# Measure of Distance

- We wish to define the distance between two objects
- Distance metric between points:
  - Euclidean distance (EUC)
  - Manhattan distance (MAN)
  - Pearson sample correlation (COR)
  - Angle distance (EISEN – considered by Eisen et al., 1998.)
  - Spearman sample correlation (SPEAR)
  - Kandall's $\tau$ sample correlation (TAU)
  - Mahalanobis distance
- Distance metric between distributions:
  - Kullback-Leibler information
  - Hamming's mutual information

# R: Distance Metric Between Points

"dist" function in *stat* package:

- Euclidean
- Manhattan

*hopach* package:

- disscosangle(X, na.rm = TRUE) **

*bioDist* package:

- cor.dist
- spearman.dist
- tau.dist

$g_1 = (-1.76, -1.45, 0.33)$

$g_2 = (0.04, -0.75, 0.29)$

$g_3 = (1.51, -1.60, 2.07)$

## Euclidean distance:

g1 vs g2: $\sqrt{(-1.76-0.04)^2+(-1.45-(-0.75))^2+(0.33-0.29)^2} = 1.93$

g1 vs g3: $\sqrt{(-1.76-1.51)^2+(-1.45-(-1.60))^2+(0.33-2.07)^2} = 3.70$

g2 vs g3: $\sqrt{(1.51-0.04)^2+(-1.60-(-0.75))^2+(2.07-0.29)^2} = 2.45$

| | g1 | g2 |
|---|---|---|
| g2 | 1.93 | |
| g3 | 3.70 | 2.45 |

```
> g1 = c(-1.76,-1.45,0.33)
> g2 = c(0.04,-0.75,0.29)
> g3 = c(1.51,-1.60,2.07)
> g = rbind(g1,g2,g3)
> dist(g)
          g1       g2
g2 1.931735
g3 3.707155 2.460041
```

$$g_1 = (-1.76, -1.45, 0.33)$$
$$g_2 = (0.04, -0.75, 0.29)$$
$$g_3 = (1.51, -1.60, 2.07)$$

Manhattan distance:

$$g1 \text{ vs } g2: |-1.76 - 0.04| + |-1.45 - (-0.75)| + |0.33 - 0.29| = 2.54$$
$$g1 \text{ vs } g3: |-1.76 - 1.51| + |-1.45 - (-1.60)| + |0.33 - 2.07| = 5.16$$
$$g2 \text{ vs } g3: |0.04 - 1.51| + |-0.75 - (-1.60)| + |0.29 - 2.07| = 4.10$$

| | g1 | g2 |
|---|---|---|
| g2 | 2.54 | |
| g3 | 5.16 | 4.10 |

```
> dist(g,method="manh")
      g1   g2
g2 2.54
g3 5.16 4.10
```

# Cosine Correlation Distance

- Note: disscosangle(hopach)

$$d_\alpha(\mathbf{x},\mathbf{y}) = \sqrt{1-\left(\frac{\mathbf{x}\cdot\mathbf{y}}{\|\mathbf{x}\|\,\|\mathbf{y}\|}\right)}$$

where $\mathbf{x}\cdot\mathbf{y} = \sum_{i=1}^{n} x_i y_i,\ \|\mathbf{x}\| = \sqrt{\sum_{i=1}^{n} x_i^2}$

```
> library(hopach)
> disscosangle(g)
             g1        g2        g3
[1,] 0.0000000 0.6325593 0.9748645
[2,] 0.6325593 0.0000000 0.4846881
[3,] 0.9748645 0.4846881 0.0000000
```

## Correlation-based distance :

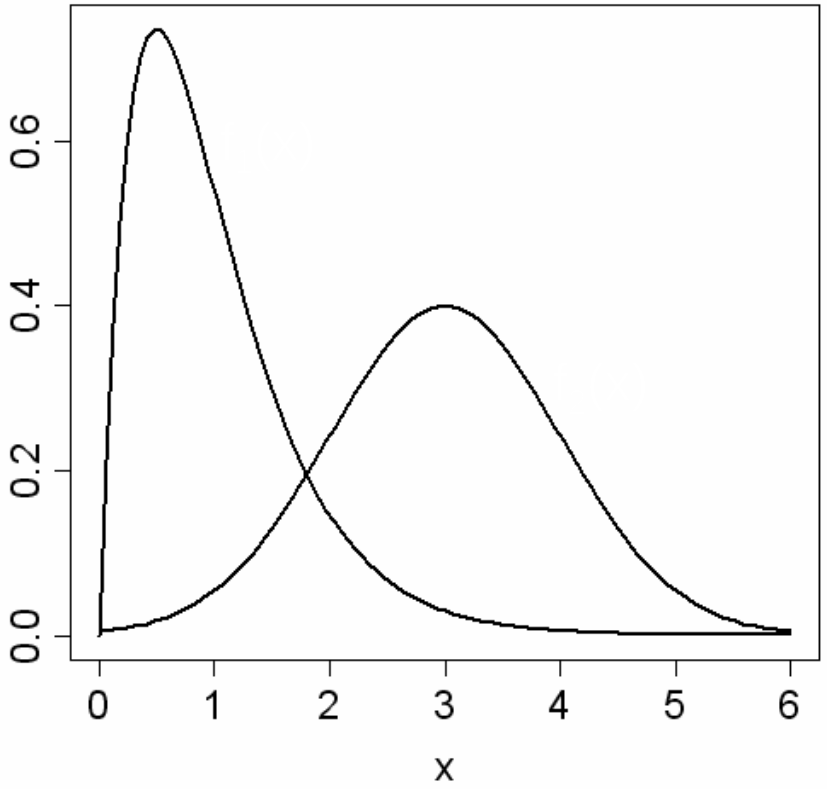
```
> library(bioDist)
> cor.dist(g)
             g1           g2
g2 0.420527385
g3 0.496338511 0.004069727
> spearman.dist(g)
    g1  g2
g2 0.5
g3 0.5 0.0
> tau.dist(g)
          g1        g2
g2 0.6666667
g3 0.6666667 0.0000000
```

# Measure of Distance

- We wish to define the distance between two objects
- Distance metric between points:
  - Euclidean distance (EUC)
  - Manhattan distance (MAN)
  - Pearson sample correlation (COR)
  - Angle distance (EISEN – considered by Eisen et al., 1998.)
  - Spearman sample correlation (SPEAR)
  - Kandall's $\tau$ sample correlation (TAU)
  - Mahalanobis distance
- Distance metric between distributions:
  - Kullback-Leibler information
  - Hamming's mutual information

# Kullback-Leibler Information

- Kullback-Leibler information (KLI) considers if the shape of the distribution of features is similar between two genes.



# Kullback-Leibler Information

$$KLI(f_1, f_2) = \int \log\left[\frac{f_1(x)}{f_2(x)}\right] f_1(x)dx$$

$$d_{KLD}(f_1, f_2) = [KLI(f_1, f_2) + KLI(f_2, f_1)]/2$$

Note:

1. KLI ($d_{KLD}$) = 0 if $f_1(x) = f_2(x)$.
2. KLI is not symmetric but $d_{KLD}$ is.
3. $d_{KLD}$ does not satisfy the triangle inequality
4. KLI or $d_{KLD}$ is not defined when $f_1(x) \neq 0$ but $f_2(x) = 0$ for some x.

# Mutual Information

- Mutual information(MI) attempts to measure the distance from independence.

$$MI(f_1, f_2) = \int_x \int_y \log\left[\frac{f(x,y)}{f_1(x)\,f_2(y)}\right] f(x,y)\,dy\,dx$$

Note:

1. If x and y are independent then $f(x,y) = f_1(x)f_2(y)$ so that MI = 0.
2. Does not satisfy the triangle inequality

# Mutual Information

- (Joe, 1989) Transformation:

$$\delta^* = [1 - \exp(-2MI)]^{1/2}$$

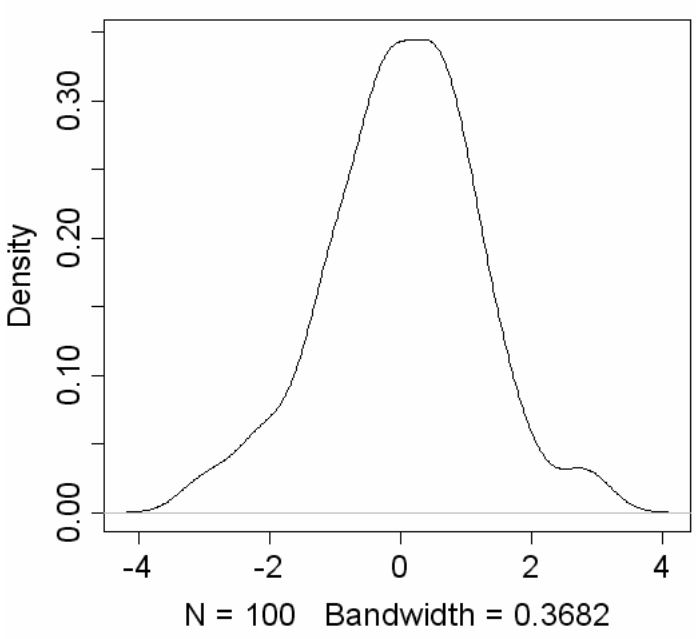
$$0 \leq \delta^* \leq 1$$

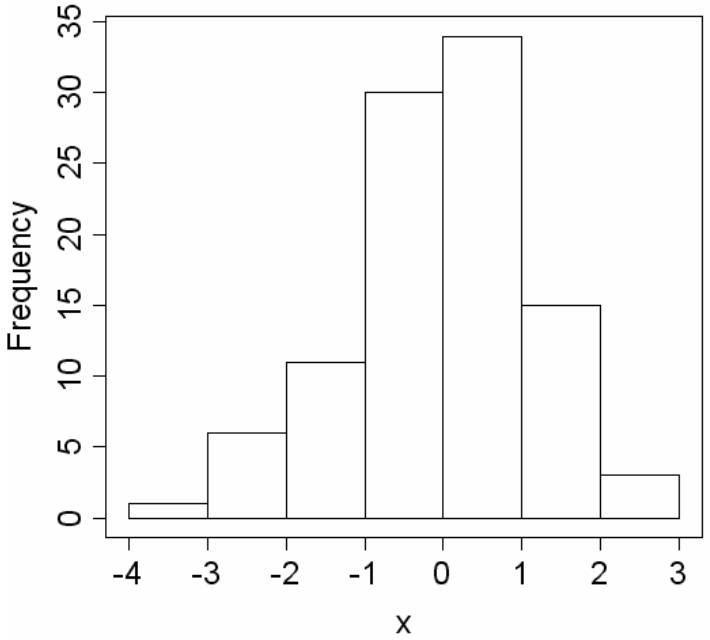
$\delta^*$ can be interpreted as a generalization of the correlation!

# R: Distance Between Distributions

*bioDist* package:

- KLD.matrix (kernel density)
- KLdist.matrix (binning)
- mutualInfo



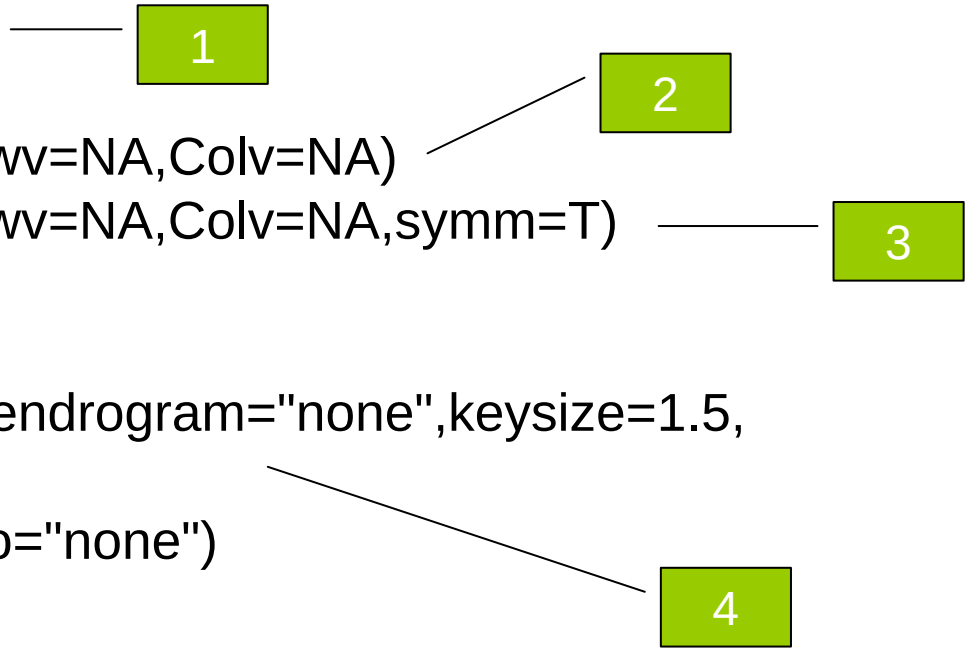
# Exercise: Apop.xls

http://homepage.ntu.edu.tw/~lyliu/IntroBioinfo/Apop.xls

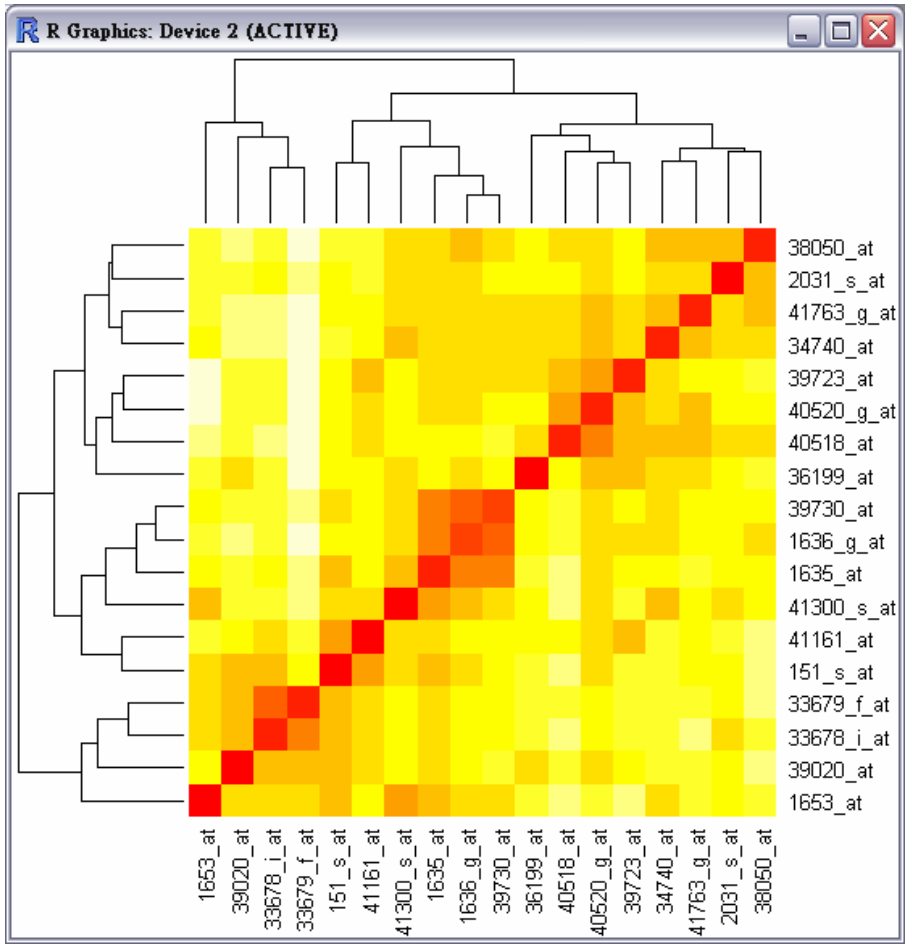
Try to compute the distances between the rows (genes).

# Distance: Visualization

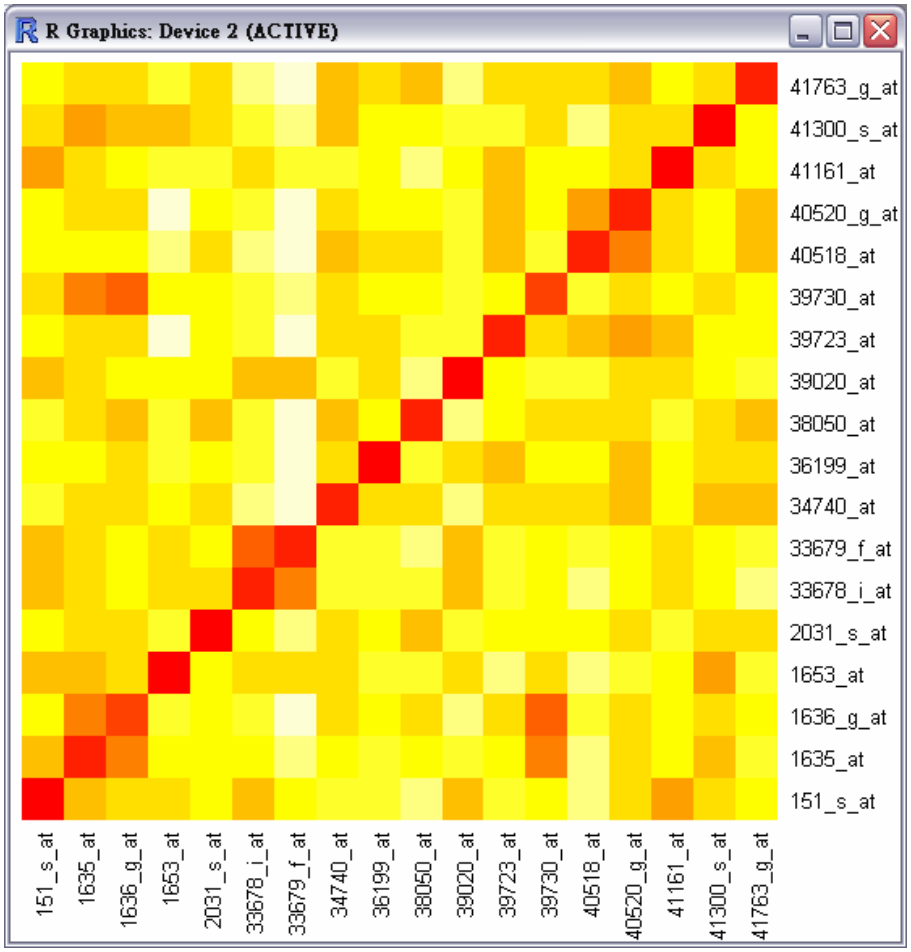
```
man = dist(Apop,"manhattan")

heatmap(as.matrix(man))

heatmap(as.matrix(man),Rowv=NA,Colv=NA)
heatmap(as.matrix(man),Rowv=NA,Colv=NA,symm=T)

library(gplots)
heatmap.2(as.matrix(man),dendrogram="none",keysize=1.5,
      Rowv=F,Colv=F,
      trace="none",density.info="none")
```



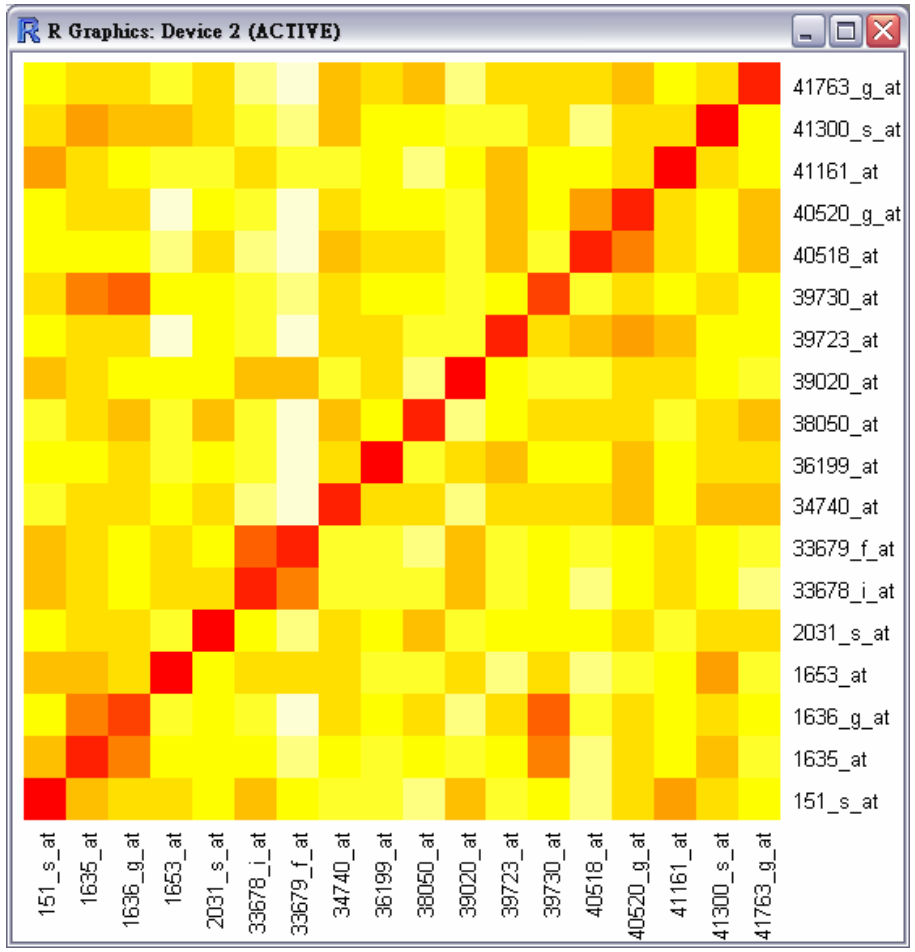
1


R Graphics: Device 2 (ACTIVE)
38050_at
2031_s_at
41763_g_at
34740_at
39723_at
40520_g_at
40518_at
36199_at
39730_at
1636_g_at
1635_at
41300_s_at
41161_at
151_s_at
33679_f_at
33678_i_at
39020_at
1653_at
1653_at
39020_at
33678_i_at
33679_f_at
151_s_at
41161_at
41300_s_at
1635_at
1636_g_at
39730_at
36199_at
40518_at
40520_g_at
39723_at
34740_at
41763_g_at
2031_s_at
38050_at

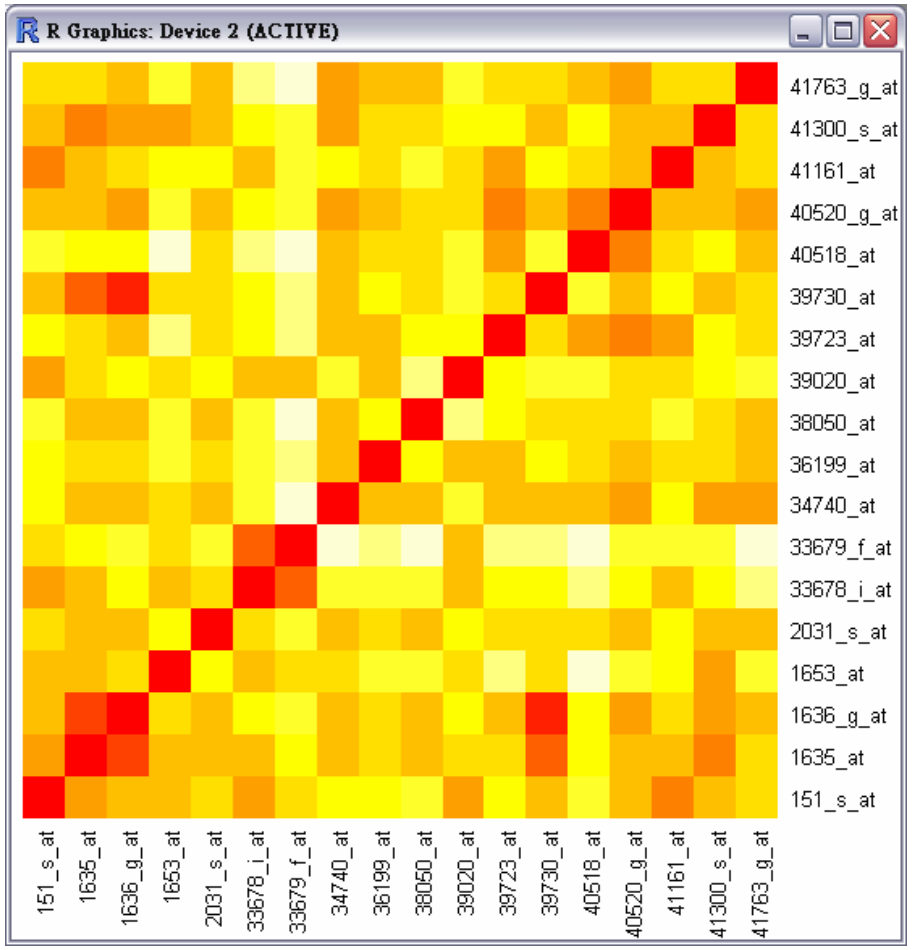
2


R Graphics: Device 2 (ACTIVE)
41763_g_at
41300_s_at
41161_at
40520_g_at
40518_at
39730_at
39723_at
39020_at
38050_at
36199_at
34740_at
33679_f_at
33678_i_at
2031_s_at
1653_at
1636_g_at
1635_at
151_s_at
151_s_at
1635_at
1636_g_at
1653_at
2031_s_at
33678_i_at
33679_f_at
34740_at
36199_at
38050_at
39020_at
39723_at
39730_at
40518_at
40520_g_at
41161_at
41300_s_at
41763_g_at

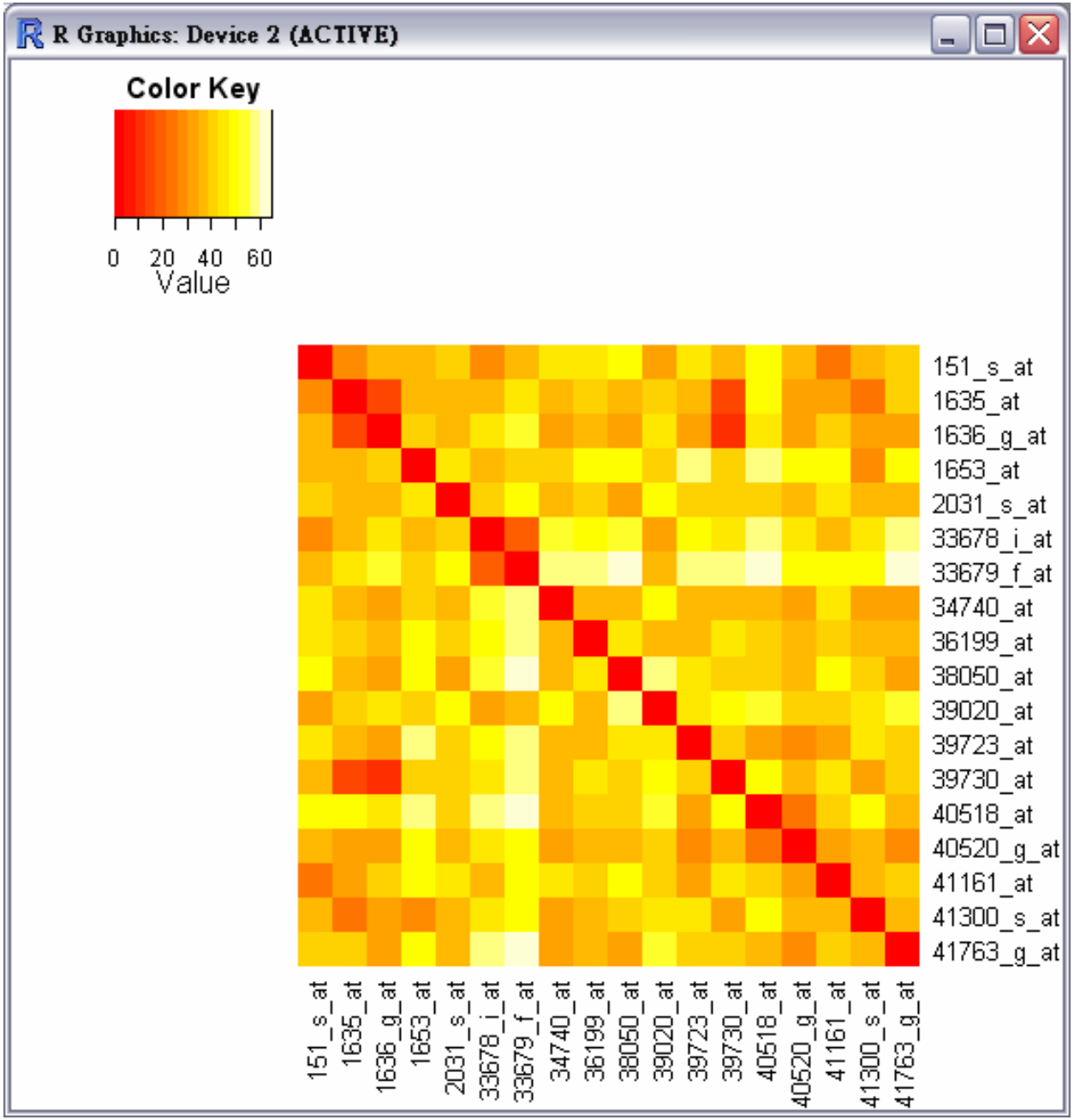
2


R Graphics: Device 2 (ACTIVE)
41763_g_at
41300_s_at
41161_at
40520_g_at
40518_at
39730_at
39723_at
39020_at
38050_at
36199_at
34740_at
33679_f_at
33678_i_at
2031_s_at
1653_at
1636_g_at
1635_at
151_s_at
151_s_at
1635_at
1636_g_at
1653_at
2031_s_at
33678_i_at
33679_f_at
34740_at
36199_at
38050_at
39020_at
39723_at
39730_at
40518_at
40520_g_at
41161_at
41300_s_at
41763_g_at

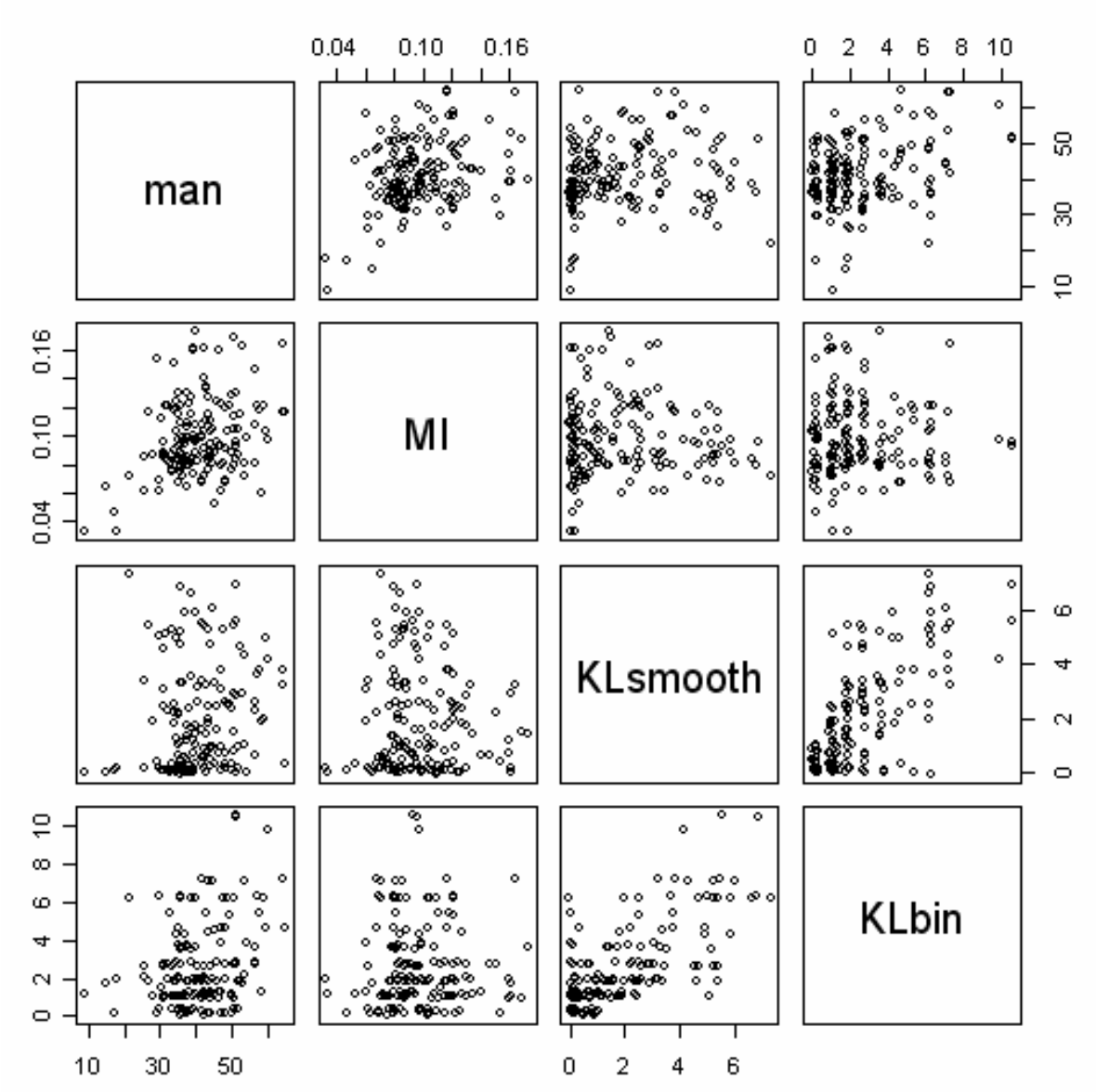
3


R Graphics: Device 2 (ACTIVE)
41763_g_at
41300_s_at
41161_at
40520_g_at
40518_at
39730_at
39723_at
39020_at
38050_at
36199_at
34740_at
33679_f_at
33678_i_at
2031_s_at
1653_at
1636_g_at
1635_at
151_s_at
151_s_at
1635_at
1636_g_at
1653_at
2031_s_at
33678_i_at
33679_f_at
34740_at
36199_at
38050_at
39020_at
39723_at
39730_at
40518_at
40520_g_at
41161_at
41300_s_at
41763_g_at

4


R Graphics: Device 2 (ACTIVE)
Color Key
0 20 40 60
Value
151_s_at
1635_at
1636_g_at
1653_at
2031_s_at
33678_i_at
33679_f_at
34740_at
36199_at
38050_at
39020_at
39723_at
39730_at
40518_at
40520_g_at
41161_at
41300_s_at
41763_g_at

pairs(cbind(man,MI,KLsmooth,KLbin))



# Cluster Analysis

- Clustering is the process of grouping together similar entities.
  - It is appropriate when there is no prior knowledge about the data.
  - In a machine learning framework, it is known as unsupervised learning since there is no known desired answer for any particular *gene* or *experiment*.

# Cluster Analysis

- The entities that are similar to each other are grouping together and form a cluster.
  - Step 1: Defining the similarity between entities
    → distance metric
  - Step 2: Forming clusters
    → clustering algorithms

# Measure of Distance

- Distance metric between points:
  - Euclidean distance (EUC)
  - Manhattan distance (MAN)
  - Pearson sample correlation (COR)
  - Angle distance (EISEN – considered by Eisen et al., 1998.)
  - Spearman sample correlation (SPEAR)
  - Kandall's $\tau$ sample correlation (TAU)
  - Mahalanobis distance
- Distance metric between distributions:
  - Kullback-Leibler information
  - Hamming's mutual information

# Cluster Analysis

- The entities that are similar to each other are grouping together and form a cluster.
  - Step 1: Defining the similarity between entities
    → distance metric
  - Step 2: Forming clusters
    → clustering algorithms

# Clustering

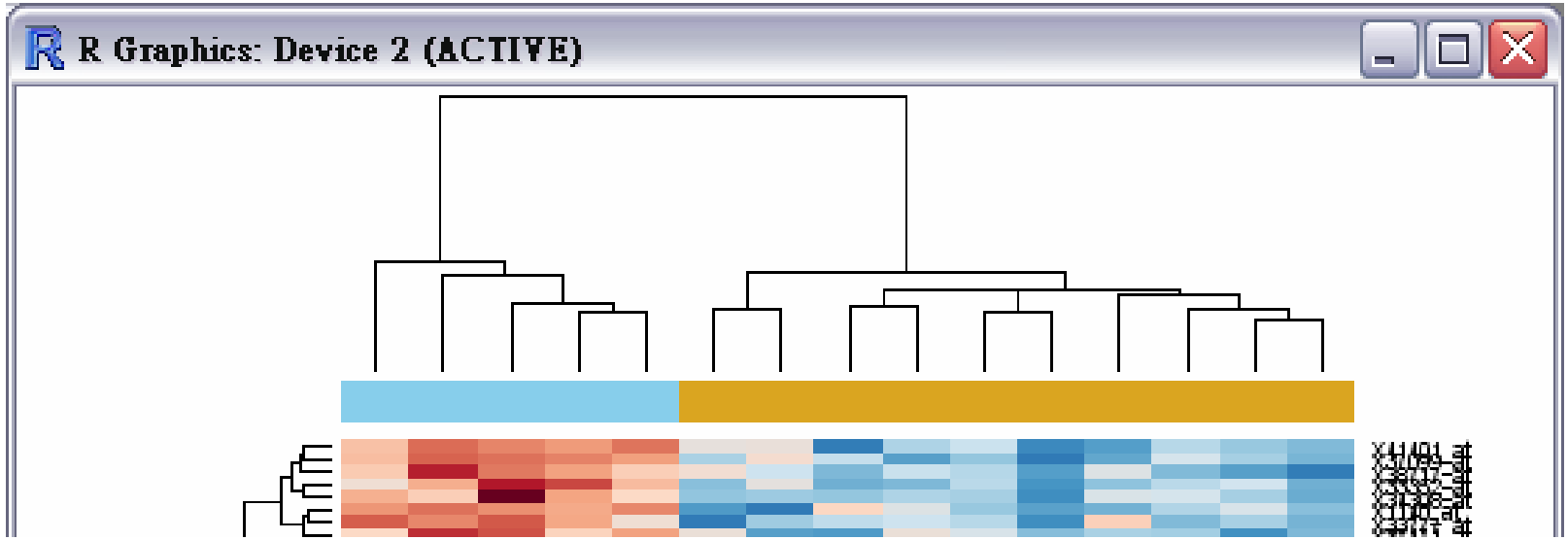
- According to distance between two objects, the entities that are closer to each other are grouping together and form a cluster.

  → clustering algorithms

Note: Anything can be clustered. The clustering results may not be related to any biological meanings between the members of a given cluster.

# Clustering

- Usually the results of clustering is shown in a clustering tree, or a dendrogram.



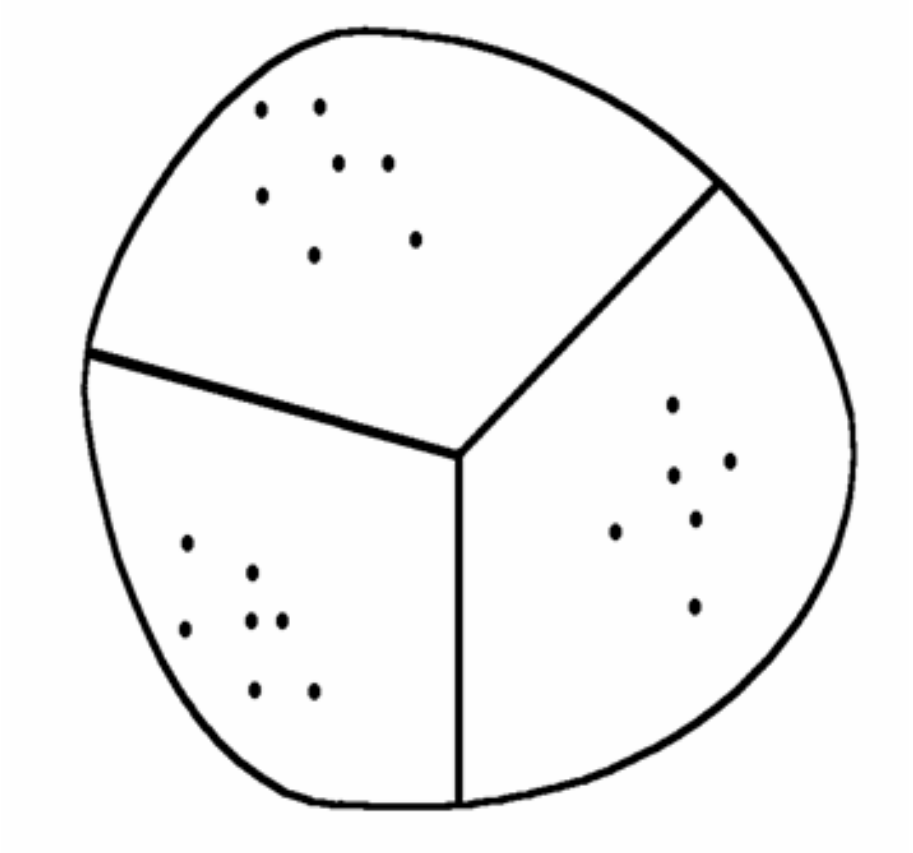
# Clustering Algorithm

- Partitioning: k-means, PAM
- Hierarchical clustering
- Model based: SOM

# Partitioning Algorithms

- Partitioning method: Construct a partition of ***n*** objects into a set of ***k*** clusters

k = 3



# Partitioning Algorithms

- Given a *k*, find a partition of *k clusters* that optimizes the chosen partitioning criterion
  - *k-means*: Each cluster is represented by the center of the cluster
  - *k-medoids* or PAM (Partition around medoids): Each cluster is represented by one of the objects in the cluster

# *K*-means Clustering

Step 1: Determine the number of clusters, *k*.

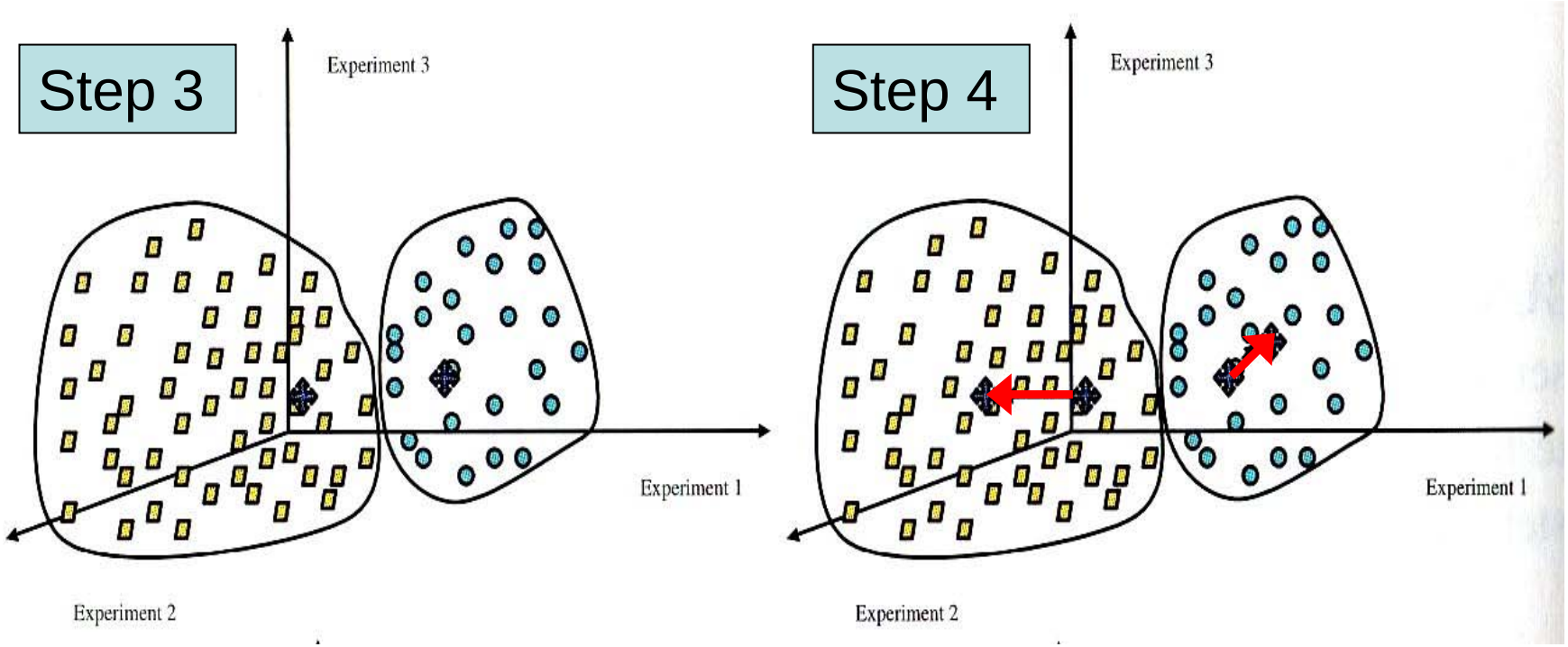
Step 2: Randomly choose *k* point as the centers of clusters.

Step 3: Calculate the distance from each pattern to *k* centers and associate every object with the closest cluster center.

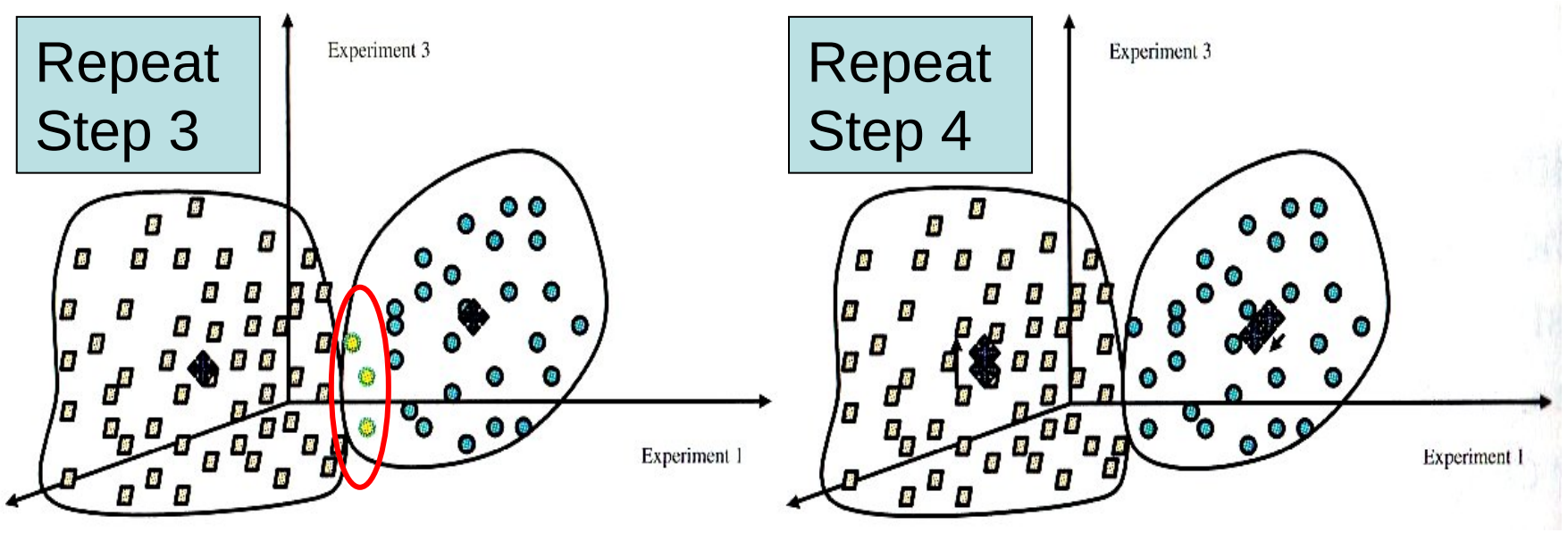
Step 4: Calculate a new center for the updated clusters.

Step 5: Repeat steps 3 and 4 until no objects are relocated.

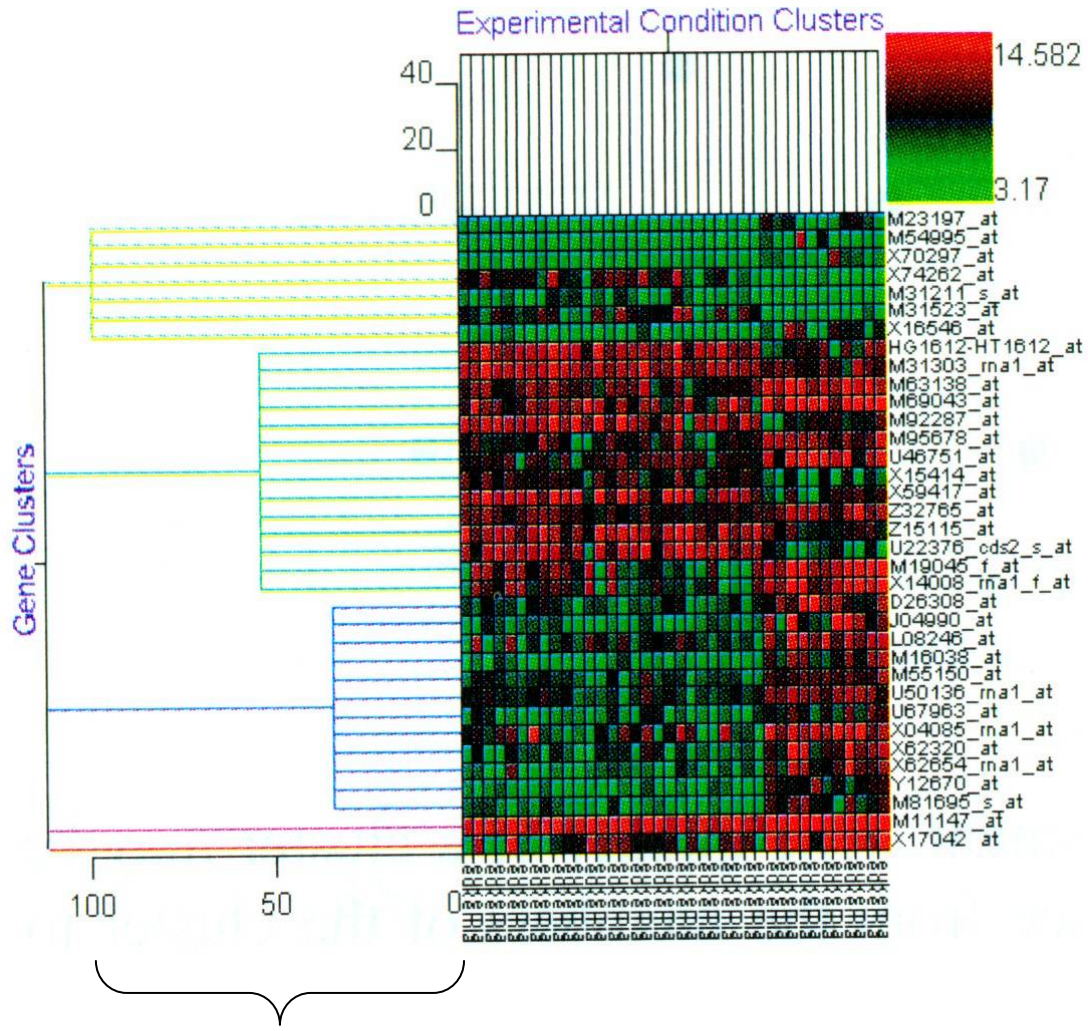
# *K*-means Clustering Example: $k = 2$



# *K*-means Clustering Example: *k* = 2



# Example of *K*-means Clustering Result




Average distance to the center of clusters

# *k*-mean Clustering: Properties

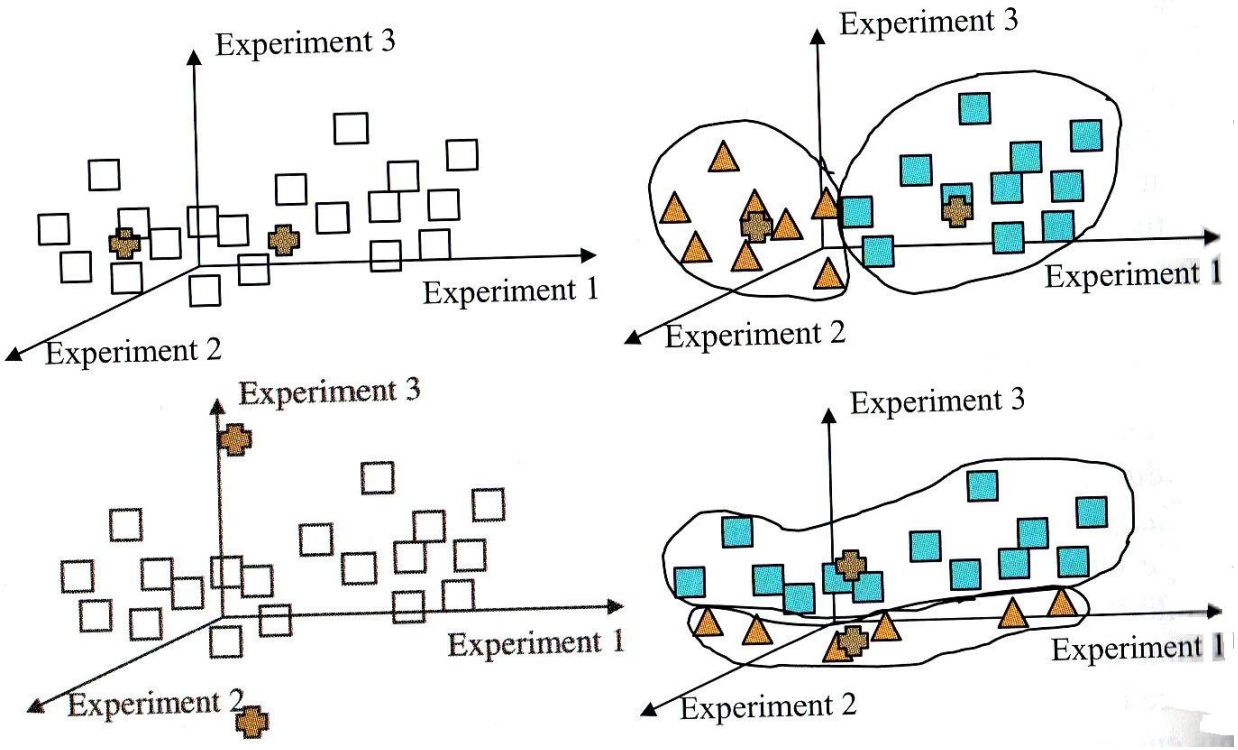
1. It is possible to produce empty clusters. To avoid such situation, one can:

   (i) let the starting cluster centers be in the general area populated by the given data.

   (ii) randomly choose $k$ points as initial centers.

# *k*-mean Clustering: Properties

2. The results of the algorithm can change between successive runs of the algorithm.



# PAM

- *PAM* (Partitioning Around Medoids):
  - starts from an initial set of medoids (objects)
  - iteratively replaces one of the medoids by one of the non-medoids if it improves the total distance of the resulting clustering
  - provides a novel graphical display, the silhouette plot, which allows the user to select the optimal number of clusters.
  - works effectively for small data sets, but does not scale well for large data sets

# PAM

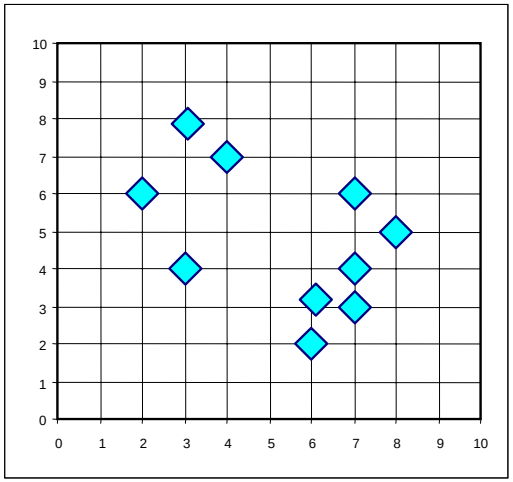
Step 1: Select ***k*** representative objects arbitrarily.

Step 2: For each pair of non-selected object ***h*** and selected object ***i***, calculate the total swapping cost $TC_{ih}$

- If $TC_{ih} < 0$, ***i*** is replaced by ***h***
- Then assign each non-selected object to the most similar representative object

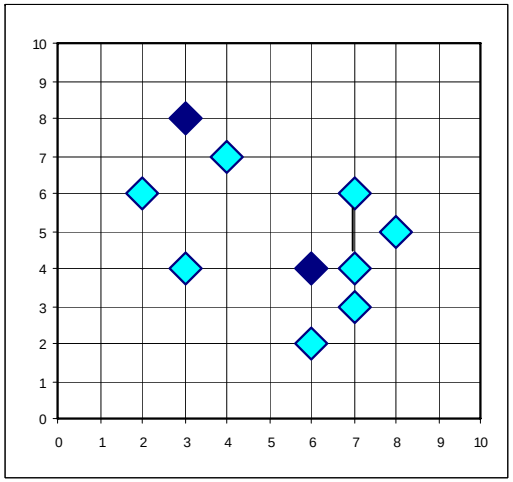
Step 3: Repeat Step 2 until there is no change.

# PAM



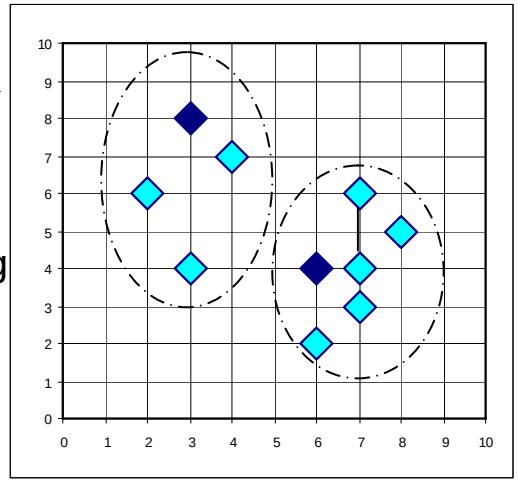
K=2

Arbitrary choose k object as initial medoids

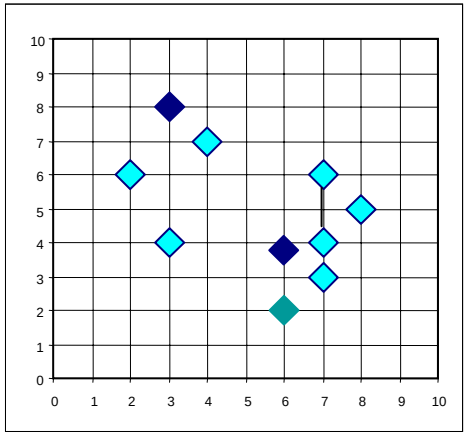


Assign each remaining object to nearest medoids

Total Cost = 20

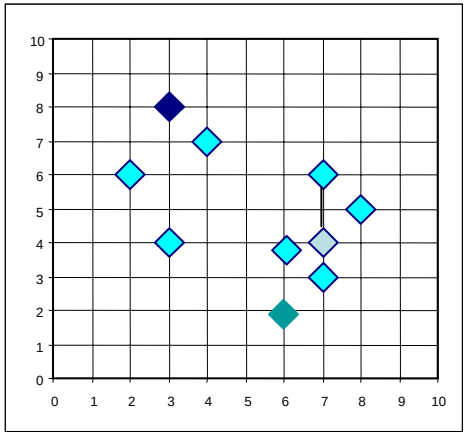


Randomly select a nonmedoid object,$O_{ramdom}$



Compute total cost of swapping

Total Cost = 26



Swapping O and $O_{ramdom}$

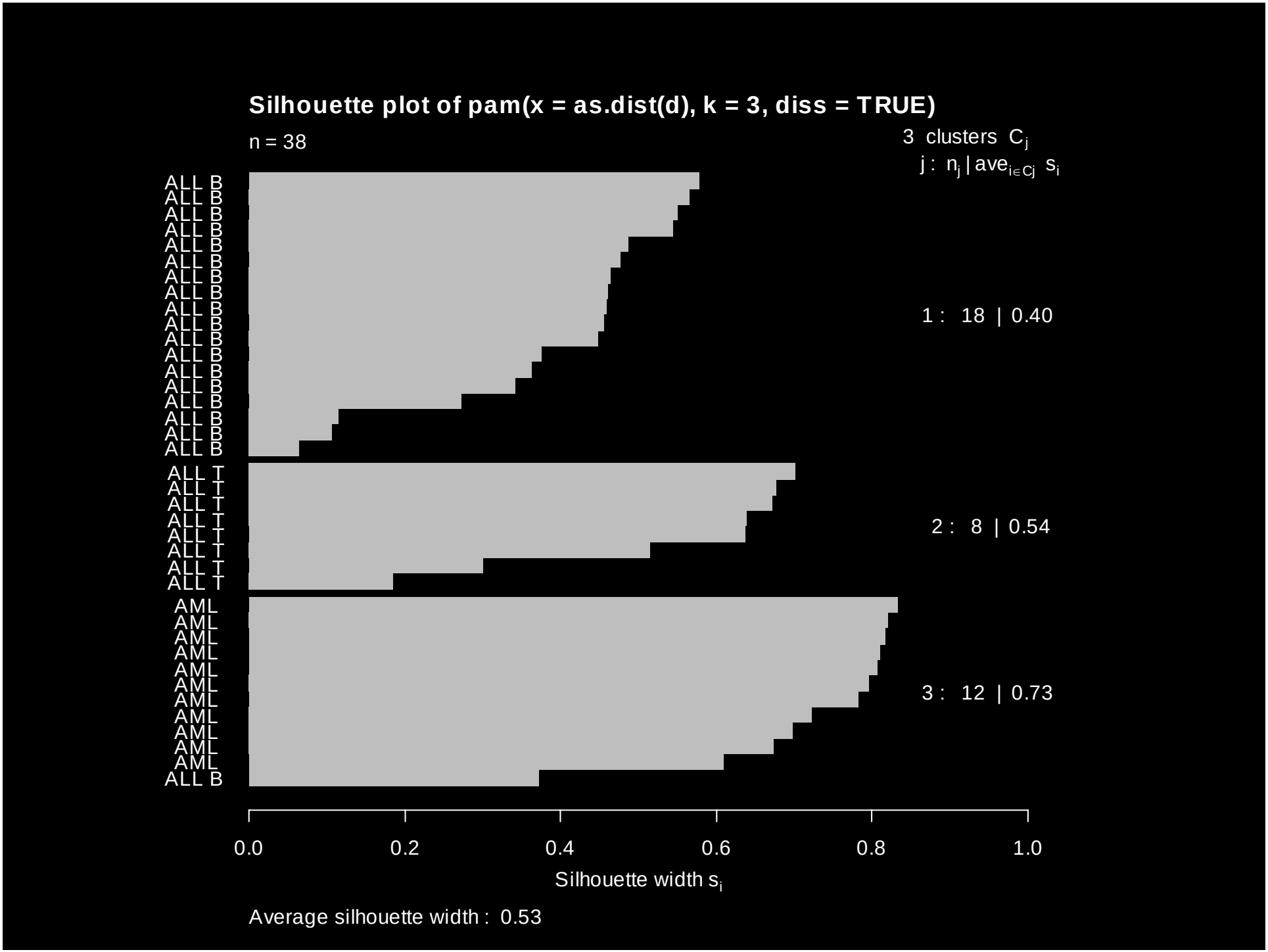
If quality is improved.

**Do loop**

**Until no change**

# PAM

- the next plot is called a silhouette plot
- each observation is represented by a horizontal bar
- the groups are slightly separated
- the length of a bar is a measure of how close the observation is to its assigned group (versus the others)


Silhouette plot of pam(x = as.dist(d), k = 3, diss = TRUE)
n = 38
3 clusters $C_j$
j : $n_j$ | ave$_{i \in Cj}$ $s_i$
ALL B
ALL B
ALL B
ALL B
ALL B
ALL B
ALL B
ALL B
ALL B
ALL B
ALL B
ALL B
ALL B
ALL B
ALL B
ALL B
ALL B
ALL T
ALL T
ALL T
ALL T
ALL T
ALL T
ALL T
ALL T
AML
AML
AML
AML
AML
AML
AML
AML
AML
AML
AML
AML
ALL B
1 : 18 | 0.40
2 : 8 | 0.54
3 : 12 | 0.73
0.0
0.2
0.4
0.6
0.8
1.0
Silhouette width $s_i$
Average silhouette width : 0.53

# Partitioning Methods: Comment

- Number of clusters, *k*:
  - If there are features that clearly distinguish between the classes (e.g. cancer and healthy), the algorithm might use them to construct meaningful clusters.
  - If the analysis has an exploratory character, one could repeat the clustering for several values of *k*.

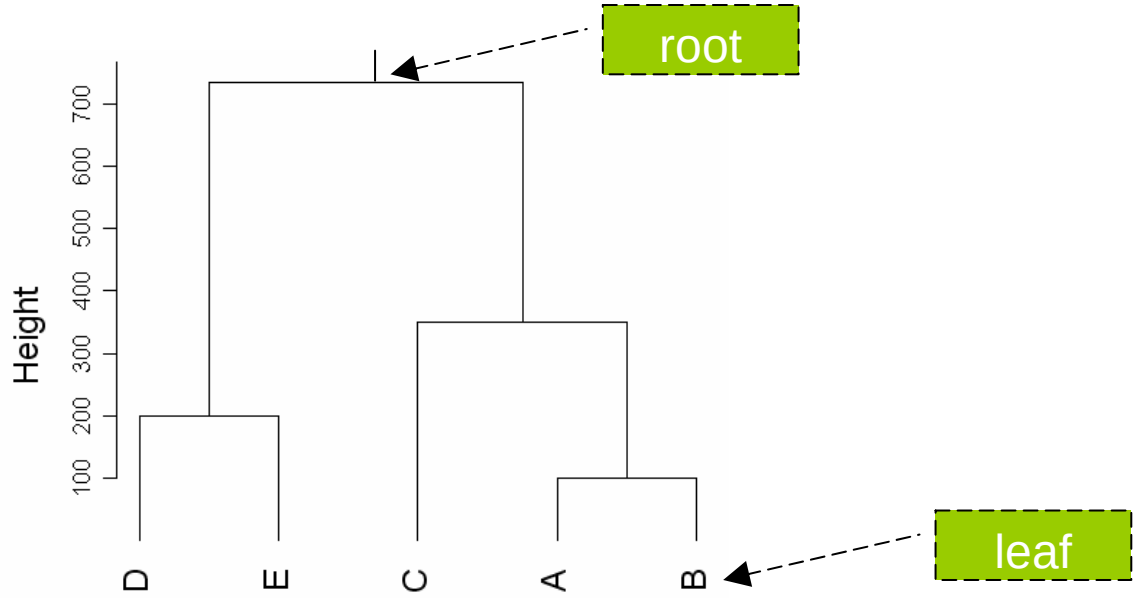
# Clustering Algorithm

- Partitioning: k-means, PAM
- Hierarchical clustering
- Model based: SOM

# Hierarchical Clustering

- *k*-means clustering returns a set of *k* clusters.
- Hierarchical clustering returns a complete tree with individual patterns as leaves and the convergence points of all branches as the root.

complete tree



# Hierarchical Clustering

Step 1: Choose one distance measurement

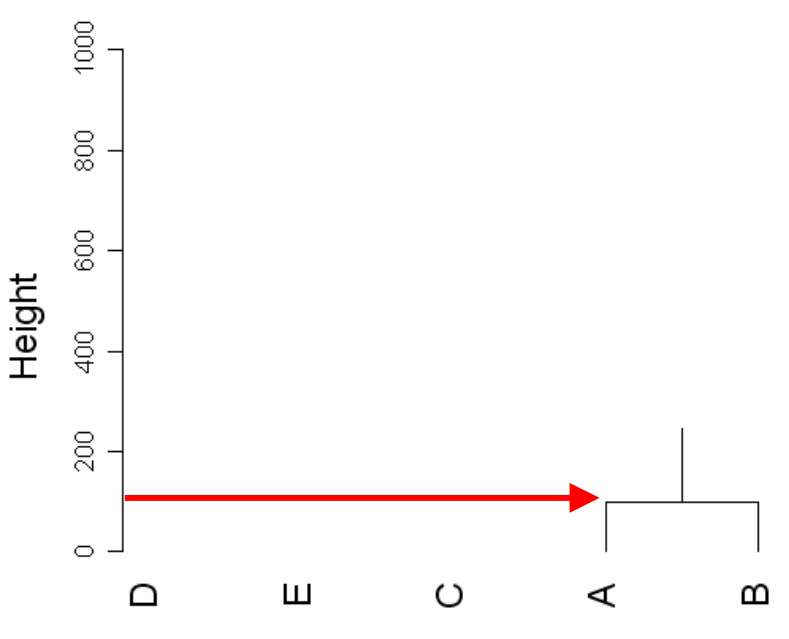
Step 2: Construct the hierarchical tree:

- Bottom-up (agglomerative) method: $n \rightarrow 1$; starting from the individual patterns and putting smaller clusters together to form bigger clusters.
- Top-down (divisive) method: $1 \rightarrow n$; starting at the root and splitting clusters into smaller ones by non-hierarchical algorithms (e.g., *k*-means with k = 2).

# Hierarchical Clustering: Example

- Example: Consider 5 experiments (A, B, C, D, E) with the following distance metric:

Bottom-up (agglomerative) method: putting similar clusters together to form bigger clusters.



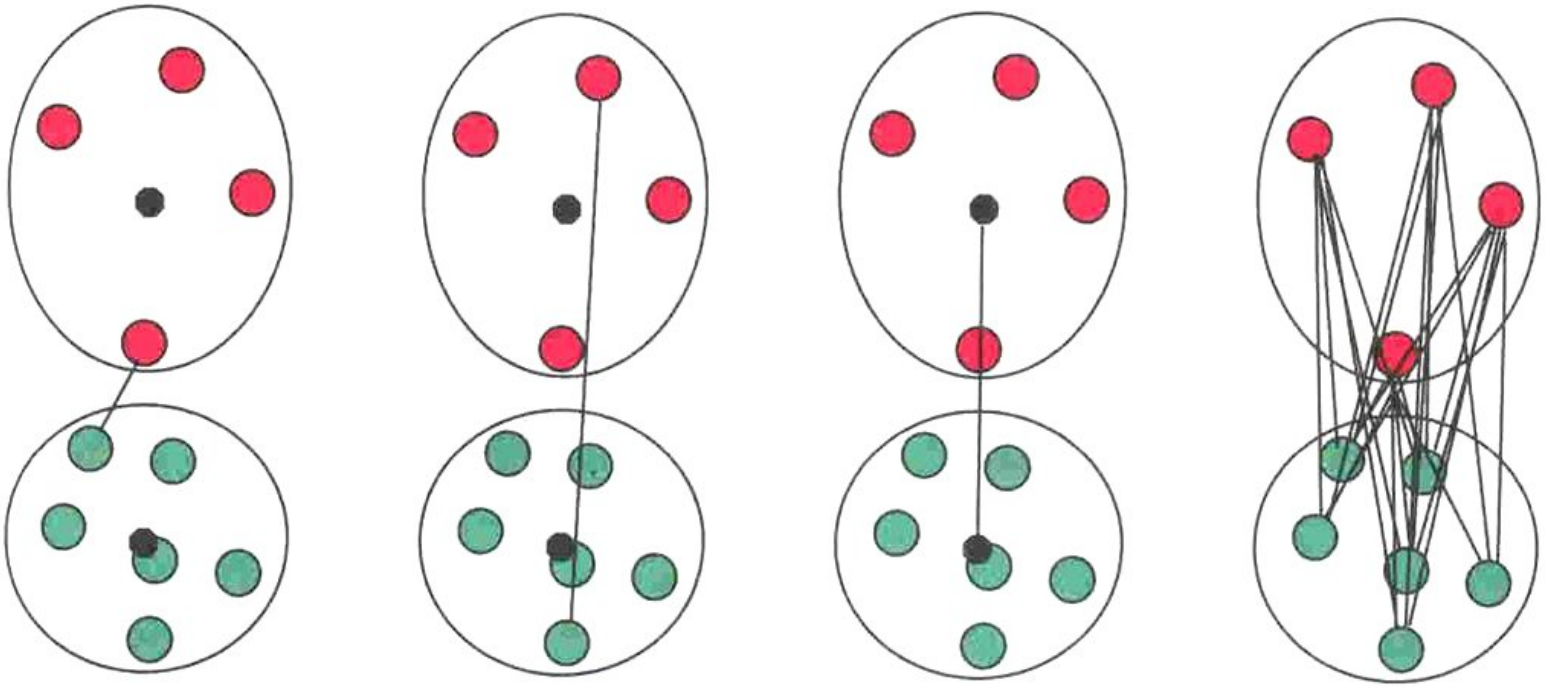

| | A | B | C | D | E |
|---|---|---|---|---|---|
| A | – | | | | |
| B | 100 | – | | | |
| C | 400 | 300 | – | | |
| D | 800 | 700 | 400 | – | |
| E | 1000 | 900 | 600 | 200 | – |

# Hierarchical Clustering: Example

| | {A,B} | C | D | E |
|---|---|---|---|---|
| {A,B} | – | | | |
| C | ? | – | | |
| D | ? | 400 | – | |
| E | ? | 600 | 200 | – |

$\Rightarrow$ Need to define inter-cluster distances

# Inter-Cluster Distances



Single Linkage

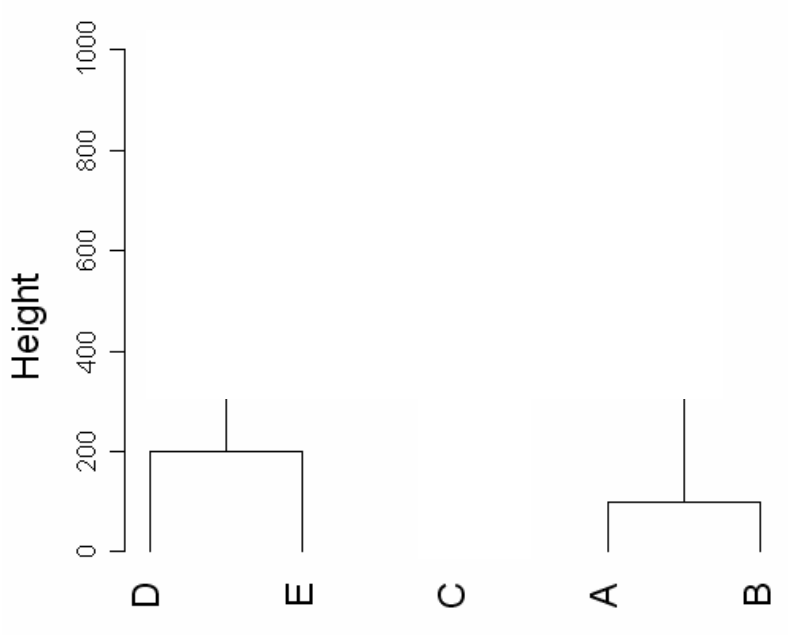
Complete Linkage

Centroid Linkage

Average Linkage

# Hierarchical Clustering: Example

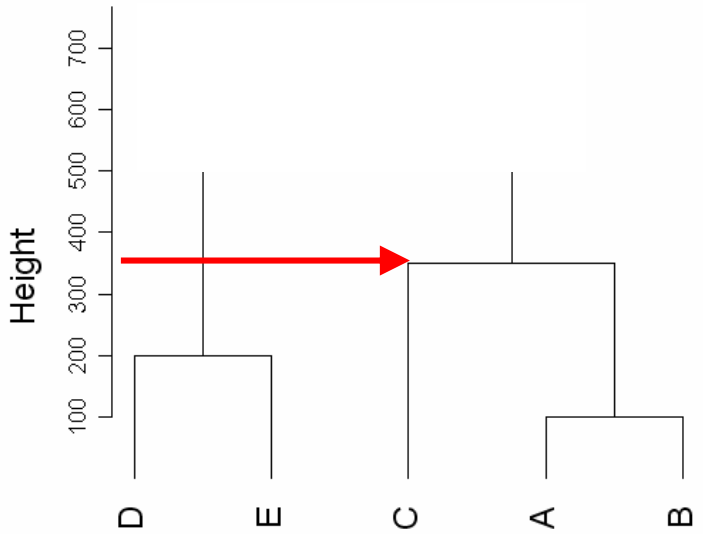
If we use average linkage: $d_{\{A,B\},C} = (d_{A,C} + d_{B,C}) / 2$, etc.




$$
\begin{array}{c} \\ \{A,B\} \\ C \\ D \\ E \end{array}
\begin{array}{c}
\begin{array}{cccc} \{A,B\} & C & D & E \end{array} \\
\begin{pmatrix}
- & & & \\
350 & - & & \\
750 & 400 & - & \\
950 & 600 & 200 & -
\end{pmatrix}
\end{array}
$$

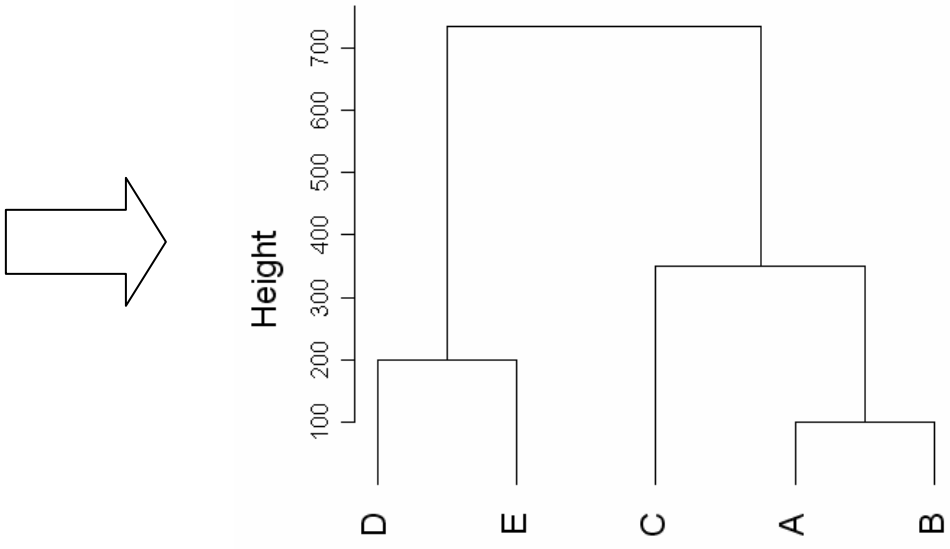
$d_{\{A,B\},C} = (d_{A,C} + d_{B,C}) / 2 = (400+300)/2$

# Hierarchical Clustering: Example



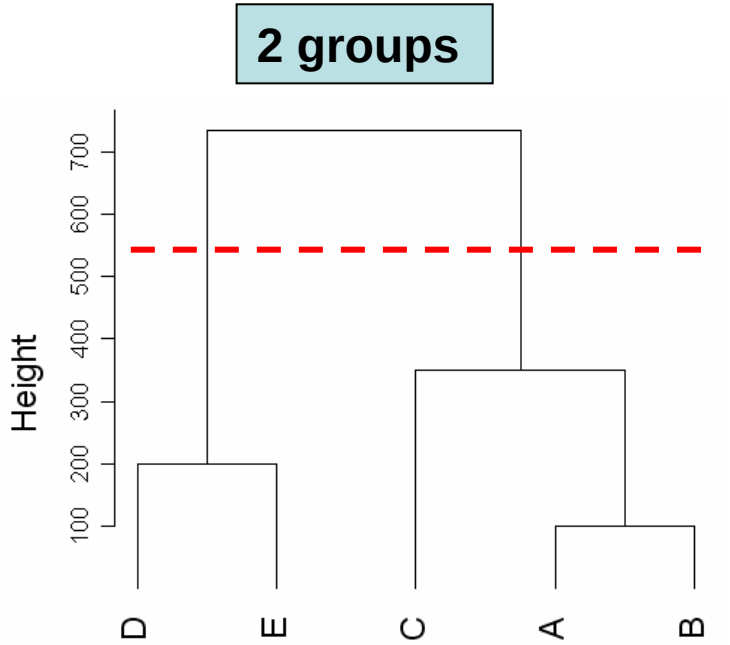

* use average linkage

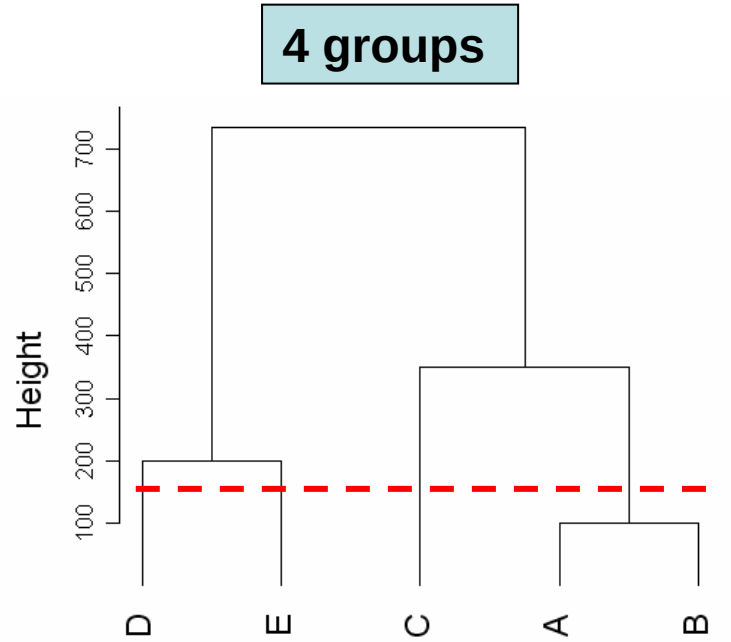
| | {A,B} | C | {D,E} |
|---|---|---|---|
| {A,B} | – | | |
| C | 350 | – | |
| {D,E} | 850 | 500 | – |



# Cutting Tree Diagrams

A hierarchical clustering diagram can be used to divide the data into a pre-determined number of clusters by cutting the tree at a certain depth.



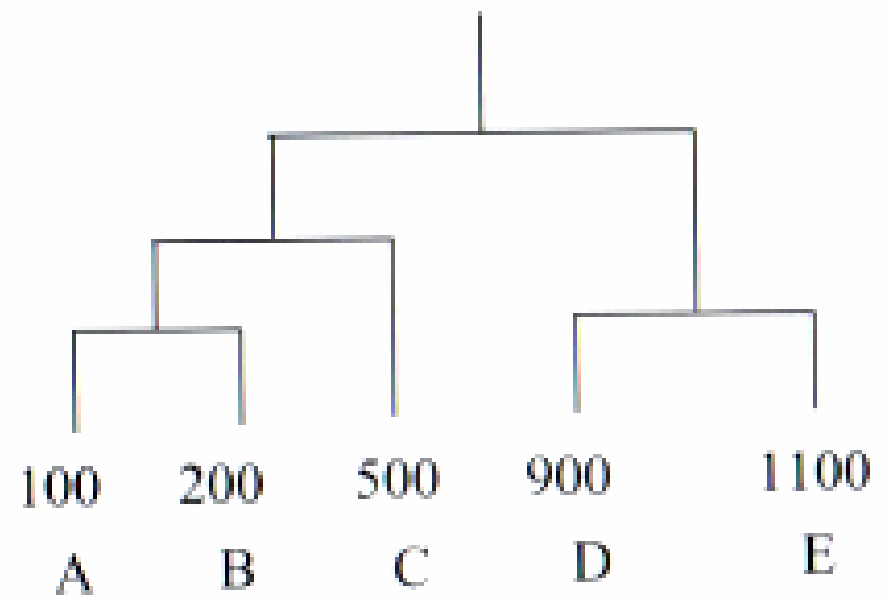
# Properties of Hierarchical Clustering

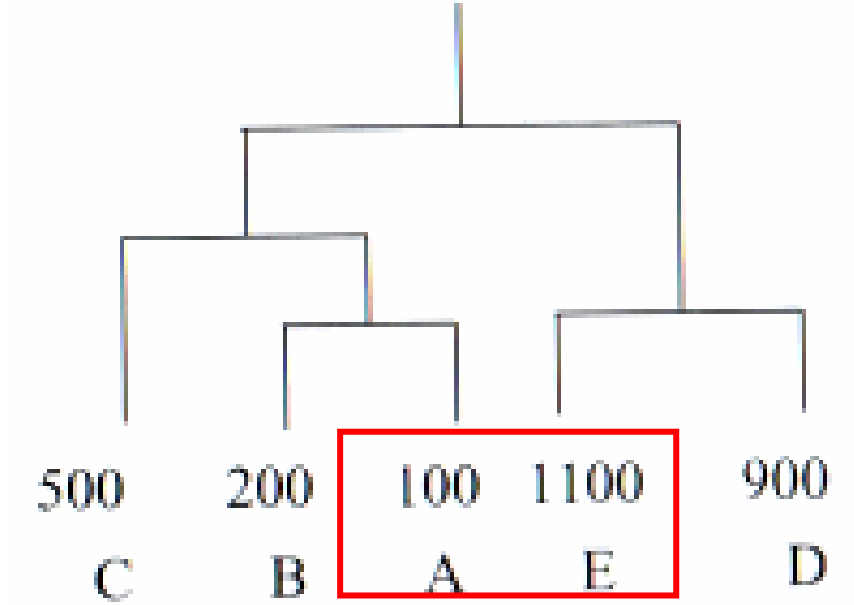
- Different tree-constructing methods:
  - The same data and the same process obtain the same results by running the same bottom-up method.
  - The same data and the same process obtain two different results by running the same top-down method.
- Different linkage type produce different results.

# Hierarchical Clustering: Comments

- Objective of the research: To obtain a clustering that reflects the structure of the data. The dendrogram itself is almost never the answer to the research question.
- Various implementations of hierarchical clustering should not be judged simply by their speed; slower algorithms may be trying to do a better job pf extracting the data features.
- The order of the objects and clusters in the dendrogram may be misleading.

# Orders in Dendrogram



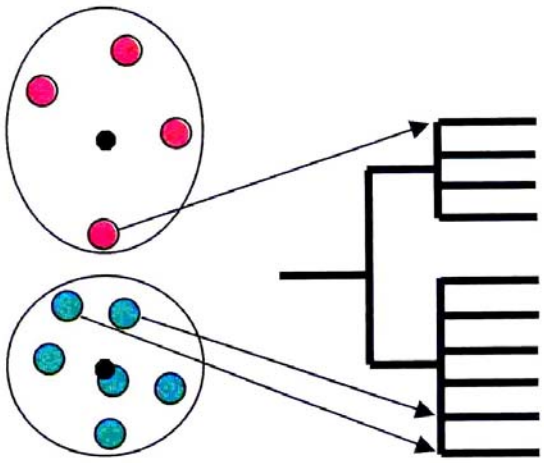
# Clustering Algorithm

- Partitioning: k-means, PAM
- Hierarchical clustering
- Model based: SOM

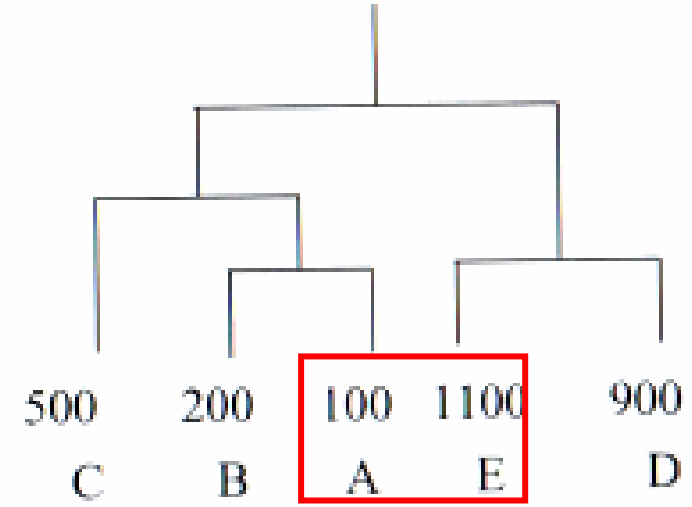
# SOM: Motivation

- Misleading dendrograms:

*K*-means



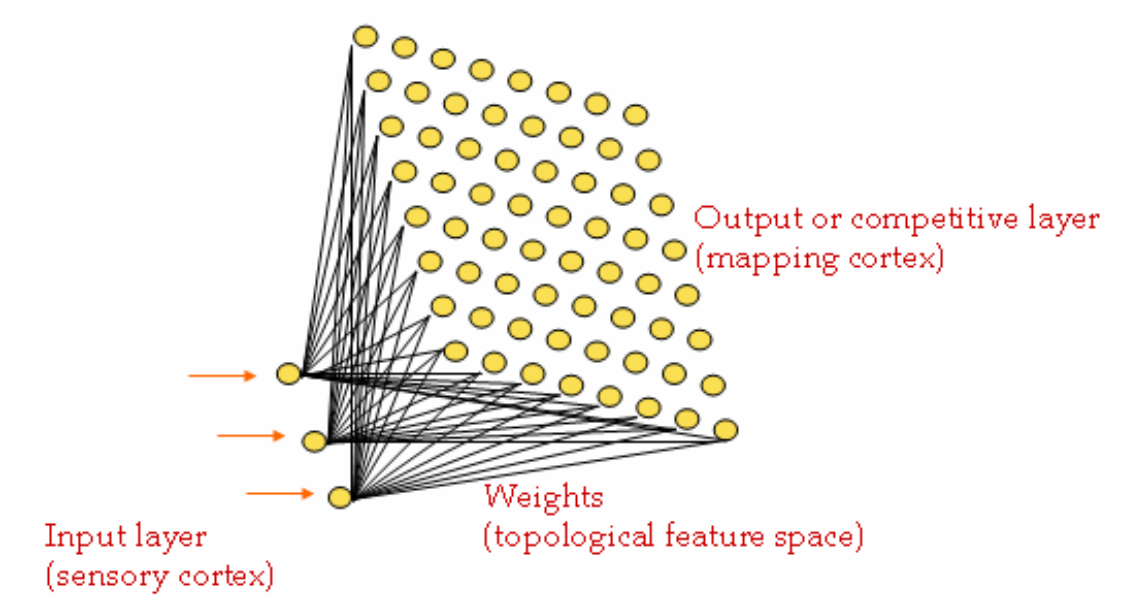
Hierarchical




- The SOM clustering is designed to create a plot in which similar patterns are plotted next to each other.

# Self-Organizing Feature Maps (SOM)

- SOM: A map consists of many simple elements (nodes or neurons); it is constructed by training.
  - SOMs are believed to resemble processing that can occur in the brain
  - Useful for visualizing high-dimensional data in 2- or 3-D space

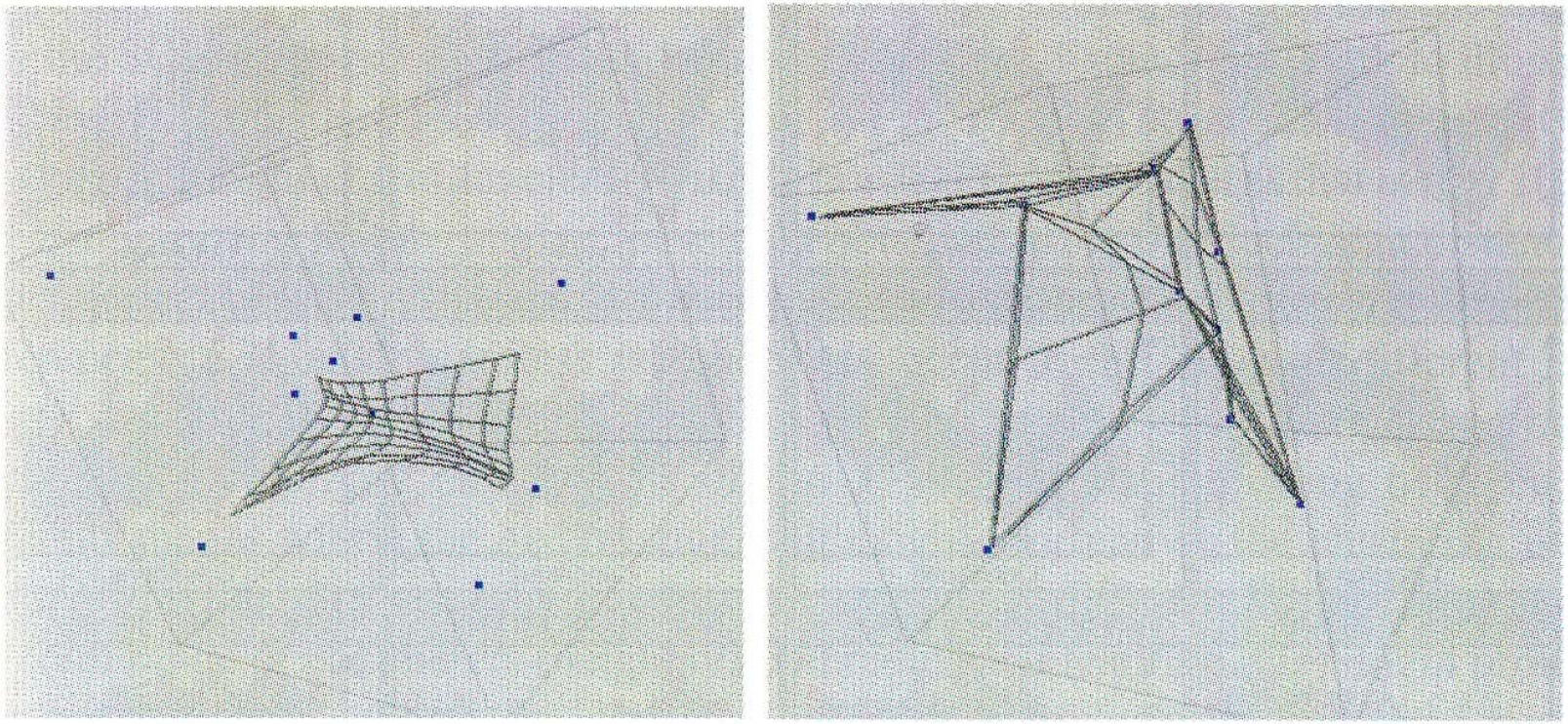


# Self-Organizing Feature Maps (SOM)

- Clustering is performed by having several units competing for the current object
- The unit whose weight vector is closest to the current object wins
- The winner and its neighbors learn by having their weights adjusted

# Self-Organizing Feature Maps (SOM)

- This process can be visualized by imagining all SOM units being connected to each other by rubber bands.



A 2D SOFM trained on 3-dimensional data.

- paper:
  - Eisen 1998
  - Algorithmic Approaches to Clustering Gene Expression Data
    http://citeseer.nj.nec.com/shamir01algorithmic.html
  - Tibshirani, Hastie, Narasimhan and Chu (2002)
    http://www.pnas.org/cgi/reprint/99/10/6567
  - Rousseeuw, P.J. (1987) Silhouettes: A graphical aid to the interpretation and validation of cluster analysis.
    *J. Comput. Appl. Math.*, **20**, 53–65