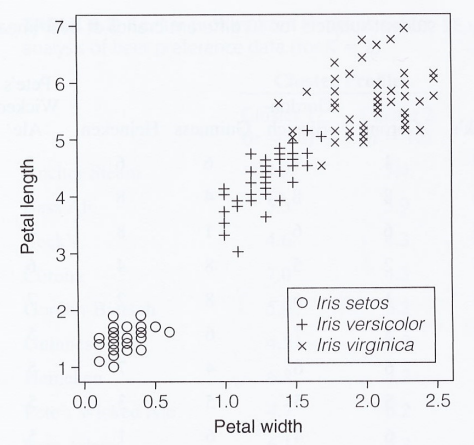
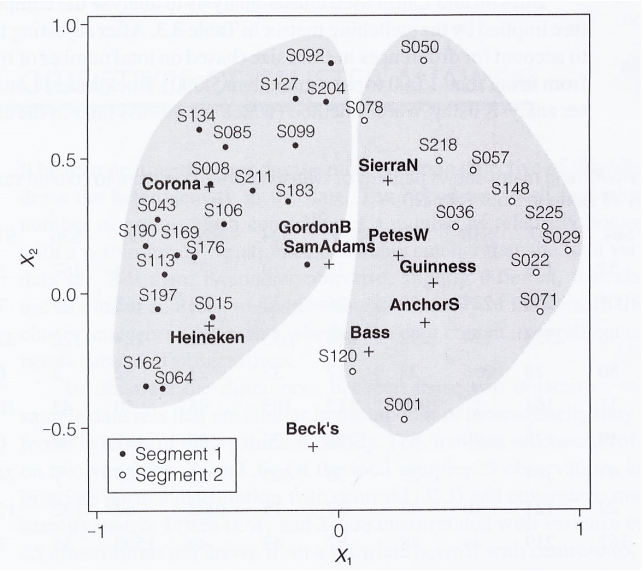
Cluster Analysis

Clustering

- Search data for a structure of "natural" groupings
  - assess dimensionality
  - identify outliers
  - suggest interesting hypotheses concerning relationship
- Distinct from **Classification**
  - a known number of groups
  - To assign new observations to one of the groups



- Potential application
  - Numerical taxonomy
  - Market segmentation
    - Divide the market into smaller groups that are more homogeneous and more easily served by a a particular type of product or a particular promotional campaign
  - Market structure analysis
    - Identify groups of similar products according to competitive measure of similarity



Similarity Measure

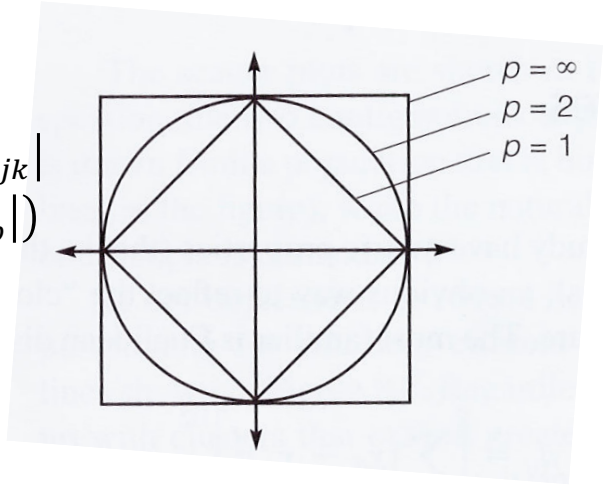
- Euclidean Distance

$$d(\boldsymbol{x},\boldsymbol{y}) = \sqrt{(\boldsymbol{x}-\boldsymbol{y})'(\boldsymbol{x}-\boldsymbol{y})}$$

- Minkowski p-Metric

$$d(\boldsymbol{x},\boldsymbol{y}) = \left[\sum_k |x_k - y_k|^p\right]^{1/p}$$

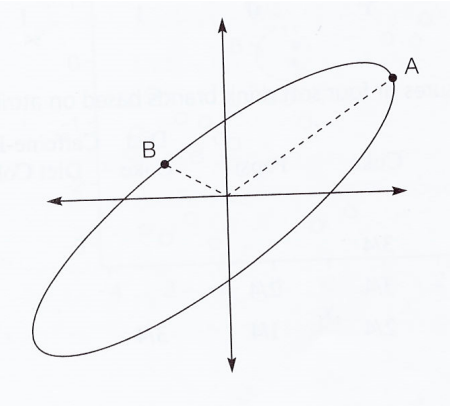
  - p=2: Euclidean distance
  - p=1: City-block distance (manhattan distance) $d_{ij}(1) = \sum_k |x_{ik} - x_{jk}|$
  - p= ∞ : sup metric $d_{ij}(\infty) = \max\left(|x_{i1} - x_{j1}|, |x_{i2} - x_{j2}|, \ldots, |x_{ip} - x_{jp}|\right)$



Similarity Measure

- Statistical distance (Mahalanobis distance)

$$d(\boldsymbol{x}, \boldsymbol{y}) = \sqrt{(\boldsymbol{x}-\boldsymbol{y})'\boldsymbol{S}^{-1}(\boldsymbol{x}-\boldsymbol{y})}$$




```
d=dist(x,method="euclidean")
```

- "dist" function
  - method="euclidean": Euclidean distance
  - method="manhattan": City block distance
  - method="minkowski": Minkowski diatance
- "mahalanobis" function
  - Mahalanobis distance

Hierarchical Clustering (Agglomerative Clustering)

- Start with each object in its own separate cluster
- Find two "closest" clusters and join them together
- Continue until one cluster of size *n* remains
- *Dendrogram* illustrates the mergers that have been made at successive levels
- Algorithm

Clusters $C_1, C_2, \ldots, C_n$ each containing a single individual

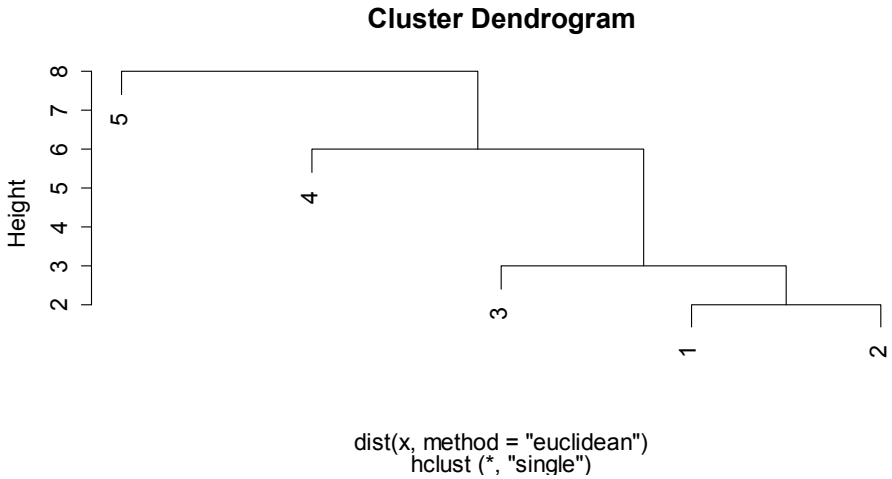
Step 1. Find the nearest pair of distinct clusters, say C_i and C_j, and merge them. Decrease the number of clusters by one.

Step 2. If the number of clusters equals one, then stop; otherwise return to Step 1.

Hierarchical Clustering : Single Linkage

- Merge nearest neighbors
- $d_{(UV)W} = \min\{d_{UW}, d_{VW}\}$

```
> x=c(1,3,6,12,20)
> dist(x,method="euclidean")
   1  2  3  4
2  2
3  5  3
4 11  9  6
5 19 17 14  8
> hc1=hclust(dist(x,method="euclidean"),method="single") # single linkage
> plot(hc1)
```



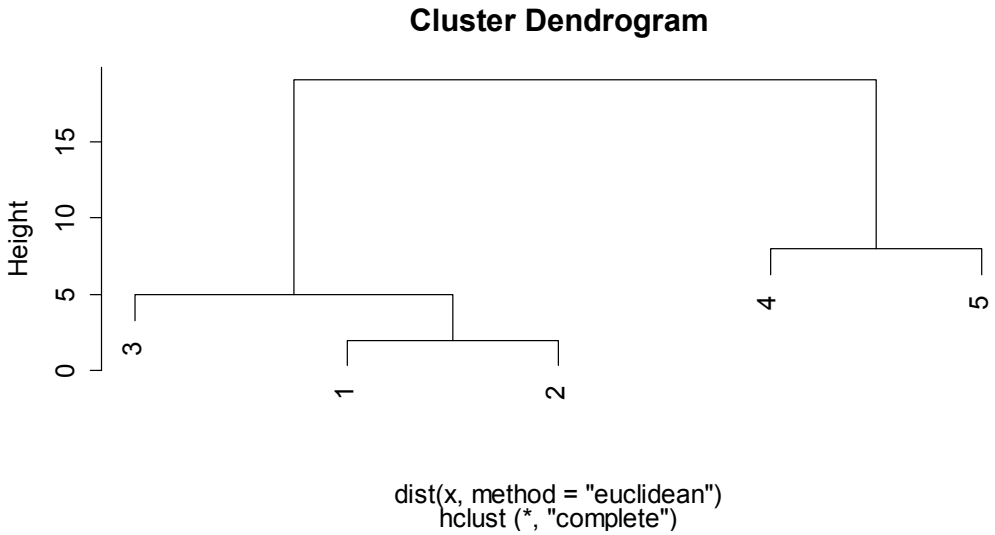
- Cannot distinguish poorly separated clusters
- Tend to pick out long stringlike clusters (chaining)

Hierarchical Clustering : Complete Linkage

- At each stage, the distance between clusters is determined by the distance between the two elements that are *most distant*
- $d_{(UV)W} = \max\{d_{UW}, d_{VW}\}$

```
> hc2=hclust(dist(x,method="euclidean"),method="complete") # complete linkage
> plot(hc2)
```

- All items in a cluster are within some maximum distance of each other
- Sensitive to outliers

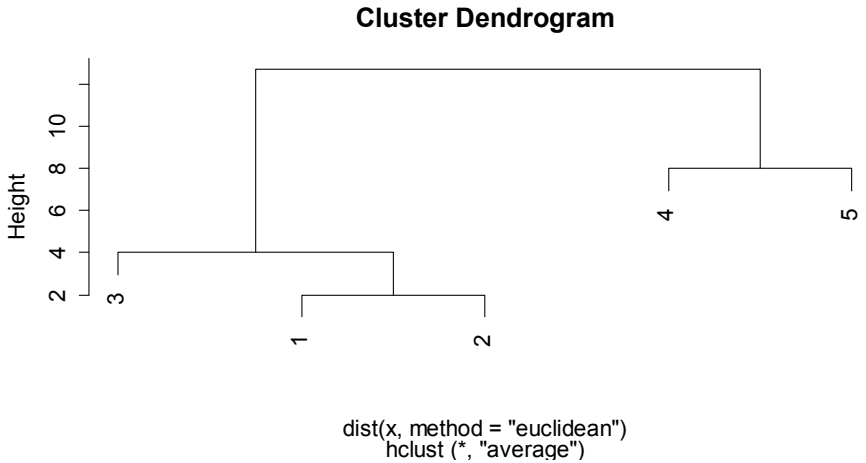


Hierarchical Clustering : Average Linkage

- The distance between two clusters is determined by the average distance between all pairs of items where one member of a pair belongs to each cluster.
- $d_{(UV)W} = \frac{\sum_i \sum_k d_{ik}}{N_{UV} N_W}$

```
> hc3=hclust(dist(x,method="euclidean"),method="average") # average linkage
> plot(hc3)
```

- Changes in the assignment of distances can affect the arrangement of the final configuration of clusters unlink single or complete linkage



Hierarchical Clustering : Ward's Method

- Minimize the 'loss of information' from joining two groups
- Minimum variance method
- Error sum of squares criterion (ESS)
  - ESS_k: the sum of the squared deviations of every item in the cluster from the cluster mean

$$ESS = ESS_1 + ESS_2 + \cdots + ESS_K$$

  - If there are *N* items and each cluster consists of a single item, $ESS = 0$
  - If all the clusters are combined in a single group of *N* items, $ESS = \sum_{j=1}^{N} (\boldsymbol{x_j} - \bar{\boldsymbol{x}})'(\boldsymbol{x_j} - \bar{\boldsymbol{x}})$
- The two clusters whose combination results in *the smallest increase in ESS*
- Ward's method is based on the notion that the clusters of multivariate observations are expected to be roughly elliptically shaped
- Similar to nonhierarchical clustering procedures

Example: USArrests

- This data set contains statistics, in arrests per 100,000 residents for assault, murder, and rape in each of the 50 US states in 1973. Also given is the percent of the population living in urban areas.
- A data frame with 50 observations on 4 variables.

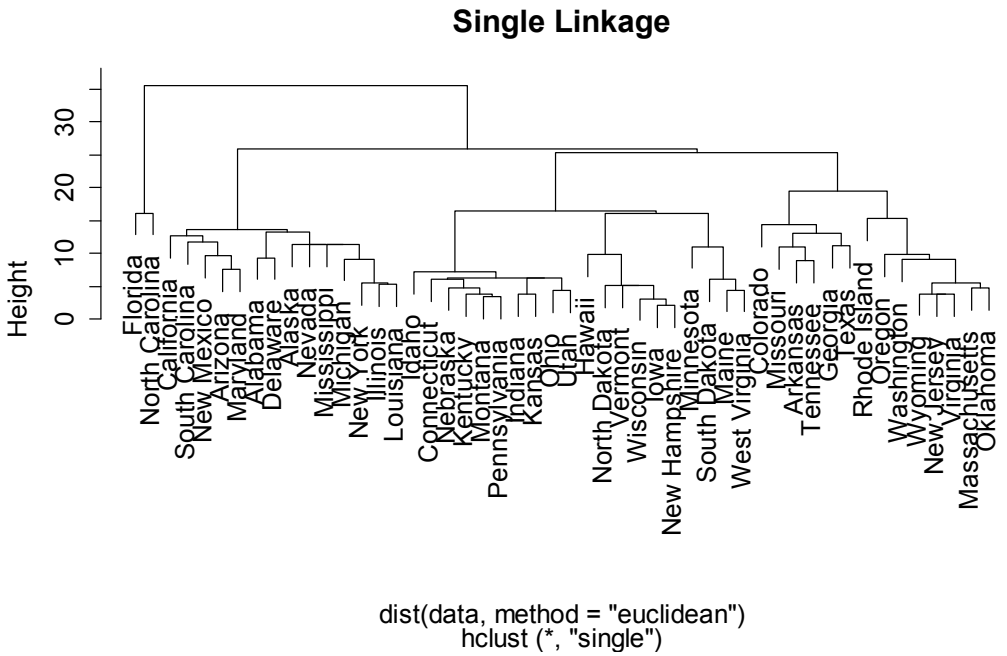
[,1] Murder numeric Murder arrests (per 100,000)

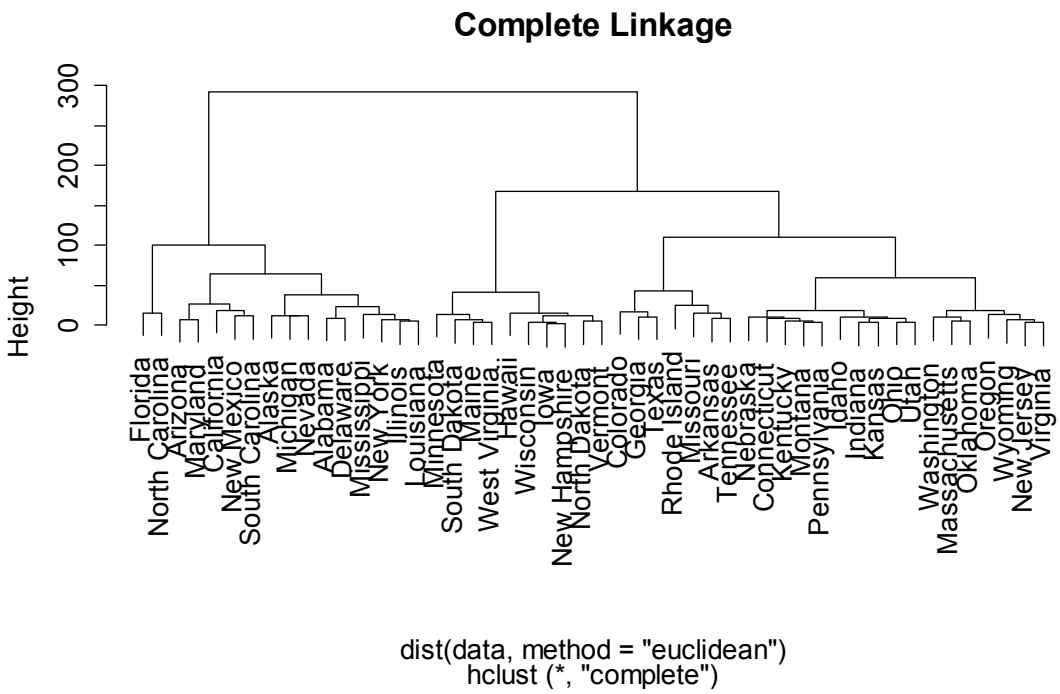
[,2] Assault numeric Assault arrests (per 100,000)

[,3] UrbanPop numeric Percent urban population

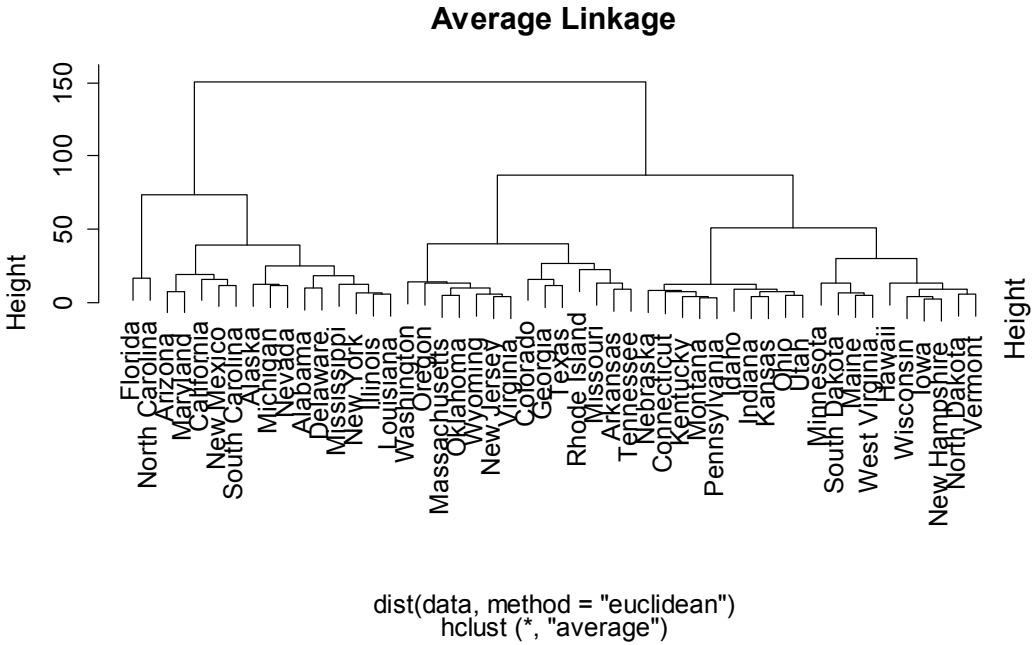
[,4] Rape numeric Rape arrests (per 100,000)

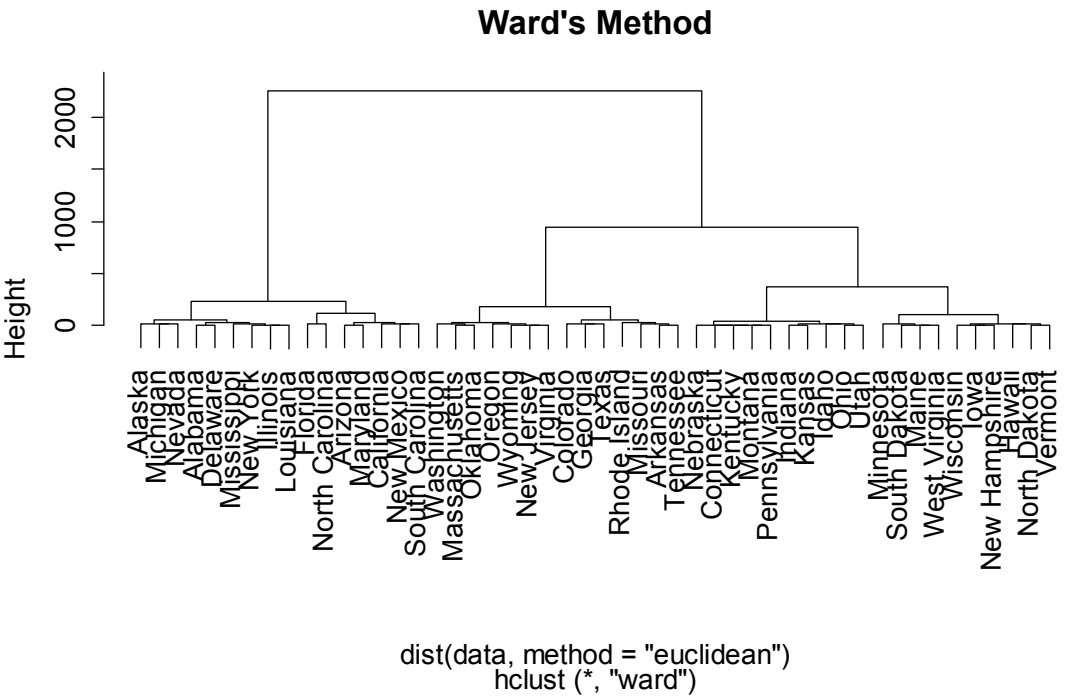
Example: USArrests



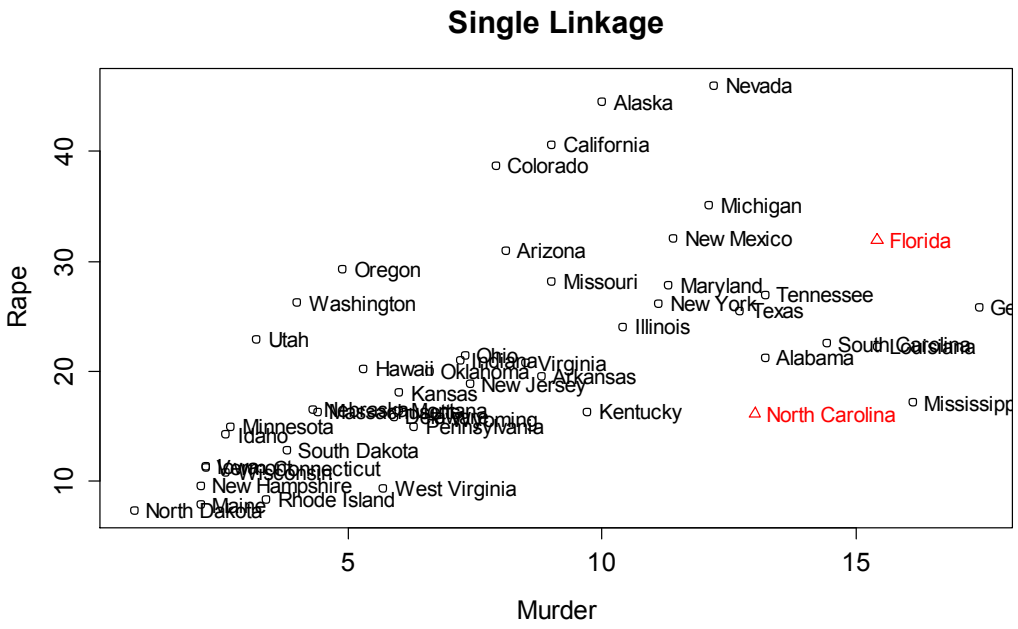


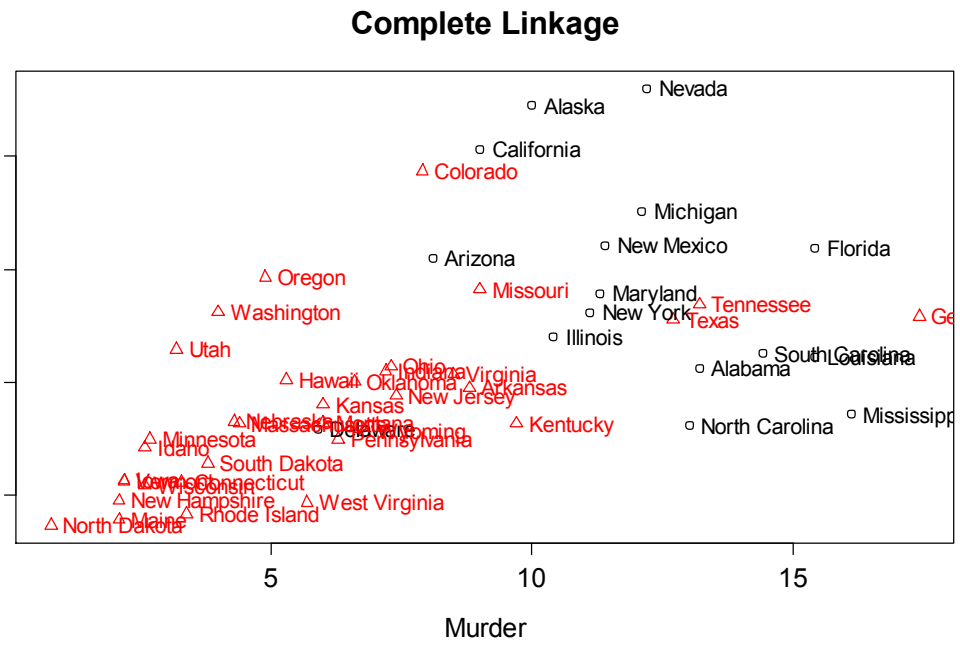
Example : USArrests





Example: USArrests





Nonhierarchical Clustering: K-means Clustering

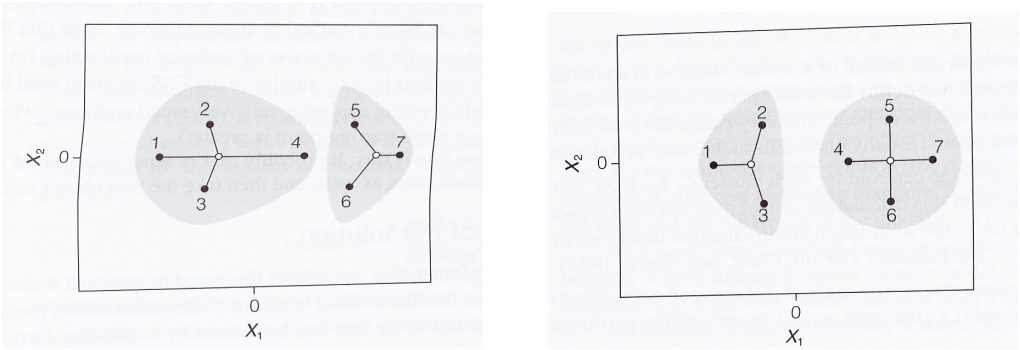
- The number of clusters *K* is specified in advanced

Step 1. Partition the items into *K* initial clusters.

Step 2. Proceed through the list of items, assigning an item to the cluster whose centroid is nearest. Recalculate the centroid for the cluster receiving the new items and for the cluster losing the item.

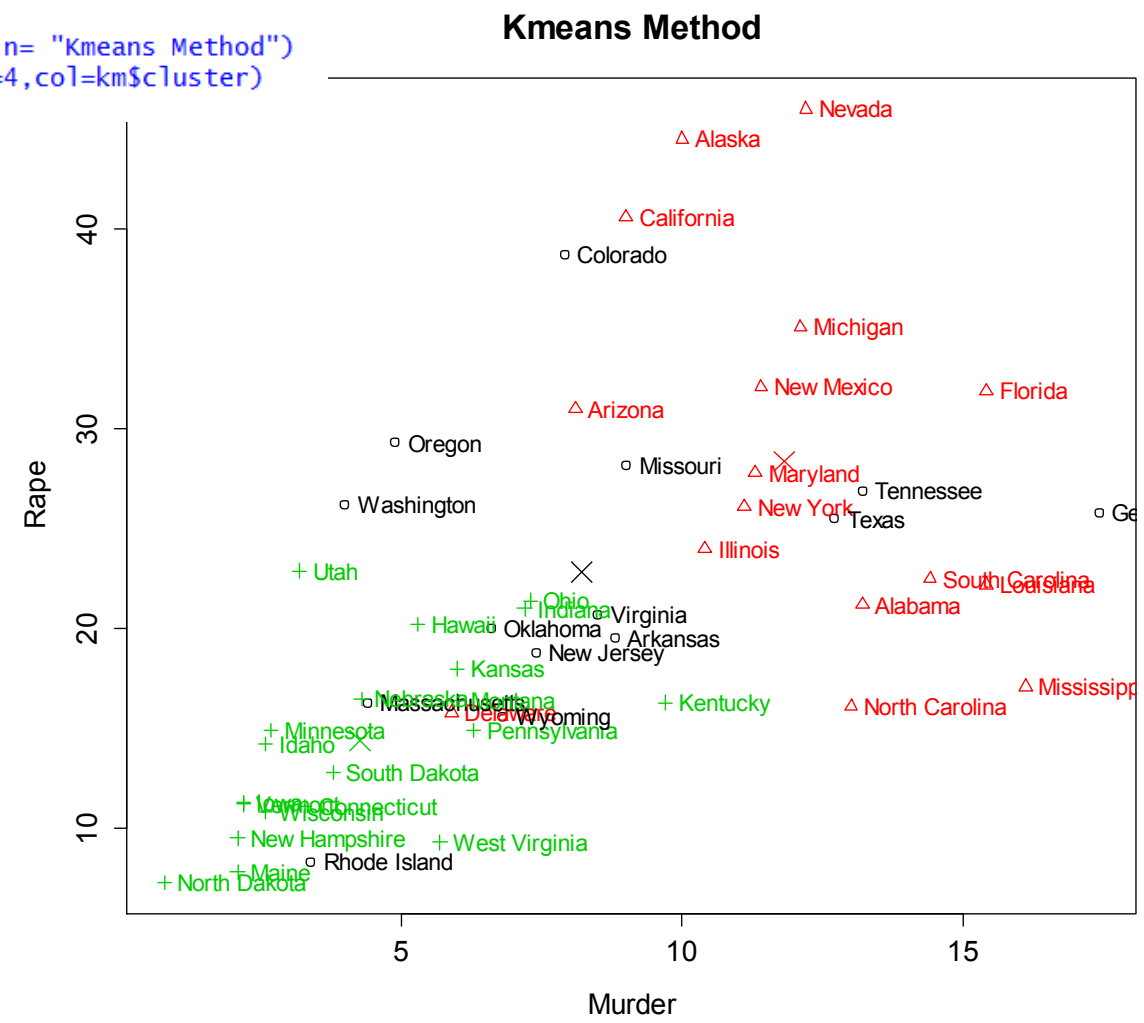
Step 3. Repeat Step 2 until no more reassignments take place.

- The final assignment of items to clusters will be dependent upon the initial partition



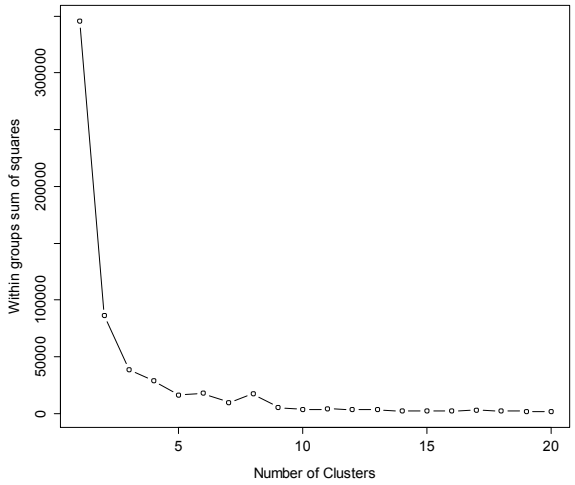
Example: USArrests

```
> km=kmeans(data,centers=3)
> plot(Murder,Rape,pch=km$cluster,col=km$cluster,main= "Kmeans Method")
> text(Murder,Rape,labels=state, cex=0.8, adj=0,pos=4,col=km$cluster)
> points(km$centers[,c(1,3)],col=1:3,pch=4,cex=2)
```



Number of Clusters

- Cluster analysis is most often used as an exploratory tool ➔ No clear answer for the number of clusters
- Hierarchical clustering
  - Where to cut the dendrogram?
  - Examine the sizes of the changes in height in the dendrogram and take a "large" change
  - Use cutree function in R
- K-means clustering
  - As the number of clusters increases, the within-groups sum of squares decreases
  - Find an obvious "elbow"



Model-based Clustering

- Cluster *k* has expected proportion p_k of the objects and the corresponding measurements are generated by a probability density function $f_k(\boldsymbol{x})$
- The mixing distribution

$$f(x) = \sum_{k=1}^{K} p_k f_k(\boldsymbol{x}), \qquad \sum_{k} p_k = 1$$

- The most commo mixture model is a mixture of multivariate normal distributions

$$f_k(\boldsymbol{x}) \text{ is the } N_p(\mu_k, \Sigma_k)$$

- Find the maximum likelihood estimates $\widehat{p_1}, \dots, \widehat{p_k}, \widehat{\mu_1}, \dots, \widehat{\mu_k}, \widehat{\Sigma_1}, \dots, \widehat{\Sigma_k}$
- Assumed form for Σ_k

Identifier	Model	# Covariance parameters	Distribution
EII	λI	1	Spherical
VII	$\lambda_k I$	G	Spherical
EEI	λA	d	Diagonal
VEI	$\lambda_k A$	$G + (d-1)$	Diagonal
EVI	λA_k	$1 + G(d-1)$	Diagonal
VVI	$\lambda_k A_k$	Gd	Diagonal
EEE	$\lambda DAD^\top$	$d(d+1)/2$	Ellipsoidal
EEV	$\lambda D_k A D_k^\top$	$1 + (d-1) + G[d(d-1)/2]$	Ellipsoidal
VEV	$\lambda_k D_k A D_k^\top$	$G + (d-1) + G[d(d-1)/2]$	Ellipsoidal
VVV	$\lambda_k D_k A_k D_k^\top$	$G[d(d+1)/2]$	Ellipsoidal

Model-based Clustering

- How do we decide the number of clusters?
  - Maximize $2 \ln L_{max} - Penalty$
    - Akaike information criterion (AIC)

$$AIC = 2 \ln L_{max} - 2N \left(K \frac{1}{2} (p+1)(p+2) - 1 \right)$$

    - Bayesian information criterion (BIC)

$$BIC = 2 \ln L_{max} - 2 \ln N \left(K \frac{1}{2} (p+1)(p+2) - 1 \right)$$

- Assign to the cluster k for which the conditional probability of membership

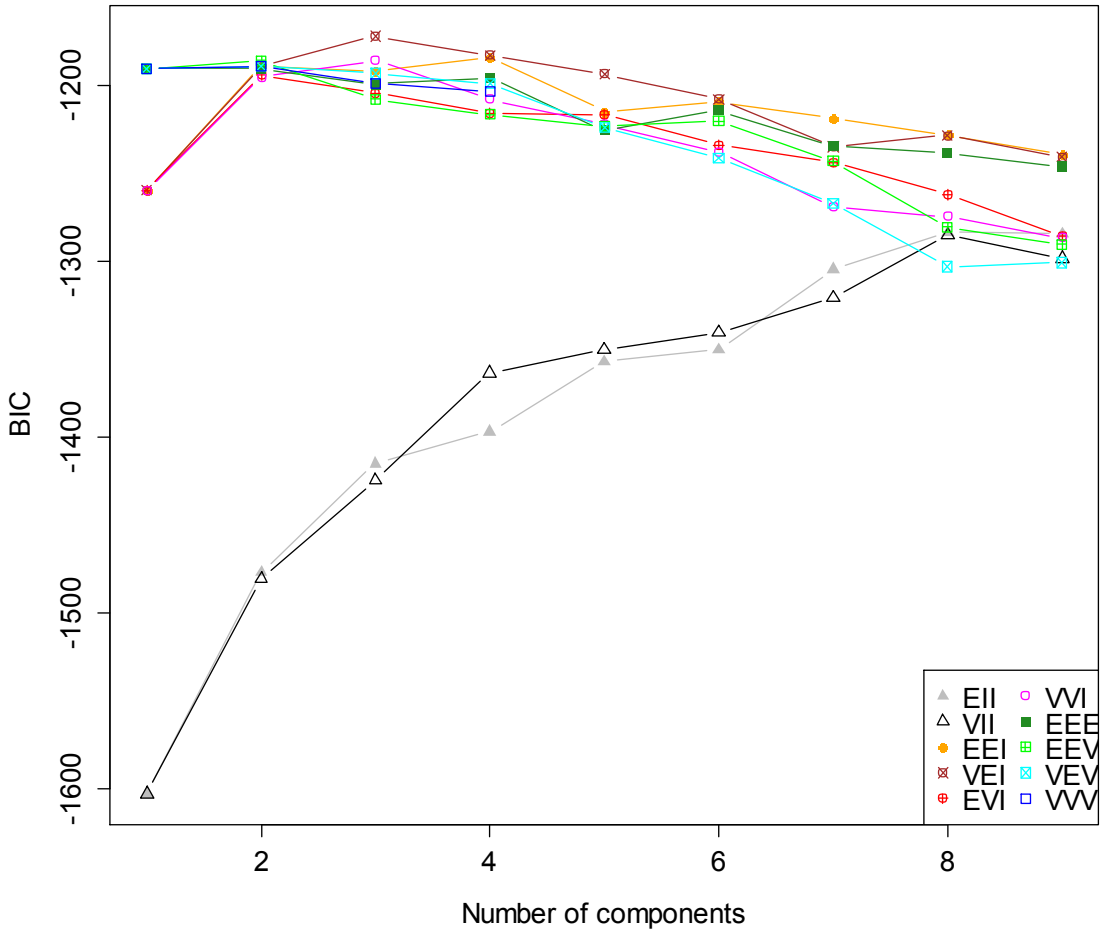
$$p(k|\boldsymbol{x}_j) = \frac{\widehat{p_j} f(\boldsymbol{x}_j|k)}{\sum_{i=1}^{K} \widehat{p_i} f(\boldsymbol{x}_i|k)}$$

is the largest

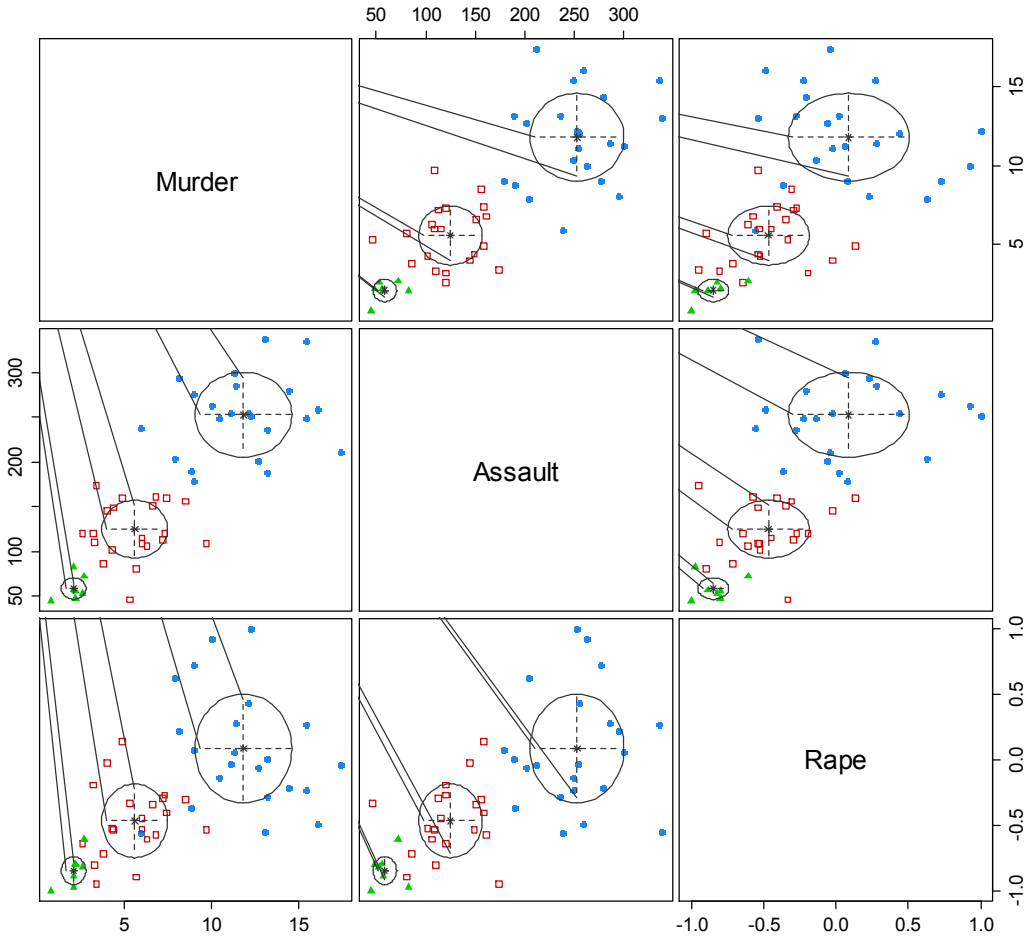
Example: USArrests

```
> library(mclust)
> mc=Mclust(data)
> summary(mc)
----------------------------------------------------
Gaussian finite mixture model fitted by EM algorithm
----------------------------------------------------

Mclust VEI (diagonal, equal shape) model with 3 components:

 log.likelihood  n df       BIC       ICL
      -554.5497 50 16 -1171.692 -1174.609

Clustering table:
 1  2  3
22 21  7
```

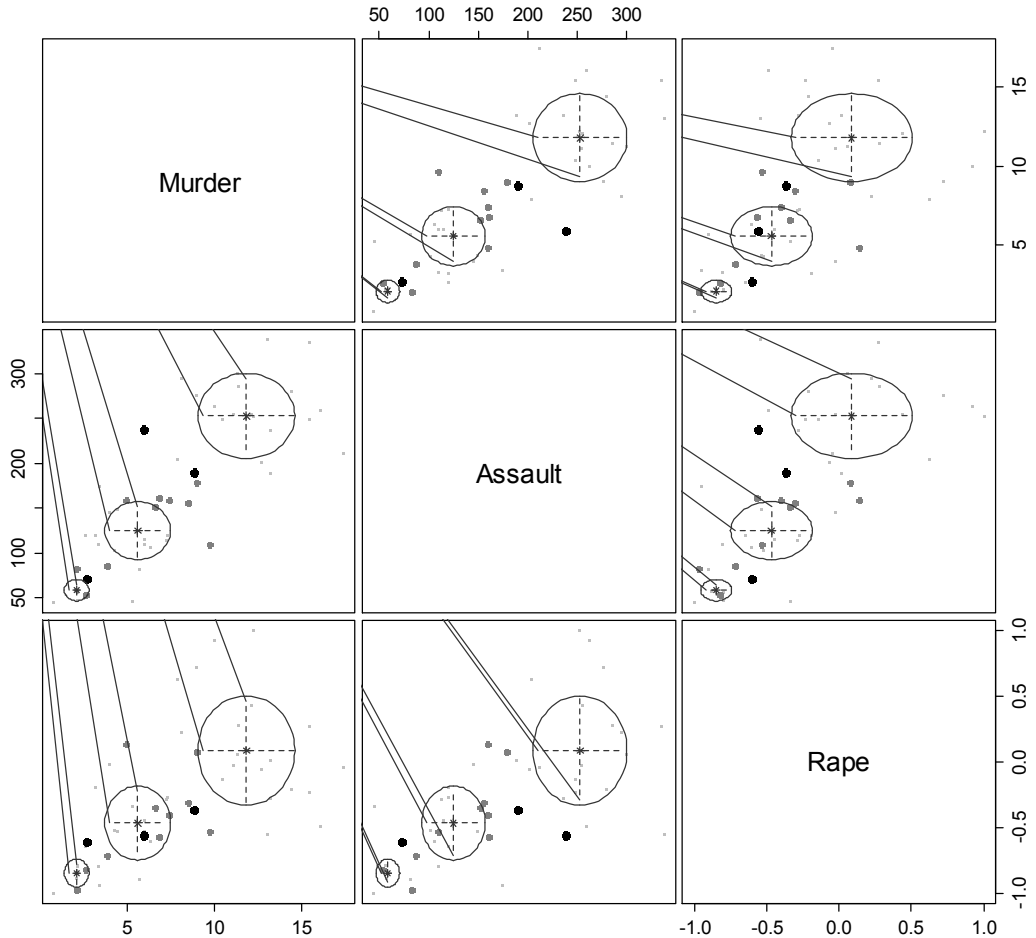
```
> plot(mc)
```



Clustering result



Classification uncertainty



Estimated density