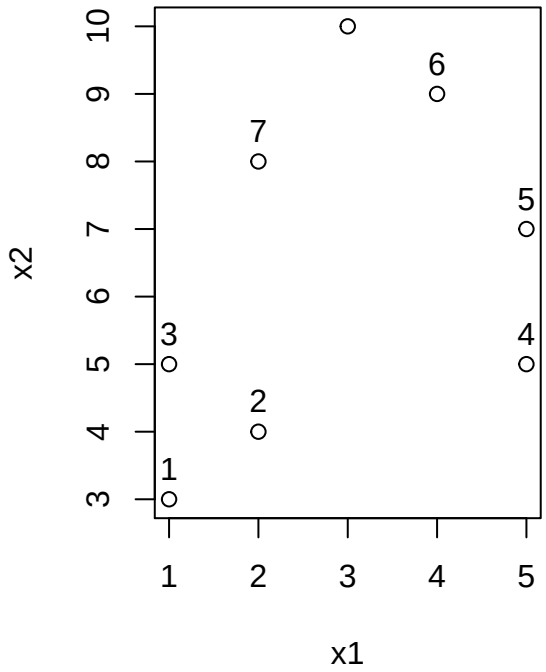
Cluster Analysis: metodi gerarchici

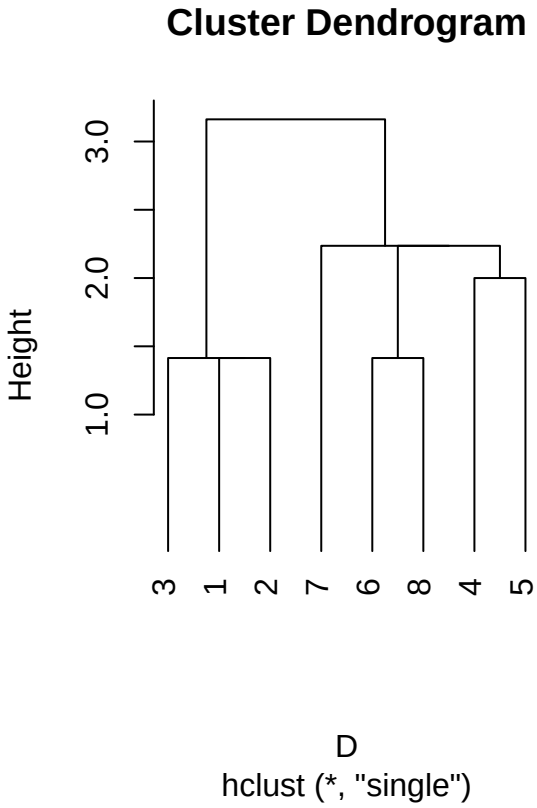
Esempio

```
X = matrix(c(1,3,2,4,1,5,5,5,5,7,4,9,2,8,3,10), ncol=2, nrow=8, byrow=T)
n = nrow(X)
colnames(X) = c("x1","x2")
rownames(X) = 1:n

( D = dist(X,method="euclidean") )
```

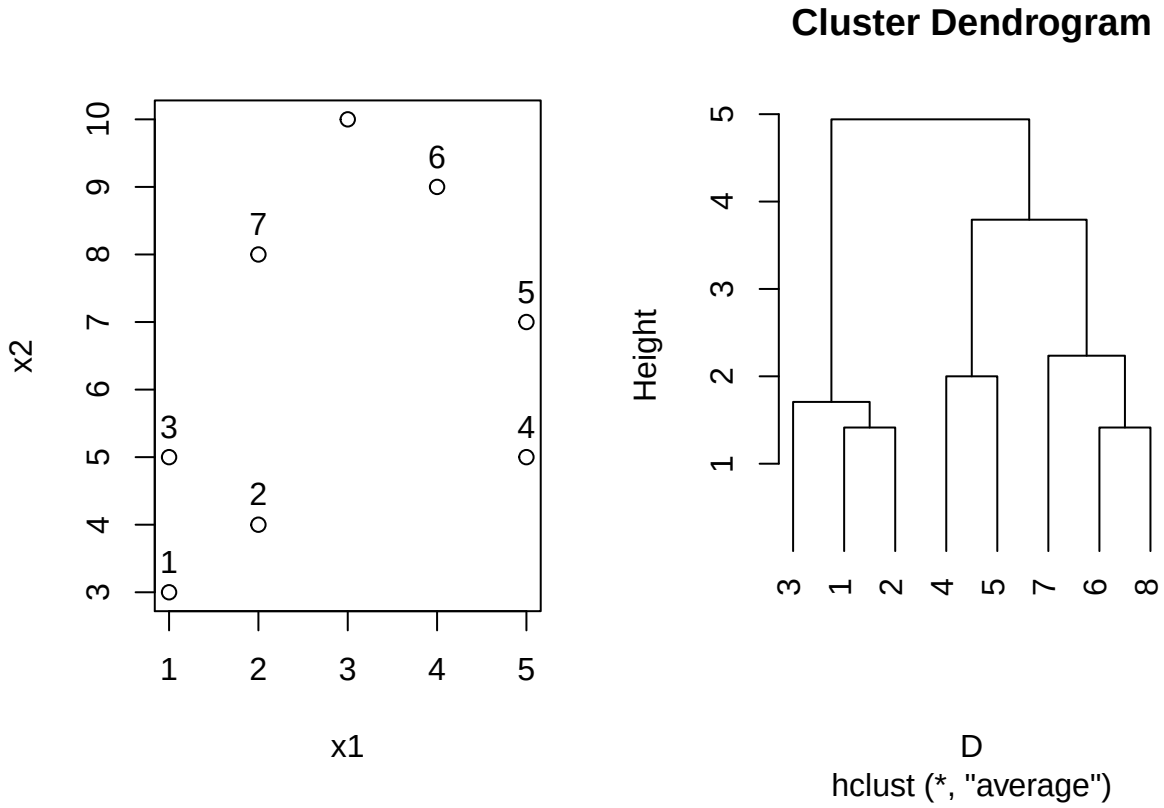
```
##          1        2        3        4        5        6        7
## 2 1.414214
## 3 2.000000 1.414214
## 4 4.472136 3.162278 4.000000
## 5 5.656854 4.242641 4.472136 2.000000
## 6 6.708204 5.385165 5.000000 4.123106 2.236068
## 7 5.099020 4.000000 3.162278 4.242641 3.162278 2.236068
## 8 7.280110 6.082763 5.385165 5.385165 3.605551 1.414214 2.236068
```

```
hc.single=hclust(D, method="single")
hc.complete=hclust(D, method="complete")
hc.average=hclust(D, method="average")


op <- par(mfrow = c(1, 2))
plot(X)
text(x2~x1, X, labels=rownames(X), pos=3)
plot(hc.single, hang=-1)
```

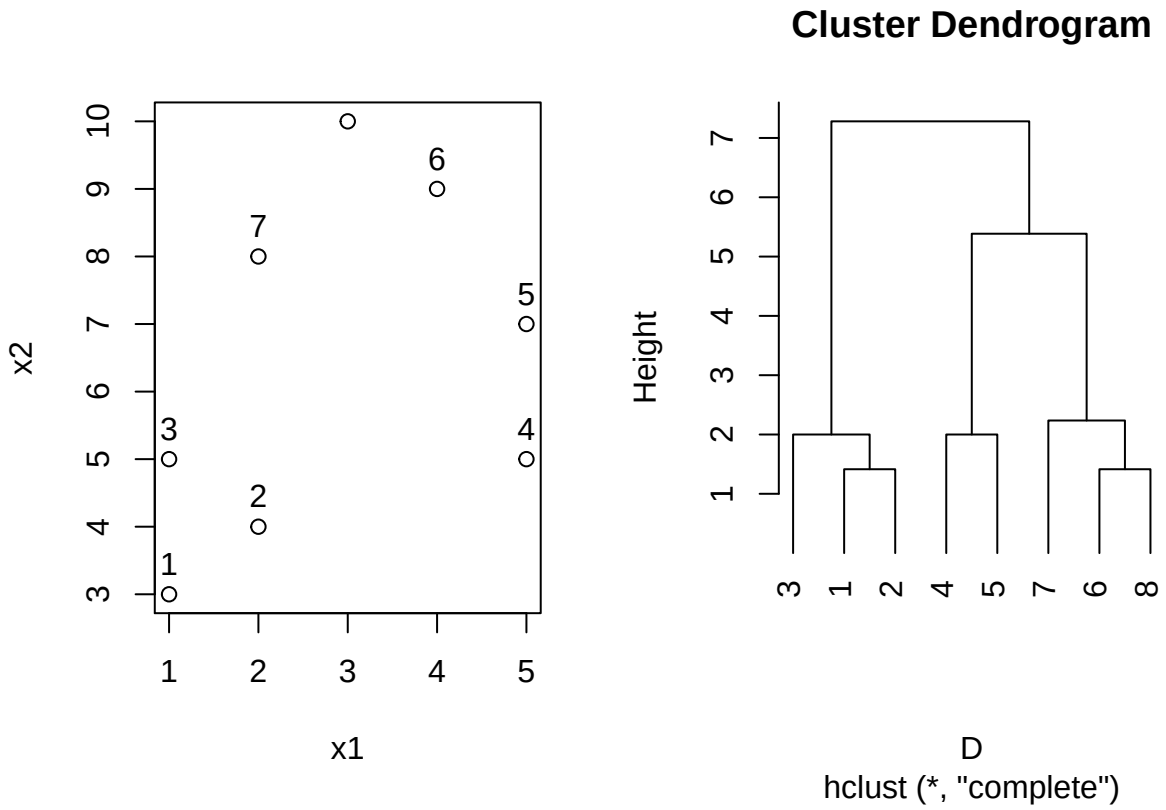





```
par(op)

op <- par(mfrow = c(1, 2))
plot(X)
text(x2~x1, X, labels=rownames(X), pos=3)
plot(hc.average, hang=-1)
```




```
par(op)

op <- par(mfrow = c(1, 2))
plot(X)
text(x2~x1, X, labels=rownames(X), pos=3)
plot(hc.complete, hang=-1)
```

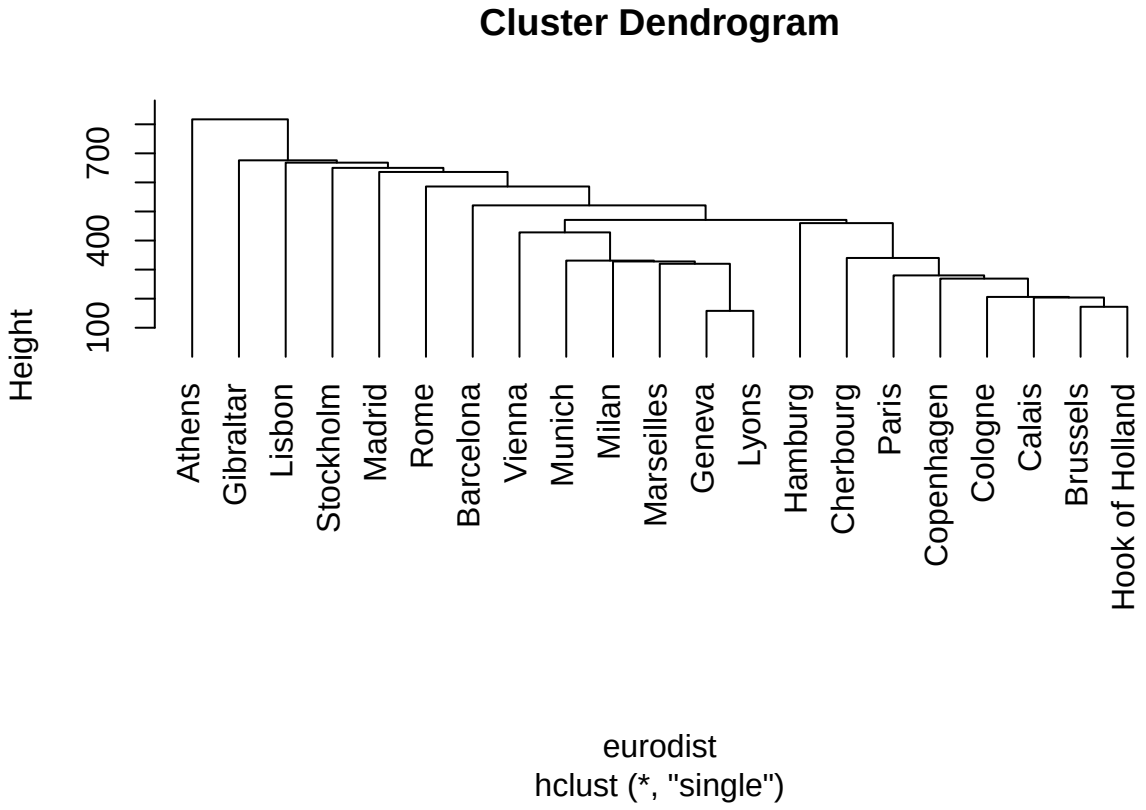



```
par(op)
```

Data set Eurodist

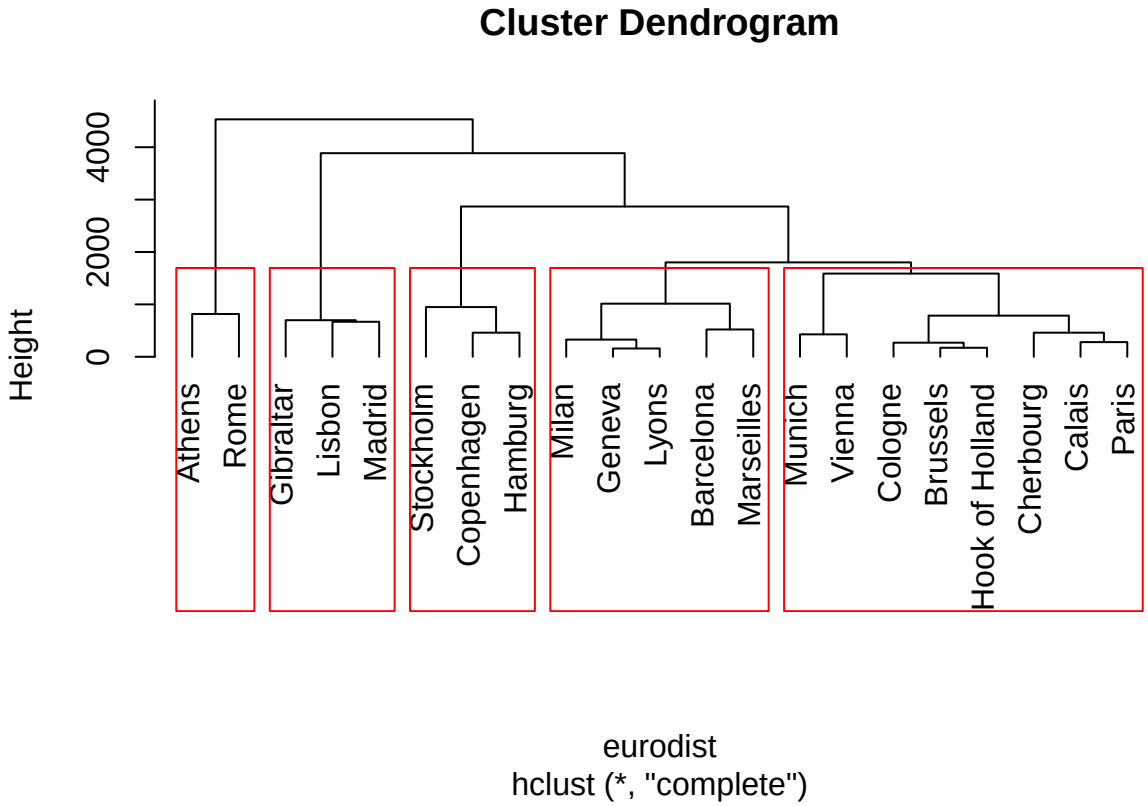
Il più ovvio esempio di distanza è quella geografica. Consideriamo allora la matrice di distanza `eurodist`, che riporta le distanze (in km) di 21 città europee. Si costruisca il dendogramma utilizzando il metodo del legame singolo, del legame completo e del legame medio, commentando i risultati.

```
data(eurodist)

hc.single=hclust(eurodist, method="single")
plot(hc.single, hang=-1)
```



Il metodo del legame singolo produce una gerarchia da cui non è possibile individuare dei gruppi (le distanze tra i bracci orizzontali del dendrogramma sono più o meno uniformi, non ci sono salti).

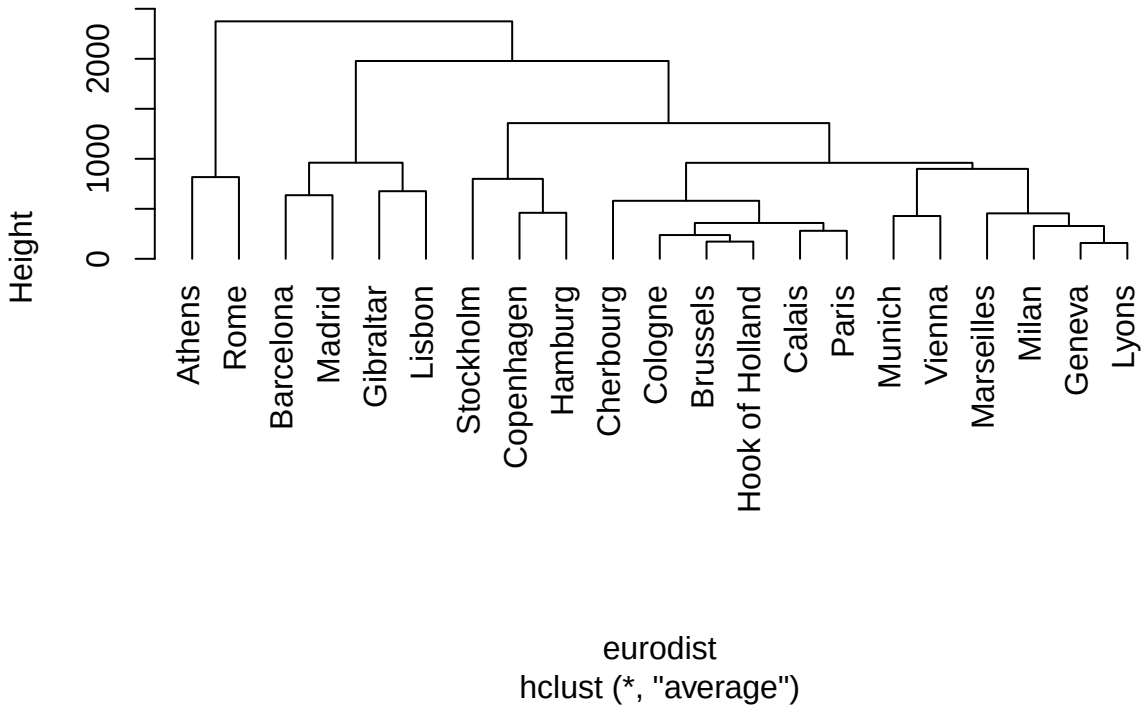
```
hc.complete=hclust(eurodist, method="complete")
plot(hc.complete, hang=-1)
rect.hclust(hc.complete,k=5)
```



Col legame completo, e considerando un numero di gruppi pari a 5, individuiamo geograficamente in Europa

- il Sud (Roma, Atene)
- la penisola iberica (Barcellona, Madrid, Gibilterra e Lisbona)
- la regione Nord-Est (Amburgo, Copenaghen e Stoccolma)
- il centro-Sud (Ginevra, Lione, Marsiglia e Milano)
- il centro-Est (Monaco e Vienna), assieme al centro-Nord continentale (Bruxelles, Hook of Holland, Colonia) e il Nord della Francia (Calais, Parigi, Cherbourg)

```
hc.average=hclust(eurodist, method="average")
plot(hc.average, hang=-1)
```



Col legame medio vengono individuati gli stessi gruppi più estremi (Sud, penisola Iberica, Nord-Ovest) mentre le suddivisioni nella zona centrale sono leggermente diverse.

Data set Flower

1. Caricare il dataset `flower` presente nella libreria `cluster`. Costruire la matrice di dissimilarità con l'indice di Gower, specificando che le variabili `V1` e `V2` sono binarie simmetriche, mentre la variabile `V3` è binaria assimetrica.

```
require(cluster)
```

```
## Loading required package: cluster
```

```
data(flower)
str(flower)
```

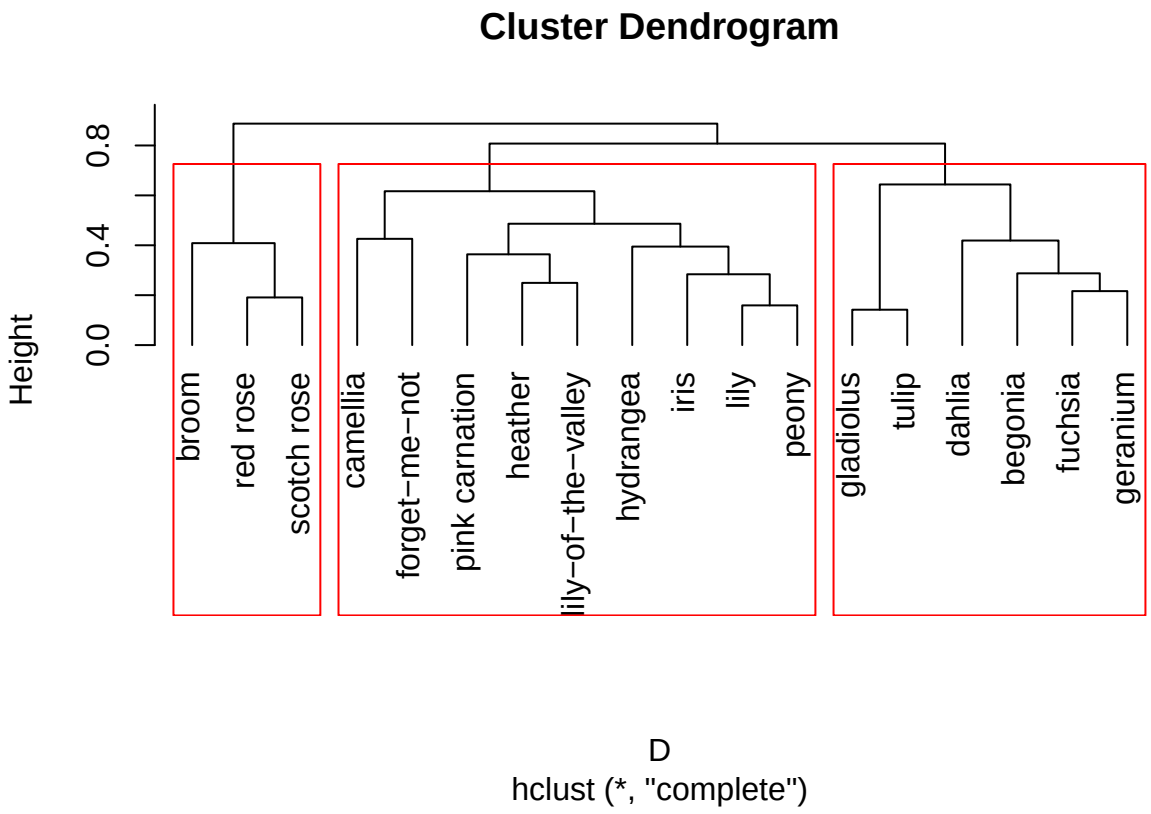
```
## 'data.frame':    18 obs. of  8 variables:
##  $ V1: Factor w/ 2 levels "0","1": 1 2 1 1 1 1 1 2 2 ...
##  $ V2: Factor w/ 2 levels "0","1": 2 1 2 1 2 2 1 1 2 2 ...
##  $ V3: Factor w/ 2 levels "0","1": 2 1 1 2 1 1 1 2 1 1 ...
##  $ V4: Factor w/ 5 levels "1","2","3","4",..: 4 2 3 4 5 4 4 2 3 5 ...
##  $ V5: Ord.factor w/ 3 levels "1"<"2"<"3": 3 1 3 2 2 3 3 2 1 2 ...
##  $ V6: Ord.factor w/ 18 levels "1"<"2"<"3"<"4"<..: 15 3 1 16 2 12 13 7 4 14 ...
##  $ V7: num  25 150 150 125 20 50 40 100 25 100 ...
##  $ V8: num  15 50 50 50 15 40 20 15 15 60 ...
```

```
row.names(flower) = c("begonia","broom","camellia","dahlia","forget-me-not","fuchsia",
                      "geranium","gladiolus","heather","hydrangea","iris","lily",
                      "lily-of-the-valley","peony","pink carnation","red rose","scotch rose", "tulip")
```

```
D = daisy(flower, metric="gower", type=list(symm=c(1,2),asymm=3))
```

2. Costruire il dendogramma utilizzando il metodo del legame completo ed evidenziare la partizione con 3 gruppi.

```
hc.complete = hclust(D, method="complete")
plot(hc.complete, hang=-1)

# numero di gruppi
K = 3
rect.hclust(hc.complete,k=K)
```

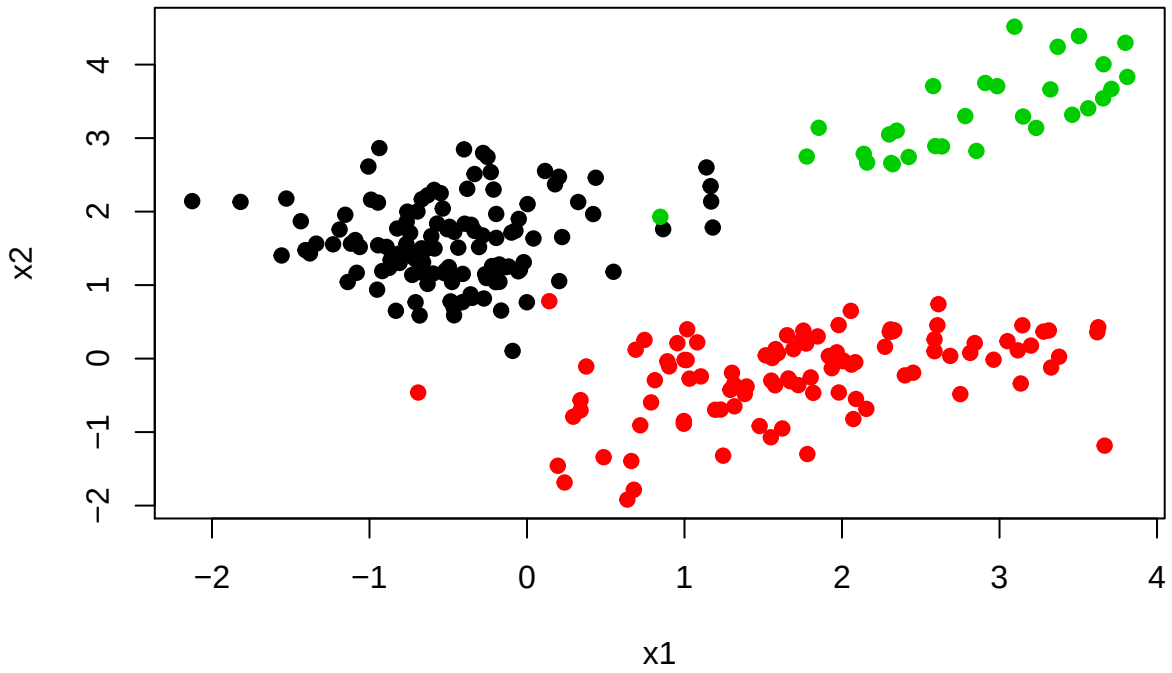


Dataset simulati C1 e C2

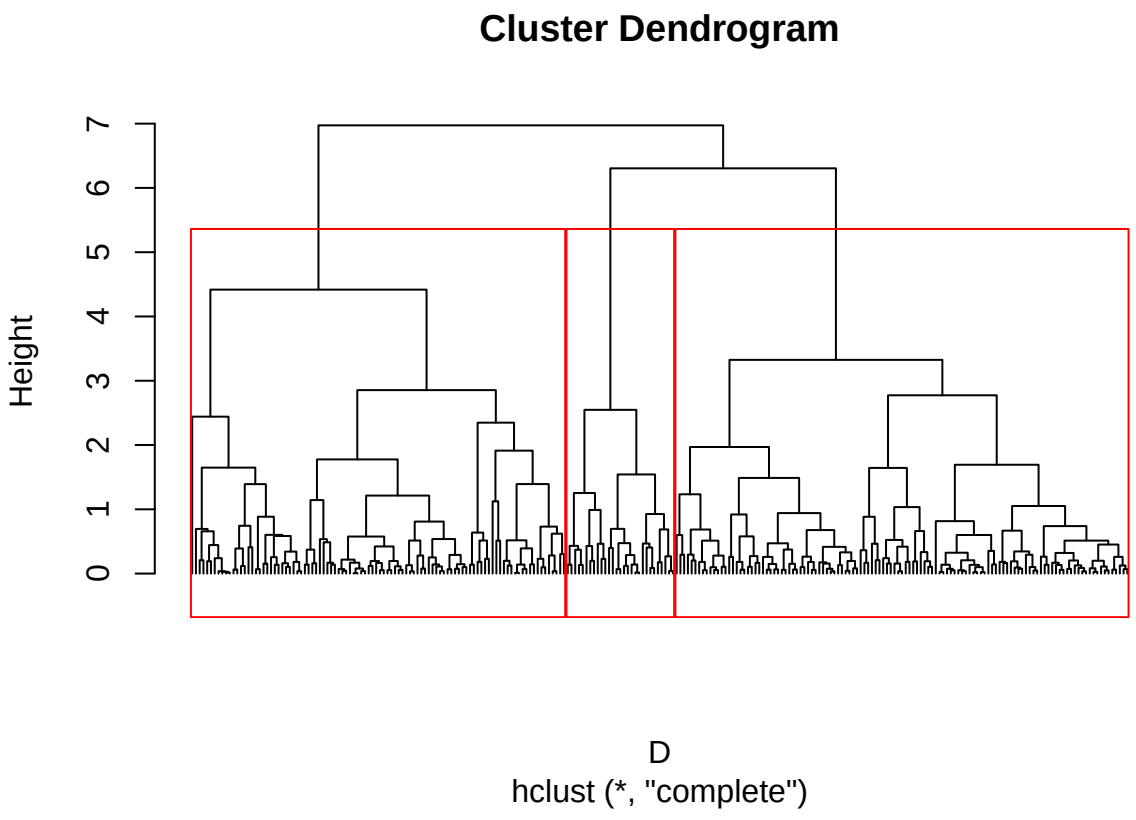
Importare il data set `C1.dat` presente sul sito http://azzalini.stat.unipd.it/Libro-DM/dati.html e costruire la matrice di distanza Euclidea. Questo dataset contiene le vere assegnazioni delle unità a 3 gruppi.

Effettuare l'analisi dei cluster gerarchica utilizzando il legame singolo, completo e medio, ricavandone 3 gruppi e commentare i risultati.

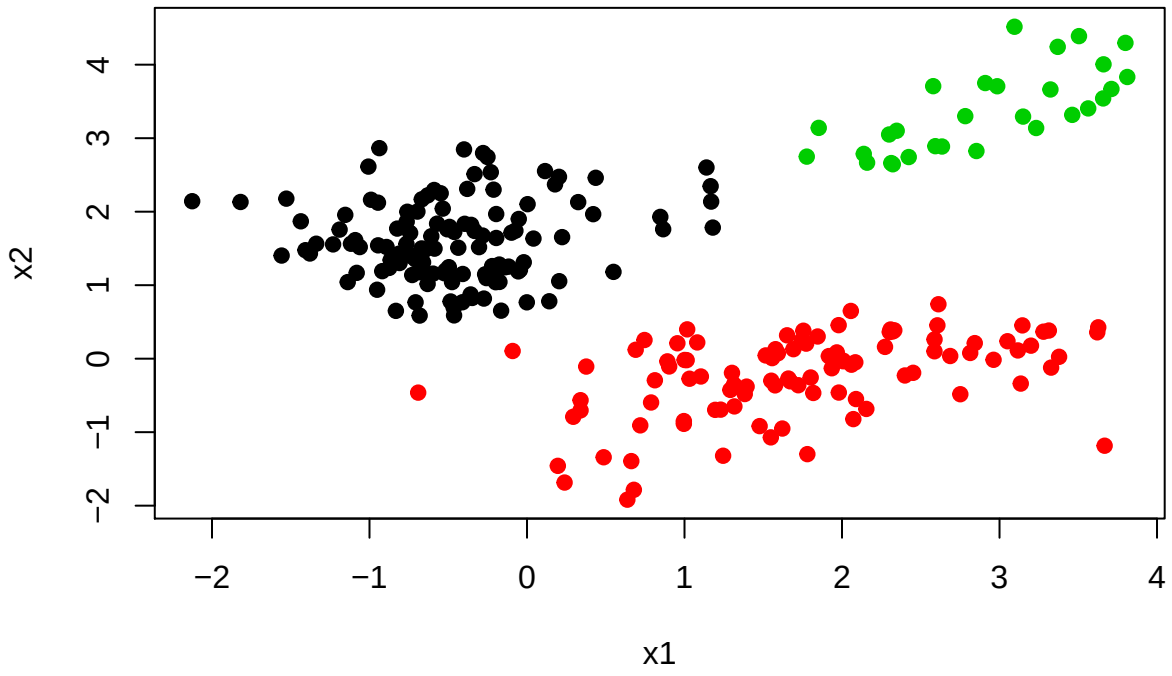
```
C1 <- read.table("http://azzalini.stat.unipd.it/Libro-DM/C1.dat", sep="", header=TRUE)

# vere assegnazioni delle unità ai 3 gruppi
plot(x2 ~ x1, col=gruppo, C1, pch=19)
```




```r
# matrice di distanza Euclidea
D = dist(C1[,c("x1","x2")], method = "euclidean")

# analisi dei cluster gerarchica
hc = hclust(D, method="complete")

# dendogramma
plot(hc, labels=FALSE, hang=-1)
rect.hclust(hc,k=3)
```




```
# i 3 gruppi risultanti
g3 = cutree(hc, k=3)
plot(x2 ~ x1, col=g3, C1, pch=19)
```

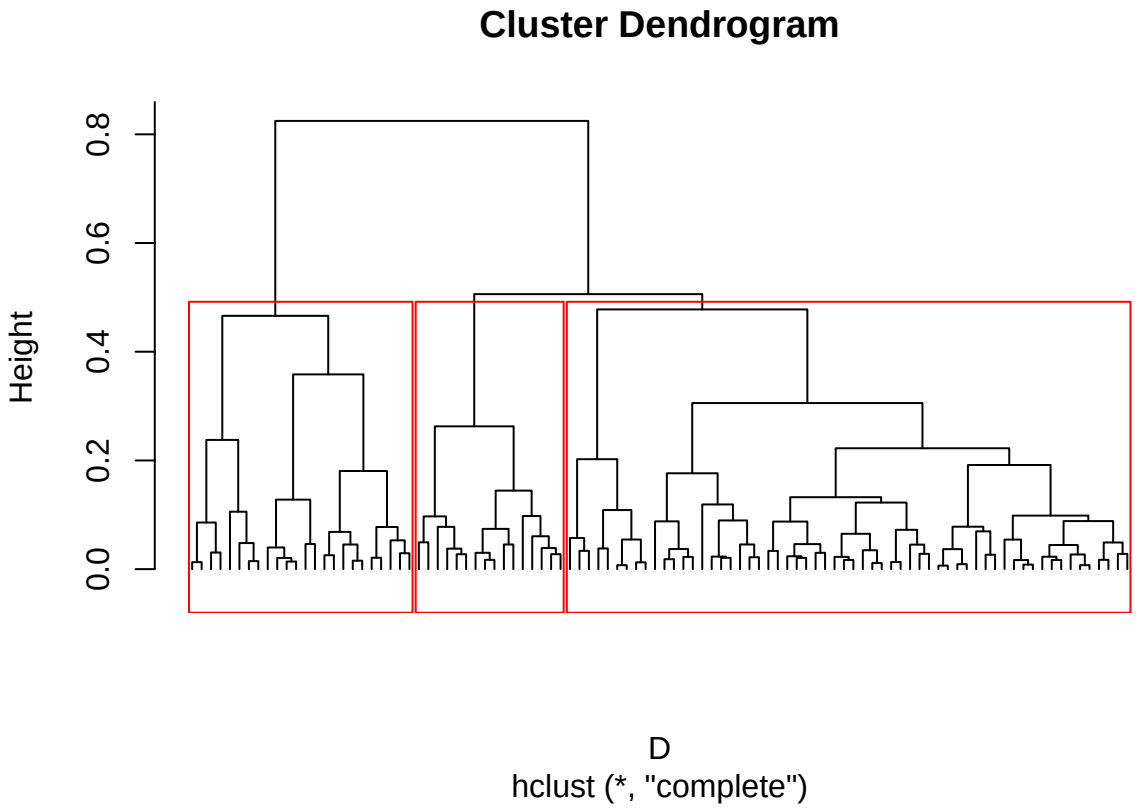



```
# confronto
table(g3, C1$gruppo)
```

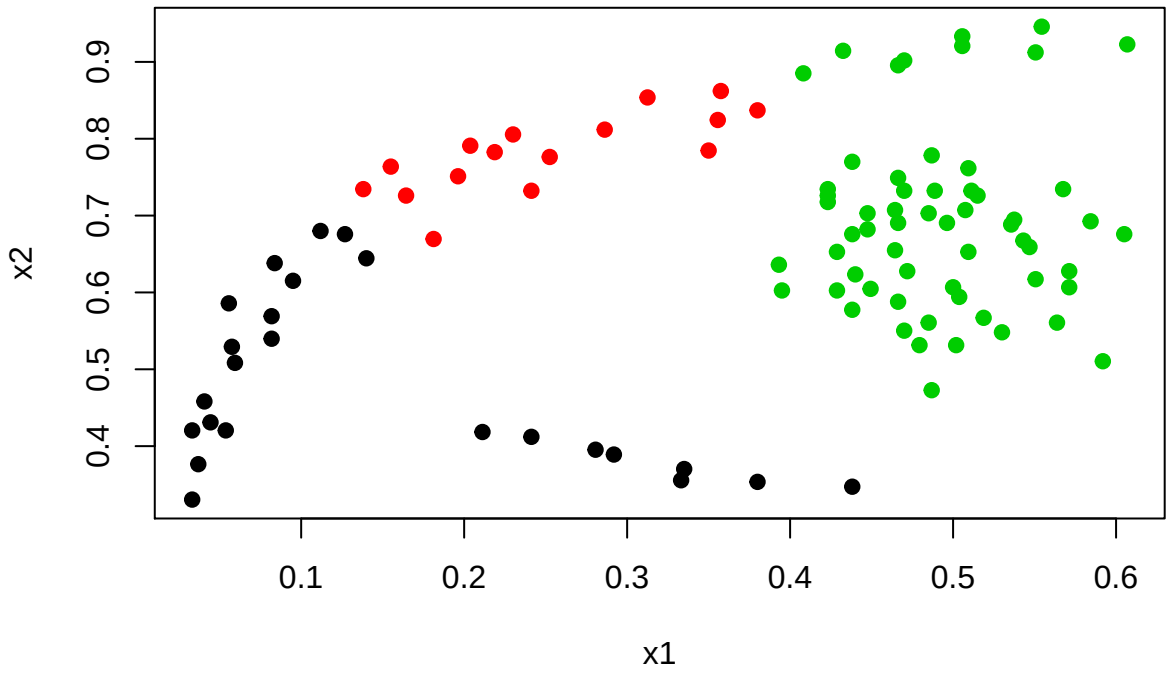
```
##
## g3    1   2   3
##   1 119   1   1
##   2   1  99   0
##   3   0   0  29
```

Importare il data set `C2.dat` e costruire la matrice di distanza Euclidea. Effettuare l'analisi dei cluster utilizzando il legame singolo, completo e medio, ricavandone 3 gruppi e commentare i risultati.

```
rm(list=ls())
C2 <- read.table("http://azzalini.stat.unipd.it/Libro-DM/C2.dat", sep="", header = TRUE)

D = dist(C2[,c("x1","x2")], method = "euclidean")
hc = hclust(D, method="complete")

plot(hc, labels=FALSE, hang=-1)
rect.hclust(hc,k=3)
```




```
g3 = cutree(hc, k=3)
plot(x2 ~ x1, col=g3, C2, pch=19)
```



Dati Iris

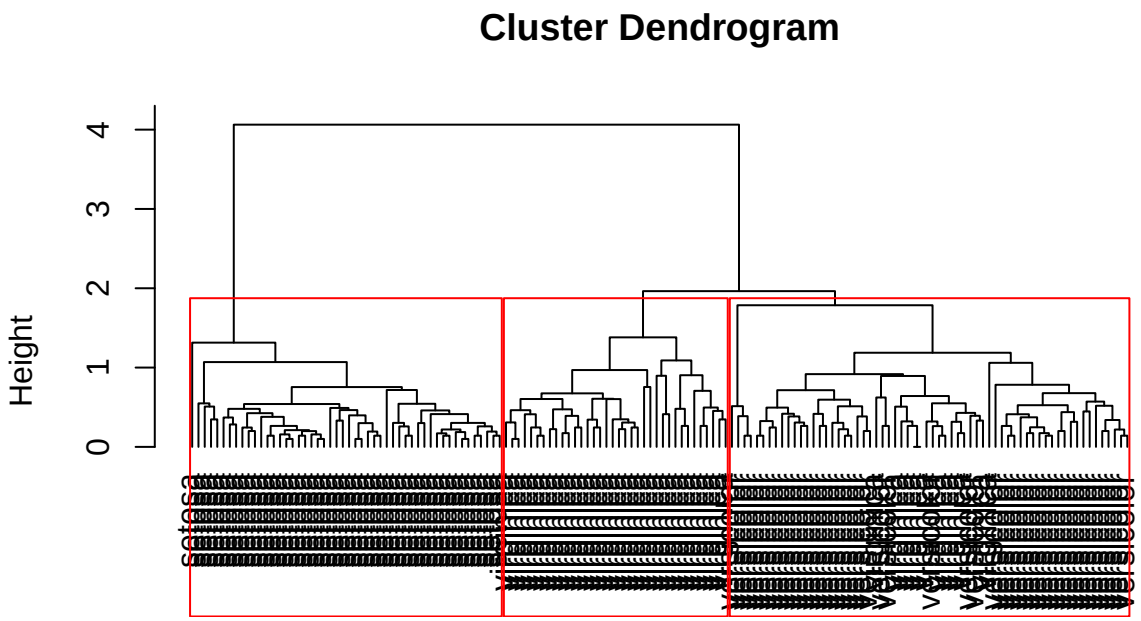
1. Caricare il dataset `iris` presente nella libreria `datasets`. Costruire la matrice di distanze Euclidee sulla base delle prime 4 variabili `Sepal.Length`, `Sepal.Width`, `Petal.Length`, `Petal.Width` e utilizzare l'algoritmo agglomerativo gerarchico con il metodo del legame medio. Costruire il dendogramma e determinare 3 gruppi. Confrontare i gruppi ottenuti con la variabile `Species` e commentare il risultato.

```
rm(list=ls())

# carico i dati
data(iris)
# guardo i dati
head(iris)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa
```

```
# considero le prime 4 variabili
X = iris[,1:4]
n = nrow(X)

# matrice di distanze Euclidee
D = dist(X, method="euclidean")

# metodo agglomerativo gerarchico, legame medio
hc = hclust(D, method="average")
plot(hc, hang= -1, label=iris$Species)
```

```
rect.hclust(hc, k=3)
```



D
hclust (*, "average")

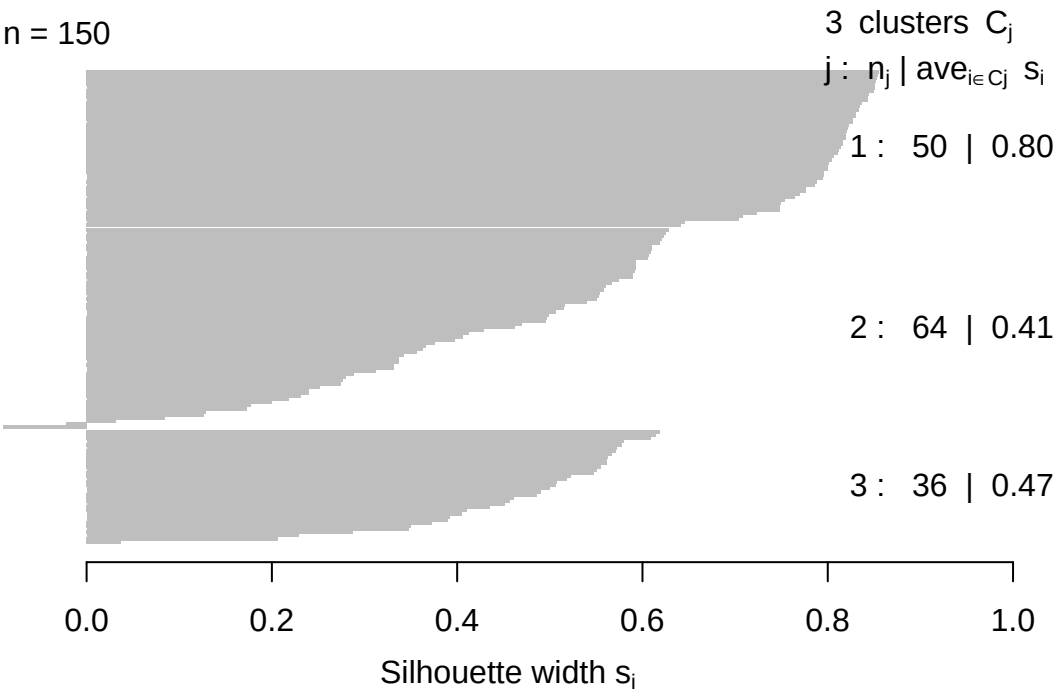
```
# determino 3 gruppi
hc3 = cutree(hc, k=3)

# confronto i gruppi con Species
table(hc3, iris$Species)
```

```
##
## hc3 setosa versicolor virginica
##   1     50          0         0
##   2      0         50        14
##   3      0          0        36
```

Il gruppo 1 contiene solo fiori `setosa`, il gruppo 3 solo fiori `virginica` e il gruppo 2 in maggioranza (50) `versicolor` ma anche 14 `setosa`.

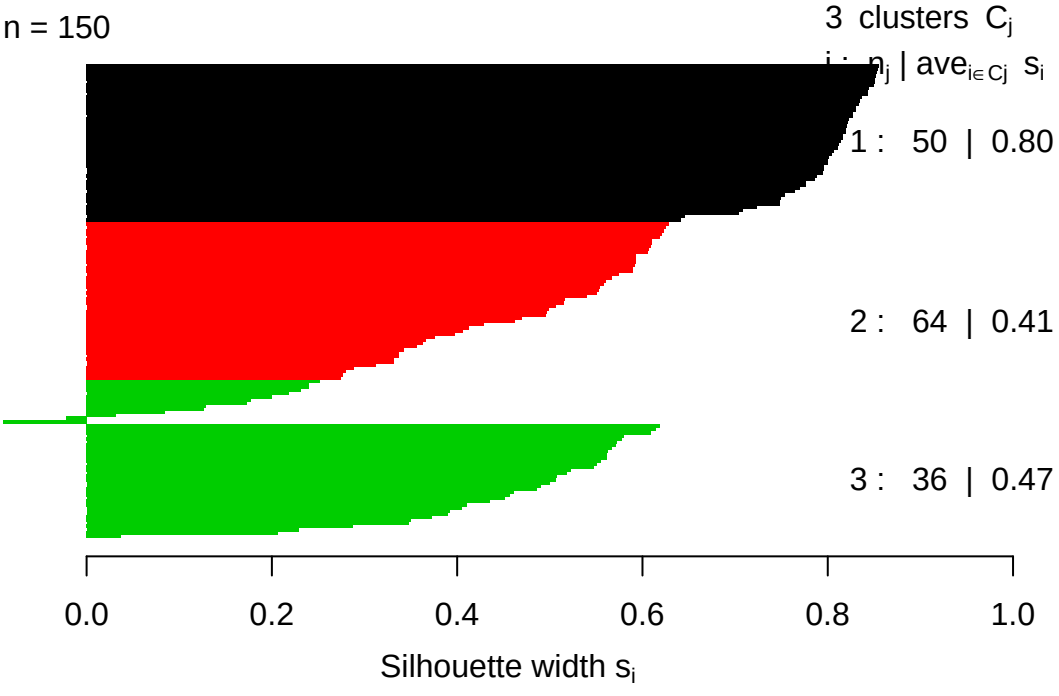
2. Costruire il grafico *silhouette* per i tre gruppi individuati con il comando `silhouette` presente nella libreria `cluster()` e commentare.

```
require(cluster)
sil <- silhouette(hc3, dist=D)
plot(sil)
```



Average silhouette width : 0.55

```
# coloro secondo la variabile Species
plot(sil, col=iris$Species)
```



Average silhouette width : 0.55

Commento.

3. Utilizzare l'algoritmo delle K-medie (argomento `algorithm = Lloyd`) specificando 3 gruppi, inizializzando i centroidi con le osservazioni di riga 25, 75 e 125. Eseguire l'algoritmo una sola volta. Determinare i 3 centroidi e l'indice CH. Confrontare i gruppi ottenuti con la variabile `Species` e commentare il risultato relativamente al raggruppamento ottenuto al punto 1.

```
K = 3
km <- kmeans(X, centers = X[c(25,75,125),], nstart=1, algorithm = "Lloyd")

# centroidi
km$centers
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width
## 1     5.006000    3.428000     1.462000    0.246000
## 2     5.901613    2.748387     4.393548    1.433871
## 3     6.850000    3.073684     5.742105    2.071053
```

```
# decomposizione
W = km$tot.withinss
B = km$betweenss
km$totss
```

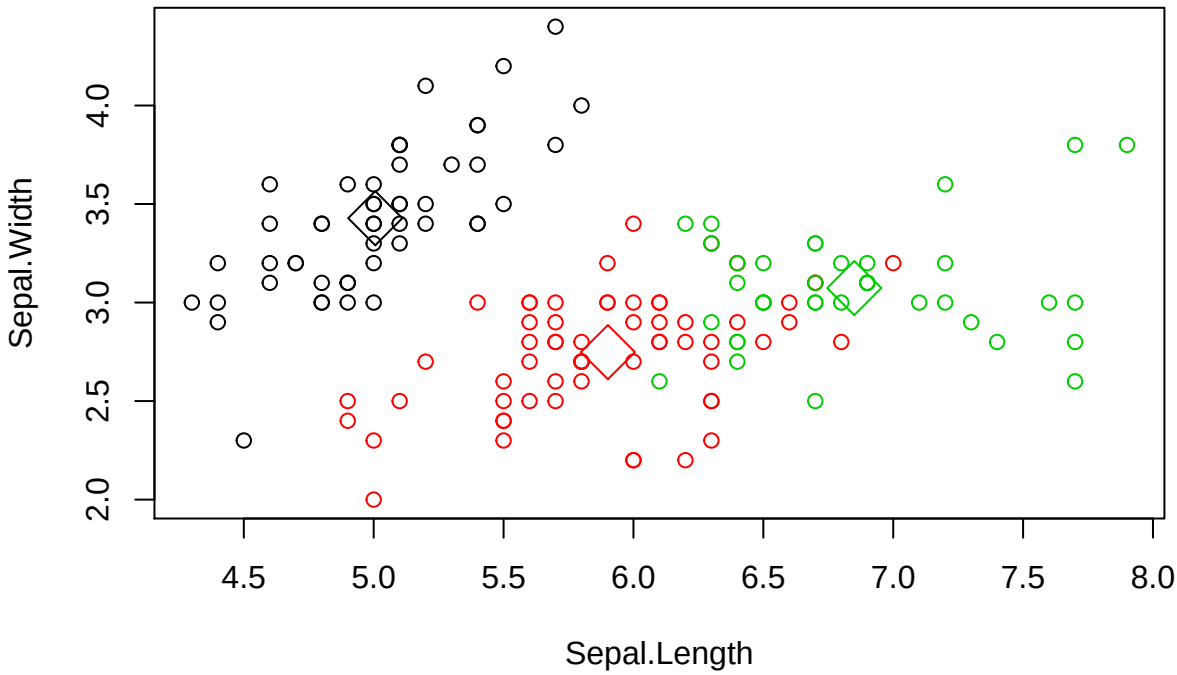
```
## [1] 681.3706
```

```
# gruppi
km3 = km$cluster

# confronto i gruppi con Species
table(km3, iris$Species)
```

```
##
## km3 setosa versicolor virginica
##   1     50          0         0
##   2      0         48        14
##   3      0          2        36
```

4. Costruire il diagramma di dispersione per le variabili `Sepal.Length` e `Sepal.Width`, colorando le unità secondo i gruppi ottenuti e aggiungendo i centrodi relativi alle 2 variabili.

```
plot(X[c("Sepal.Length", "Sepal.Width")], col=km$cluster)
points(km$centers[,c("Sepal.Length", "Sepal.Width")], col=1:3, pch=23, cex=3)
```



5. Si consideri la matrice dei dati standardizzati e su questi determinare i punteggi delle prime due componenti principali. Utilizzare l'algoritmo delle K-medie sui punteggi ottenuti, specificando 3 gruppi e inizializzando i centroidi con le osservazioni di riga 25, 75 e 125. Confrontare i gruppi ottenuti con la variabile `Species` e commentare.

```
# standardizzo i dati
S = var(X)*((n-1)/n)
Z = scale(X, center=TRUE, scale=diag(S)^(1/2))

# determino i punteggi delle prime due componenti principali
pca <- prcomp(Z, center = FALSE, scale. = FALSE)
Y <- pca$x[,1:2]

# algoritmo K medie
km.pca <- kmeans(Y, centers = Y[c(25,75,125),], nstart=1, algorithm = "Lloyd")

# confronto i gruppi con Species
table(km.pca$cluster, iris$Species)
```

```
##
##     setosa versicolor virginica
##   1     50          0         0
##   2      0         39        16
##   3      0         11        34
```