Introduzione alla chemiometria e disegno sperimentale

Modulo 5: Clustering in R software

Docente: Dr. Sabina Licen (slicen@units.it)

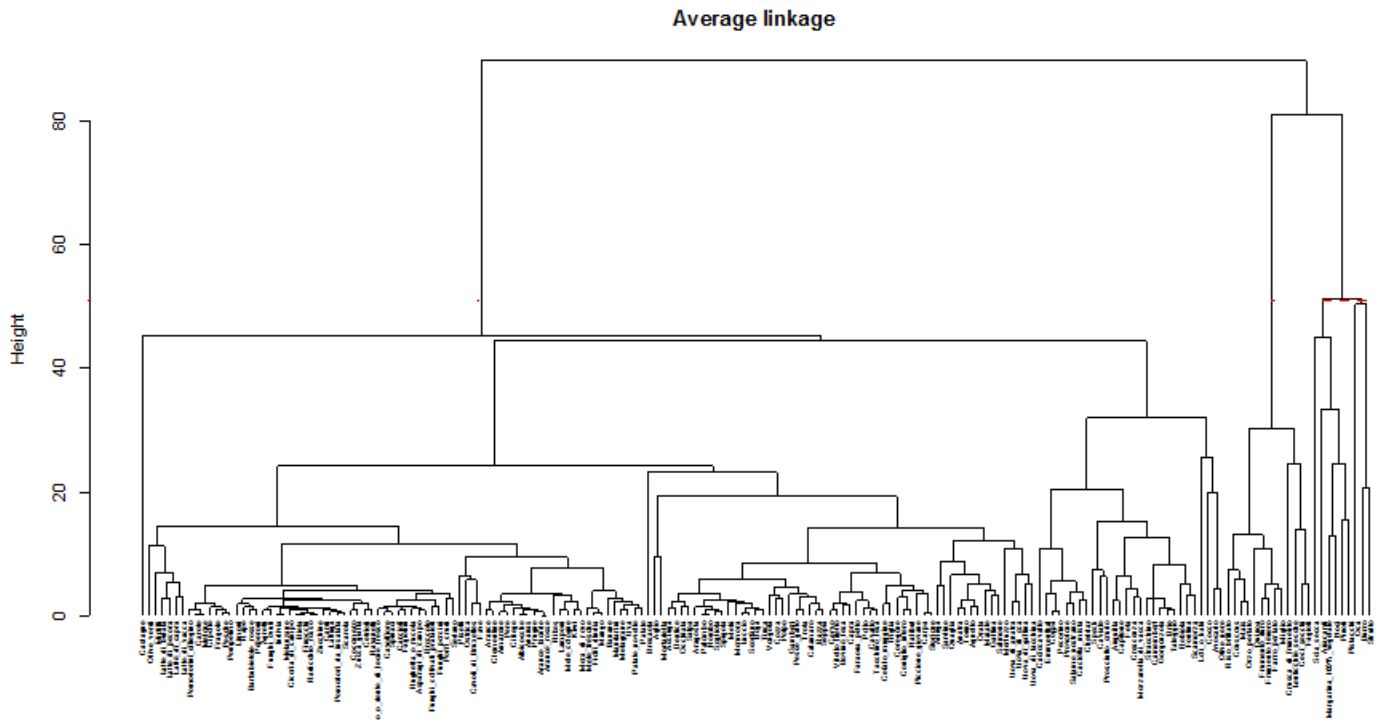
Il raggruppamento gerarchico in R

```
Dist<-dist(nomematrice, method="euclidean")
Hc<- hclust(Dist, method="complete")
Dend<-as.dendrogram(Hc)
```

→ Qua si decide il tipo di distanza

→ Qua si decide il tipo di linkage

```
plot(Dend)
```



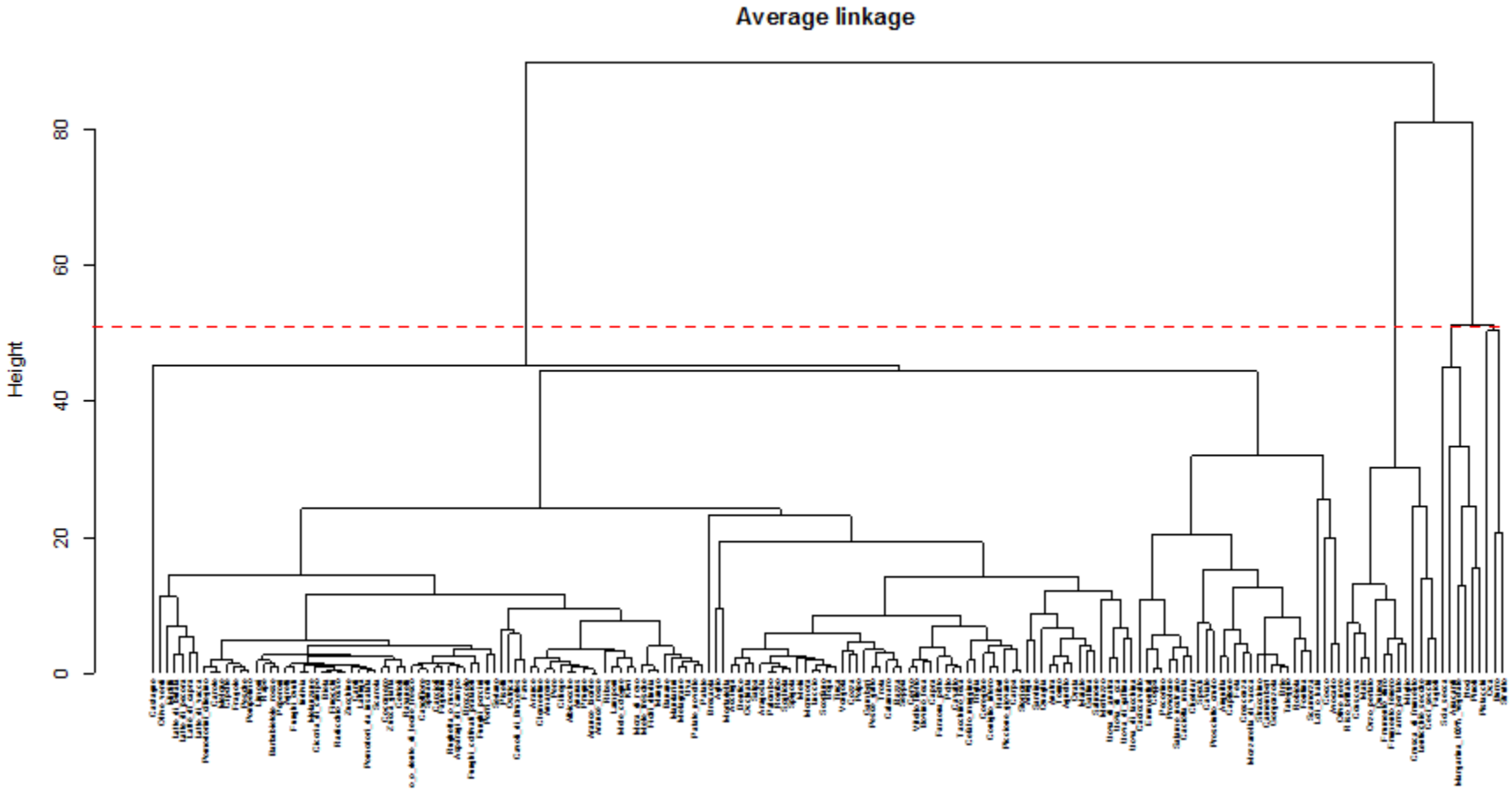
Fonte: http://www.sthda.com/english/wiki/beautiful-dendrogram-visualizations-in-r-5-must-known-methods-unsupervised-machine-learning#plot.dendrogram-function

Scegliere il numero di gruppi

```
Cut<-cutree(Hc, h= 51)
```

↓

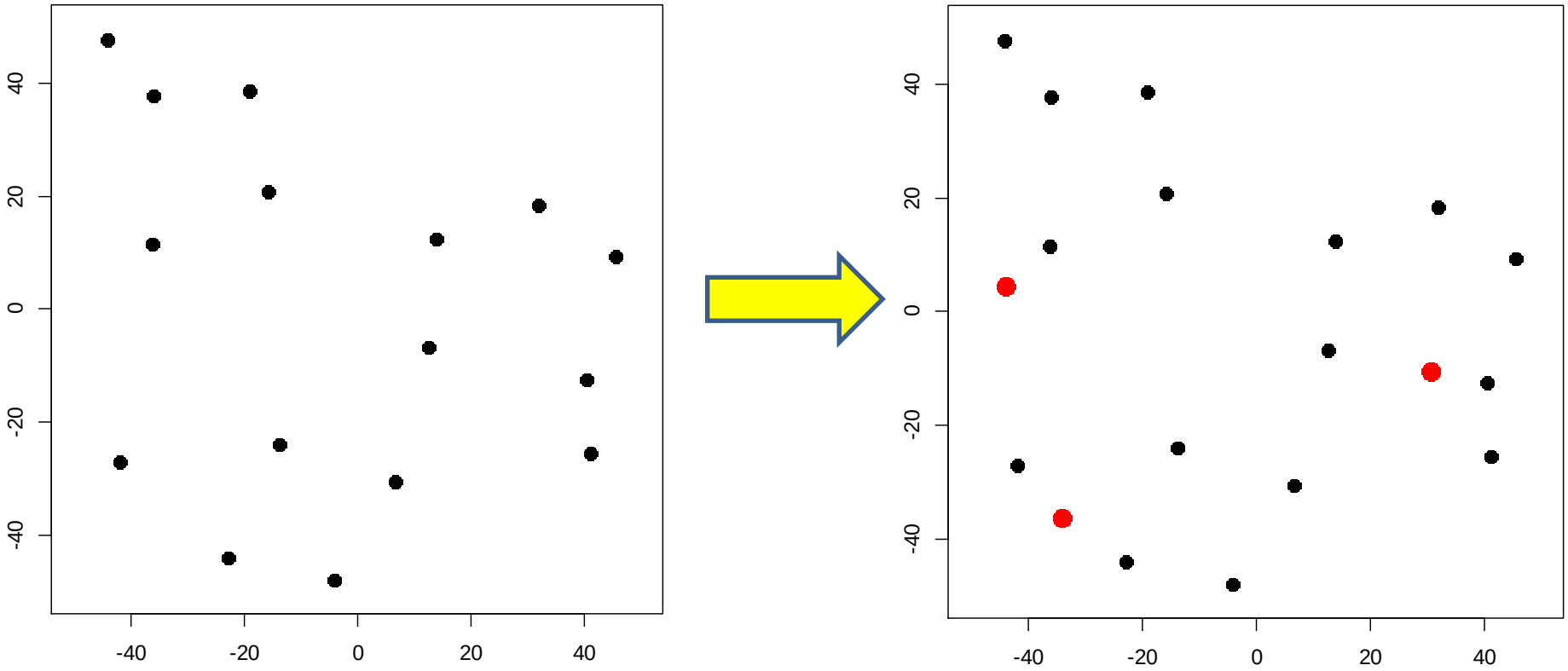
Scelgo la distanza a cui "tagliare" i rami



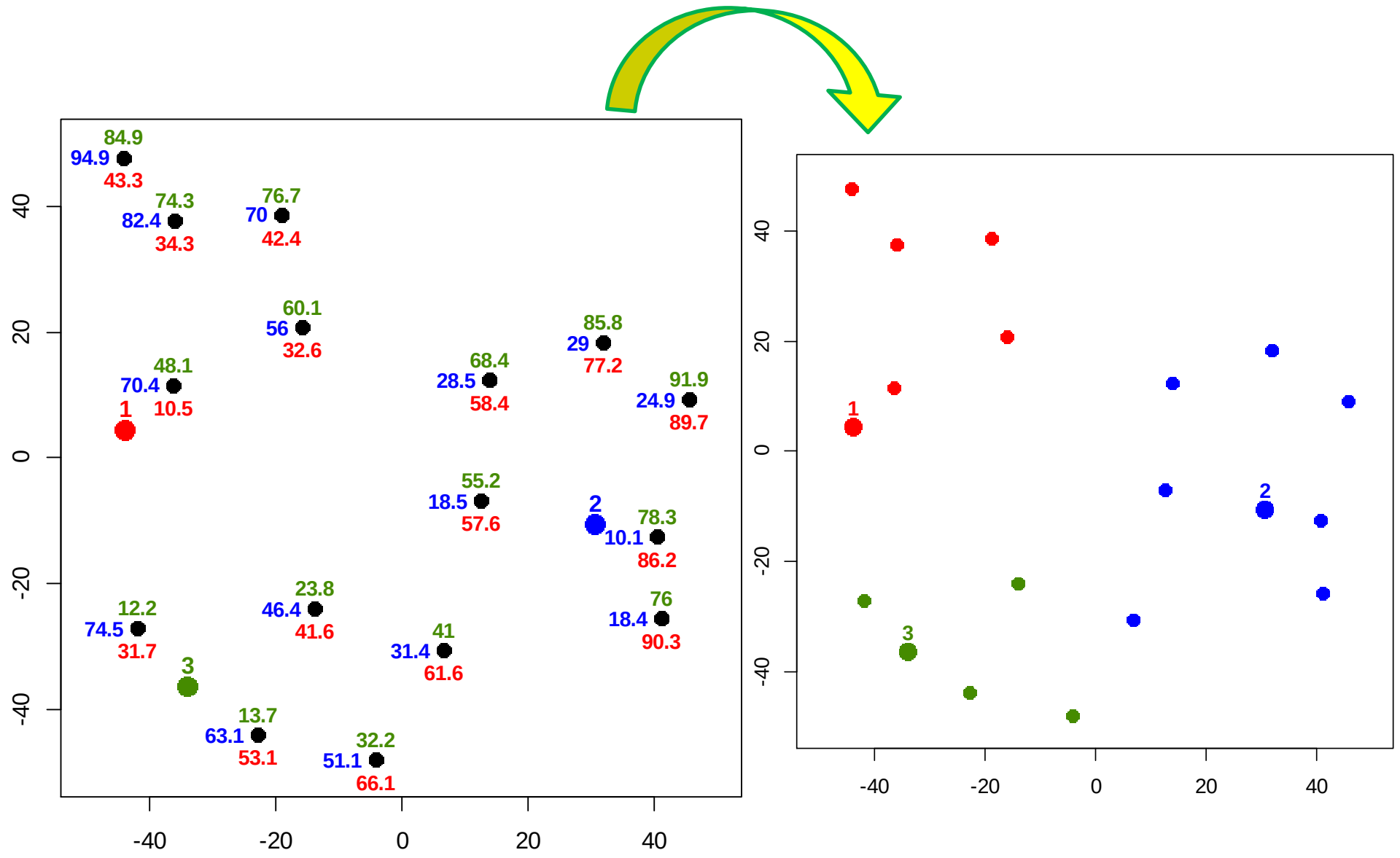
K-means clustering

L'algoritmo **k**-means clustering consente di dividere le osservazioni in un numero predefinito di cluster **k**.

Come funziona? Viene inizializzato random scegliendo **k** "punti" con numero di variabili uguale a quello delle osservazioni e calcolando la distanza tra di essi e tutte le osservazioni:

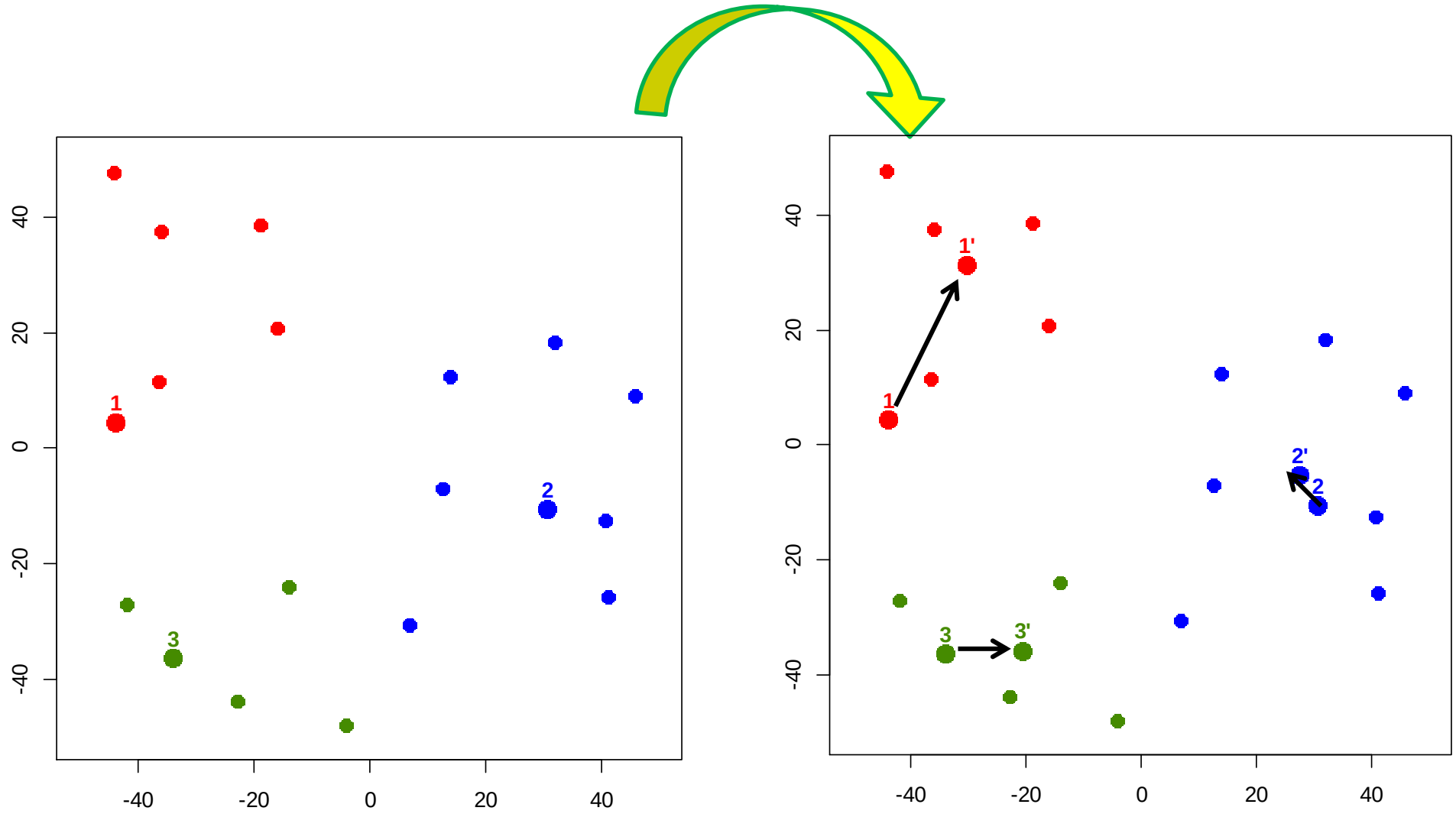


K-means clustering (2)



K-means clustering (3)

Una volta individuati i cluster vengono calcolati i loro centroidi ("punti" con valori di variabili uguali al valore medio di tutte le osservazioni di quel cluster)



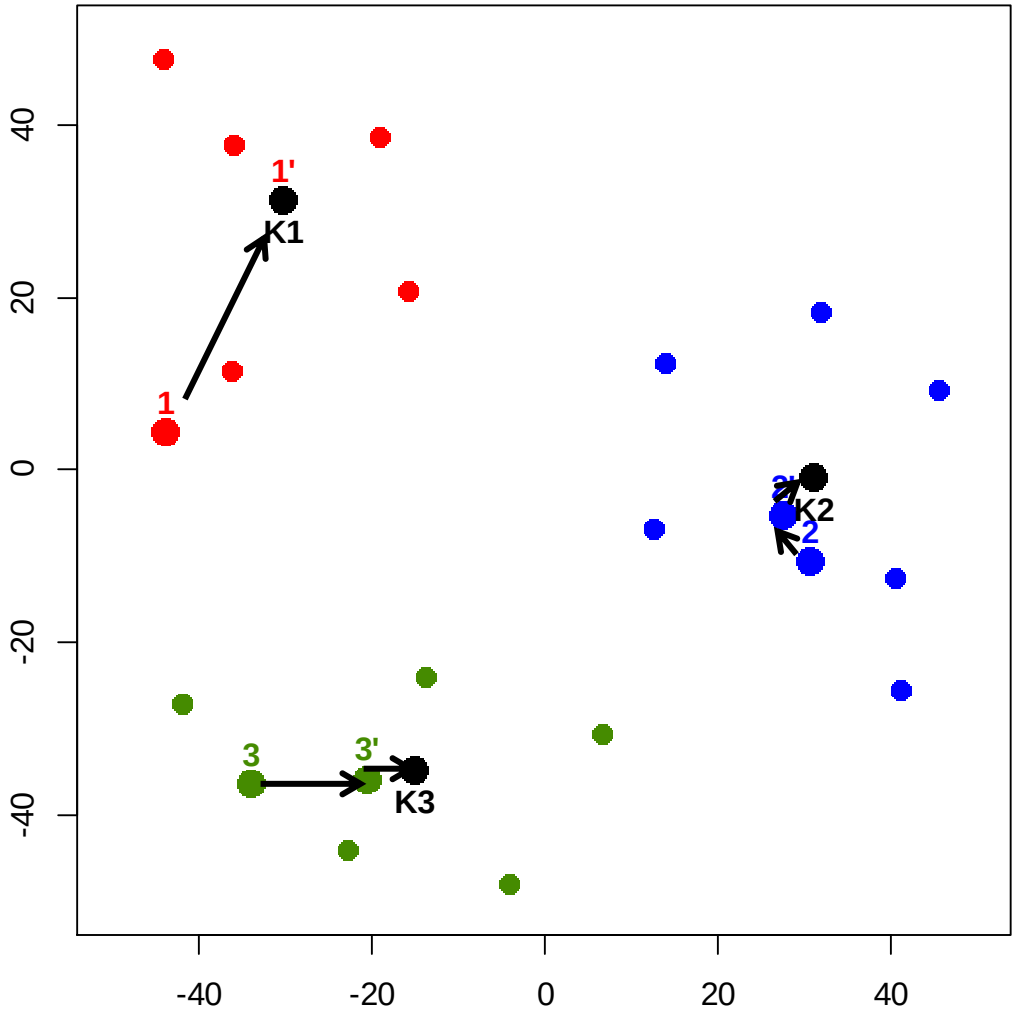
K-means clustering (4)

Vengono quindi calcolate le distanze di tutte le osservazioni dai nuovi centroidi e riassegnate le appartenenze ai cluster. Quindi vengono calcolati i nuovi centroidi dei cluster.

Il processo è iterativo e finisce quando, tra una iterazione ed un'altra, a nessuna osservazione viene più assegnata una appartenenza ad un cluster diverso dal precedente (**convergenza**)

oppure

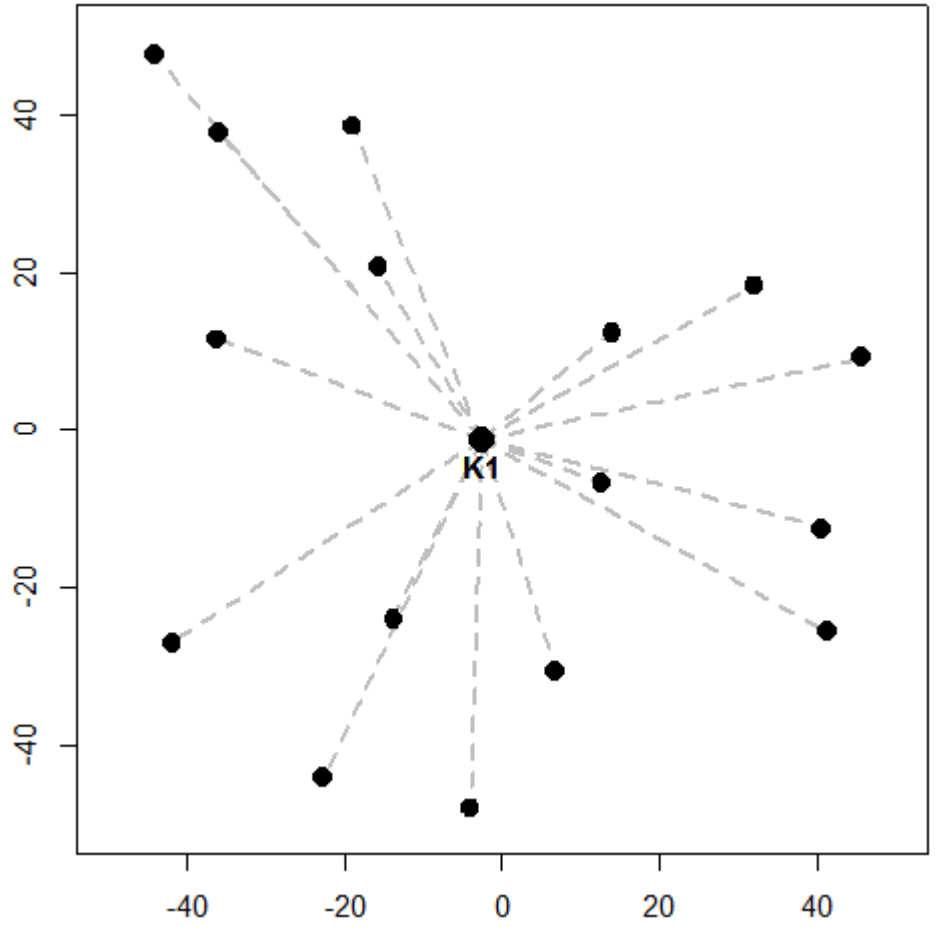
fino ad un massimo numero di iterazioni stabilito dall'utilizzatore.



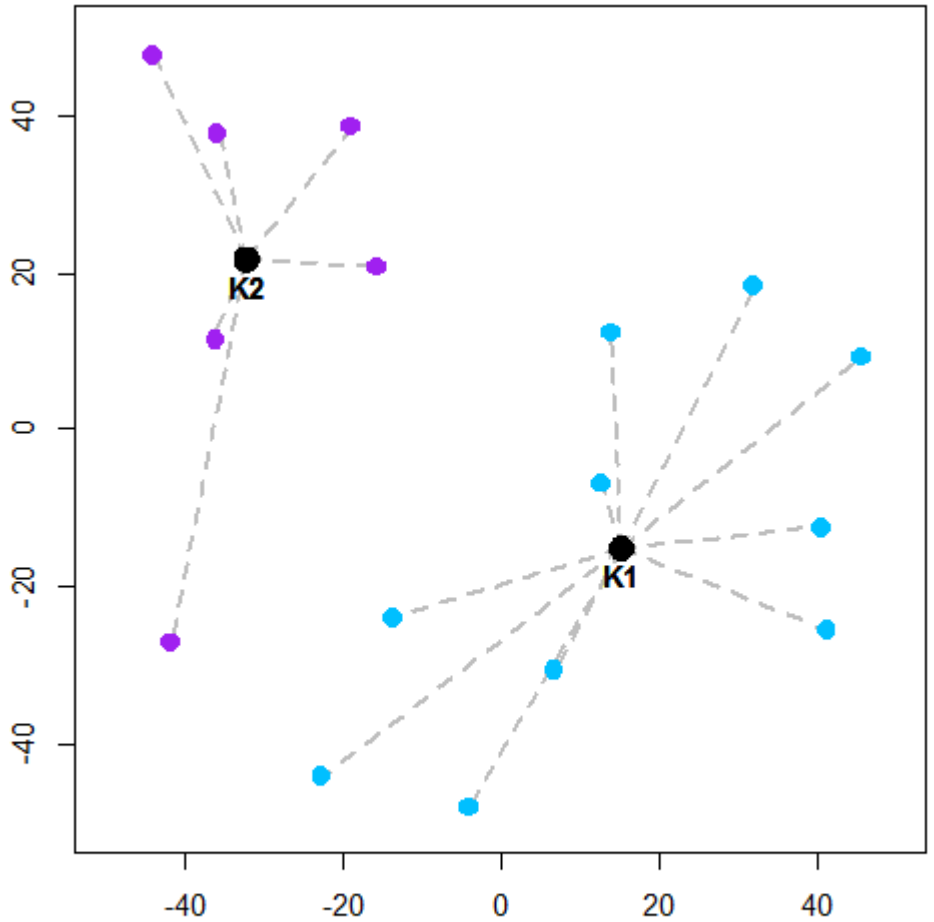
Come scegliere il numero appropriato di cluster?

Parametro: somma dei quadrati delle distanze intra-cluster (*total within-cluster sum of squares*)

Si utilizza l'algoritmo sui dati impostando diversi **k**, poi si calcola il parametro di cui sopra.

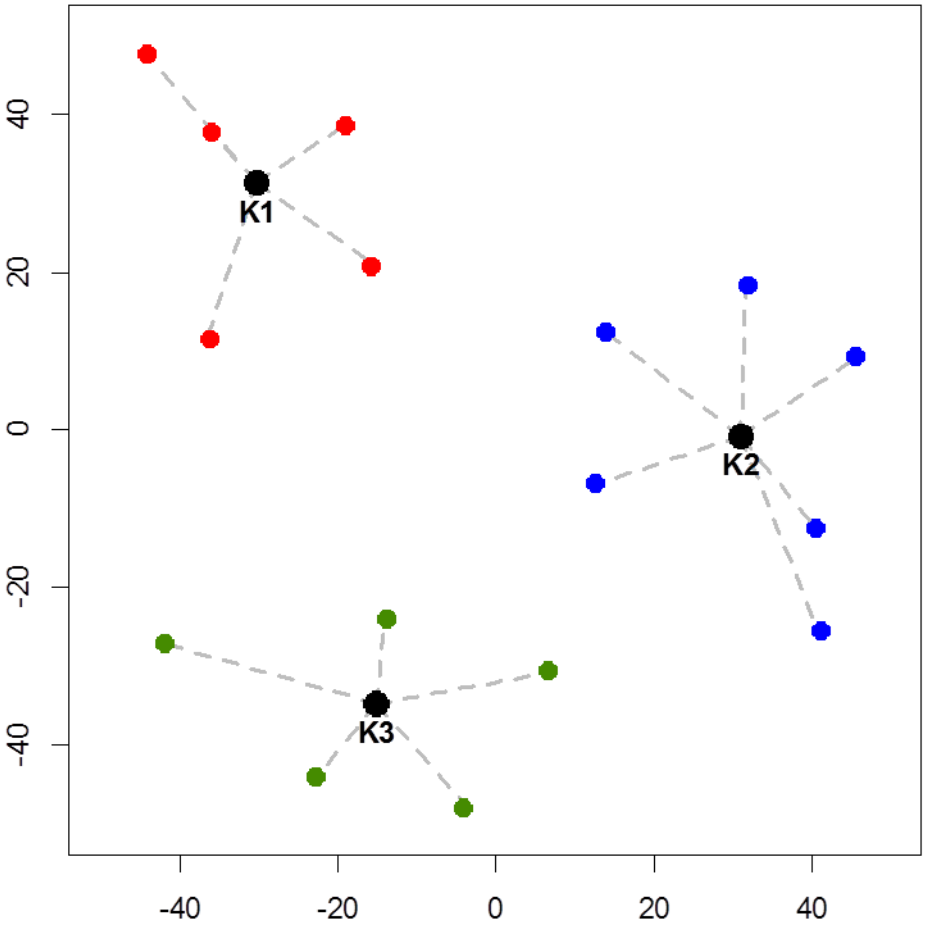


k=1

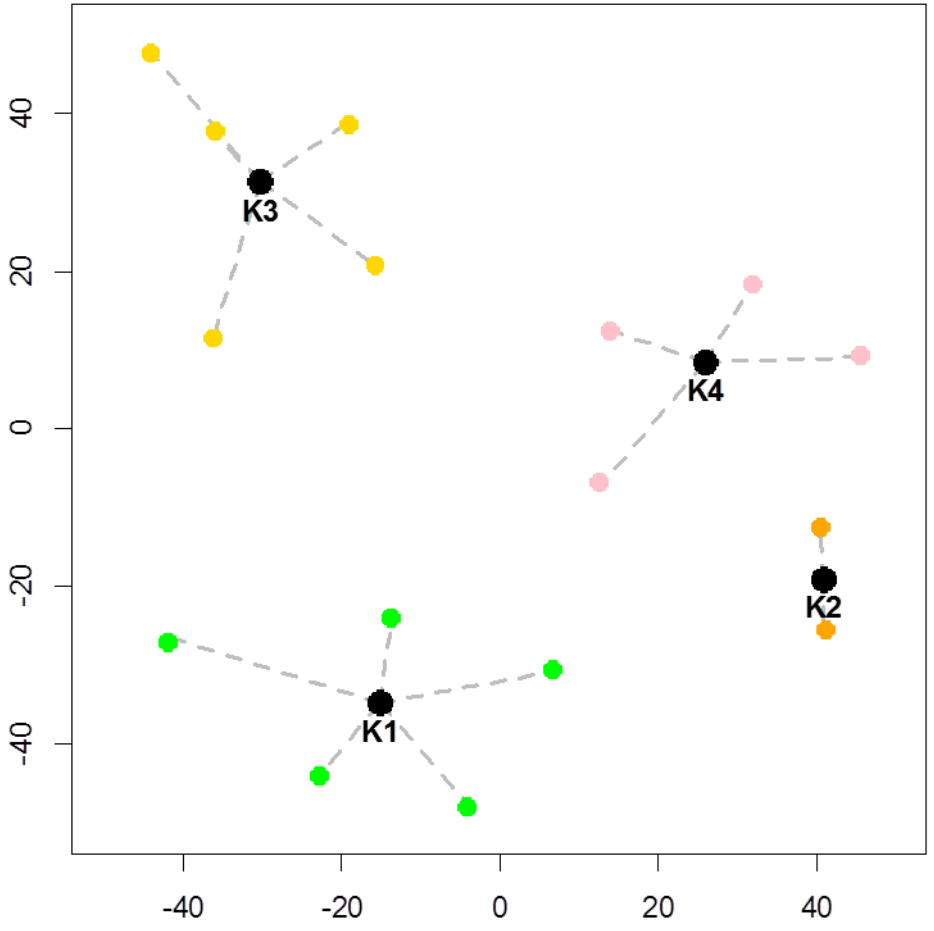


k=2

Come scegliere il numero appropriato di cluster? (2)

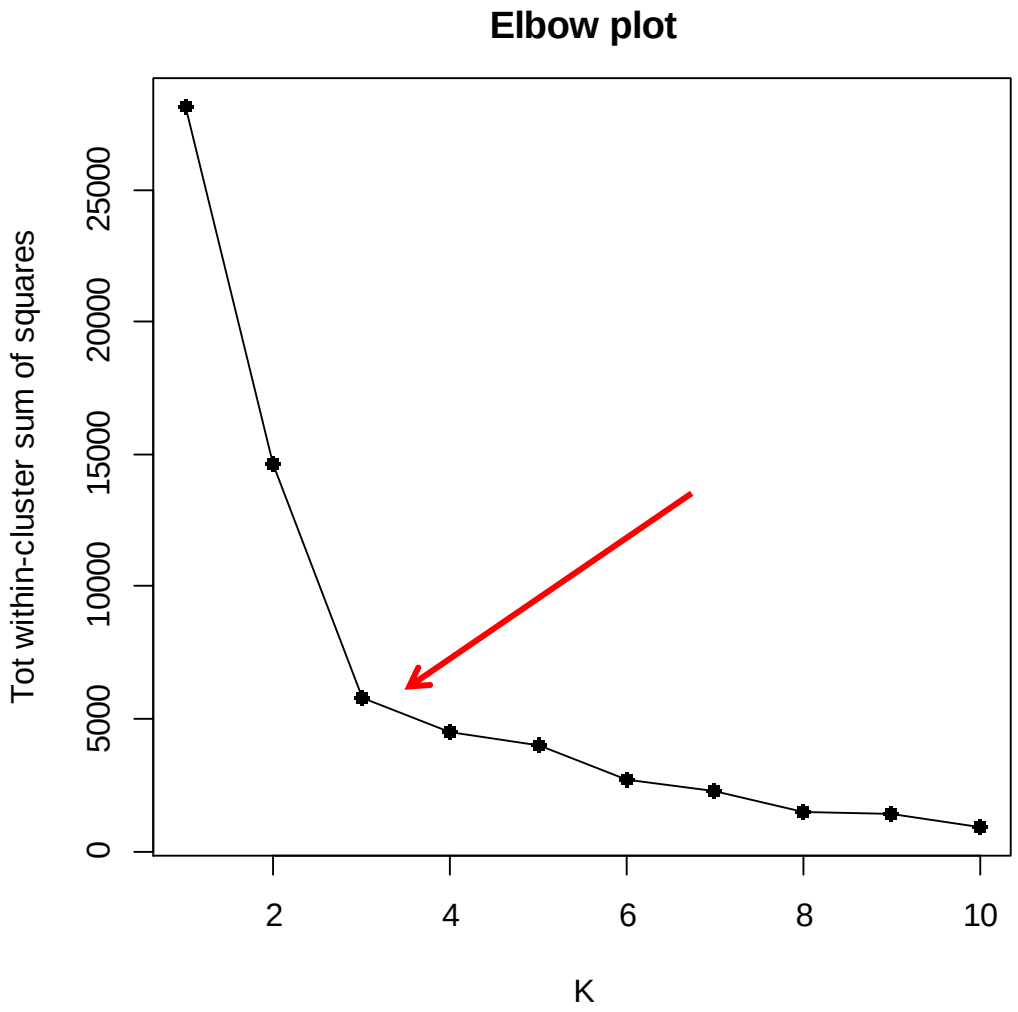


k=3



k=4

Elbow plot



Si pongono in grafico i valori di somma dei quadrati delle distanze intra-cluster contro il numero di cluster **k** e si osserva dove il grafico fa un sensibile cambio di pendenza ("gomito")

Altro metodo possibile: **Silhouette coefficient**

https://towardsdatascience.com/clustering-analysis-in-r-using-k-means-73eca4fb7967

K-means in R

KM<-kmeans(nomematrice, centers = 5 , iter.max = 20)

Scelgo il numero presunto di cluster

Scelgo il numero massimo di iterazioni (default =10)

Attenzione!!! Dato che l'inizializzazione (ovvero la scelta iniziale dei presunti centroidi) è random, ogni volta che si calcola **KM** pur con lo stesso numero di cluster può accadere che:

- I numeri dei cluster a cui le osservazioni sono associate cambino pur essendo evidente ad es. a livello grafico che si tratta dello stesso cluster;
- I valori delle coordinate dei centroidi cambino.

Per assicurare risultati riproducibili (per esempio ogni volta che si riapre il software) è utile utilizzare la seguente funzione, subito prima di calcolare **KM**:

set.seed(numerointero)

KM<- etc...

Numero a piacere

K-means in R (2)

```
> attributes(KM)
$names
[1] "cluster"      "centers"      "totss"        "withinss"
[5] "tot.withinss" "betweenss"    "size"         "iter"
[9] "ifault"

$class
[1] "kmeans"
```

KM$cluster → Vettore che contiene l'assegnazione di ogni osservazione ad un numero di cluster

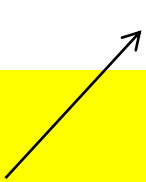
KM$centers → Matrice che contiene le coordinate a n-dimensioni (con n = numero di variabili sperimentali) dei centroidi di ogni cluster

KM$size → Vettore contenente il numero di osservazioni assegnate ad ogni cluster

KM$tot.withinss → Somma dei quadrati delle distanze intra-cluster

Elbow plot in R

Per calcolare la somma dei quadrati delle distanze intra-cluster per diversi valori di **k** e quindi ottenere l'Elbow plot si può utilizzare la funzione **for**.

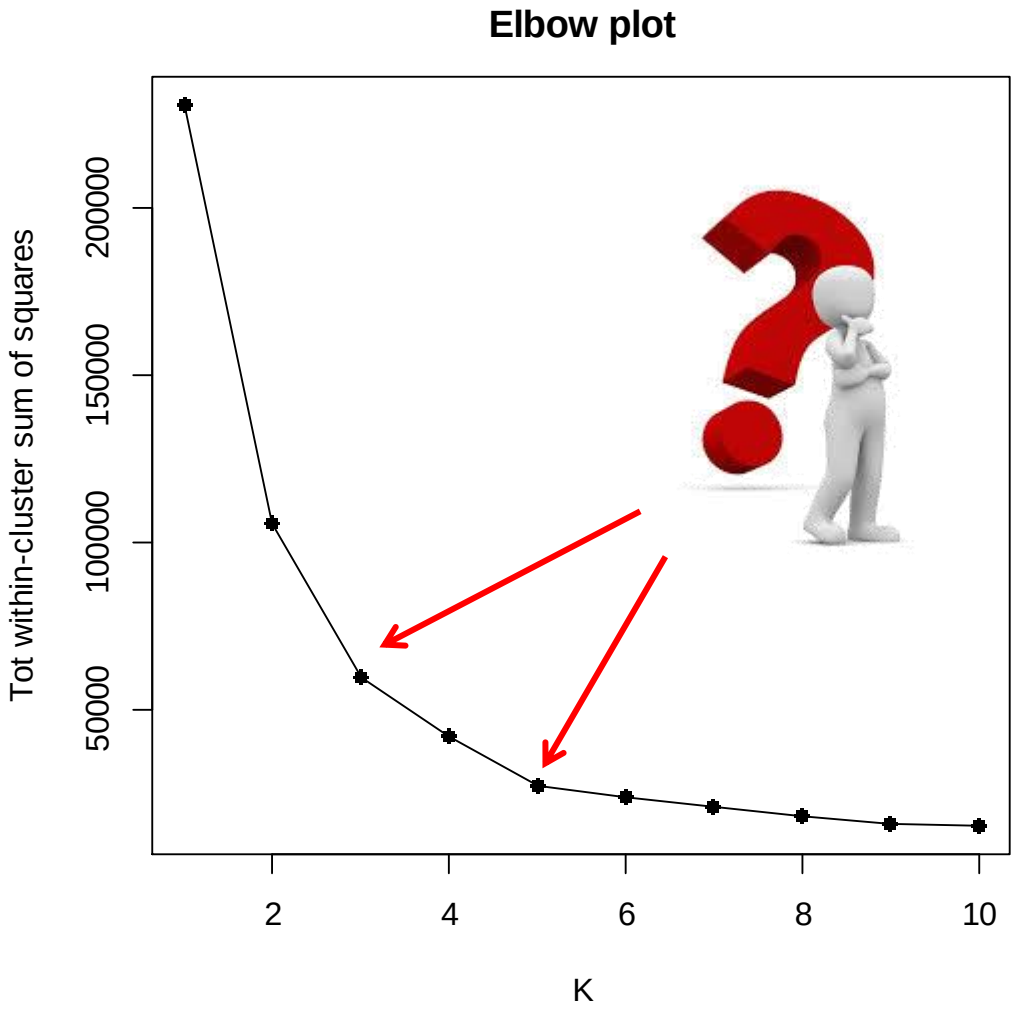



```
Tot<-NULL
for(i in c(1:numerointero)) {set.seed(numerointero); km<- kmeans(nomematrice, centers=i);
Tot<-c(Tot,km$tot.withinss)}
```

```
plot(Tot, xlab="K",ylab="Somma dei quadrati delle distanze intra-cluster", pch=16, type="o",
main="Elbow plot")
```

Interpretare dataset sperimentali con k-means clustering

(Estratto database CREA (vedi esercitazioni) , variabili = Acqua, Proteine, Lipidi, Carboidrati)

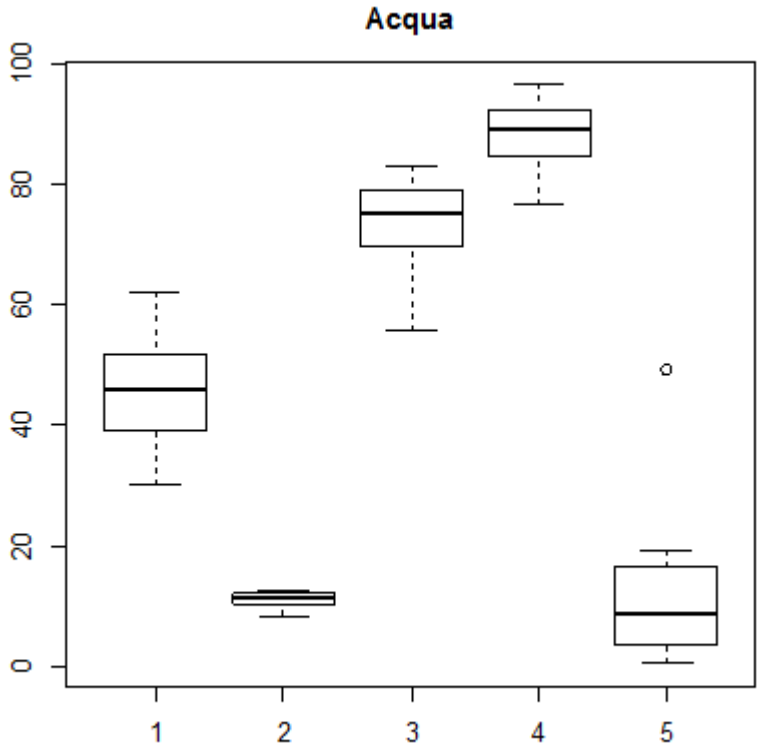


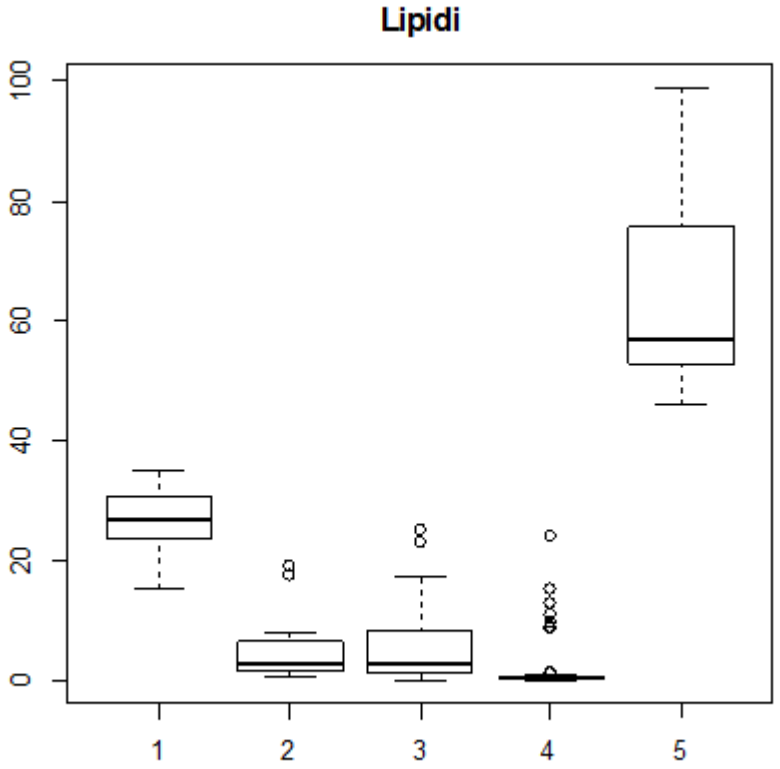
Interpretare dataset sperimentali con k-means clustering

```
set.seed(7)

KM<-kmeans(nomematrice, centers = 5 , iter.max = 100)
```

```
Newdata<-data.frame(nomematrice, K5= KM$cluster)
```



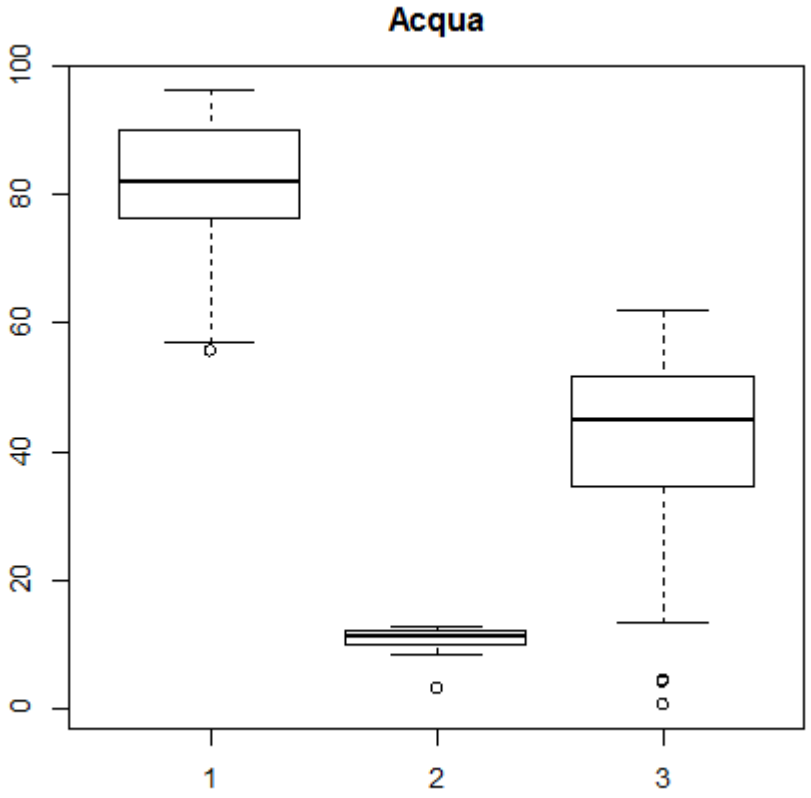


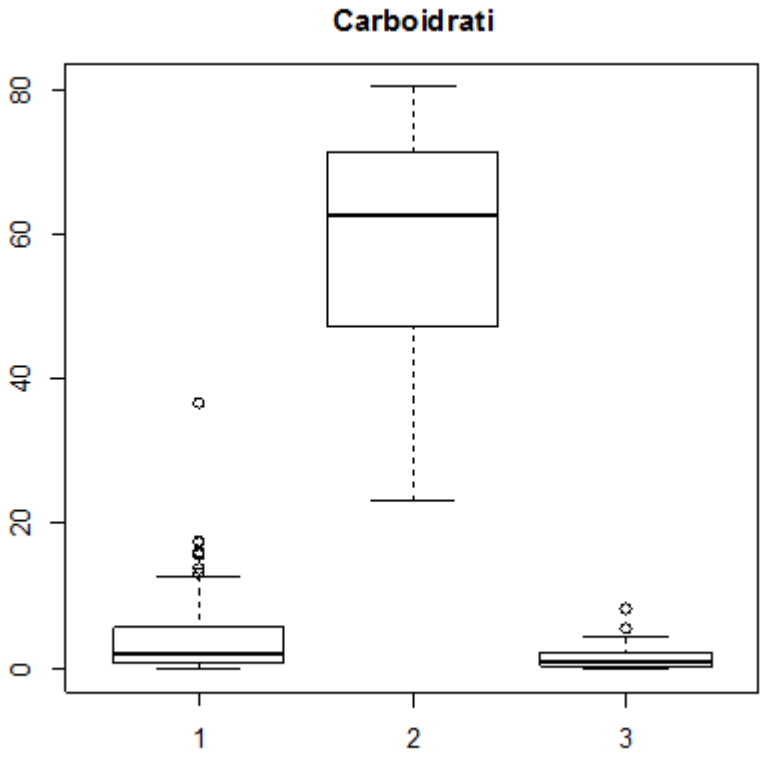
Interpretare dataset sperimentali con k-means clustering

```
set.seed(7)

KM<-kmeans(nomematrice, centers = 3 , iter.max = 100)
```

```
Newdata2<-data.frame(Newdata, K3= KM$cluster)
```





Difficoltà interpretative con dataset sperimentali

Le variabili sperimentali sono numerose?

I cluster sono di difficile interpretazione?

Suggerimento:

a) analizzare i dati con Principal Component Analysis;
b) applicare un algoritmo di clusterizzazione alle prime 2 o 3 componenti principali.