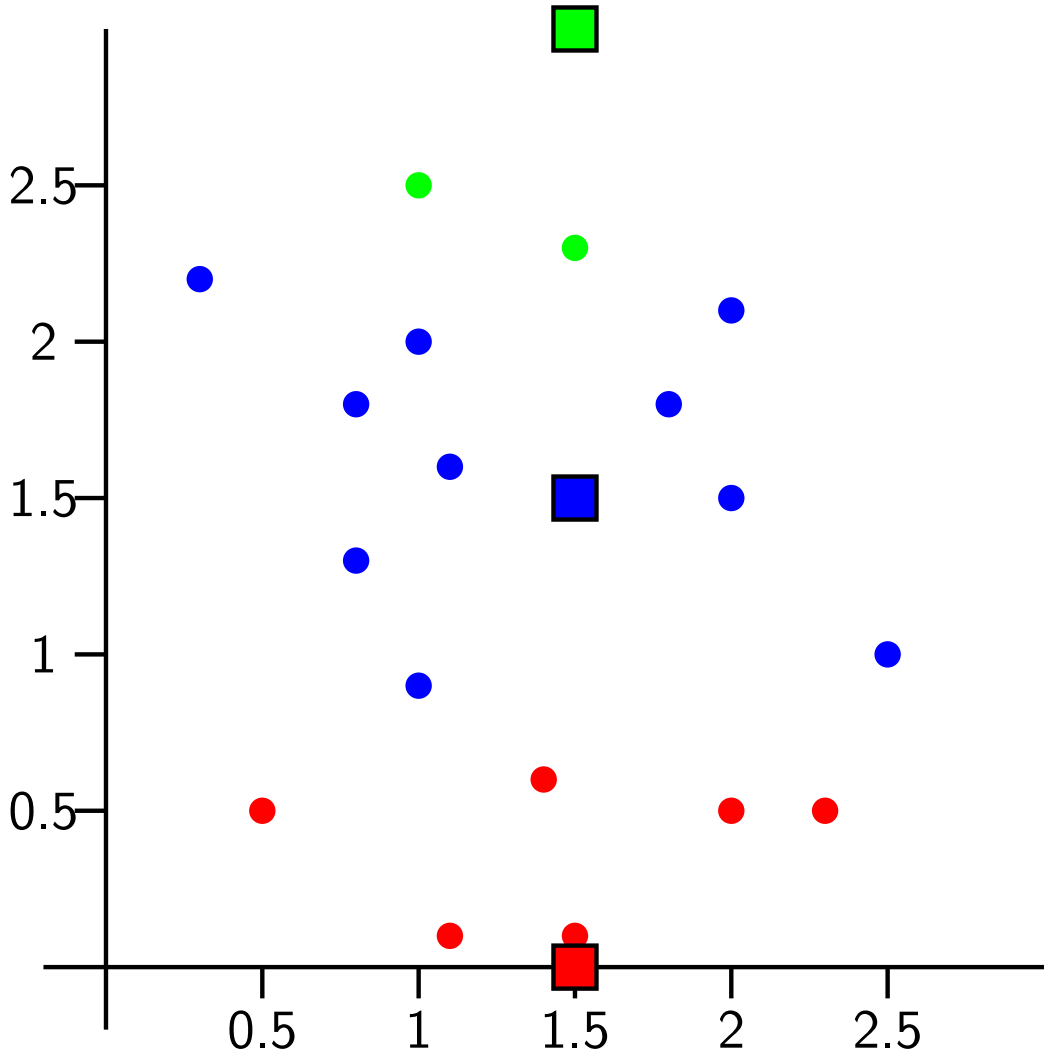
El resultat depèn dels $\mathbf{c}_1, \ldots, \mathbf{c}_k$ inicials.

Aquest algoritme no té perquè donar un clustering òptim.
Convé repetir-lo diverses vegades amb diferents inicialitzacions.

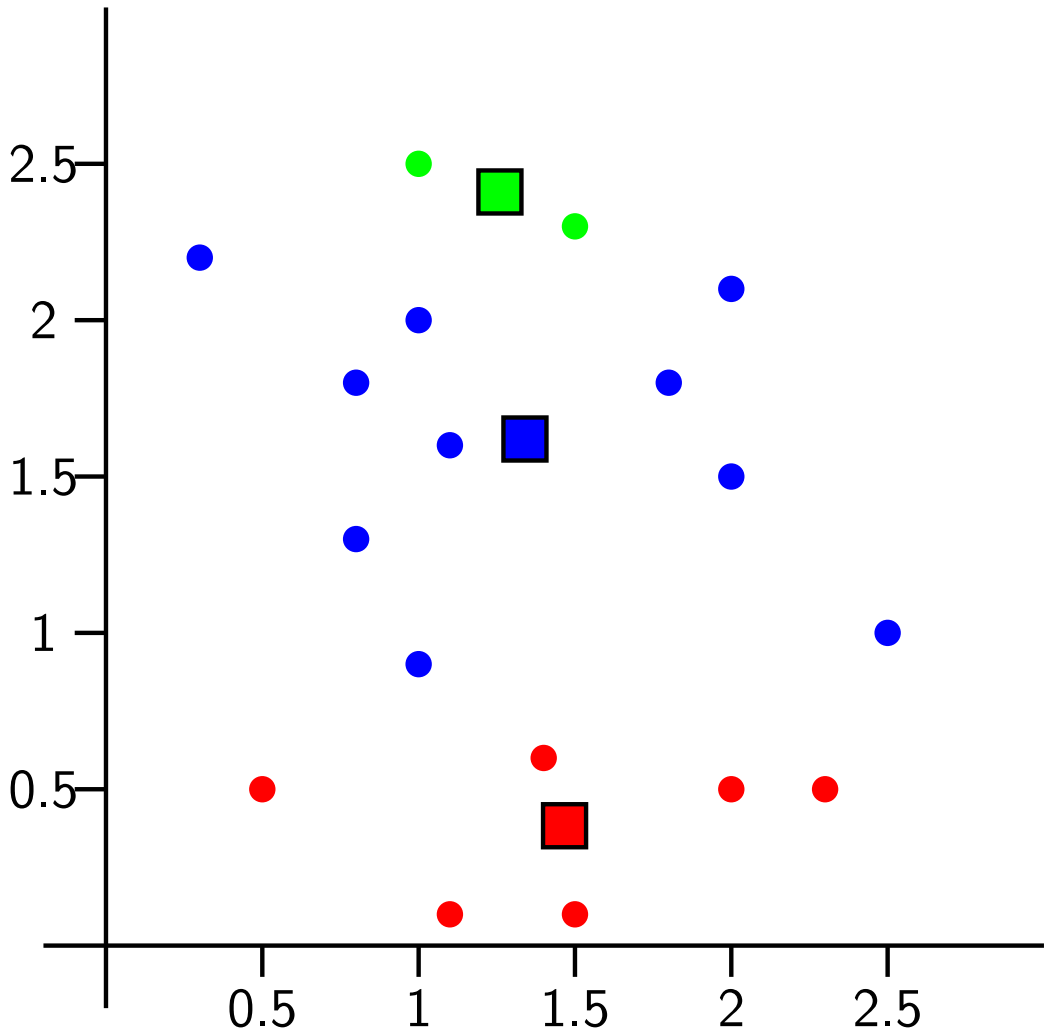
Exemple



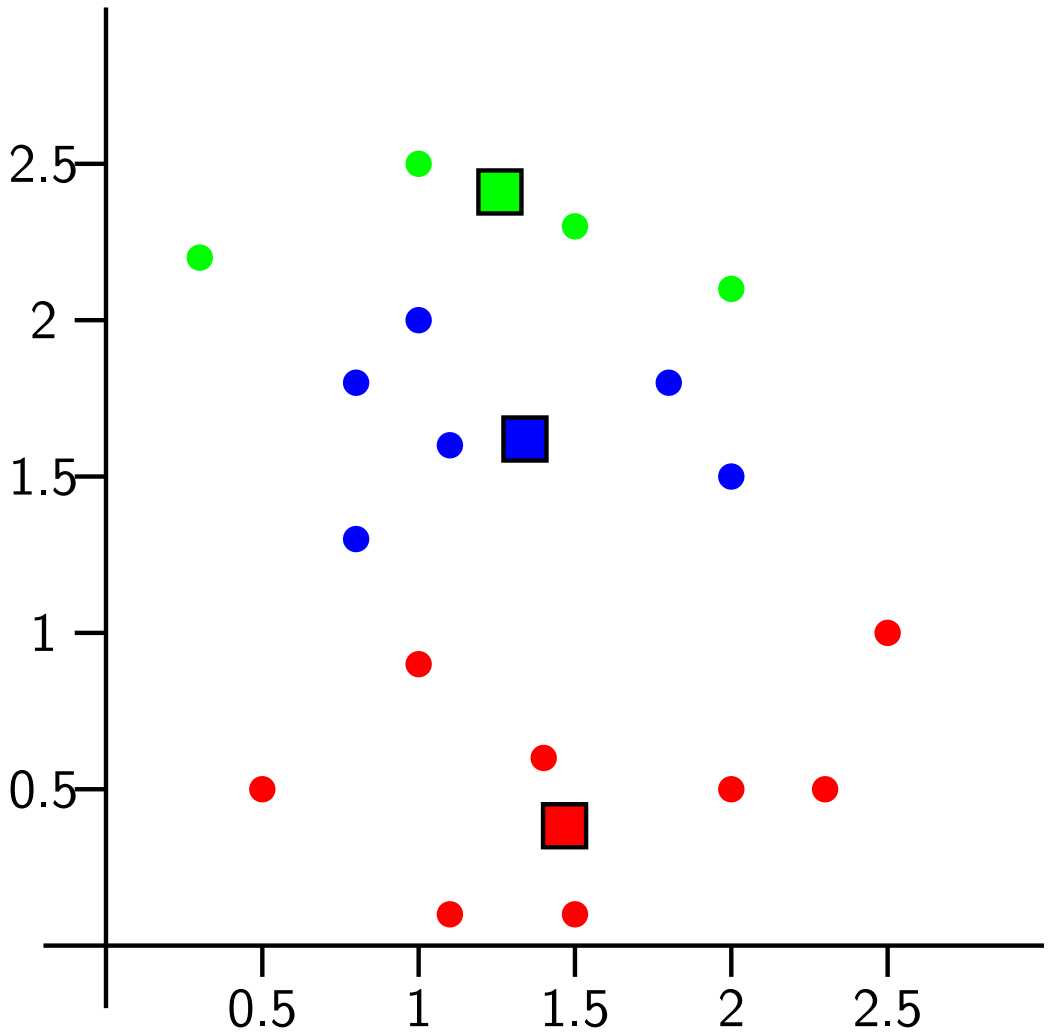
Exemple



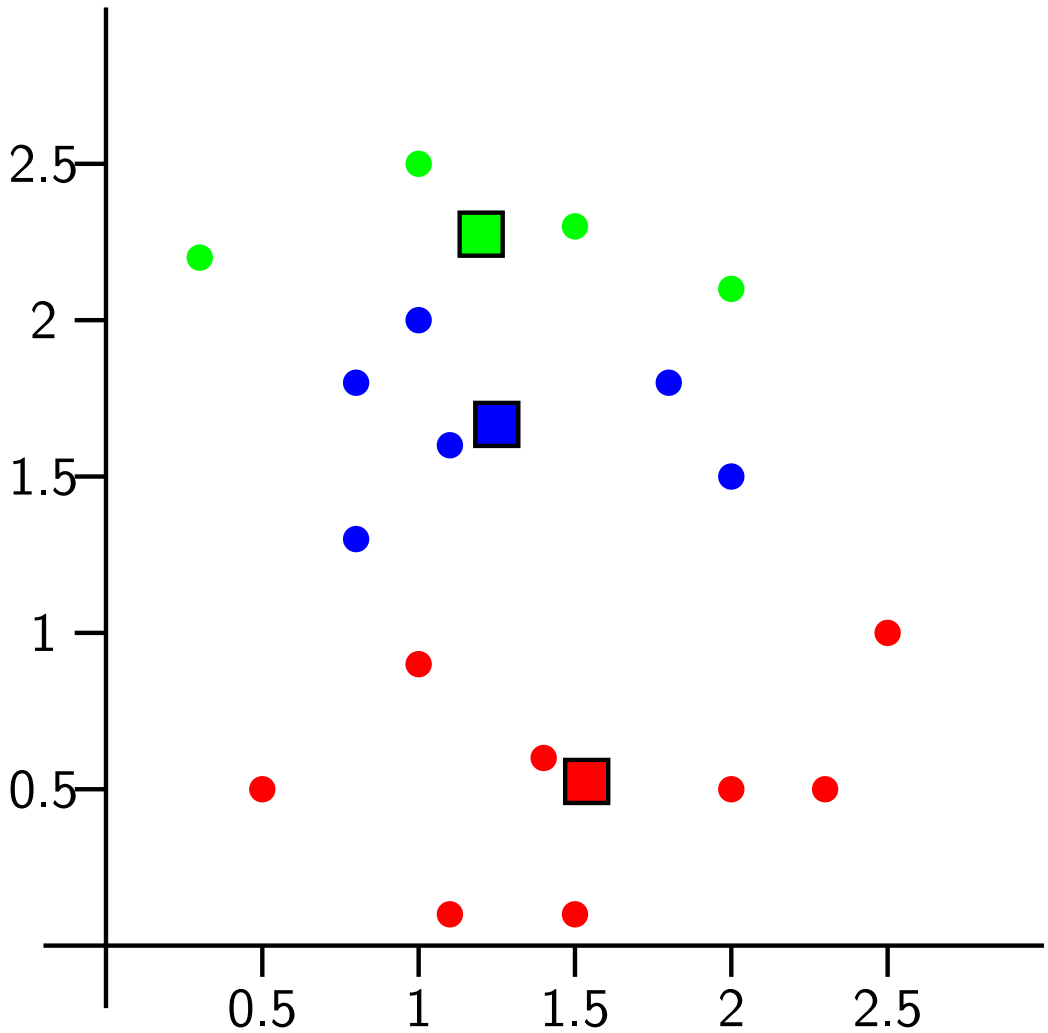
Exemple



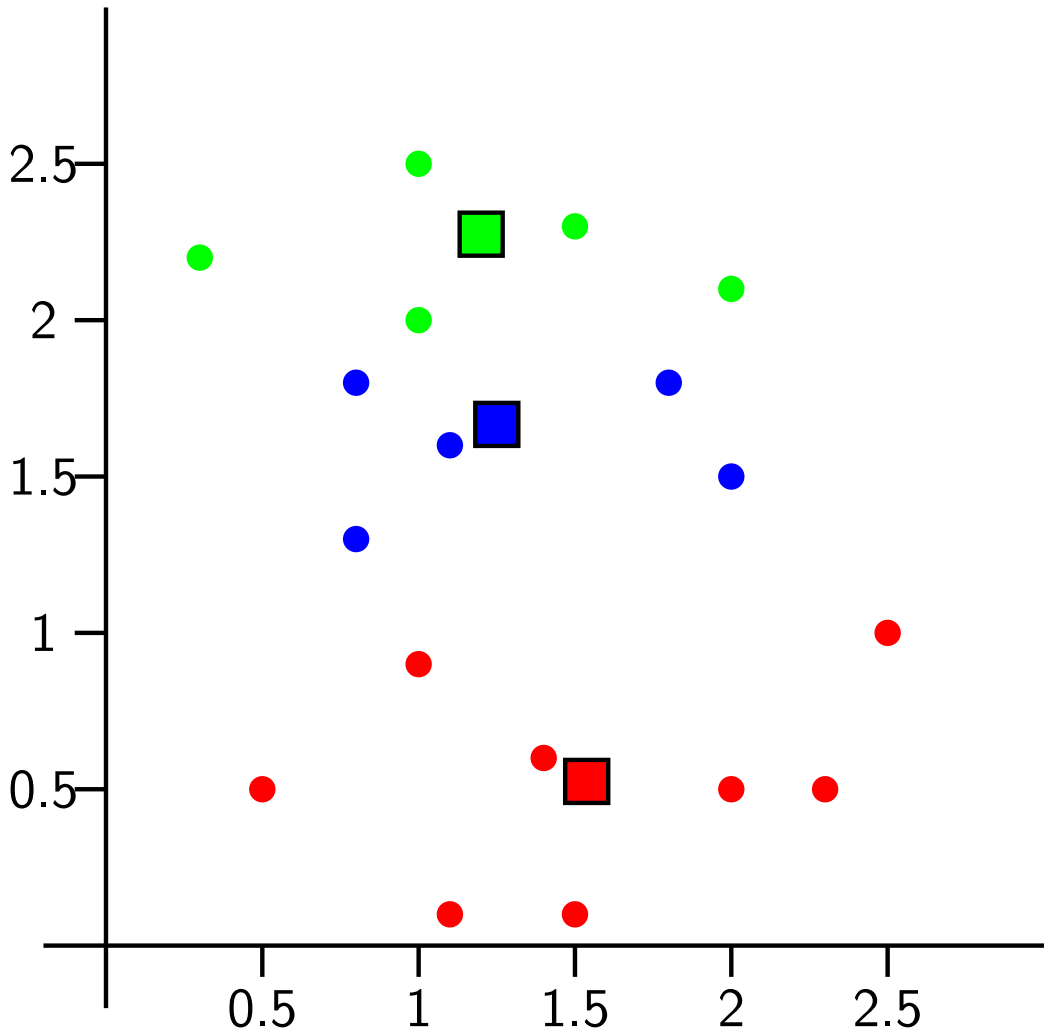
Exemple



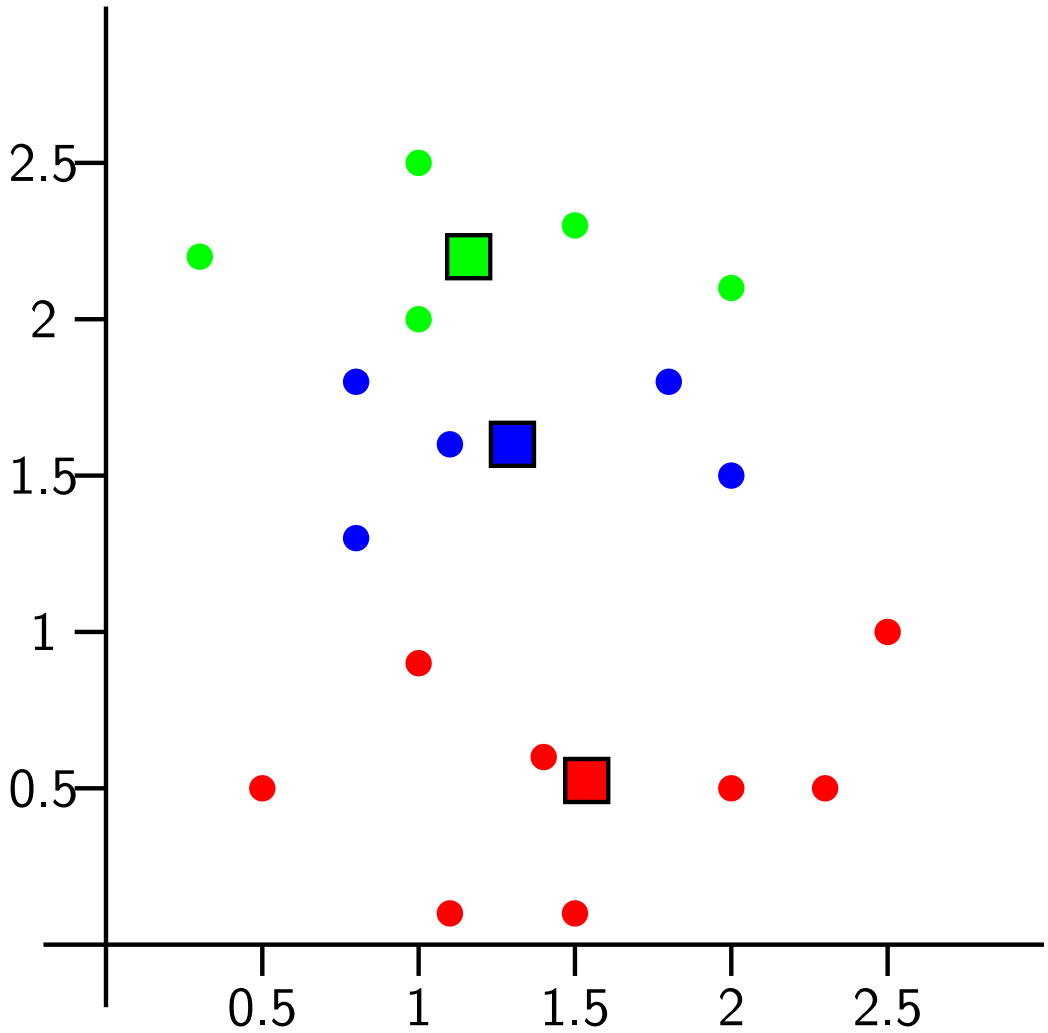
Exemple



Exemple



Exemple



I s'atura: $SS_C = 7.25375$

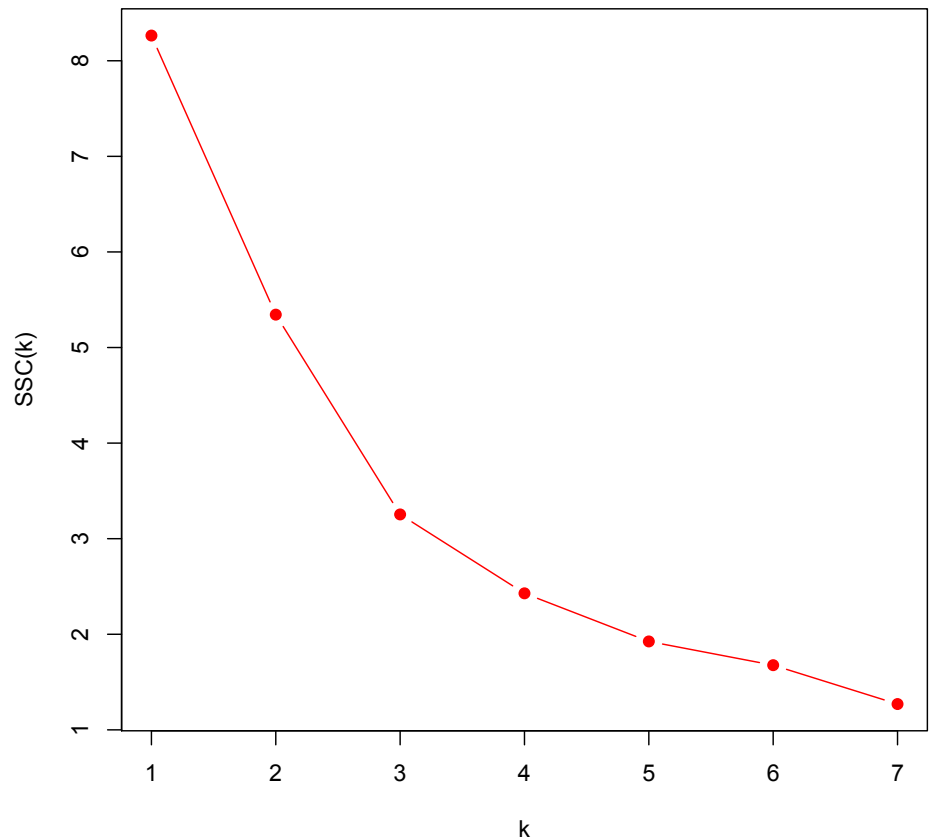
k-means

Limitacions de k-means:

- No hi ha un mètode eficient i universal de triar els centres de partida
- No es pot garantir un òptim global
- No es pot determinar de manera efectiva el nombre k *a priori*
- No és invariant per canvi d'escala (convé estandarditzar dades)
- Sensible a *outliers*
- Només aplicable dins $\mathbb{R}^n$ amb distància euclidiana
- Troba clusters esfèrics

Quina k? El mètode del colze

La SS_C òptima $SS_C(k)$ minva amb k seguint una funció més o menys còncava. Si podem detectar un k a partir del qual SS_C minva molt més lentament que abans d'ell, aquest serà el k recomanable.



$k = 3$ és el recomanable

Quina k? Test F

Es calcula

$$F_k = \frac{SS_C(k) - SS_C(k+1)}{\frac{SS_C(k+1)}{p-k-1}}$$

Es pren com a p-valor

$$P(F_{n,n(p-k-1)} > F_k)$$

amb $F_{n,n(p-k-1)}$ una F de Fisher amb n i $n(p-k-1)$ graus de llibertat, i triam el k amb p-valor més petit.

Cal dir que és un mètode molt emprat, però no massa justificable

Quina k? Test F

En l'exemple del gràfic anterior

k	2	3	4	5	6	7	8
$SS_C(k)$	8.264	5.344	3.254	2.428	1.925	1.677	1.27
F_k	8.2	9	4.42	3.14	1.63	3.2	
p-valor	0.0014	0.001	0.02	0.06	0.229	0.06	

$k = 3$ torna a ser el més recomanable

Si el conjunt de punts és molt gran, tots els p-valors són propers a 0 i aquest mètode no és útil.

k-means amb R

La instrucció bàsica per executar un k-means amb R és

```
kmeans(x,centres,iter.max=...)
```

amb

- `x`, una matriu amb els punts $\mathbf{x}_i$ com a fileres
- `centres`, una matriu amb els centres $\mathbf{c}_i$ de partida com a fileres, o el nombre k
- `iter.max` el nombre màxim d'iteracions

Aquesta instrucció no segueix exactament el nostre algoritme, si voleu que executi l'algoritme explicat hi heu d'entrar, a més, `algorithm="Lloyd"`

k-means amb R

```
> dades=matrix(c(0.8,1.3,0.8,1.8,1.0,0.9,1.1,
0.1,1.1,1.6,1.4,0.6,1.5,0.1,2,2.1,1.5,2.3,1.8,
1.8,2.3,0.5,0.3,2.2,1,2.5,2,0.5,2,1.5,2.5,1,
0.5,0.5,1,2),
nrow=18,byrow=TRUE)
> cent=matrix(c(0.5,0,0.5,1.5,0.5,3),
nrow=3,byrow=TRUE)
```

k-means amb R

```
> kmeans(dades,cent,algorithm="Lloyd")
$k$-means clustering with 3 clusters of sizes
8, 5, 5
Cluster means:
    [,1]  [,2]
1 1.5375 0.525
2 1.3000 1.600
3 1.1600 2.220
Clustering vector:
 [1] 2 2 1 1 2 1 1 3 3 2 1 3 3 1 2 1 1 3
Within cluster sum of squares by cluster:
[1] 4.03375 1.46000 1.76000
 (between_SS / total_SS =  57.9 %)
...
```

k-means amb R

Components de la `list` `kmeans`:

- `cluster`: assignacions d'elements a clusters

```
> km=kmeans(dades,cent,algorithm="Lloyd")
> km$cluster
 [1] 2 2 1 1 2 1 1 3 3 2 1 3 3 1 2 1 1 3
```

- `centers`: els centres dels clusters

```
> km$centers
    [,1]  [,2]
1 1.5375 0.525
2 1.3000 1.600
3 1.1600 2.220
```

k-means amb R

Components de la `list` `kmeans`:

- `totss`: suma dels quadrats de les distàncies dels punts al punt mig de tots aquests punts.

```
> km$totss
[1] 17.20944
```

- `withinss`: vector de les sumes, per a cada cluster, dels quadrats de les distàncies dels seus punts al seu centre

```
> km$withinss
[1] 4.03375 1.46000 1.76000
```

k-means amb R

Components de la `list` `kmeans`:

- `tot.withinss`: suma de `withinss`, SS_C

  ```
  > km$tot.withinss
  [1] 7.25375
  ```

- `betweenss`: diferència `totss` − `tot.withinss`. És la suma, ponderada pel nombre d'objectes del cluster corresponent, dels quadrats de les distàncies dels centres dels clusters al punt mig de tots els punts.

  ```
  > km$betweenss
  [1] 9.955694
  ```

k-means amb R

Components de la `list` `kmeans`:

- Ens interessa `betweenss/totss`, que mesura la fracció de la variabilitat de les dades que expliquen els clusters.
  Com més gran millor
  Al resultat de `kmeans` és `between_SS / total_SS`

```
> km
...
 (between_SS / total_SS =  57.9 %)
...
> 9.955694/17.20944  #betweenss/totss
[1] 0.5785019
```

k-means amb R

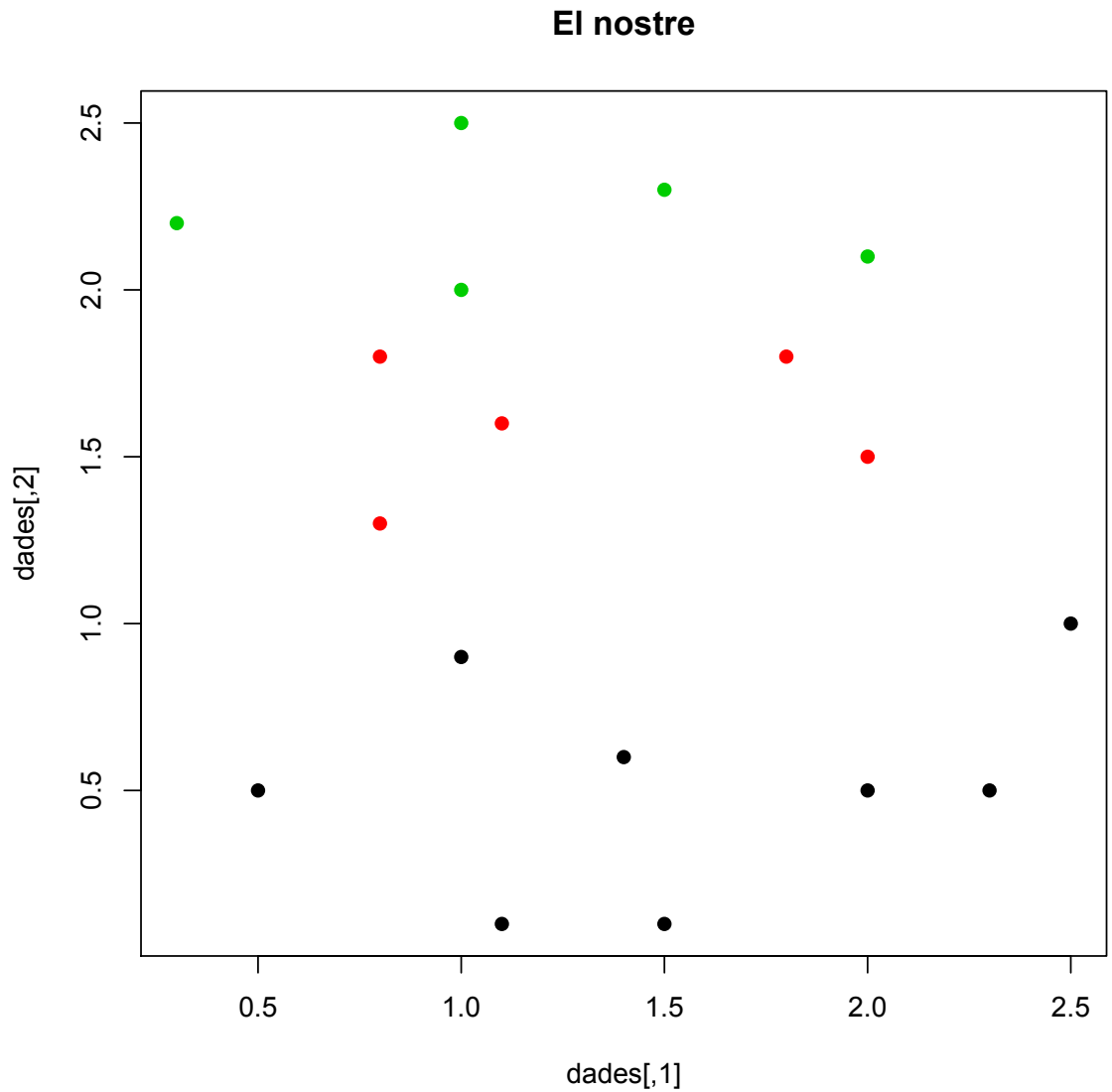
```
> km.rand=kmeans(dades,3,algorithm="Lloyd")
> km.rand
$k$-means clustering with 3 clusters of sizes
6, 5, 7
Cluster means:
       [,1]     [,2]
1 2.1000000 1.233333
2 1.1000000 0.440000
3 0.9285714 1.957143
Clustering vector:
 [1] 3 3 2 2 3 2 2 1 3 1 1 3 3 1 1 1 2 3
Within cluster sum of squares by cluster:
[1] 2.593333 1.092000 1.851429
 (between_SS / total_SS =  67.8 %)
> km.rand$tot.withinss
[1] 5.965111
```

k-means amb R

```
> km2=kmeans(dades,3)  #5a repeticio ;-)
> km2
K-means clustering with 3 clusters of sizes
5, 4, 9
Cluster means:
      [,1]     [,2]
1 1.100000 0.440000
2 2.200000 0.875000
3 1.144444 1.955556
Clustering vector:
 [1] 3 3 1 1 3 1 1 3 3 3 2 3 3 2 2 2 1 3
Within cluster sum of squares by cluster:
[1] 1.092000 0.867500 3.384444
 (between_SS / total_SS =  68.9 %)
> km2$tot.withinss
[1] 5.343944
```

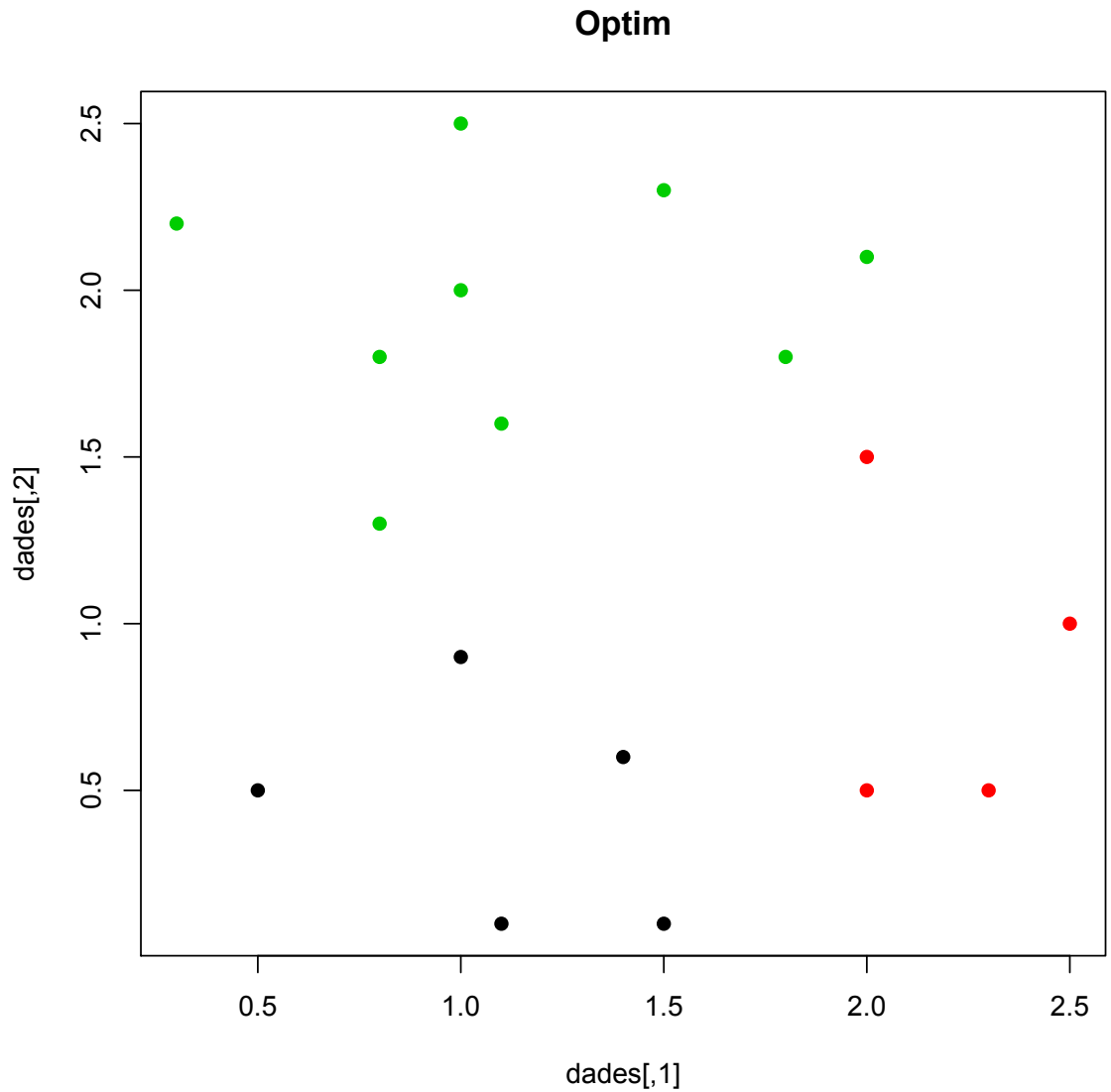
k-means amb R

```
> plot(dades,col=km$cluster,pch=19,
 main="El nostre")
```



k-means amb R

```
> plot(dades,col=km2$cluster,pch=19,
 main="Optim")
```



Mètodes jeràrquics

Els mètodes jeràrquics parteixen d'una matriu D de semblances o de distàncies entre els objectes

Si tenim p objectes, necessitam una matriu

$$D = \begin{pmatrix} d_{11} & d_{12} & \dots & d_{1p} \\ d_{21} & d_{22} & \dots & d_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ d_{p1} & d_{p2} & \dots & d_{pp} \end{pmatrix}$$

on cada d_{ij} és la distància o la semblança entre l'objecte i i l'objecte j

Semblances

Una semblança sobre un conjunt X és una aplicació $\sigma : X \times X \to [0, 1]$ que és:

- Reflexiva: Si $x = y$, aleshores $\sigma(x, y) = 1$
- Simètrica: $\sigma(x, y) = \sigma(y, x)$

Dos objectes x, y són més semblants com més gran és $\sigma(x, y)$

Distàncies

Una distància sobre un conjunt X és una aplicació $d : X \times X \to [0, \infty[$ que satisfà:

- Separació: $d(x, y) = 0$ si, i només si, $x = y$
- Simetria: $d(x, y) = d(y, x)$
- Desigualtat triangular: $d(x, z) \leqslant d(x, y) + d(y, z)$

Dos objectes x, y són més semblants com més petita és $d(x, y)$

El primer problema és escollir la semblança o la distància a emprar, segons el significat que vulguem que tingui el clustering. És una decisió molt important!

Dades binàries

Partim de p objectes, dels quals hem pres n medicions, i els organitzam en fileres d'una matriu

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{p1} & x_{p2} & \dots & x_{pn} \end{pmatrix}$$

Suposem que les medicions són binàries (0 o 1)

Exemple: Propietats dicotòmiques d'organismes

	Pèl	Pulmons	Ovípar	Llet
Ca	1	1	0	1
Granot	0	1	1	0
Puput	0	1	1	0
Ornitorrinc	1	1	1	1
Salmó	0	0	1	0

Dades binàries

Donades dues fileres (objectes)

$$\mathbf{x}_i = (x_{i1}, \ldots, x_{in}), \quad \mathbf{x}_j = (x_{j1}, \ldots, x_{jn}),$$

definim les quantitats següents:

$$\begin{aligned} a_0 &= \left|\{k \mid x_{ik} = x_{jk} = 0\}\right| \\ a_1 &= \left|\{k \mid x_{ik} = x_{jk} = 1\}\right| \\ a_2 &= \left|\{k \mid x_{ik} \neq x_{jk} = 1\}\right| \end{aligned}$$

Una semblança entre els objectes i i j es pot definir mitjançant la fórmula genèrica

$$\sigma_{ij} = \frac{a_1 + \delta a_0}{\alpha a_1 + \beta a_0 + \lambda a_2}$$

Dades binàries

Els paràmetres δ i λ són factors que donen pes a característiques. Els més comuns:

Nom	δ	λ	α	β	Definició
Hamming	1	1	1	1	$\frac{a_1 + a_0}{n}$
Jaccard	0	1	1	0	$\frac{a_1}{a_1 + a_2}$
Tanimoto	1	2	1	1	$\frac{a_1 + a_0}{a_1 + 2a_2 + a_0}$
Rusell–Rao	0	1	1	1	$\frac{a_1}{n}$
Diu	0	0.5	1	0	$\frac{2a_1}{2a_1 + a_2}$
Kulczynski	0	1	0	0	$\frac{a_1}{a_2}$

Exemple

De 3 organismes hem observat si contenen o no gens homòlegs a 8 gens prototipus. Els resultats són els de la taula següent (1=Sí, 0=No)

	Gens							
Organisme	A	B	C	D	E	F	G	H
X	0	1	1	0	1	1	0	0
Y	1	0	0	1	0	0	1	1
Z	0	0	1	0	1	0	1	0

La matriu de semblances de Hamming és

$$\mathbf{D}_H = \begin{pmatrix} 1.000 & 0.000 & 0.625 \\ & 1.000 & 0.375 \\ & & 1.000 \end{pmatrix}$$

Matrius de contingència

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{p1} & x_{p2} & \cdots & x_{pn} \end{pmatrix}$$

on cada entrada és una freqüència

Siguin

$$x_{i\bullet} = \sum_{k=1}^{n} x_{ik}, \quad x_{\bullet k} = \sum_{i=1}^{p} x_{ik}, \quad x_{\bullet\bullet} = \sum_{i=1}^{p} x_{i\bullet} = \sum_{k=1}^{n} x_{\bullet k}$$

Se recomana prendre com a distància

$$d_{ij} = \sqrt{\sum_{k=1}^{n} \frac{x_{\bullet\bullet}}{x_{\bullet k}} \left(\frac{x_{ik}}{x_{i\bullet}} - \frac{x_{jk}}{x_{j\bullet}} \right)^2}$$

Exemple

A 3 boscos s'hi ha escollit una àrea de la mateixa superfície i s'hi han comptat els nombres d'exemplars de 5 plantes.

	Planta				
Bosc	A	B	C	D	E
X	12	3	8	0	24
Y	3	22	15	8	11
Z	0	7	12	20	6

Taula amb freqüències marginals:

	Planta					
Bosc	A	B	C	D	E	$x_{i\bullet}$
X	12	3	8	0	24	47
Y	3	22	15	8	11	59
Z	0	7	12	20	6	45
$x_{\bullet j}$	15	32	35	28	41	151

Exemple

	Planta					
Bosc	A	B	C	D	E	$x_{i\bullet}$
X	12	3	8	0	24	47
Y	3	22	15	8	11	59
Z	0	7	12	20	6	45
$x_{\bullet j}$	15	32	35	28	41	151

$$
\begin{aligned}
d^2_{XY} &= \frac{151}{15}\left(\frac{12}{47}-\frac{3}{59}\right)^2+\frac{151}{32}\left(\frac{3}{47}-\frac{22}{59}\right)^2 \\
&\quad +\frac{151}{35}\left(\frac{8}{47}-\frac{15}{59}\right)^2+\frac{151}{28}\left(\frac{0}{47}-\frac{8}{59}\right)^2 \\
&\quad +\frac{151}{41}\left(\frac{24}{47}-\frac{11}{59}\right)^2=\ldots
\end{aligned}
$$

Exemple

	Planta					
Bosc	A	B	C	D	E	$x_{i\bullet}$
X	12	3	8	0	24	47
Y	3	22	15	8	11	59
Z	0	7	12	20	6	45
$x_{\bullet j}$	15	32	35	28	41	151

$$D = \begin{pmatrix} 0 & 1.178 & 1.525 \\ & 0 & 0.880 \\ & & 0 \end{pmatrix}$$

Dades contínues

Quan tenim els objectes descrits com a vectors de $\mathbb{R}^n$ i cada entrada correspon a l'observació d'una variable contínua, se solen emprar distàncies basades en les normes L_r: Donats

$$\mathbf{x}_i = (x_{i1}, \ldots, x_{in}), \quad \mathbf{x}_j = (x_{j1}, \ldots, x_{jn}),$$

la distància L_r entre aquests és

$$d_{ij} = \|\mathbf{x_i} - \mathbf{x_j}\|_r = \left(\sum_{k=1}^{n} |x_{ik} - x_{jk}|^r \right)^{1/r}$$

Dades contínues

Quan $r = 1$,

$$d_{ij} = \sum_{k=1}^{n} |x_{ik} - x_{jk}|$$

se'n diu la distància de Manhattan

Quan $r = 2$,

$$d_{ij} = \sqrt{\sum_{k=1}^{n} (x_{ik} - x_{jk})^2}$$

és la distància euclidiana

Escalat de les dades

De vegades és convenient que les dades estiguin en la mateixa escala, per evitar diferències en les contribucions de les diferents columnes

Quan s'empra la distància euclidiana, per escalar es divideix cada entrada x_{ik} per la desviació típica $s_{\bullet k}$ de la columna corresponent abans d'aplicar la distància.

Queda

$$d_{ij} = \sqrt{\sum_{k=1}^{n} \frac{(x_{ik} - x_{jk})^2}{s_{\bullet k}^2}}$$

Clustering jeràrquic

Existeixen dos tipus de mètodes de clustering jeràrquic:

- Els algoritmes aglomeratius comencen amb la partició més fina possible (cada objecte constitueix un cluster) i els van agrupant.
- Els algoritmes de divisió comencen amb la partició més grollera possible (tots els objectes constitueixen un cluster) i van dividint els clusters en clusters més petits.

Els algoritmes aglomeratius són més populars, perquè en general requereixen menys temps de càlcul

Algoritme bàsic de clustering jeràrquic aglomeratiu

Algoritme bàsic

1. Partim de p objectes, i de la matriu $p \times p$ de distàncies entre ells
2. Formam un cluster amb cada objecte
3. Trobam dos clusters a distància mínima C_1 i C_2
4. Unim C_1 i C_2 en un cluster nou $C_1 + C_2$
5. Eliminam C_1 i C_2 de la llista de clusters
6. Recalculam la distància de $C_1 + C_2$ als altres clusters
7. Repetim (3)–(6) fins que només queda un únic cluster

Clustering jeràrquic aglomeratiu

El càlcul de la distància entre clusters es pot fer de diverses maneres, donant lloc a resultats diferents:

- Per enllaç simple:
  $d(C, C') = \min\{d(a, b) \mid a \in C, b \in C'\}$

  En aquest cas

  $$d(C, C_1 + C_2) = \min\{d(C, C_1), d(C, C_2)\}$$

- Per enllaç complet:
  $d(C, C') = \max\{d(a, b) \mid a \in C, b \in C'\}$

  En aquest cas

  $$d(C, C_1 + C_2) = \max\{d(C, C_1), d(C, C_2)\}$$

Clustering jeràrquic aglomeratiu

El càlcul de la distància entre clusters es pot fer de diverses maneres, donant lloc a resultats diferents:

- Per enllaç mitjà:

$$d(C, C') = \frac{\sum_{a \in C, b \in C'} d(a, b)}{|C| \cdot |C'|}$$

En aquest cas,

$$\begin{aligned} & d(C, C_1 + C_2) \\ & = \frac{|C_1|}{|C_1| + |C_2|} d(C, C_1) + \frac{|C_2|}{|C_1| + |C_2|} d(C, C_2) \end{aligned}$$

- ...

Clustering jeràrquic aglomeratiu

En general, conegudes

$$d(C, C_1), \quad d(C, C_2), \quad d(C_1, C_2),$$

hi ha una fórmula genèrica per calcular $d(C, C_1 + C_2)$:

$$\begin{aligned} d(C, C_1 + C_2) = \delta_1 d(C, C_1) + \delta_2 d(C, C_2) + \delta_3 d(C_1, C_2) \\ + \delta_0 |d(C, C_1) - d(C, C_2)|, \end{aligned}$$

on els δ_i son paràmetres a triar. Cada tria dóna un algoritme diferent, amb resultats possiblement diferents.

Clustering jeràrquic aglomeratiu

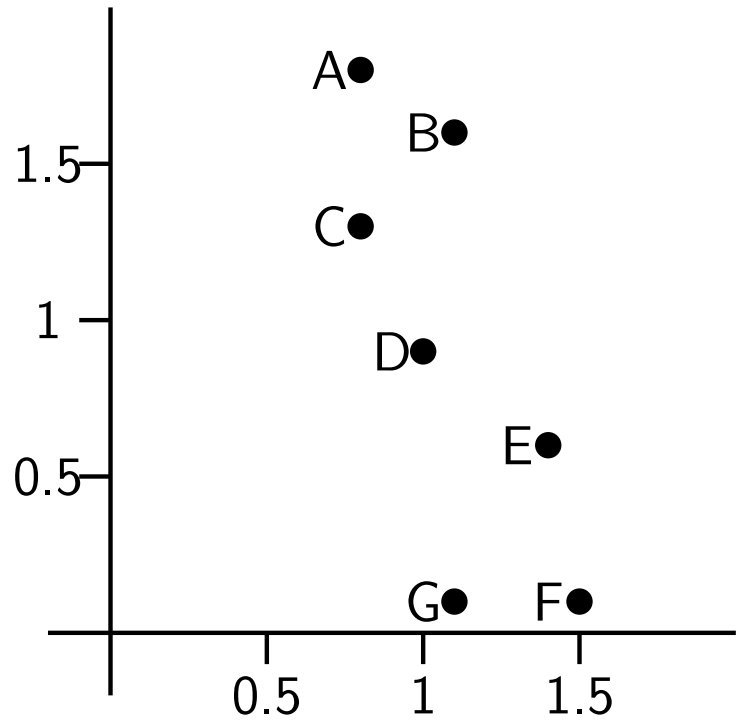
Si diem n_X al nombre d'elements d'un cluster X:

Nom	δ_1	δ_2	δ_3	δ_0
Enllaç simple	$1/2$	$1/2$	0	$-1/2$
Enllaç complet	$1/2$	$1/2$	0	$1/2$
Enllaç mitjà	$\frac{n_{C_1}}{n_{C_1}+n_{C_2}}$	$\frac{n_{C_2}}{n_{C_1}+n_{C_2}}$	0	0
Centroide	$\frac{n_{C_1}}{n_{C_1}+n_{C_2}}$	$\frac{n_{C_2}}{n_{C_1}+n_{C_2}}$	$-\frac{n_{C_1} n_{C_2}}{(n_{C_1}+n_{C_2})^2}$	0
Mediana	$1/2$	$1/2$	$-1/4$	0
Ward	$\frac{n_C+n_{C_1}}{n_C+n_{C_1}+n_{C_2}}$	$\frac{n_C+n_{C_2}}{n_C+n_{C_1}+n_{C_2}}$	$-\frac{n_C}{n_C+n_{C_1}+n_{C_2}}$	0

Exemple

A,B,C,D,E,F,G: plantes;
x, y: gens;
dades: expressió del gen en condicions de sequera

	x	y
A	0.8	1.8
B	1.1	1.6
C	0.8	1.3
D	1.0	0.9
E	1.4	0.6
F	1.5	0.1
G	1.1	0.1



Comparam les dades amb distància euclidiana. Emprarem enllaç simple.

Exemple

Matriu de distàncies

	A	B	C	D	E	F	G
A							
B	0.3606						
C	0.5000	0.4243					
D	0.9220	0.7071	0.4472				
E	1.3416	1.0440	0.9220	0.5000			
F	1.8385	1.5524	1.3892	0.9434	0.5099		
G	1.7263	1.5000	1.2369	0.8062	0.5381	0.4000	

Exemple

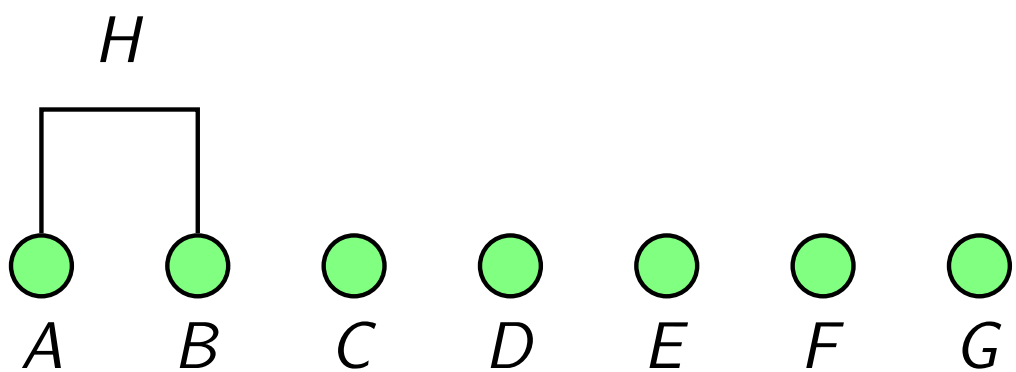
Detectam un mínim

	A	B	C	D	E	F	G
A							
B	0.3606						
C	0.5000	0.4243					
D	0.9220	0.7071	0.4472				
E	1.3416	1.0440	0.9220	0.5000			
F	1.8385	1.5524	1.3892	0.9434	0.5099		
G	1.7263	1.5000	1.2369	0.8062	0.5381	0.4000	

Substituïm {A,B} per H i recalculam

	H	C	D	E	F	G
H						
C	0.4243					
D	0.7071	0.4472				
E	1.0440	0.9220	0.5000			
F	1.5524	1.3892	0.9434	0.5099		
G	1.5000	1.2369	0.8062	0.5381	0.4000	

Exemple



Exemple

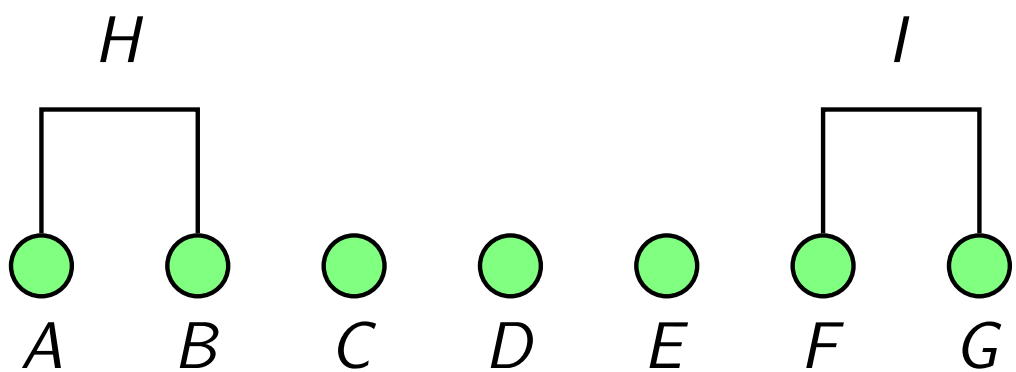
Detectam un mínim

	H	C	D	E	F	G
H						
C	0.4243					
D	0.7071	0.4472				
E	1.0440	0.9220	0.5000			
F	1.5524	1.3892	0.9434	0.5099		
G	1.5000	1.2369	0.8062	0.5381	0.4000	

Substituïm {F,G} per I i recalculam

	H	C	D	E	I
H					
C	0.4243				
D	0.7071	0.4472			
E	1.0440	0.9220	0.5000		
I	1.5000	1.2369	0.8062	0.5099	

Exemple



Exemple

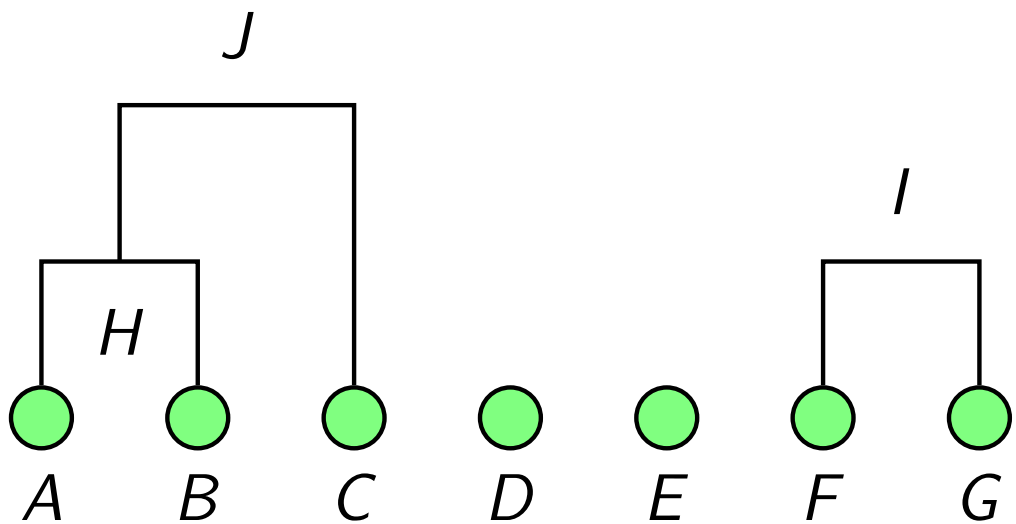
Detectam un mínim

	H	C	D	E	I
H					
C	0.4243				
D	0.7071	0.4472			
E	1.0440	0.9220	0.5000		
I	1.5000	1.2369	0.8062	0.5099	

Substituïm {H,C} per J i recalculam

	J	D	E	I
J				
D	0.4472			
E	0.9220	0.5000		
I	1.2369	0.8062	0.5099	

Exemple



Exemple

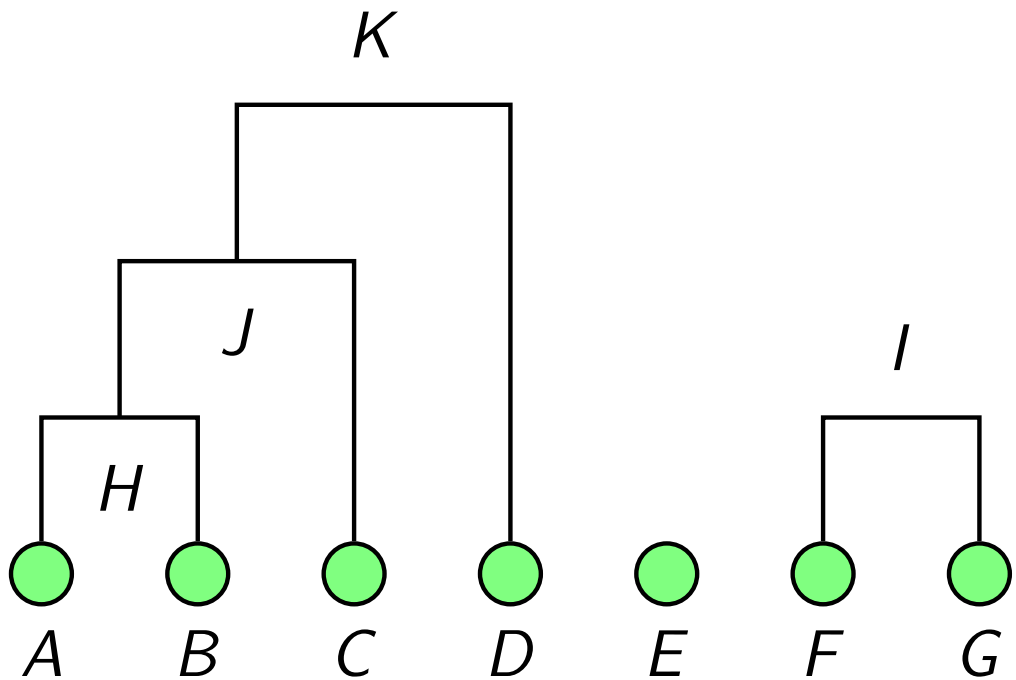
Detectam un mínim

	J	D	E	I
J				
D	0.4472			
E	0.9220	0.5000		
I	1.2369	0.8062	0.5099	

Substituïm {J,D} per K i recalculam

	K	E	I
K			
E	0.5000		
I	0.8062	0.5099	

Exemple



Exemple

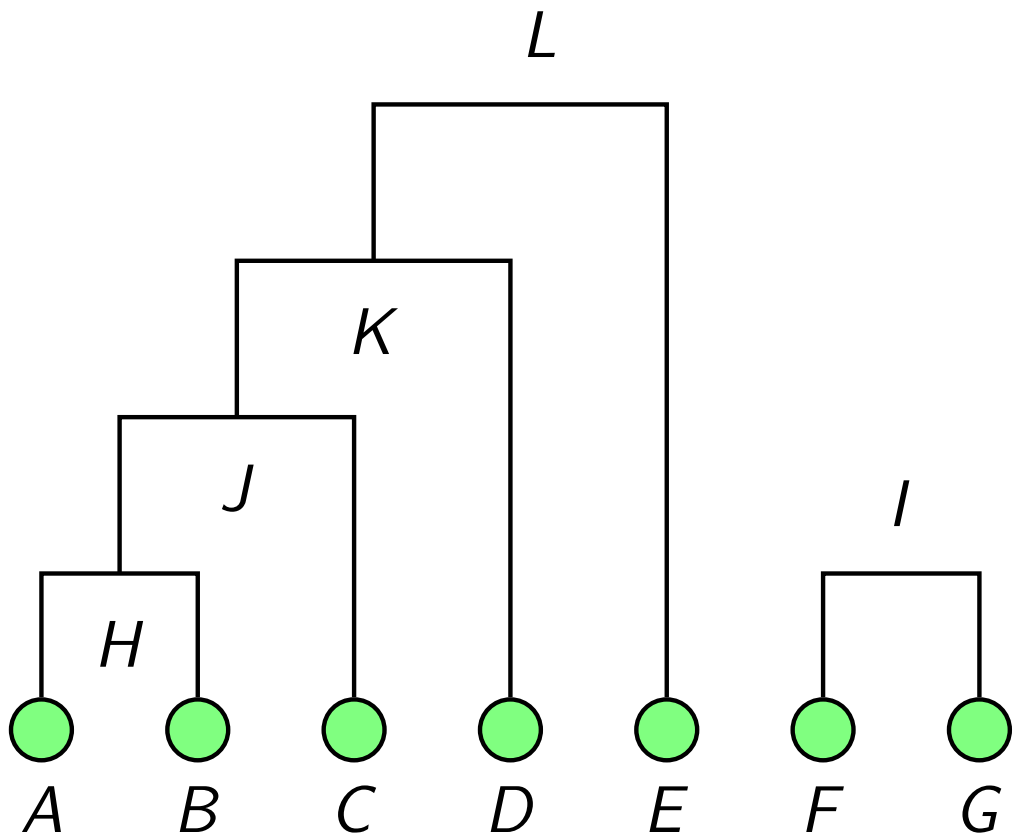
Detectam un mínim

	K	E	I
K			
E	0.5000		
I	0.8062	0.5099	

Substituïm {K,E} per L i recalculam

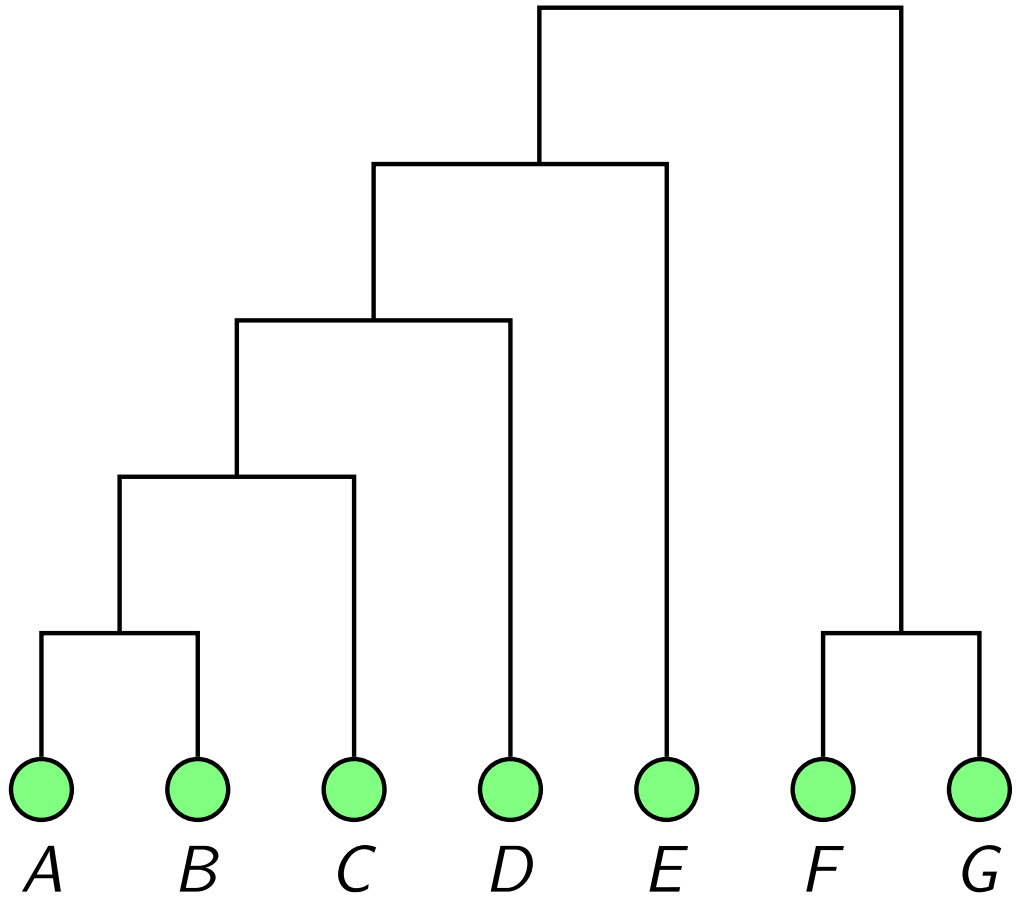
	L	I
L		
I	0.5099	

Exemple



Exemple

Finalment, unim L i I en un sol cluster



Limitacions del clustering jeràrquic aglomeratiu

- La distància que s'hi empra és molt important
- No hi ha teoria que avali quin mètode per calcular la distància entre clusters és el millor en cada cas
- Realment, no defineix directament clusters, però tallant en una alçada del dendrograma n'obtenim
- Sempre agrupa de dos en dos, i de vegades pren decisions aleatòries per aconseguir-ho

Clustering jeràrquic aglomeratiu amb R

La instrucció bàsica és

```
hclust(d, method = "...")
```

on

- `d` és una matriu de distàncies
- `method` serveix per especificar el mètode: `"single"`, `"complete"`, `"average"`, `"ward"`, `"median"`, `"centroid"`, ...

Clustering jeràrquic aglomeratiu amb R

Per representar el clustering per mitjà d'un dendrograma, cal aplicar al resultat de `hclust` la instrucció

```
plot(clust, labels=..., hang=..., ...)
```

on

- `clust` és un `hclust`
- `labels` serveix per posar noms als objectes
- `hang` serveix per especificar la posició de les etiquetes: mirau el `help`
- Altres paràmetres usuals dels `plot`

Clustering jeràrquic aglomeratiu amb R

Per calcular la distància entre les fileres d'una matriu, podem emprar

```
dist(x, method = "...")
```

on

- `x` és una matriu de dades
- `method` serveix per especificar el mètode: `"euclidean"`, `"manhattan"`, `...`

Exemple

```
>dades=matrix(data=c(0.8,1.8,1.1,1.6, 0.8,1.3,
  1.0,0.9,1.4, 0.6,1.5, 0.1,1.1,0.1),
  nrow=7,byrow=TRUE)
> dades
     [,1] [,2]
[1,]  0.8  1.8
[2,]  1.1  1.6
[3,]  0.8  1.3
[4,]  1.0  0.9
[5,]  1.4  0.6
[6,]  1.5  0.1
[7,]  1.1  0.1
```

Exemple

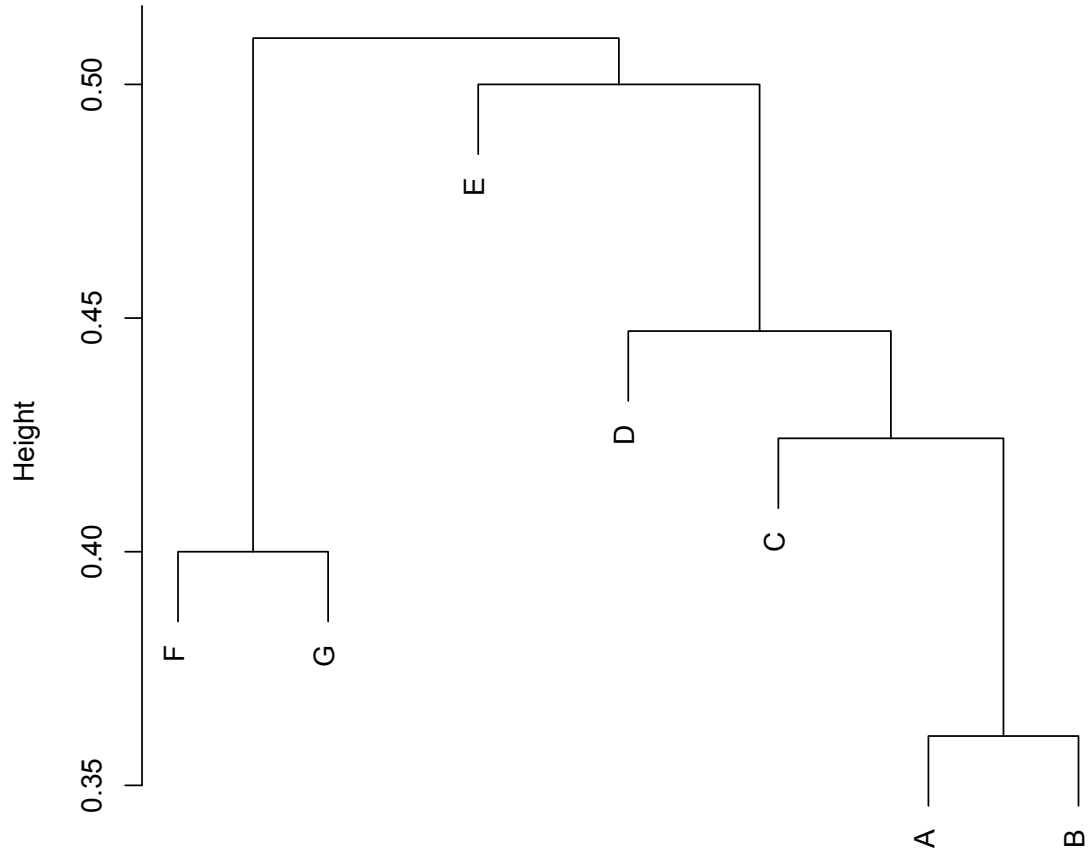
```
> distancies=dist(dades,method="euclidean")
> distancies
     1      2      3      4      5      6
2 0.3606
3 0.5000 0.4243
4 0.9220 0.7071 0.4472
5 1.3417 1.0440 0.9220 0.5000
6 1.8385 1.5524 1.3892 0.9434 0.5099
7 1.7263 1.5000 1.2370 0.8062 0.5831 0.4000
```

Exemple

```
> clustering=hclust(distancies,method="single")
> clustering$merge #formació de clusters
     [,1] [,2]
[1,]   -1   -2
[2,]   -6   -7
[3,]   -3    1
[4,]   -4    3
[5,]   -5    4
[6,]    2    5
> clustering$height  #distàncies mínimes
[1] 0.3605551 0.4000000 0.4242641 0.4472136
  0.5000000 0.5099020
```

Exemple

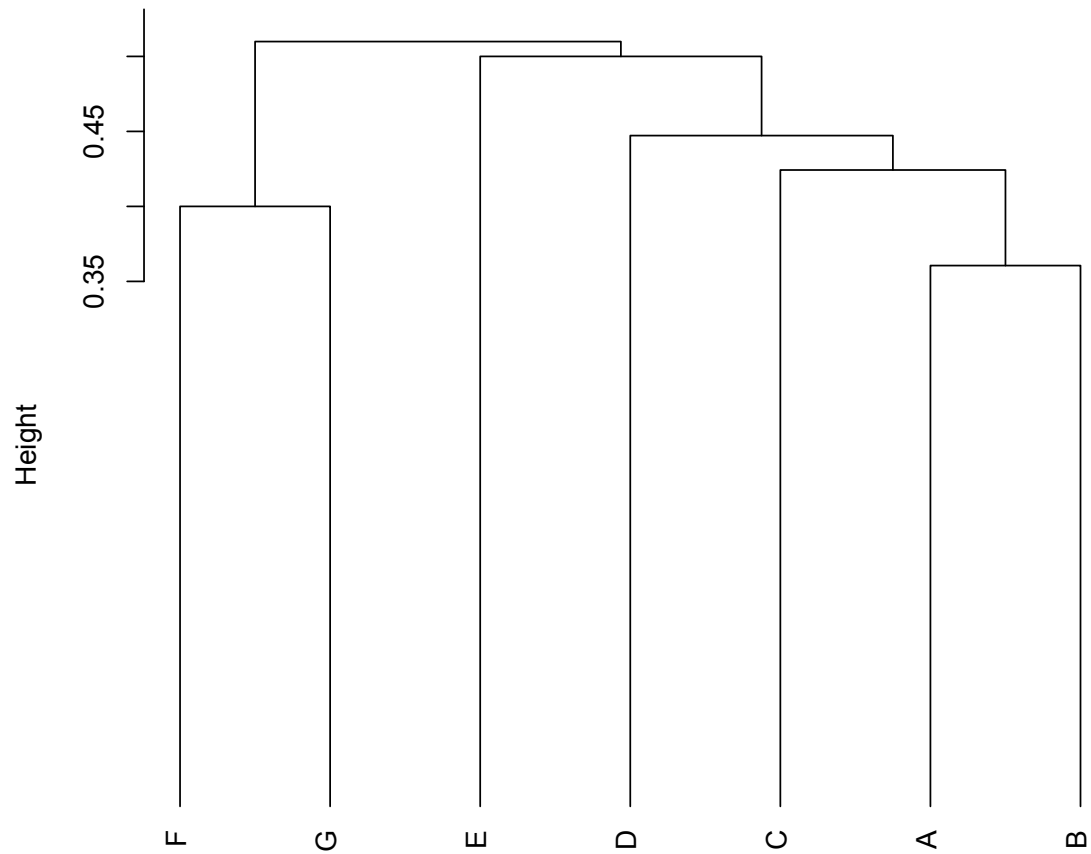
```
> especies=c("A","B","C","D","E","F","G")
> plot(clustering,labels=especies)
```



distancies

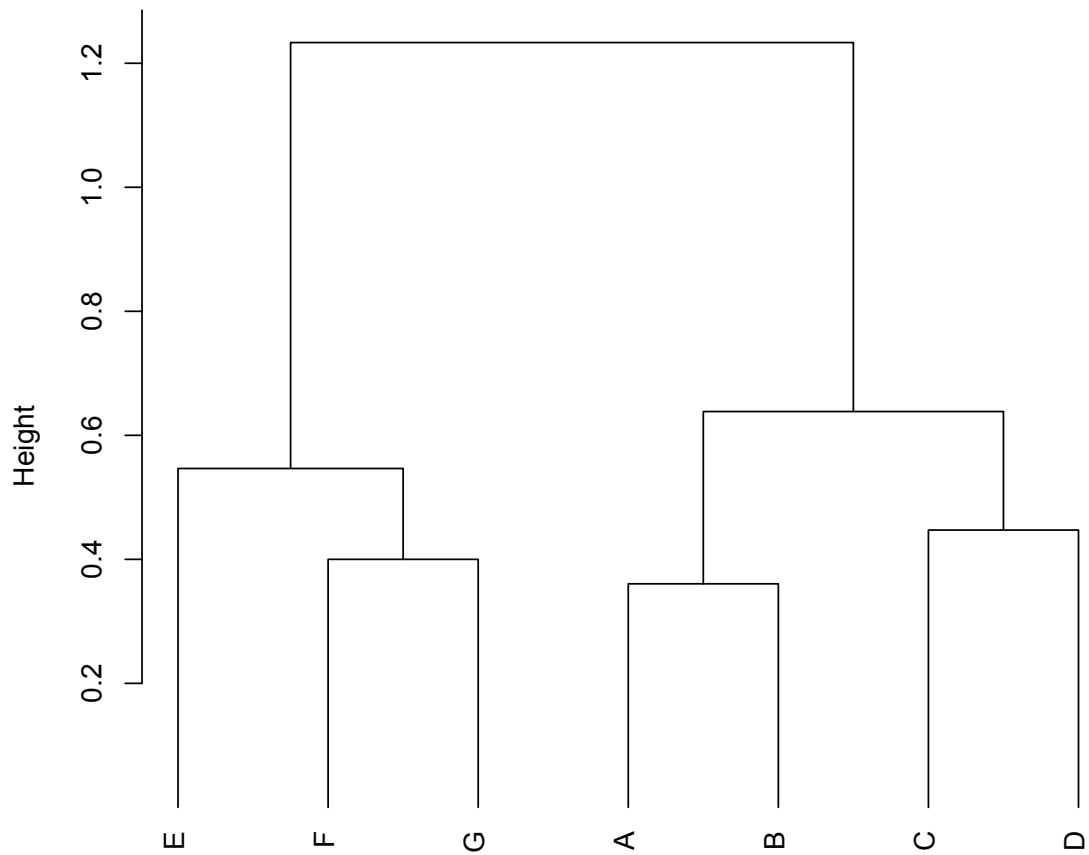
Clustering jeràrquic amb R

```
> plot(hclust(distancies,method="single"),
 labels=especies,hang=-1)
```



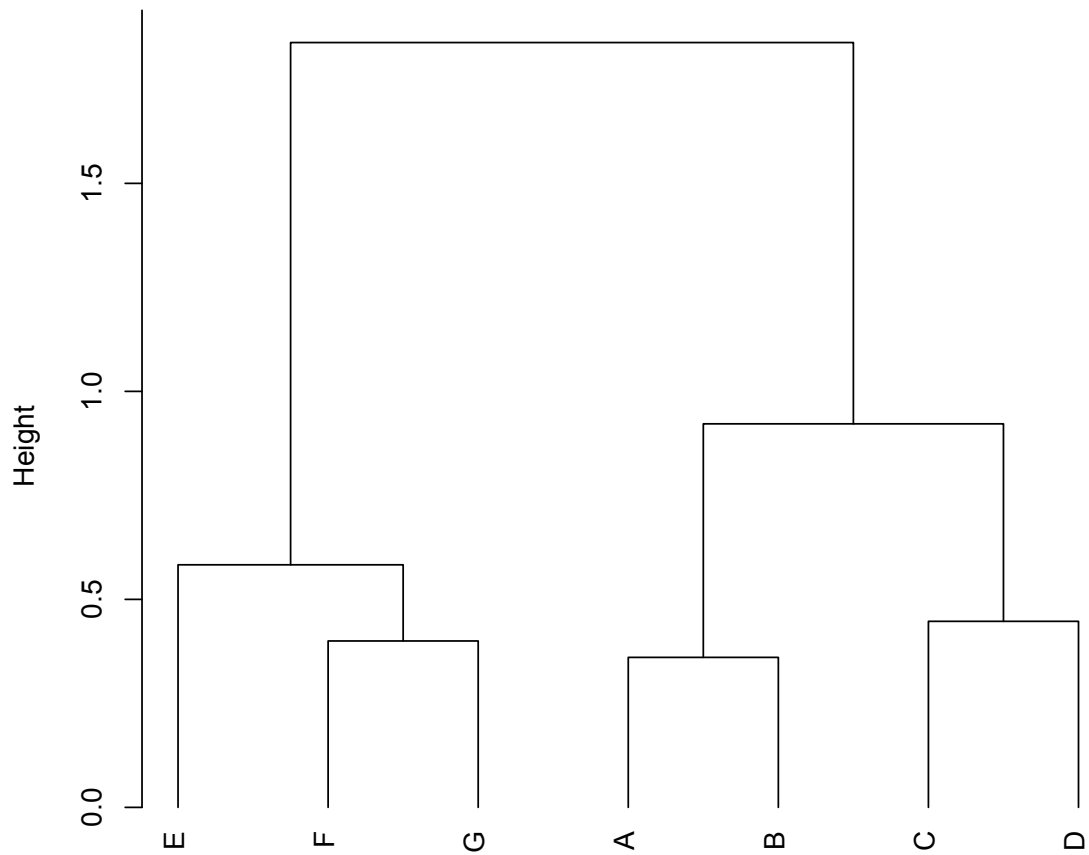
Clustering jeràrquic amb R

```
> plot(hclust(distancies,method="average"),
 labels=especies,hang=-1)
```



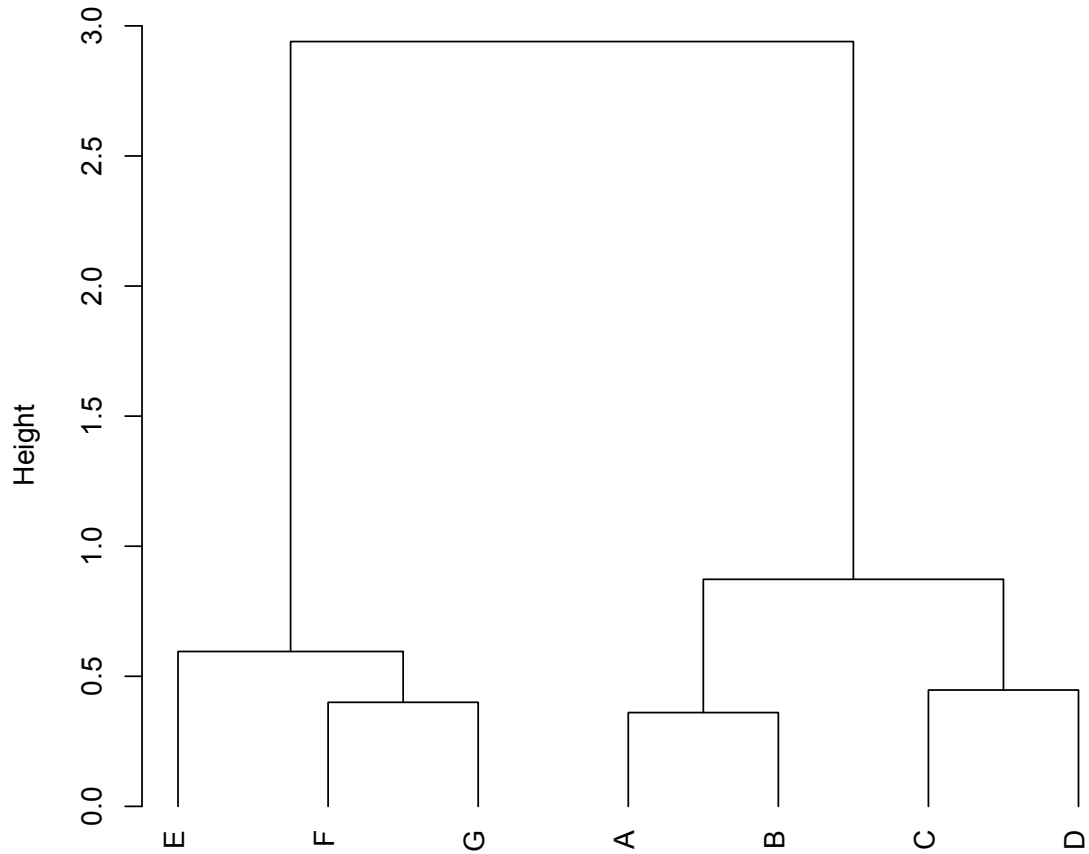
Clustering jeràrquic amb R

```
> plot(hclust(distancies,method="complete"),
 labels=especies,hang=-1)
```



Clustering jeràrquic amb R

```
> plot(hclust(distancies,method="ward"),
 labels=especies,hang=-1)
```



Algunes propietats dels mètodes

- El mètode d'enllaç simple, on

$$d(C, C_1 + C_2) = \min\big(d(C, C_1), d(C, C_2)\big),$$

tendeix a construir clusters grans: clusters que haurien de ser diferents però que tenen dos individus propers s'uneixen en un únic cluster.

- El mètode d'enllaç complet, on

$$d(C, C_1 + C_2) = \max\big(d(C, C_1), d(C, C_2)\big),$$

se'n va a l'altre extrem, i tendeix a agrupar clusters només quan tots els punts estan propers.

Algunes propietats dels mètodes

- El mètode d'enllaç mitjà, on

$$d(C, C_1 + C_2) = \frac{n_{C_1}}{n_{C_1} + n_{C_2}} d(C, C_1) + \frac{n_{C_2}}{n_{C_1} + n_{C_2}} d(C, C_2)$$

és una solució intermèdia.

És molt emprat en la reconstrucció d'arbres filogenètics a partir de matrius de distàncies (mètode UPGMA, *Unweighted Pair Group Method Using Arithmetic averages*)

Algunes propietats dels mètodes

El mètode de Ward és molt diferent.

Es defineix l'heterogeneïtat d'un cluster C com

$$I_C = \frac{1}{n_C} \sum_{x_i \in C} d^2(x_i, \mathbf{c}_C),$$

on $\mathbf{c}_C$ representa el punt mitjà del cluster C respecte de la distància emprada

Si d és la distància euclidiana, I_C és la variància del cluster C

Algunes propietats dels mètodes

Quan dos clusters s'uneixen,

$$I_{C_1+C_2} = I_{C_1} + I_{C_2} + \frac{n_{C_1} \cdot n_{C_2}}{n_{C_1} + n_{C_2}} d^2(C_1, C_2)$$

El mètode de Ward uneix els clusters de manera que l'augment de la suma de les heterogeneïtats sigui mínim.

El resultat és que els grups són (globalment) el més homogenis possible.