Statistical Methods for High Dimensional Biology

STAT/BIOF/GSAT 540

Lecture 13 – Cluster Analysis

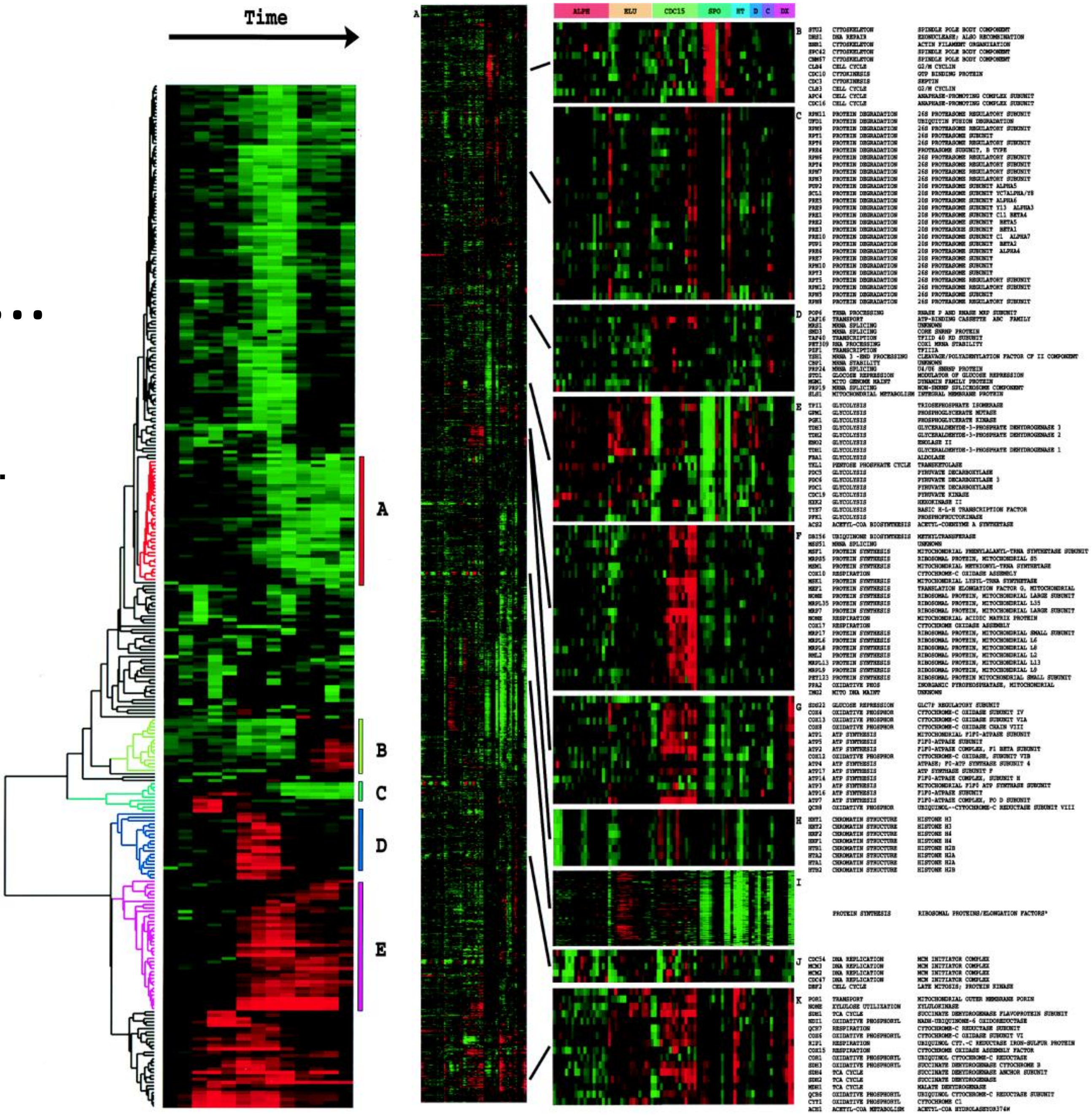
Gabriela Cohen Freue

February 24 2019

Other contributors: Drs. Jenny Bryan; Sara Mostafavi; Matias Salibian-Barrera

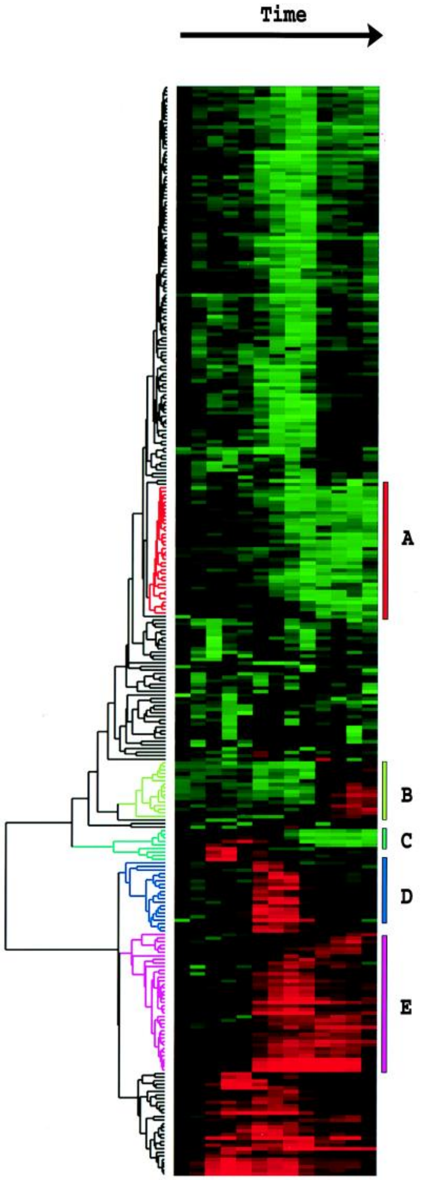
A familiar scene in an ‘omic’ paper …

Behind the scenes…
Cluster Analysis (CA)
was used to organize genes into groups (clusters) and to create dendograms.



Eisen, et al (1998)

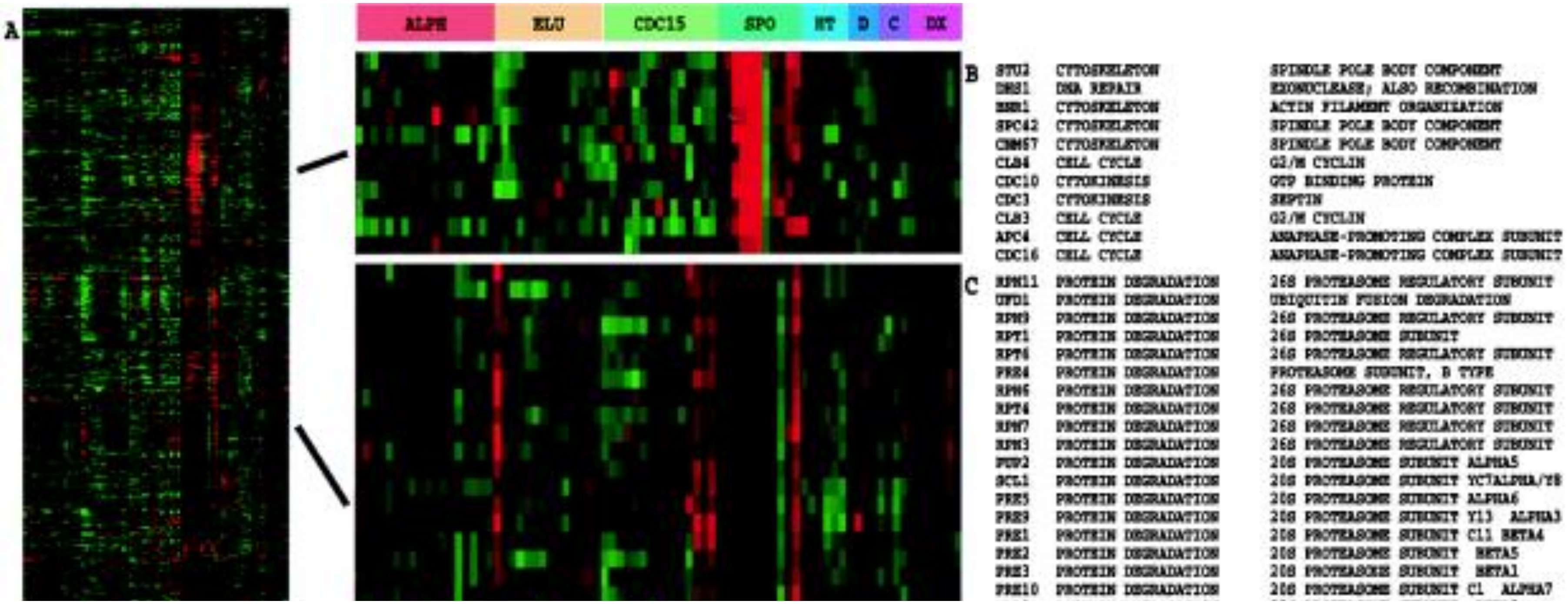
Cluster analysis in genomics



- “ Cluster analysis and display of genome-wide expression patterns ” by Eisen, et al. (PNAS, Vol. 95, pp. 14863–14868, December 1998)
- Imprinted CA on the microarray community
- This precedent + explosion of array data + ease of application = widespread (over?)use of CA
- Currently, CA is used similarly in many other –omics studies.

Utility of clustering in –omics studies

- Eisen, et al., showed that coexpression (thus, cluster co-membership) is often exhibited by genes with similar roles in cellular processes



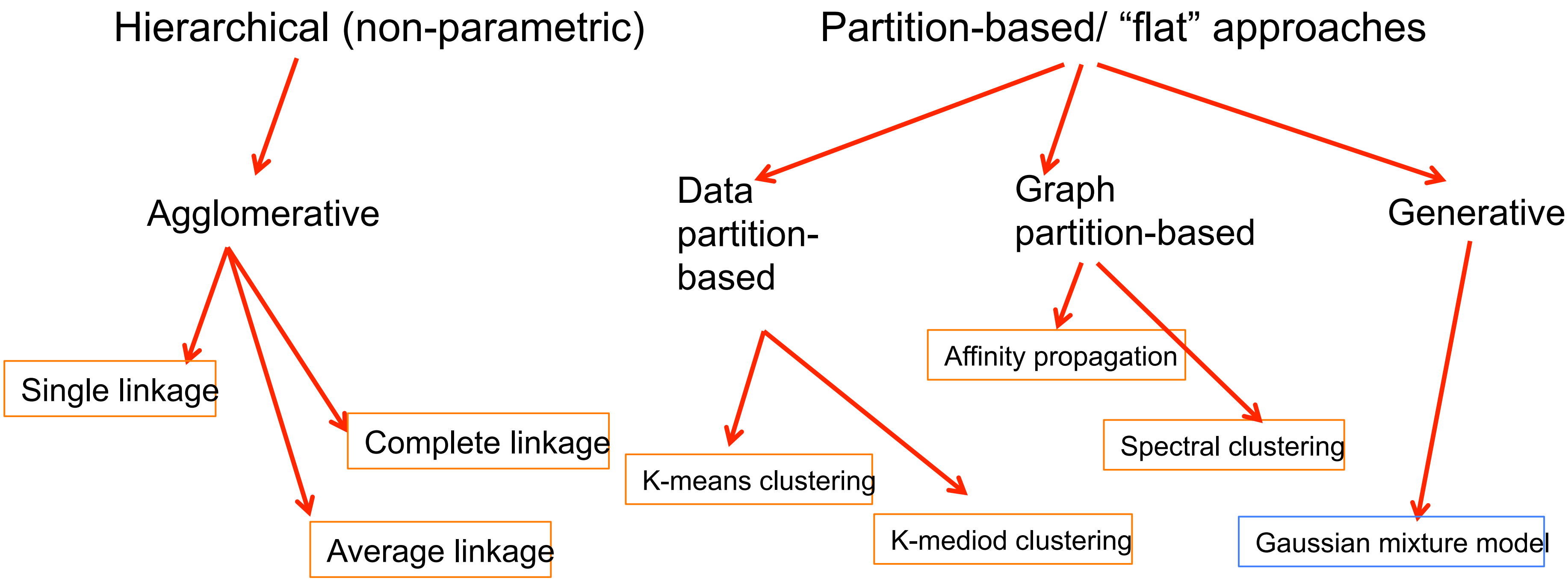
STU2	CYTOSKELETON	SPINDLE POLE BODY COMPONENT
DHS1	DNA REPAIR	EXONUCLEASE; ALSO RECOMBINATION
BNR1	CYTOSKELETON	ACTIN FILAMENT ORGANIZATION
SPC42	CYTOSKELETON	SPINDLE POLE BODY COMPONENT
CNM67	CYTOSKELETON	SPINDLE POLE BODY COMPONENT
CLB4	CELL CYCLE	G2/M CYCLIN
CDC10	CYTOKINESIS	GTP BINDING PROTEIN
CDC3	CYTOKINESIS	SEPTIN
CLB3	CELL CYCLE	G2/M CYCLIN
APC4	CELL CYCLE	ANAPHASE-PROMOTING COMPLEX SUBUNIT
CDC16	CELL CYCLE	ANAPHASE-PROMOTING COMPLEX SUBUNIT

RPN11	PROTEIN DEGRADATION	26S PROTEASOME REGULATORY SUBUNIT
UFD1	PROTEIN DEGRADATION	UBIQUITIN FUSION DEGRADATION
RPN9	PROTEIN DEGRADATION	26S PROTEASOME REGULATORY SUBUNIT
RPT1	PROTEIN DEGRADATION	26S PROTEASOME SUBUNIT
RPT6	PROTEIN DEGRADATION	26S PROTEASOME REGULATORY SUBUNIT
PRE4	PROTEIN DEGRADATION	PROTEASOME SUBUNIT, B TYPE
RPN6	PROTEIN DEGRADATION	26S PROTEASOME REGULATORY SUBUNIT
RPT4	PROTEIN DEGRADATION	26S PROTEASOME REGULATORY SUBUNIT
RPN7	PROTEIN DEGRADATION	26S PROTEASOME REGULATORY SUBUNIT
RPN3	PROTEIN DEGRADATION	26S PROTEASOME REGULATORY SUBUNIT
PUP2	PROTEIN DEGRADATION	20S PROTEASOME SUBUNIT ALPHA5
SCL1	PROTEIN DEGRADATION	20S PROTEASOME SUBUNIT YC7ALPHA/Y8
PRE5	PROTEIN DEGRADATION	20S PROTEASOME SUBUNIT ALPHA6
PRE9	PROTEIN DEGRADATION	20S PROTEASOME SUBUNIT Y13 ALPHA3
PRE1	PROTEIN DEGRADATION	20S PROTEASOME SUBUNIT C11 BETA4
PRE2	PROTEIN DEGRADATION	20S PROTEASOME SUBUNIT BETA5
PRE3	PROTEIN DEGRADATION	20S PROTEASOME SUBUNIT BETA1
PRE10	PROTEIN DEGRADATION	20S PROTEASOME SUBUNIT C1 ALPHA7

What is Clustering?

- “Clustering” Colloquially means placing/grouping a set of objects into groups/clusters.
- Clustering is a **formal problem** in Computer Science and in Statistics, with formal definitions and “solutions”.
- Clustering is often used as a tool for visualization, hypothesis generation, selection of variables for further analysis.
  - Keep in mind, with typical use of clustering: there is no measure of “strength of evidence” or “strength of clustering structure” provided.

Many clustering methods …



Hierarchical (non-parametric)

Partition-based/ “flat” approaches

Agglomerative

Data partition-based

Graph partition-based

Generative

Single linkage

Complete linkage

Average linkage

K-means clustering

K-mediod clustering

Affinity propagation

Spectral clustering

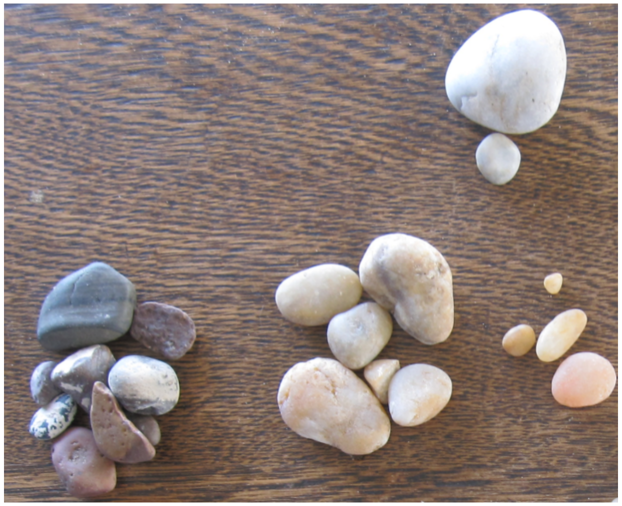
Gaussian mixture model

🟧 **Discrete clustering assignment**
🟦 **Probabilistic cluster assignment**

Clustering problem

- Goal: place a set of **objects** into groups or **clusters** in a way that **similar** objects are in the same cluster.

Cluster some rocks:

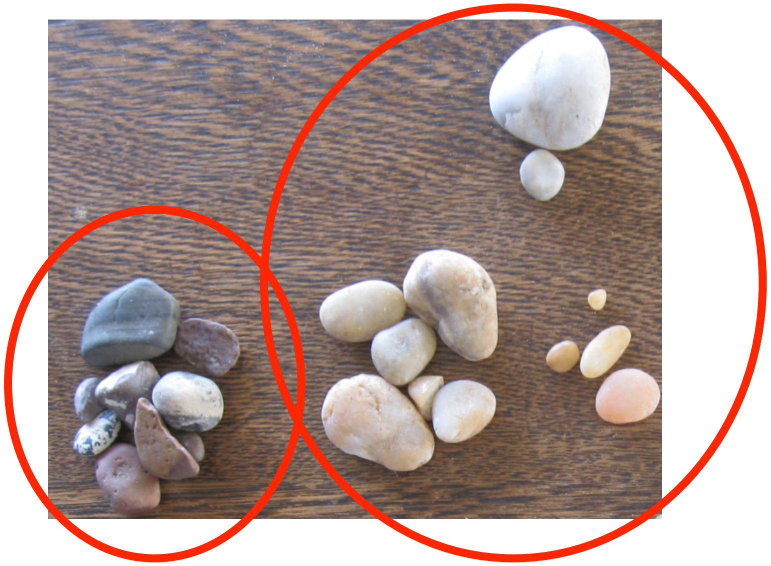


Rocks were clustered according to their color and texture.

Clustering problem (cont.)

- Goal: place a set of **objects** into groups or **clusters** in a way that **similar** objects are in the same cluster.

Cluster some rocks:

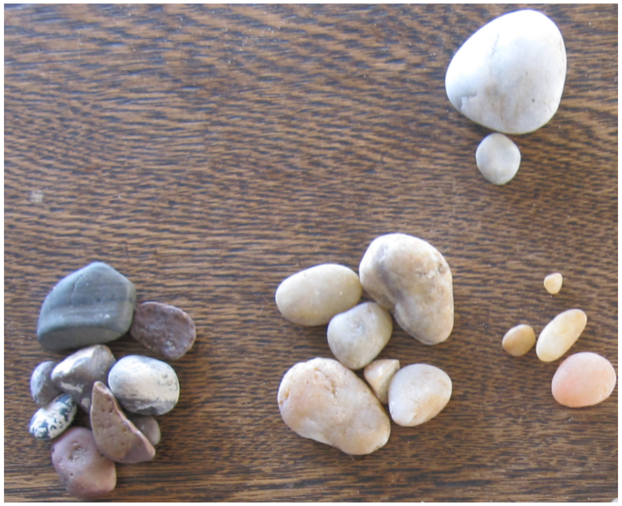


Note that you could have also considered a 2-cluster solution.

Clustering problem (cont.)

- Goal: place a set of **objects** into groups or **clusters** in a way that **similar** objects are in the same cluster.

Cluster some rocks:

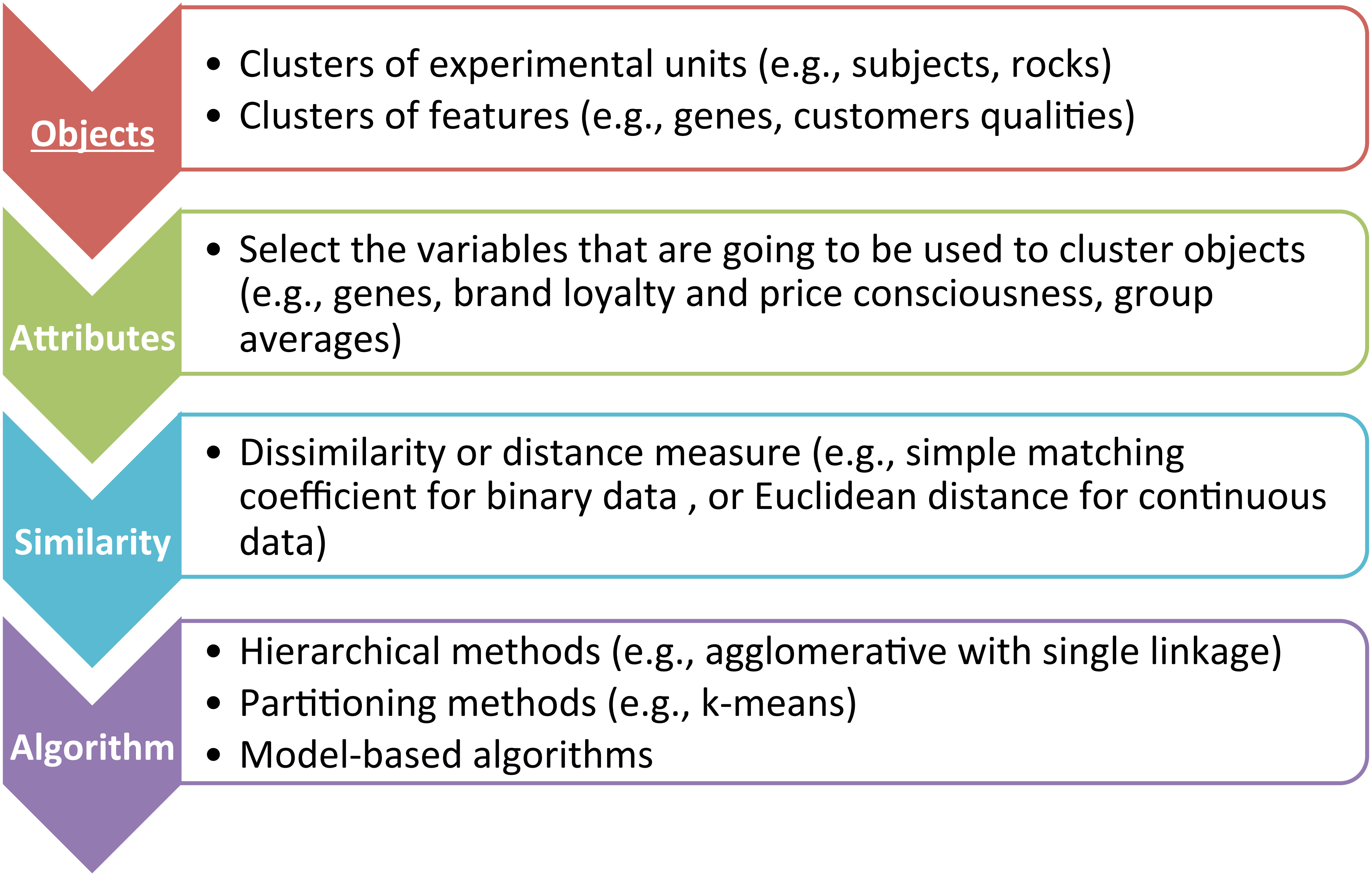


OR, we could have also clustered the rocks according to their size

Key elements

- **Goal**: place a set of objects in groups or **clusters**
- The variables used to compare objects are called the attributes (e.g., color, texture, size)
- The analysis is based on a (dis)similarity/distance measure of the attributes

Attributes of objects *within* a cluster are more 'similar' than attributes of objects among different clusters



Objects

- Clusters of experimental units (e.g., subjects, rocks)
- Clusters of features (e.g., genes, customers qualities)

Attributes

- Select the variables that are going to be used to cluster objects (e.g., genes, brand loyalty and price consciousness, group averages)

Similarity

- Dissimilarity or distance measure (e.g., simple matching coefficient for binary data , or Euclidean distance for continuous data)

Algorithm

- Hierarchical methods (e.g., agglomerative with single linkage)
- Partitioning methods (e.g., k-means)
- Model-based algorithms

Number of clusters???

Example: PhotoRec Data

- Gene expression of purified photoreceptors at distinct developmental stages and from different genetic backgrounds
- Almost 30K genes and 39 samples (mice)
- 5 developmental stages: day 16 of embryonic development (E16), postnatal days 2,6 and 10 (P2, P6 and P10) as well as 4_weeks.
- 2 genetic backgrounds: wild type Nrl mice (wt) and knockout Nrl mice (NrlKO).

A peak at the data…

	Wild Type			Knock-out			…			Wild Type			Knock-out		
	E16									4_weeks					
	Sample_20	…	Sample_23	Sample_16	…	Sample_17	…	Sample *i*	…	Sample_36	…	Sample_39	Sample_11	…	Sample_9
1415670_at	7.24	…	7.07	7.38	…	7.34	…	…	…	7.25	…	7.13	7.42	…	7.32
1415671_at	9.48	…	10.13	7.64	…	10.03	…	…	…	9.66	…	8.73	9.83	…	9.80
1415672_at	10.01	…	9.91	8.42	…	10.24	…	…	…	9.51	…	9.53	10.00	…	9.85
1415673_at	8.36	…	8.49	8.36	…	8.37	…	…	…	8.49	…	8.65	8.60	…	8.40
1415674_a_at	8.59	…	8.64	8.51	…	8.89	…	…	…	8.42	…	8.28	8.43	…	8.46
1415675_at	9.59	…	9.70	9.66	…	9.61	…	…	…	9.67	…	9.45	9.60	…	9.51
1415676_a_at	9.68	…	10.19	8.05	…	10.02	…	…	…	9.95	…	8.70	9.23	…	9.82
1415677_at	7.24	…	7.49	7.34	…	7.34	…	…	…	7.28	…	6.84	7.33	…	7.45
1415678_at	11.71	…	11.57	10.46	…	11.75	…	…	…	11.56	…	11.80	12.04	…	11.81
1415679_at	9.21	…	9.92	8.22	…	9.60	…	…	…	9.13	…	8.08	9.06	…	9.29
…	…	…	…	…	…	…	…		…	…	…	…	…	…	…
gene g	…	…	…	…	…	…	…	X_{gi}	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…		…	…	…	…	…	…	…
1460746_at	6.37	…	6.12	7.25	…	6.15	…	…	…	6.34	…	6.52	6.36	…	6.35

Column-driven analyses

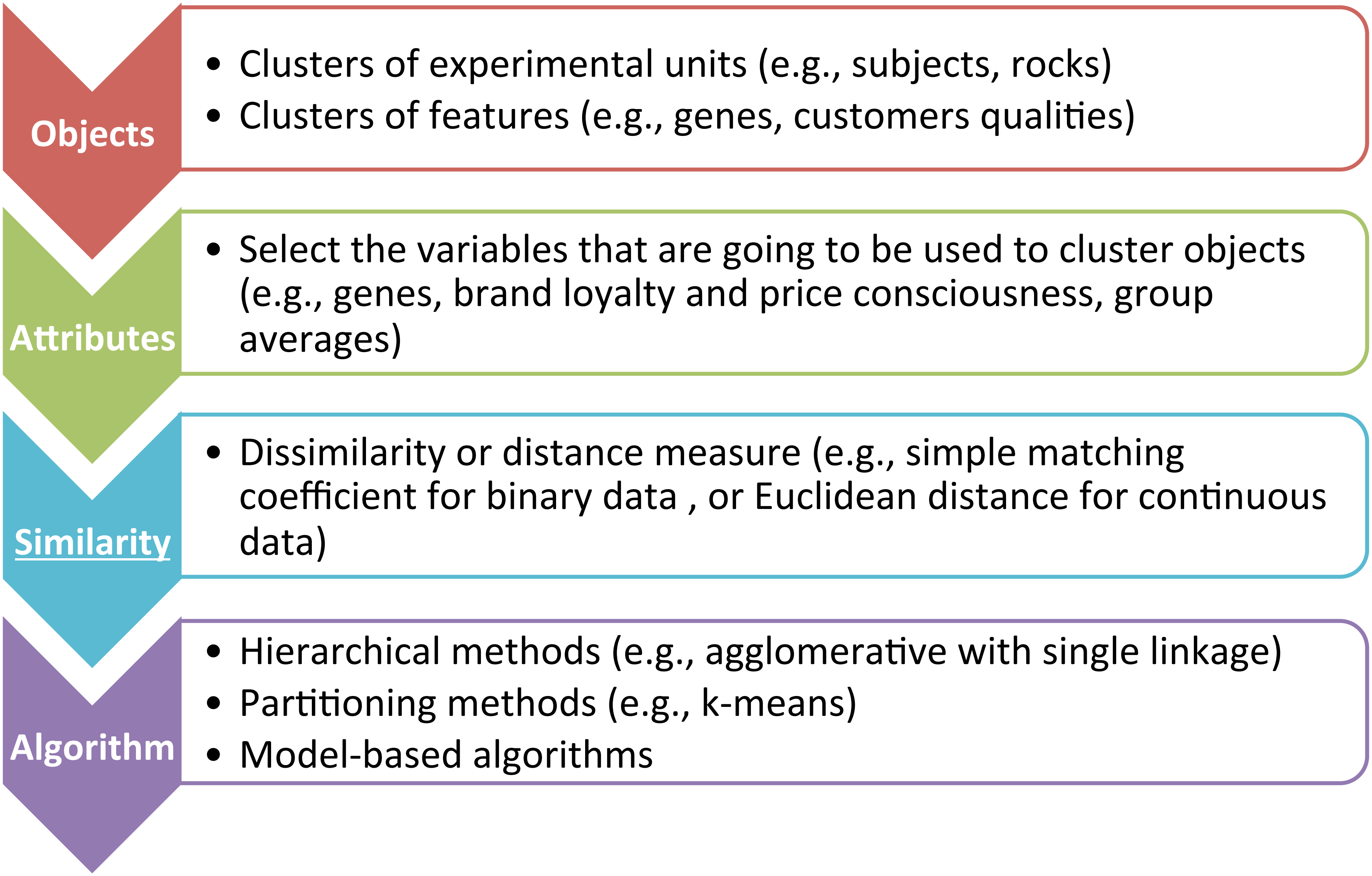
	Wild Type			Knock-out			…			Wild Type			Knock-out		
	E16									4_weeks					
	Sample_20	…	Sample_23	Sample_16	…	Sample_17	…	Sample *i*	…	Sample_36	…	Sample_39	Sample_11	…	Sample_9
1415670_at	7.24	…	7.07	7.38	…	7.34	…	…	…	7.25	…	7.13	7.42	…	7.32
1415671_at	9.48	…	10.13	7.64	…	10.03	…	…	…	9.66	…	8.73	9.83	…	9.80
1415672_at	10.01	…	9.91	8.42	…	10.24	…	…	…	9.51	…	9.53	10.00	…	9.85
1415673_at	8.36	…	8.49	8.36	…	8.37	…	…	…	8.49	…	8.65	8.60	…	8.40
1415674_a_at	8.59	…	8.64	8.51	…	8.89	…	…	…	8.42	…	8.28	8.43	…	8.46
1415675_at	9.59	…	9.70	9.66	…	9.61	…	…	…	9.67	…	9.45	9.60	…	9.51
1415676_a_at	9.68	…	10.19	8.05	…	10.02	…	…	…	9.95	…	8.70	9.23	…	9.82
1415677_at	7.24	…	7.49	7.34	…	7.34	…	…	…	7.28	…	6.84	7.33	…	7.45
1415678_at	11.71	…	11.57	10.46	…	11.75	…	…	…	11.56	…	11.80	12.04	…	11.81
1415679_at	9.21	…	9.92	8.22	…	9.60	…	…	…	9.13	…	8.08	9.06	…	9.29
…	…	…	…	…	…	…	…		…	…	…	…	…	…	…
gene g	…	…	…	…	…	…	…	X_{gi}	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…		…	…	…	…	…	…	…
1460746_at	6.37	…	6.12	7.25	…	6.15	…	…	…	6.34	…	6.52	6.36	…	6.35

Samples Clustering: based on the expression of the *G* genes (attributes), how do samples (objects) cluster?

Row-driven analyses

	Wild Type			Knock-out			...			Wild Type			Knock-out		
	E16									4_weeks					
	Sample_20	...	Sample_23	Sample_16	...	Sample_17	...	Sample *i*	...	Sample_36	...	Sample_39	Sample_11	...	Sample_9
1415670_at	7.24	...	7.07	7.38	...	7.34	...	...	...	7.25	...	7.13	7.42	...	7.32
1415671_at	9.48	...	10.13	7.64	...	10.03	...	...	...	9.66	...	8.73	9.83	...	9.80
1415672_at	10.01	...	9.91	8.42	...	10.24	...	...	...	9.51	...	9.53	10.00	...	9.85
1415673_at	8.36	...	8.49	8.36	...	8.37	...	...	...	8.49	...	8.65	8.60	...	8.40
1415674_a_at	8.59	...	8.64	8.51	...	8.89	...	...	...	8.42	...	8.28	8.43	...	8.46
1415675_at	9.59	...	9.70	9.66	...	9.61	...	...	...	9.67	...	9.45	9.60	...	9.51
1415676_a_at	9.68	...	10.19	8.05	...	10.02	...	...	...	9.95	...	8.70	9.23	...	9.82
1415677_at	7.24	...	7.49	7.34	...	7.34	...	...	...	7.28	...	6.84	7.33	...	7.45
1415678_at	11.71	...	11.57	10.46	...	11.75	...	...	...	11.56	...	11.80	12.04	...	11.81
1415679_at	9.21	...	9.92	8.22	...	9.60	...	...	...	9.13	...	8.08	9.06	...	9.29
...	...	...	...	...	...	...	...		...	...	...	...	...	...	...
gene g	...	...	...	...	...	...	...	X_{gi}	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...		...	...	...	...	...	...	...
1460746_at	6.37	...	6.12	7.25	...	6.15	...	...	...	6.34	...	6.52	6.36	...	6.35

Gene Clustering: based on the gene expressions of 39 samples (attributes), how do genes (objects) cluster?



Objects

- Clusters of experimental units (e.g., subjects, rocks)
- Clusters of features (e.g., genes, customers qualities)

Attributes

- Select the variables that are going to be used to cluster objects (e.g., genes, brand loyalty and price consciousness, group averages)

Similarity

- Dissimilarity or distance measure (e.g., simple matching coefficient for binary data , or Euclidean distance for continuous data)

Algorithm

- Hierarchical methods (e.g., agglomerative with single linkage)
- Partitioning methods (e.g., k-means)
- Model-based algorithms

Number of clusters???

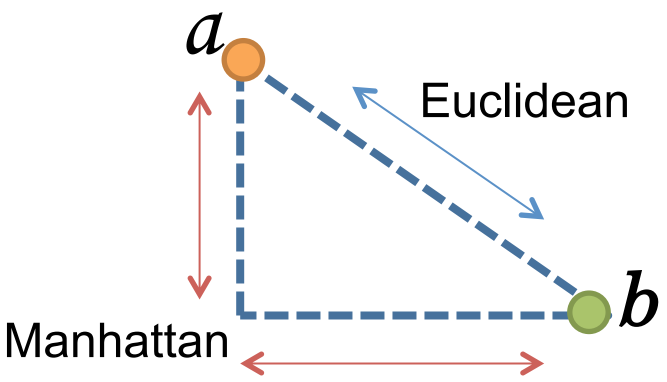
Dissimilarity and Distance

- We need to compute pairwise (dis)similarities/distances between all objects.
- Many clustering algorithms use a *distance* to measure proximity between objects
- These measures are collect into a symmetric matrix -- the distance or dissimilarity matrix

Typical **distance** measures for continuous data:

– Euclidean distance*

$$a \in R^p, b \in R^p : d(a,b) = \sqrt{\sum_{i=1}^{p} (a_i - b_i)^2}$$

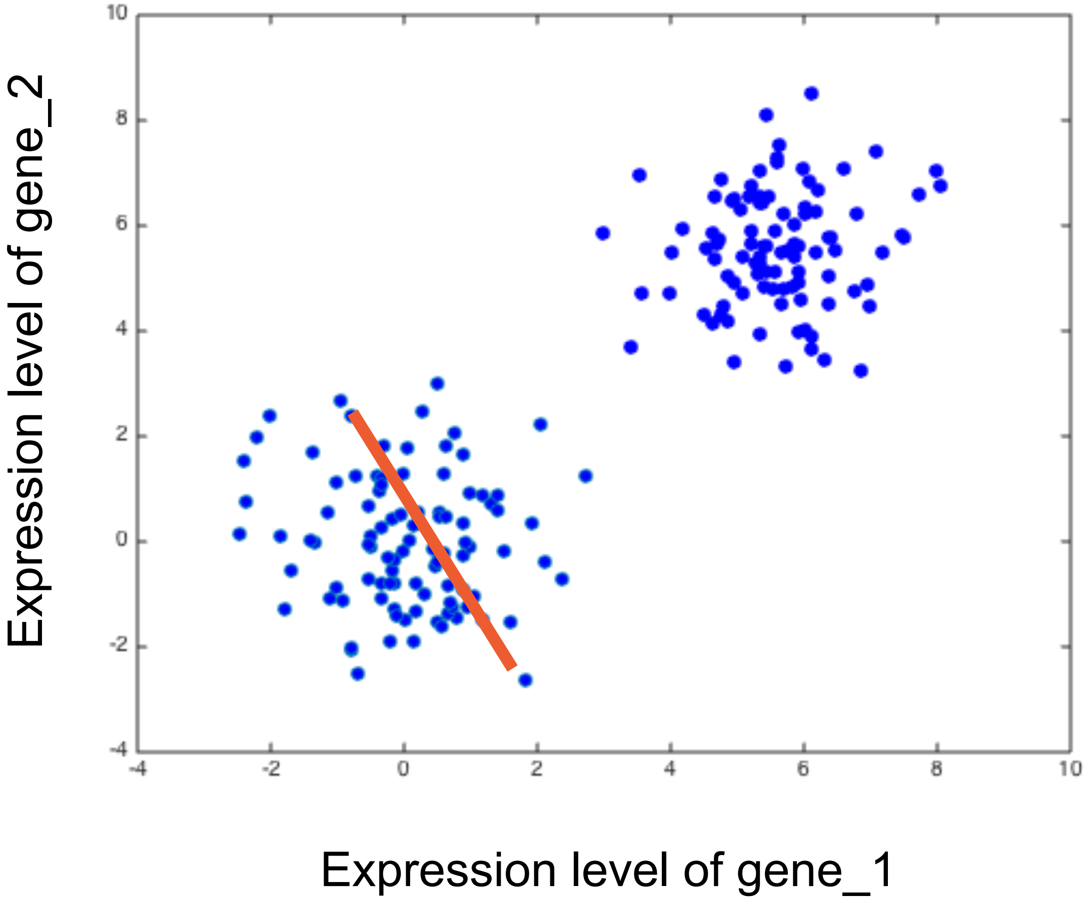


– Manhattan distance *

$$a \in R^p, b \in R^p : d(a,b) = \sum_{i=1}^{p} \left| a_i - b_i \right|$$

* The data need to be standardized if variables are measured in very different scales

Suppose you measured protein activity levels (expression) for 2 genes (gene A and gene B) for 200 individuals



How close are these individuals?

Typical **similarity** measures in genomics

- Centered correlation

$$a \in R^p, b \in R^p : \mathrm{corr}(a,b) = \frac{1}{p}\sum_{i=1}^{p}\left(\frac{a_i - \overline{a}}{\hat{\sigma}_a}\right)\left(\frac{b_i - \overline{b}}{\hat{\sigma}_b}\right)$$

- Uncentered correlation

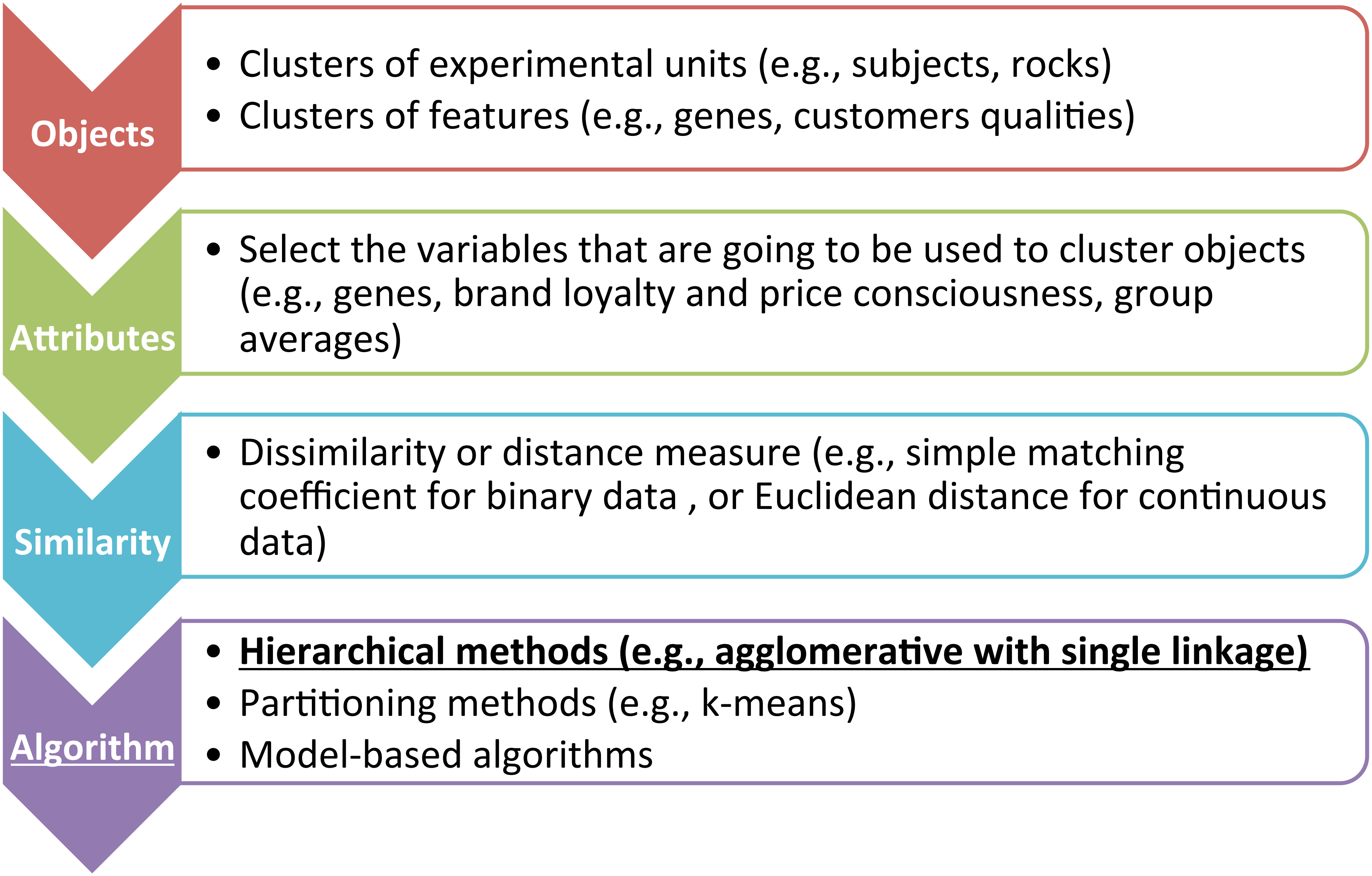
$$a \in R^p, b \in R^p : \mathrm{corr}(a,b) = \frac{1}{p}\sum_{i=1}^{p}\left(\frac{a_i}{\hat{\sigma}_a^{(0)}}\right)\left(\frac{b_i}{\hat{\sigma}_b^{(0)}}\right);$$

$\hat{\sigma}^{(0)}$: the center is set to 0

- Correlation **Distances**:

$$\mathrm{dist}(a,b) = 1 - \mathrm{corr}(a,b)$$

$$\mathrm{dist}(a,b) = 1 - \left|\mathrm{corr}(a,b)\right|$$



Objects

- Clusters of experimental units (e.g., subjects, rocks)
- Clusters of features (e.g., genes, customers qualities)

Attributes

- Select the variables that are going to be used to cluster objects (e.g., genes, brand loyalty and price consciousness, group averages)

Similarity

- Dissimilarity or distance measure (e.g., simple matching coefficient for binary data , or Euclidean distance for continuous data)

Algorithm

- **Hierarchical methods (e.g., agglomerative with single linkage)**
- Partitioning methods (e.g., k-means)
- Model-based algorithms

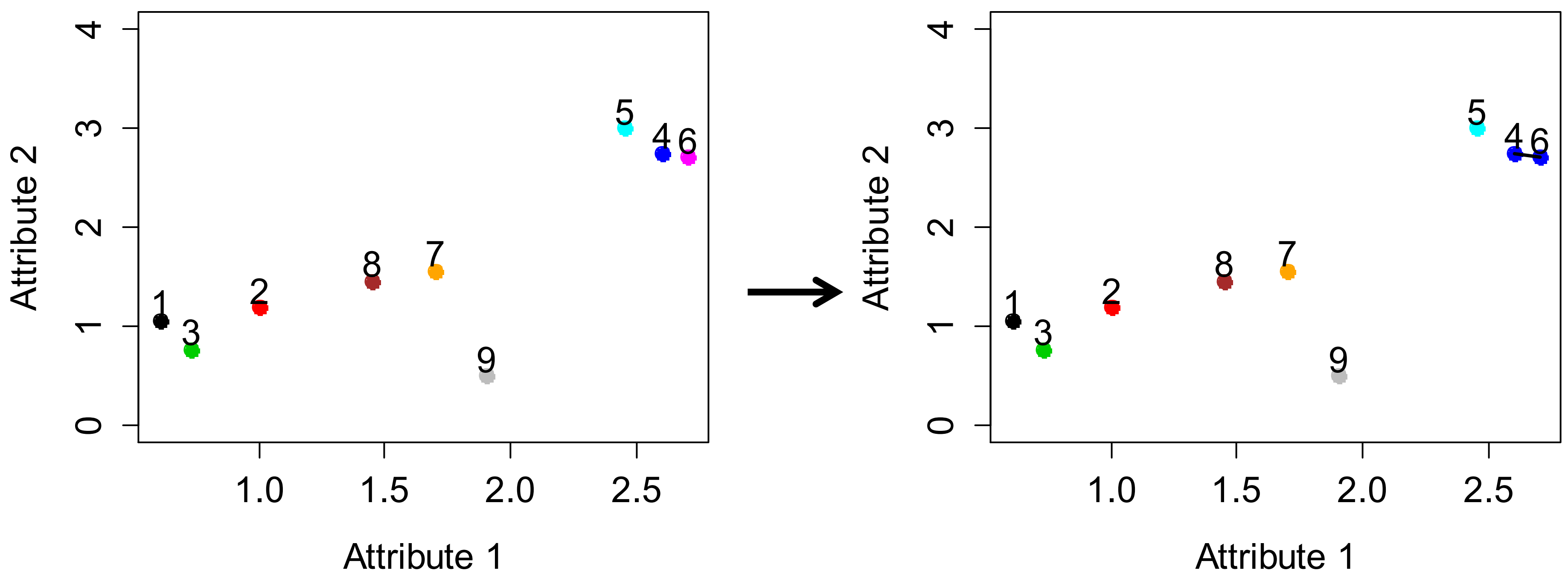
Number of clusters???

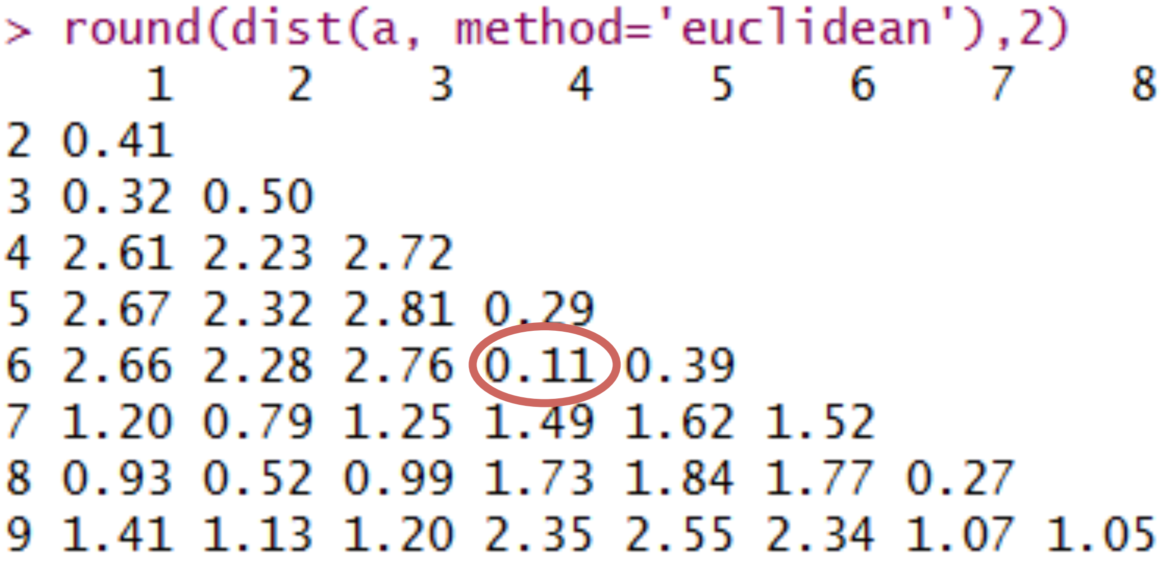
c_i

Algorithms: Hierarchical

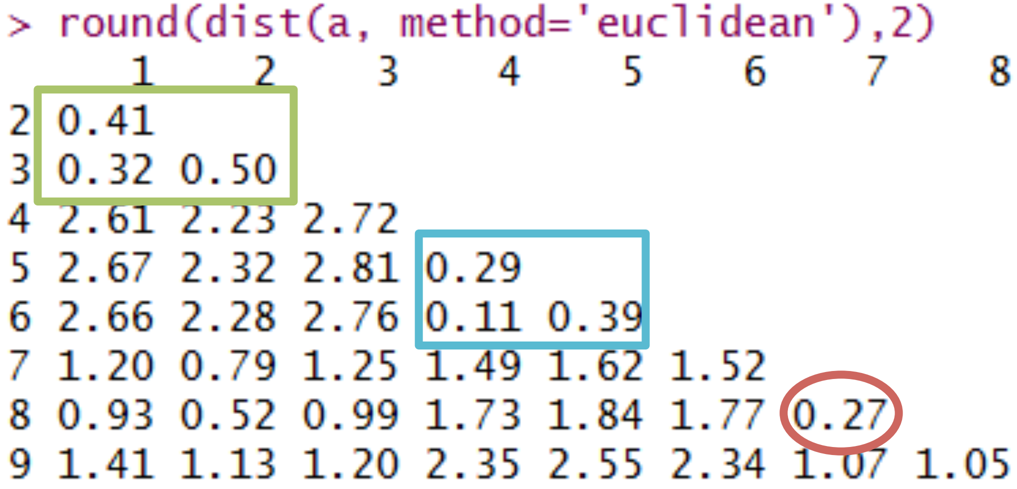
Given *N objects* with *H attributes* and a *distance metric*:

1. Assign each object to a cluster and compute the pairwise distances between all clusters
2. Find the “closest” pair of *clusters* and *merge them* into a single cluster
3. Compute new distances between clusters
4. Repeat steps 2 and 3 until all objects belong to a single cluster.

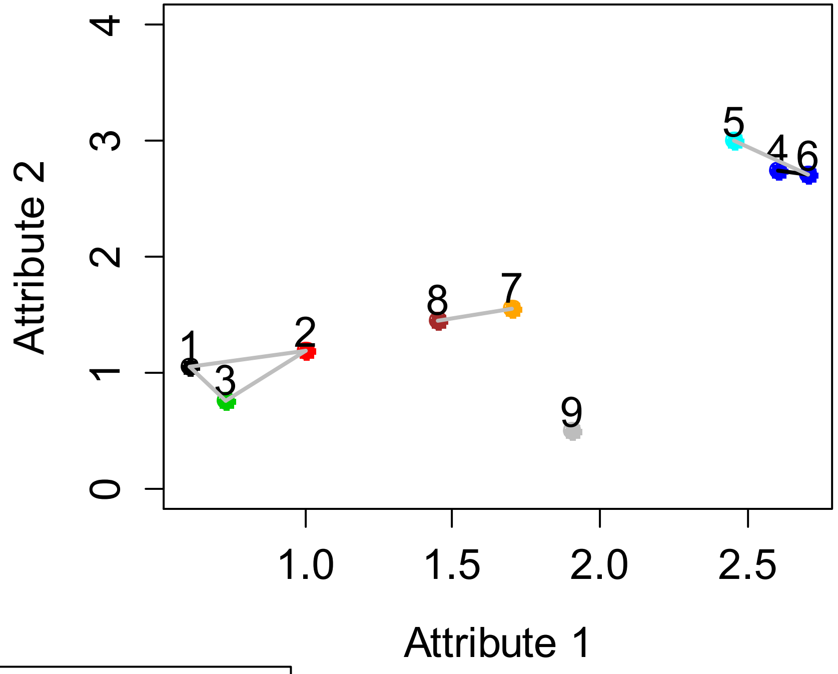


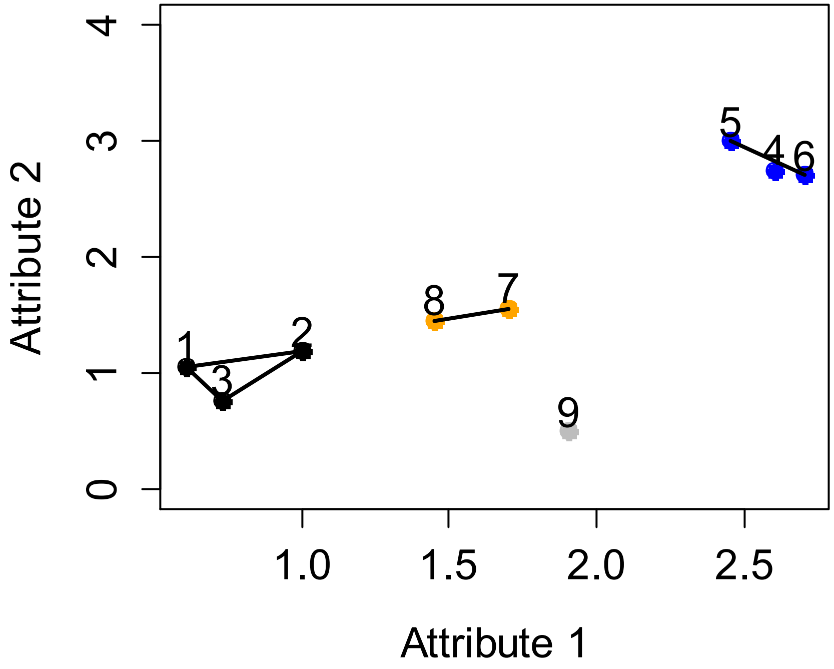


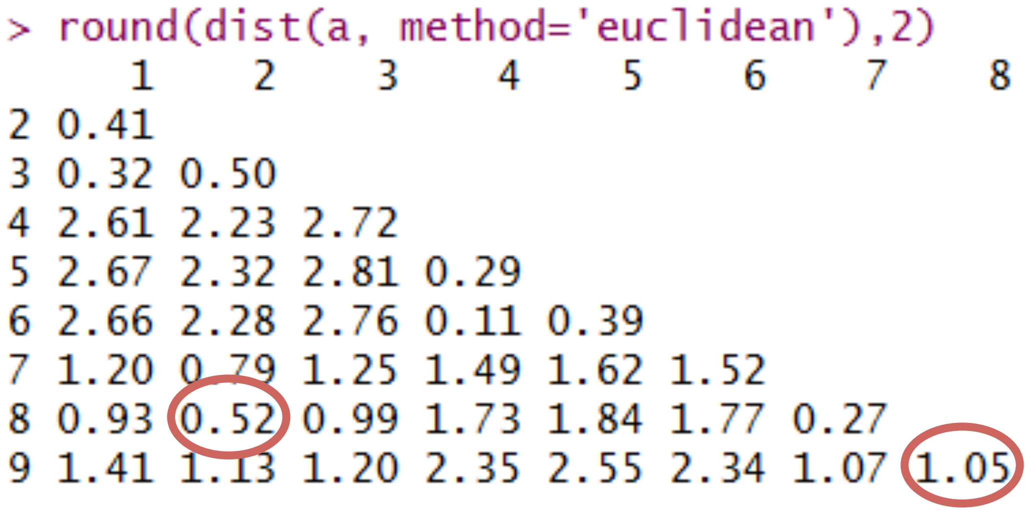
```
> round(dist(a, method='euclidean'),2)
     1    2    3    4    5    6    7    8
2 0.41
3 0.32 0.50
4 2.61 2.23 2.72
5 2.67 2.32 2.81 0.29
6 2.66 2.28 2.76 0.11 0.39
7 1.20 0.79 1.25 1.49 1.62 1.52
8 0.93 0.52 0.99 1.73 1.84 1.77 0.27
9 1.41 1.13 1.20 2.35 2.55 2.34 1.07 1.05
```



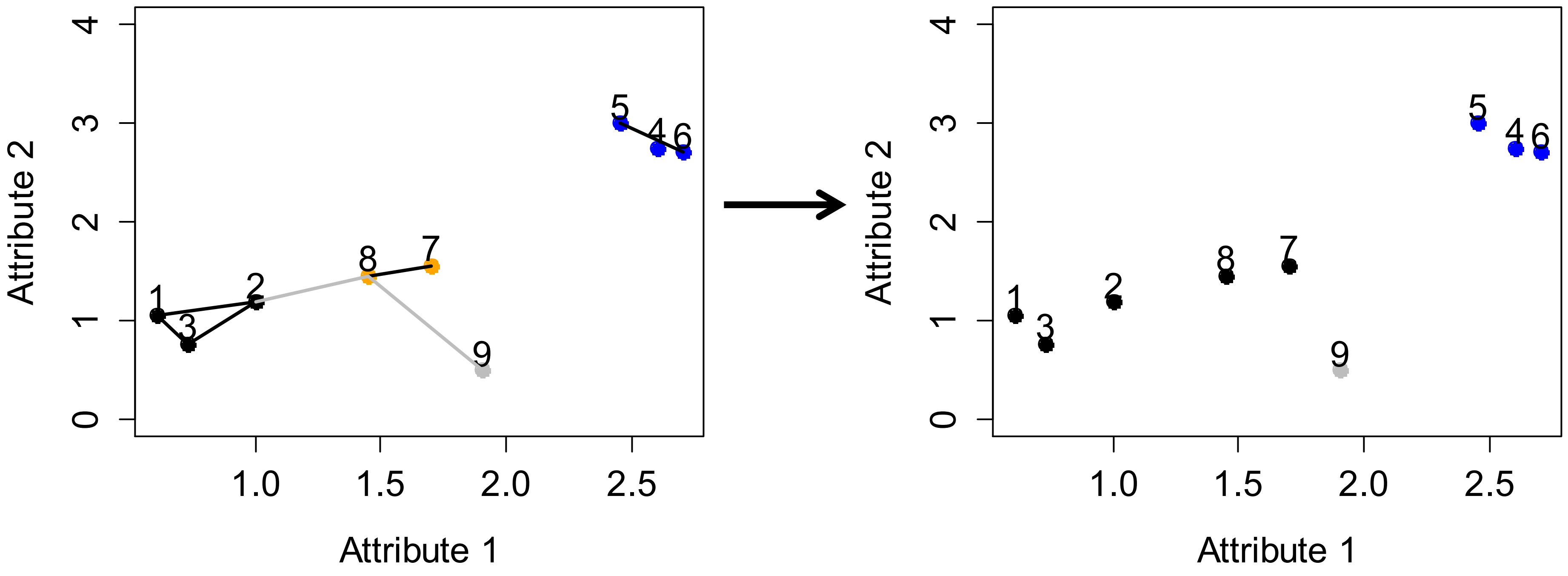
```
> round(dist(a, method='euclidean'),2)
     1    2    3    4    5    6    7    8
2 0.41                                   
3 0.32 0.50                              
4 2.61 2.23 2.72                         
5 2.67 2.32 2.81 0.29                    
6 2.66 2.28 2.76 0.11 0.39               
7 1.20 0.79 1.25 1.49 1.62 1.52          
8 0.93 0.52 0.99 1.73 1.84 1.77 0.27     
9 1.41 1.13 1.20 2.35 2.55 2.34 1.07 1.05
```

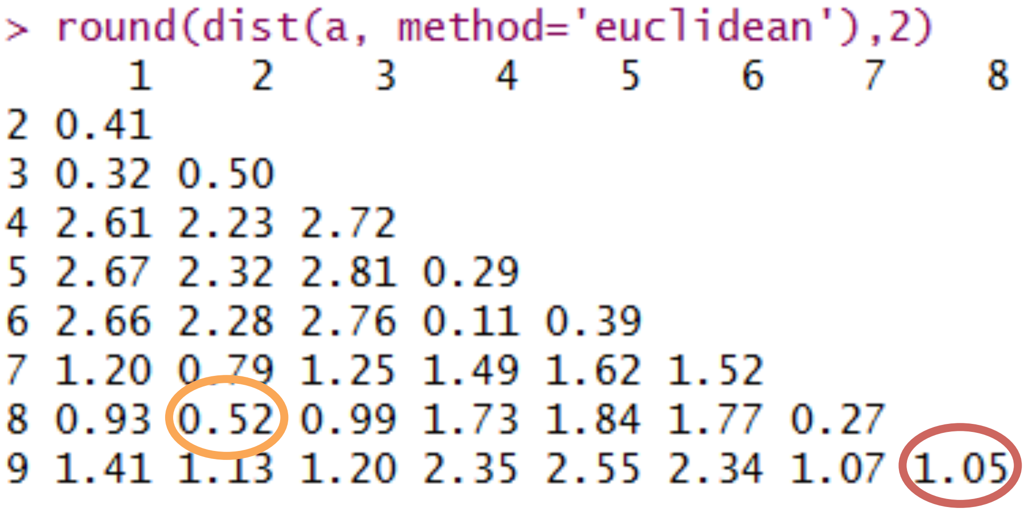




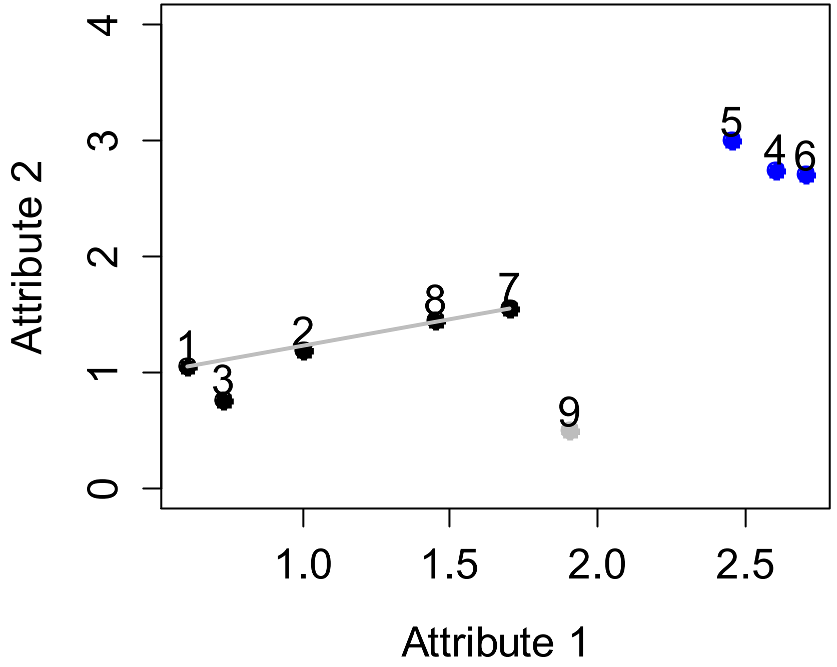


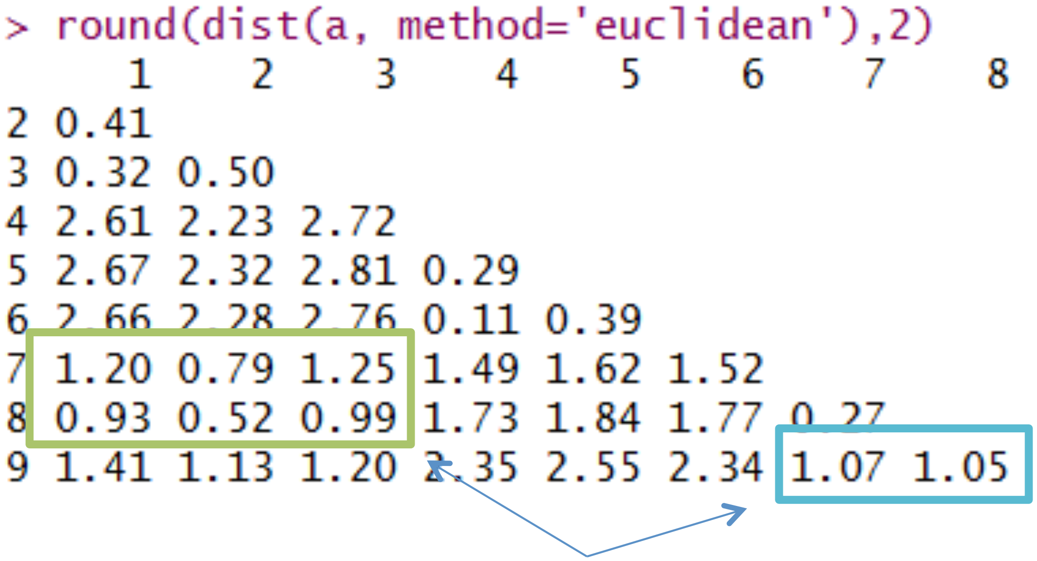
```
> round(dist(a, method='euclidean'),2)
     1    2    3    4    5    6    7    8
2 0.41
3 0.32 0.50
4 2.61 2.23 2.72
5 2.67 2.32 2.81 0.29
6 2.66 2.28 2.76 0.11 0.39
7 1.20 0.79 1.25 1.49 1.62 1.52
8 0.93 0.52 0.99 1.73 1.84 1.77 0.27
9 1.41 1.13 1.20 2.35 2.55 2.34 1.07 1.05
```



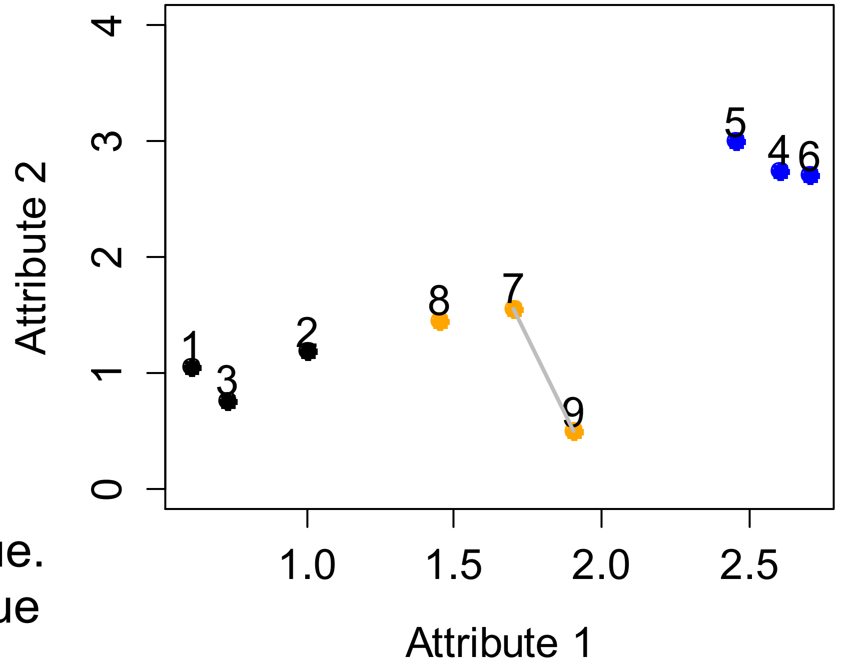


```
> round(dist(a, method='euclidean'),2)
     1    2    3    4    5    6    7    8
2 0.41
3 0.32 0.50
4 2.61 2.23 2.72
5 2.67 2.32 2.81 0.29
6 2.66 2.28 2.76 0.11 0.39
7 1.20 0.79 1.25 1.49 1.62 1.52
8 0.93 0.52 0.99 1.73 1.84 1.77 0.27
9 1.41 1.13 1.20 2.35 2.55 2.34 1.07 1.05
```



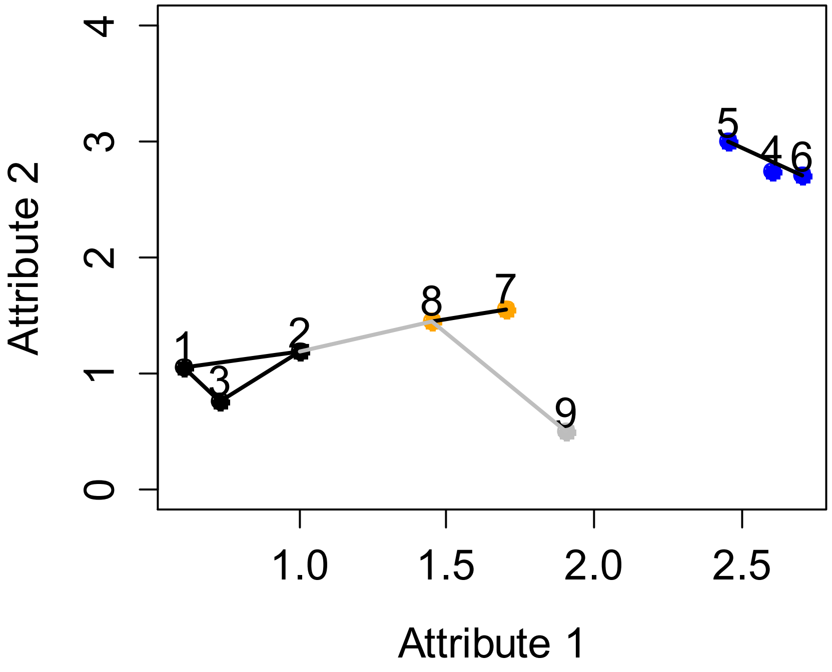


```
> round(dist(a, method='euclidean'),2)
     1    2    3    4    5    6    7    8
2 0.41
3 0.32 0.50
4 2.61 2.23 2.72
5 2.67 2.32 2.81 0.29
6 2.66 2.28 2.76 0.11 0.39
7 1.20 0.79 1.25 1.49 1.62 1.52
8 0.93 0.52 0.99 1.73 1.84 1.77 0.27
9 1.41 1.13 1.20 2.35 2.55 2.34 1.07 1.05
```



The min. in green is smaller than the min.in blue.
The max. in green is larger than the max. in blue

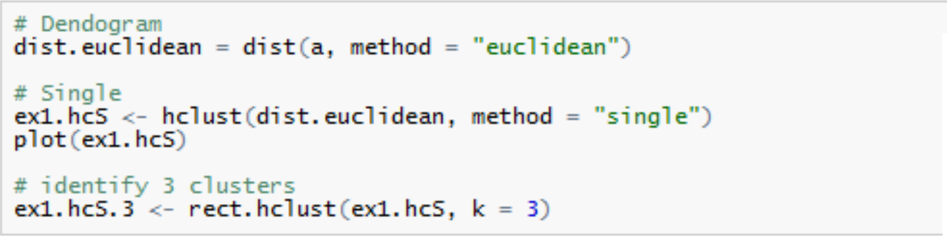
Single Linkage: The distance between two clusters is the ***minimum*** distance between any two elements



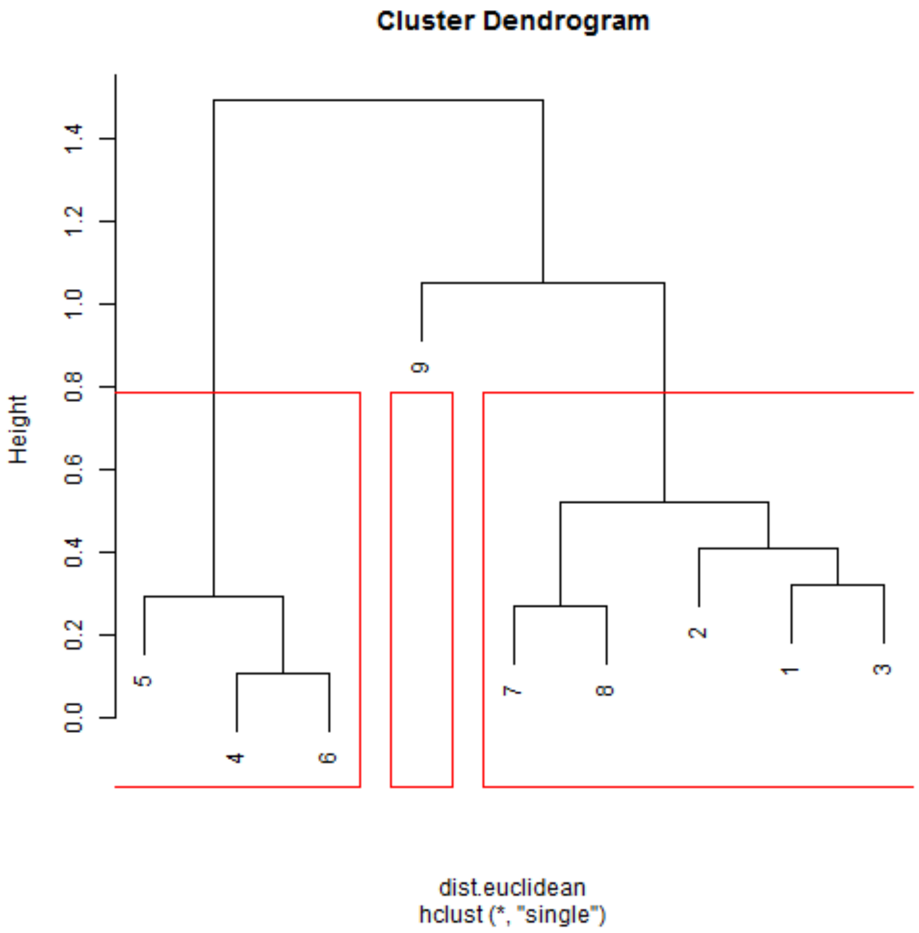
Complete Linkage: The distance between two clusters is the ***maximum*** distance between any two elements

Note: the distance between objects is the same (e.g., Euclidean)

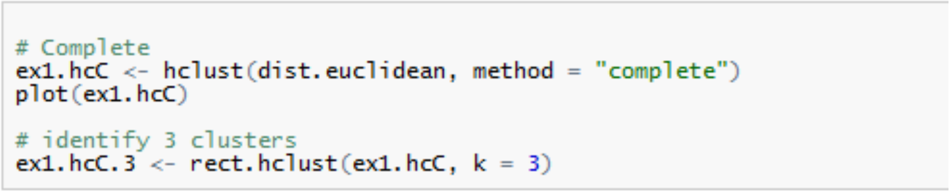
Single Linkage



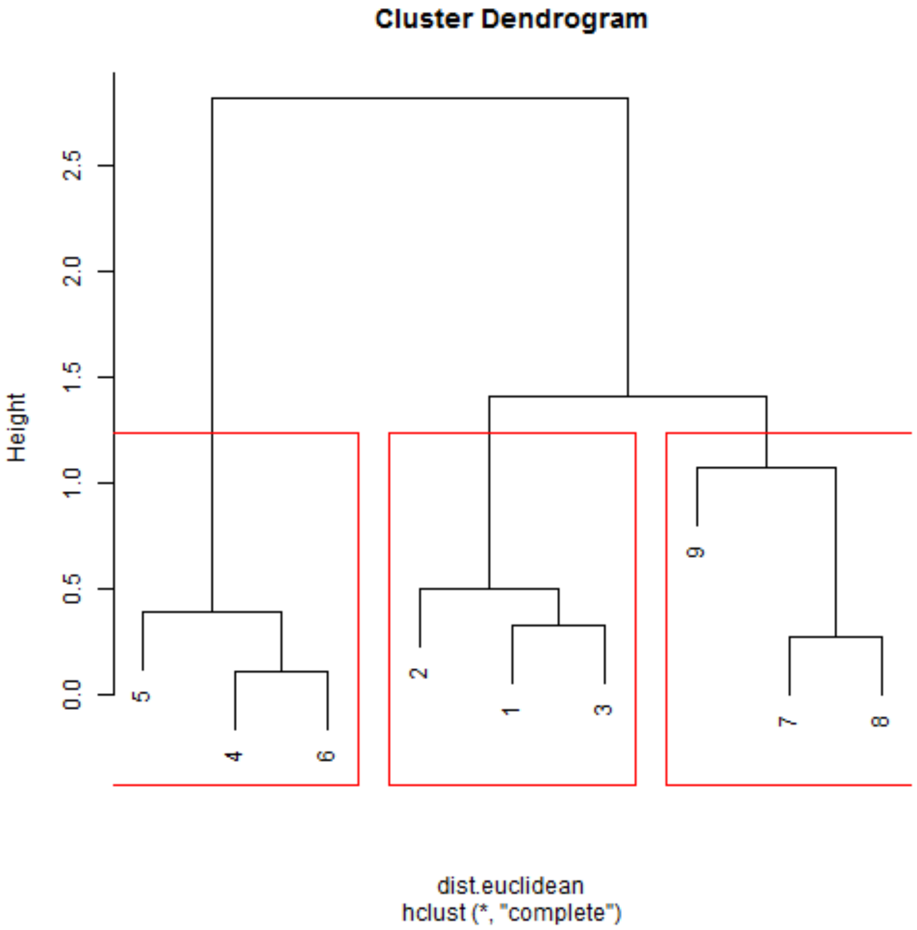
```
# Dendogram
dist.euclidean = dist(a, method = "euclidean")

# Single
ex1.hcS <- hclust(dist.euclidean, method = "single")
plot(ex1.hcS)

# identify 3 clusters
ex1.hcS.3 <- rect.hclust(ex1.hcS, k = 3)
```



Complete Linkage

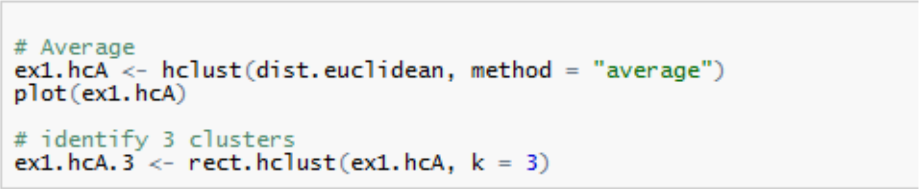


```
# Complete
ex1.hcC <- hclust(dist.euclidean, method = "complete")
plot(ex1.hcC)

# identify 3 clusters
ex1.hcC.3 <- rect.hclust(ex1.hcC, k = 3)
```

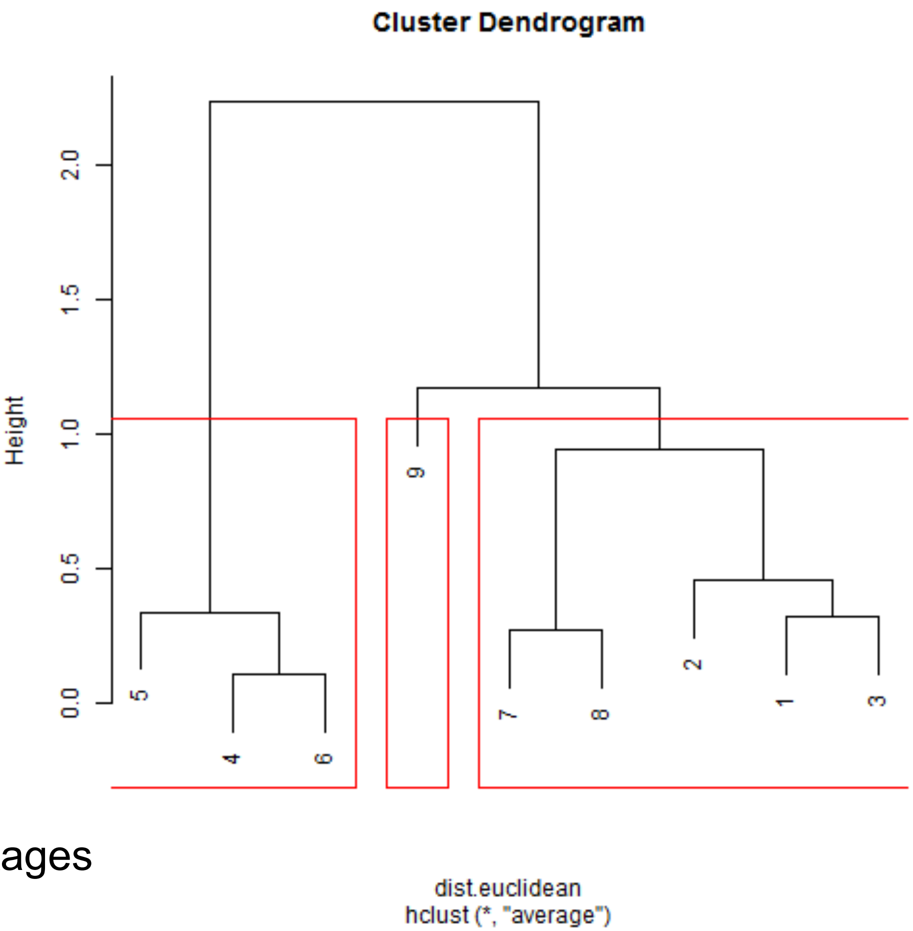


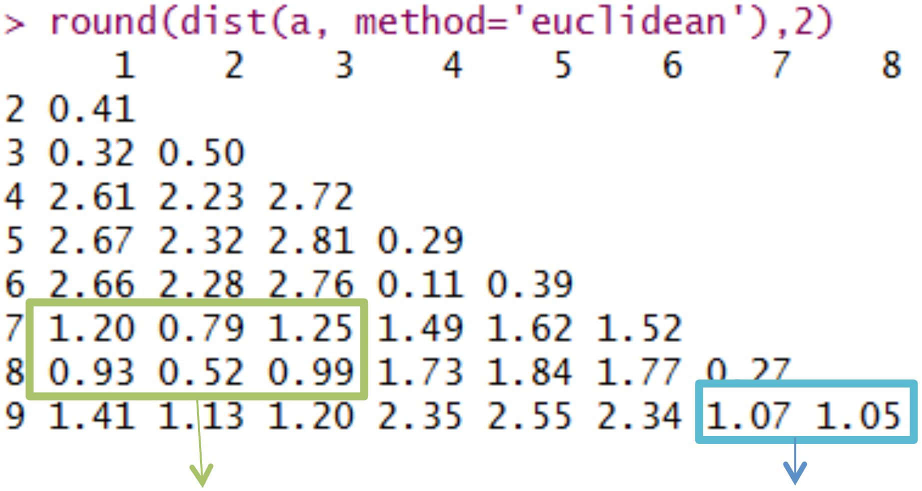
Is there an intermediate solutions?

Average Linkage: The distance between two clusters is the ***average*** of all pairwise distances between any two objects



```
# Average
ex1.hcA <- hclust(dist.euclidean, method = "average")
plot(ex1.hcA)

# identify 3 clusters
ex1.hcA.3 <- rect.hclust(ex1.hcA, k = 3)
```





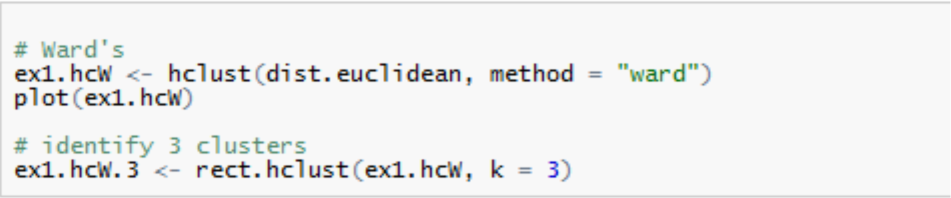
```
> round(dist(a, method='euclidean'),2)
     1    2    3    4    5    6    7    8
2 0.41
3 0.32 0.50
4 2.61 2.23 2.72
5 2.67 2.32 2.81 0.29
6 2.66 2.28 2.76 0.11 0.39
7 1.20 0.79 1.25 1.49 1.62 1.52
8 0.93 0.52 0.99 1.73 1.84 1.77 0.27
9 1.41 1.13 1.20 2.35 2.55 2.34 1.07 1.05
```

Avg=0.95

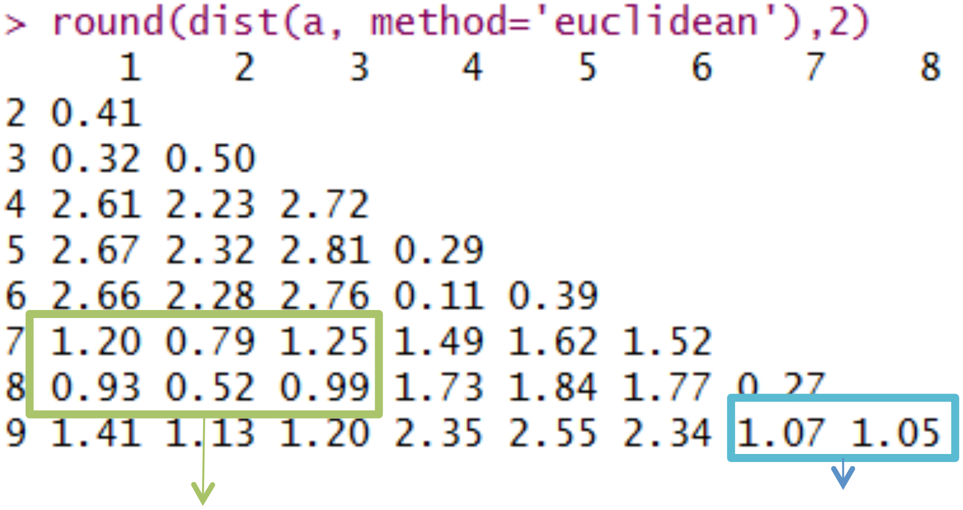
Avg=1.06

Note: in this trivial example, average and single linkages give the same result. In general, this is not true.

Ward's Criterion: The distance between two clusters is the ***sum*** of all pairwise distances between any two objects



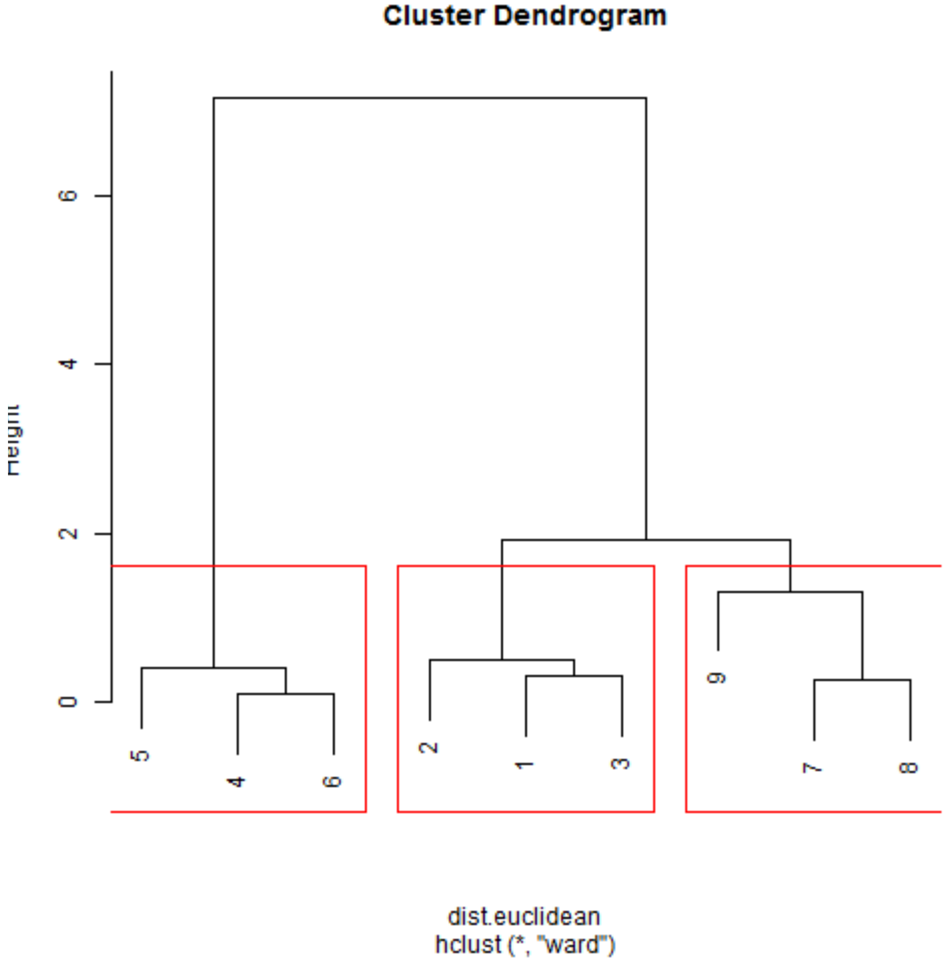
```
# Ward's
ex1.hcW <- hclust(dist.euclidean, method = "ward")
plot(ex1.hcW)

# identify 3 clusters
ex1.hcW.3 <- rect.hclust(ex1.hcW, k = 3)
```



```
> round(dist(a, method='euclidean'),2)
     1    2    3    4    5    6    7    8
2 0.41
3 0.32 0.50
4 2.61 2.23 2.72
5 2.67 2.32 2.81 0.29
6 2.66 2.28 2.76 0.11 0.39
7 1.20 0.79 1.25 1.49 1.62 1.52
8 0.93 0.52 0.99 1.73 1.84 1.77 0.27
9 1.41 1.13 1.20 2.35 2.55 2.34 1.07 1.05
```

Sum=5.68

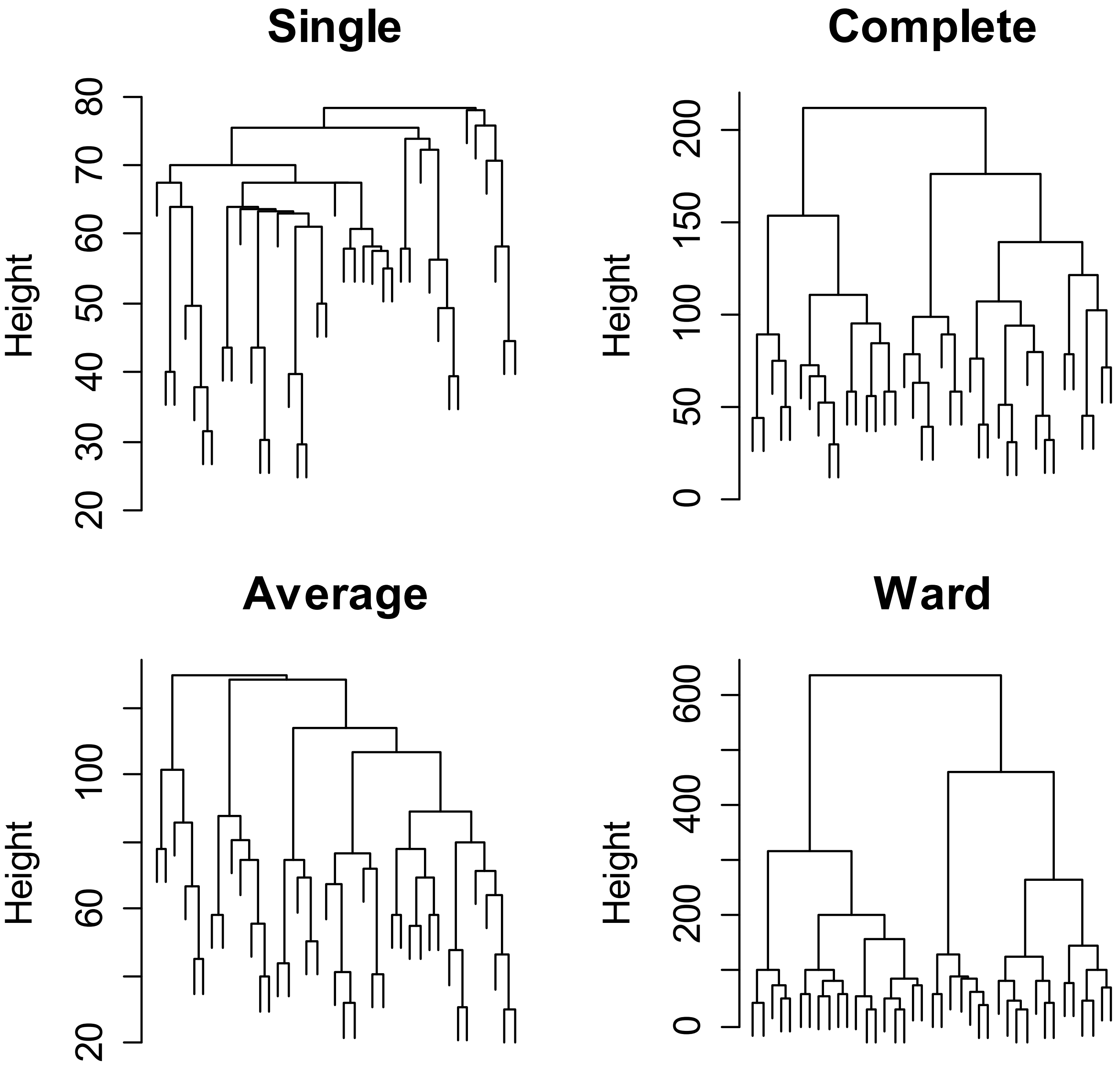
Sum=2.12

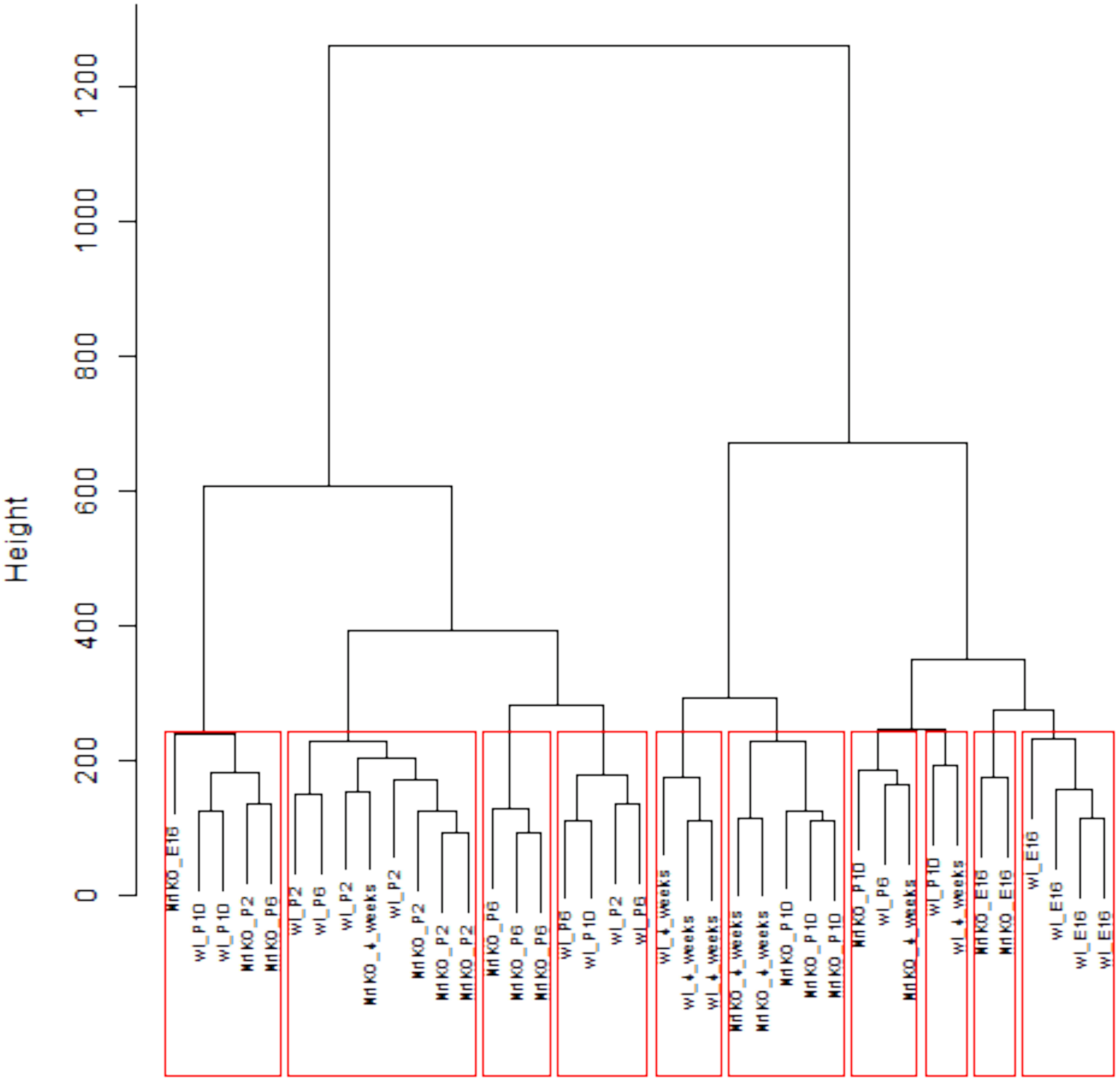


Note: in this trivial example, Ward's criterion gives the same result as complete linkage. In general, this is not true.

photoRec: Sample Clustering

- **Data**: each gene has been measured on 3 or 4 biological replicates of wild-type and knock-out mice, for each of 5 time points (~30K genes, 39 samples)
- **Objects**: 39 mice samples
- **Attributes**: ~30K genes

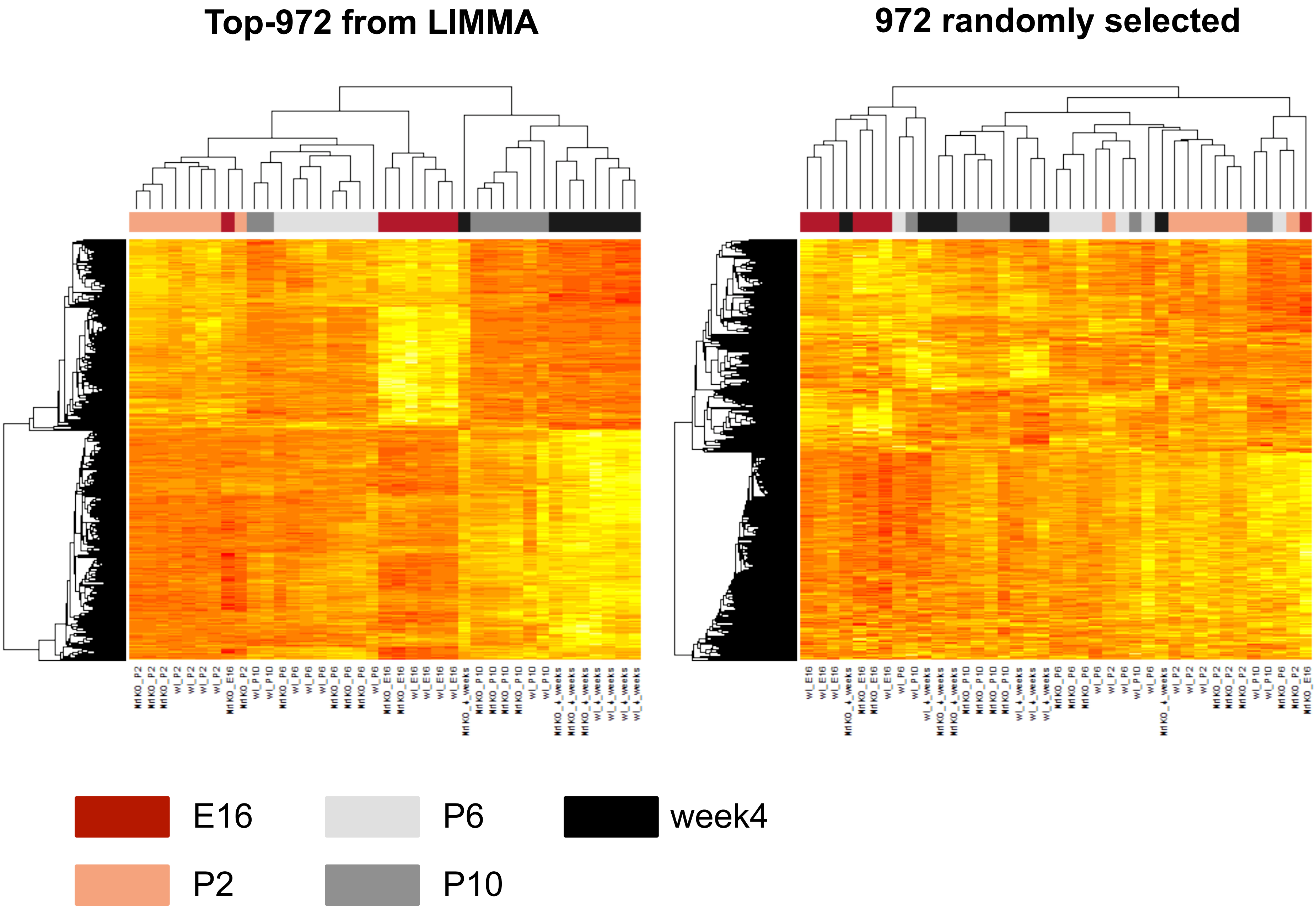

Single
Height
20
30
40
50
60
70
80
Complete
Height
0
50
100
150
200
Average
Height
20
60
100
Ward
Height
0
200
400
600

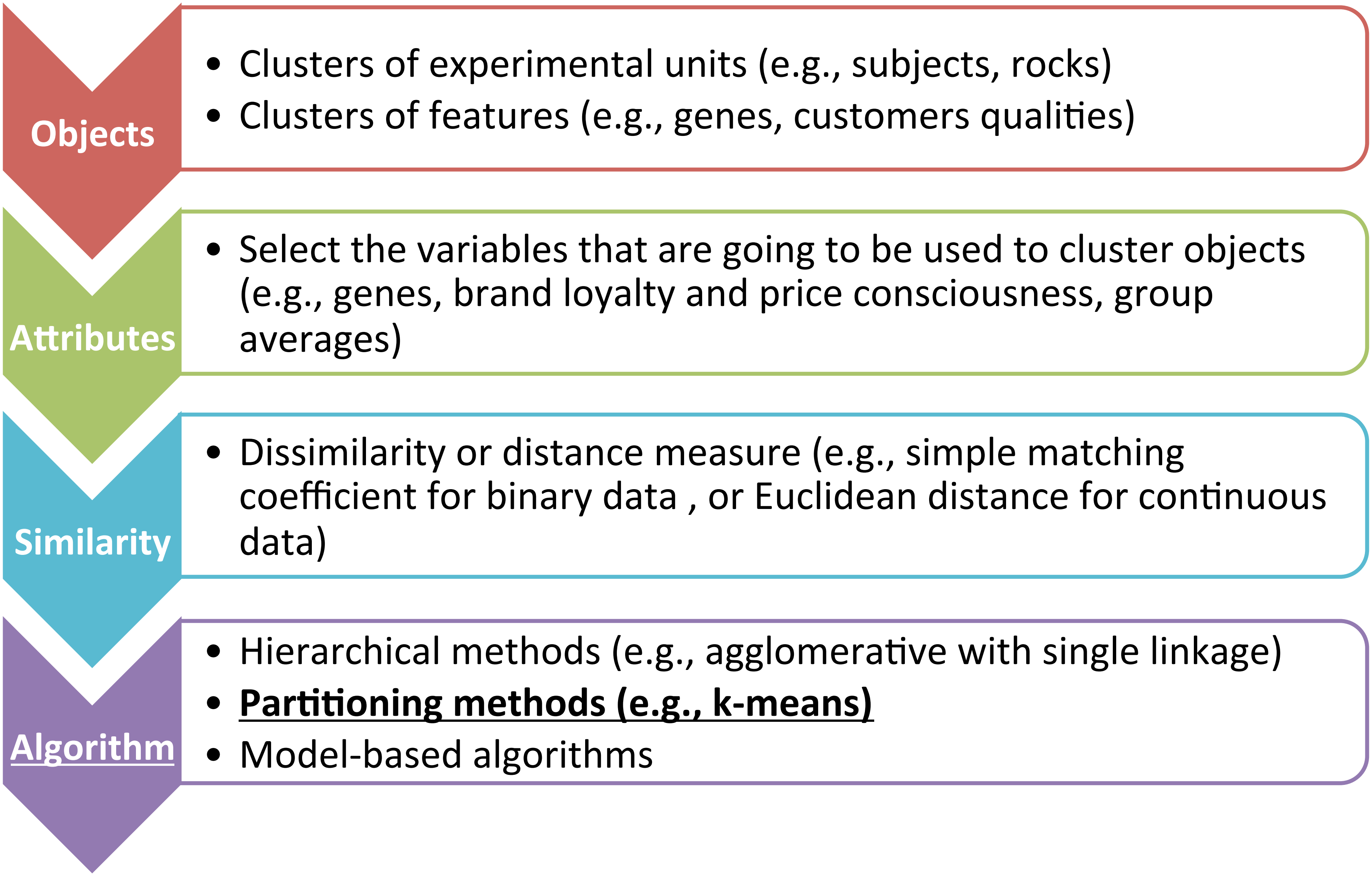


Ward showing 8 clusters

photoRec: Gene Clustering

- **Objects**: ~30K genes
- **Attributes**: 39 mice samples
- Since clustering genes is slow when you have a lot of genes, for the sake of time we will work with a smaller subset of genes.


Top-972 from LIMMA
972 randomly selected
NrlKO_P2
NrlKO_P2
NrlKO_P2
wt_P2
wt_P2
wt_P2
wt_P2
NrlKO_E16
NrlKO_P2
wt_P10
wt_P10
NrlKO_P6
wt_P6
wt_P6
wt_P6
NrlKO_P6
NrlKO_P6
NrlKO_P6
wt_P6
NrlKO_E16
NrlKO_E16
wt_E16
wt_E16
wt_E16
wt_E16
NrlKO_4_weeks
NrlKO_P10
NrlKO_P10
NrlKO_P10
NrlKO_P10
wt_P10
wt_P10
NrlKO_4_weeks
NrlKO_4_weeks
NrlKO_4_weeks
wt_4_weeks
wt_4_weeks
wt_4_weeks
wt_4_weeks
wt_E16
wt_E16
wt_E16
NrlKO_4_weeks
NrlKO_E16
NrlKO_E16
wt_E16
wt_P6
wt_P10
wt_4_weeks
NrlKO_4_weeks
NrlKO_4_weeks
NrlKO_P10
NrlKO_P10
NrlKO_P10
NrlKO_P10
wt_4_weeks
wt_4_weeks
wt_4_weeks
NrlKO_P6
NrlKO_P6
NrlKO_P6
wt_P6
wt_P2
wt_P6
wt_P10
wt_P6
NrlKO_4_weeks
wt_P2
wt_P2
wt_P2
NrlKO_P2
NrlKO_P2
NrlKO_P2
wt_P10
wt_P10
NrlKO_P6
NrlKO_P2
NrlKO_E16
E16
P6
week4
P2
P10



Objects

- Clusters of experimental units (e.g., subjects, rocks)
- Clusters of features (e.g., genes, customers qualities)

Attributes

- Select the variables that are going to be used to cluster objects (e.g., genes, brand loyalty and price consciousness, group averages)

Similarity

- Dissimilarity or distance measure (e.g., simple matching coefficient for binary data , or Euclidean distance for continuous data)

Algorithm

- Hierarchical methods (e.g., agglomerative with single linkage)
- **Partitioning methods (e.g., k-means)**
- Model-based algorithms

Number of clusters???

Algorithms: Partitioning

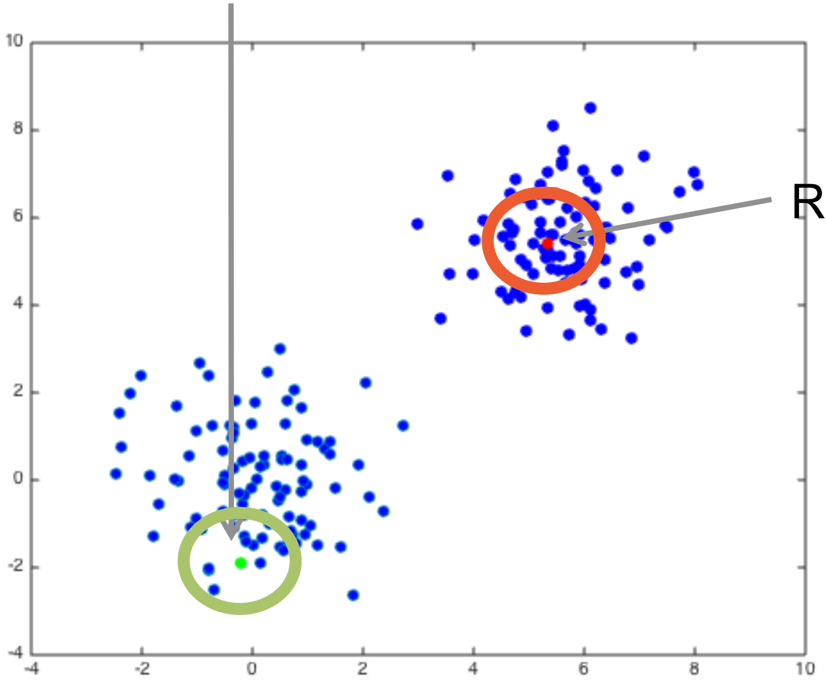
- These algorithms partition the objects into K groups
- Often motivated by an objective function that attains an extreme for the “correct” or “best” partition of the objects.
- K needs to be determined *a priori*.
- Most typical cases: k-means and partitioning around the medoids (PAM).

Partitioning algorithms: e.g., K-means

Algorithm: iterative procedure

1) Pick k random points as initial cluster centers

Randomly selected point 1



Randomly selected point 2

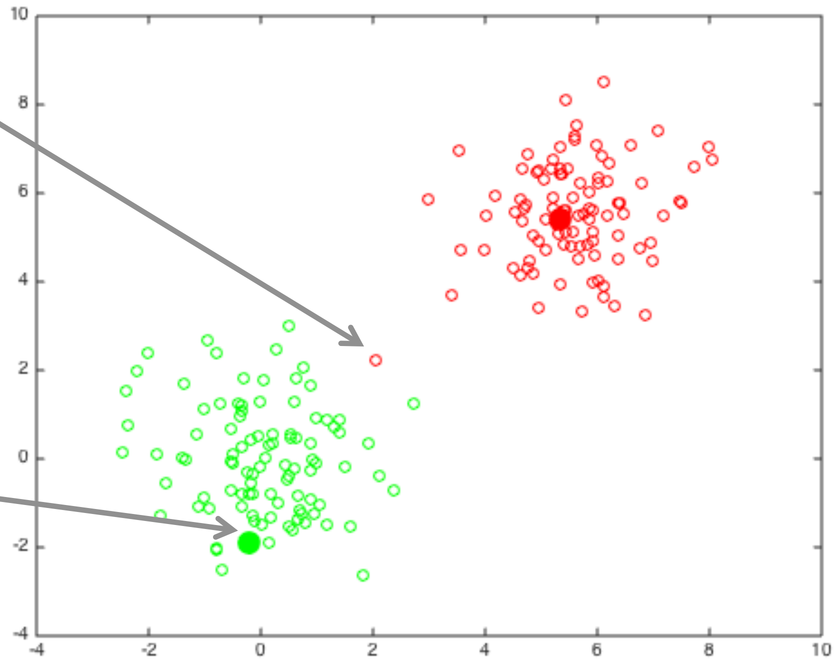
K-means algorithm (cont.)

Algorithm: iterative procedure

1) Pick k random points as initial cluster centers
2) Measure distance between all points and the cluster centers – assign points to nearest cluster

Doesn’t look right

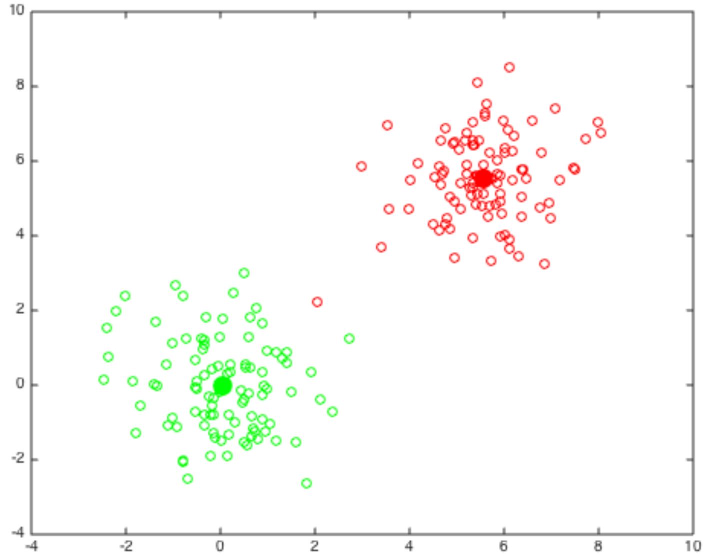
“Initialization point”



K-means algorithm (cont.)

Algorithm: iterative procedure

1) Pick k random points as initial cluster centers
2) Measure distance between all points and the cluster centers – assign points to nearest cluster
3) Compute cluster means
4) Reassign points to cluster based on distance
    - If not change from previous assignment, stop, else to go step 3



K-means algorithm: in words…

1. Divide the data into K clusters
   Initialize the "centroids" with the mean of the object attributes in each cluster
2. Assign each item to the cluster with closest centroid
3. When all objects have been assigned, recalculate the centroids (mean)
4. Repeat 2-3 until the centroids no longer move

K-means objective function

Objective function: minimize the average squared **Euclidean distance** of objects from their assigned cluster centers.

* N objects, each have p attributes: $\{x_1, x_2, x_3, \ldots, x_n\}$

Cluster center for cluster k

* k-means objective function: $J = \underbrace{\sum_{i=1}^{n} \sum_{k=1, i \in k}^{K} (x_i - \mu_k)^2}_{\text{Euclidian distance between each point to the center}}$

* Given C_1, C_2,…, C_K , the minimum of *within cluster distance* is attained estimating the center of the cluster with its sample mean

Partitioning around Medoids (PAM)

- Kaufmann and Rousseeuw provide this as a partitioning method in the `cluster` library in R
- K representative objects (= medoids) are chosen as cluster centers and objects are assigned to the center with which they have minimum dissimilarity
- Objective function is total of all object-to-medoid dissimilarities
- Similar to K-means but more robust to outliers

photoRec: Sample Clustering

- **Objects**: 39 mice samples (5 time points)
- **Attributes**: ~30K genes
- **Number of clusters**: K=5 (although another K may be "better")

k-means

	C_1	C_2	C_3	C_4	C_5
E16	0	0	6	0	1
P2	4	0	0	0	4
P6	5	1	0	0	2
P10	1	2	0	3	2
4 w	0	2	1	5	0

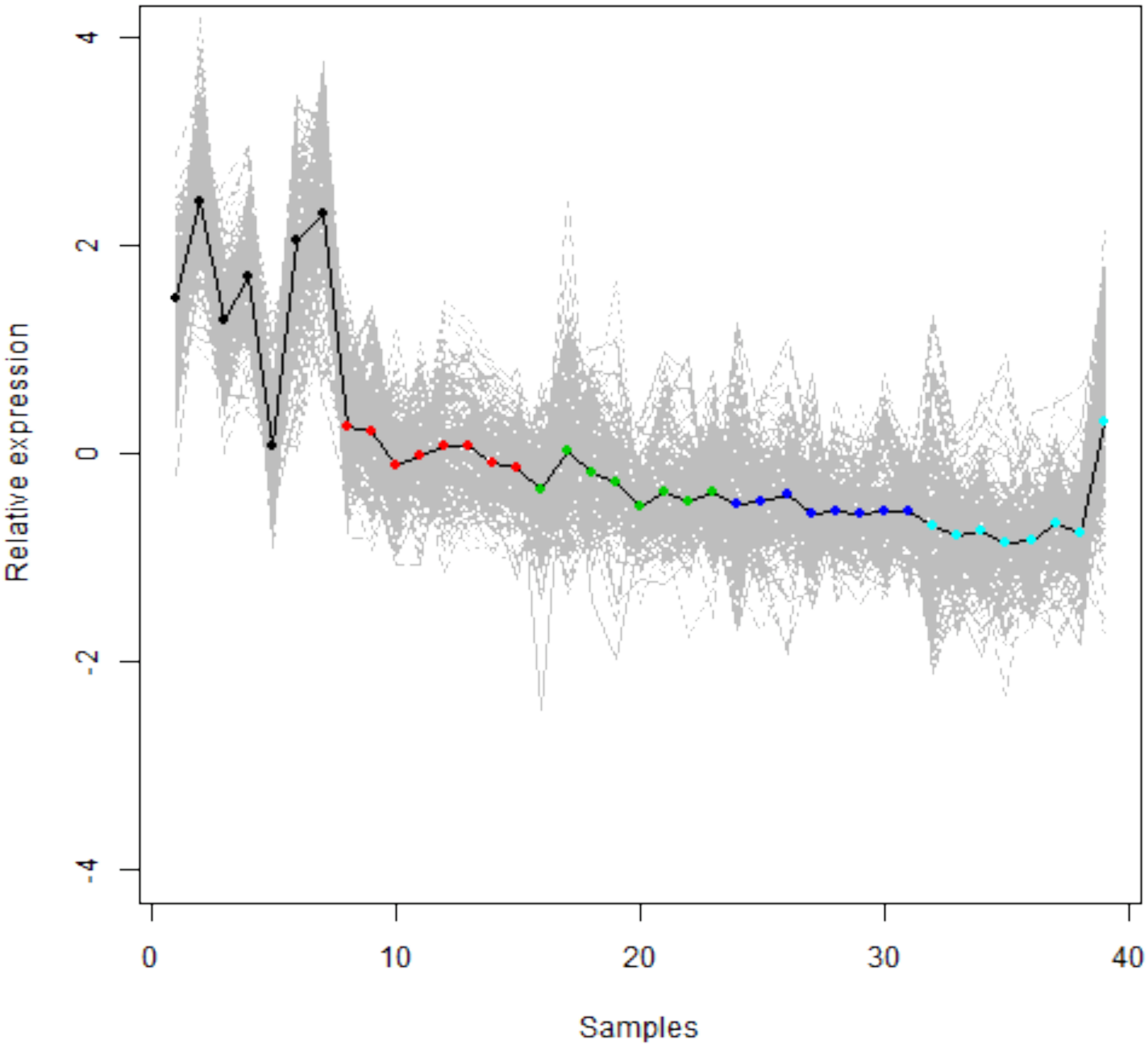
PAM

	C_1	C_2	C_3	C_4	C_5
E16	6	1	0	0	0
P2	0	1	7	0	0
P6	3	2	3	0	0
P10	0	2	1	1	4
4 w	1	0	1	4	2

photoRec: Gene Clustering

- **Objects**: ~1K genes (DE among 5 developmental stages, selected using limma)
- **Attributes**: 39 mice samples
- **Number of clusters**: K=5 (this value may be too low given that there are ~1K objects to cluster)

k-means: cluster 1



Algorithms to estimate K

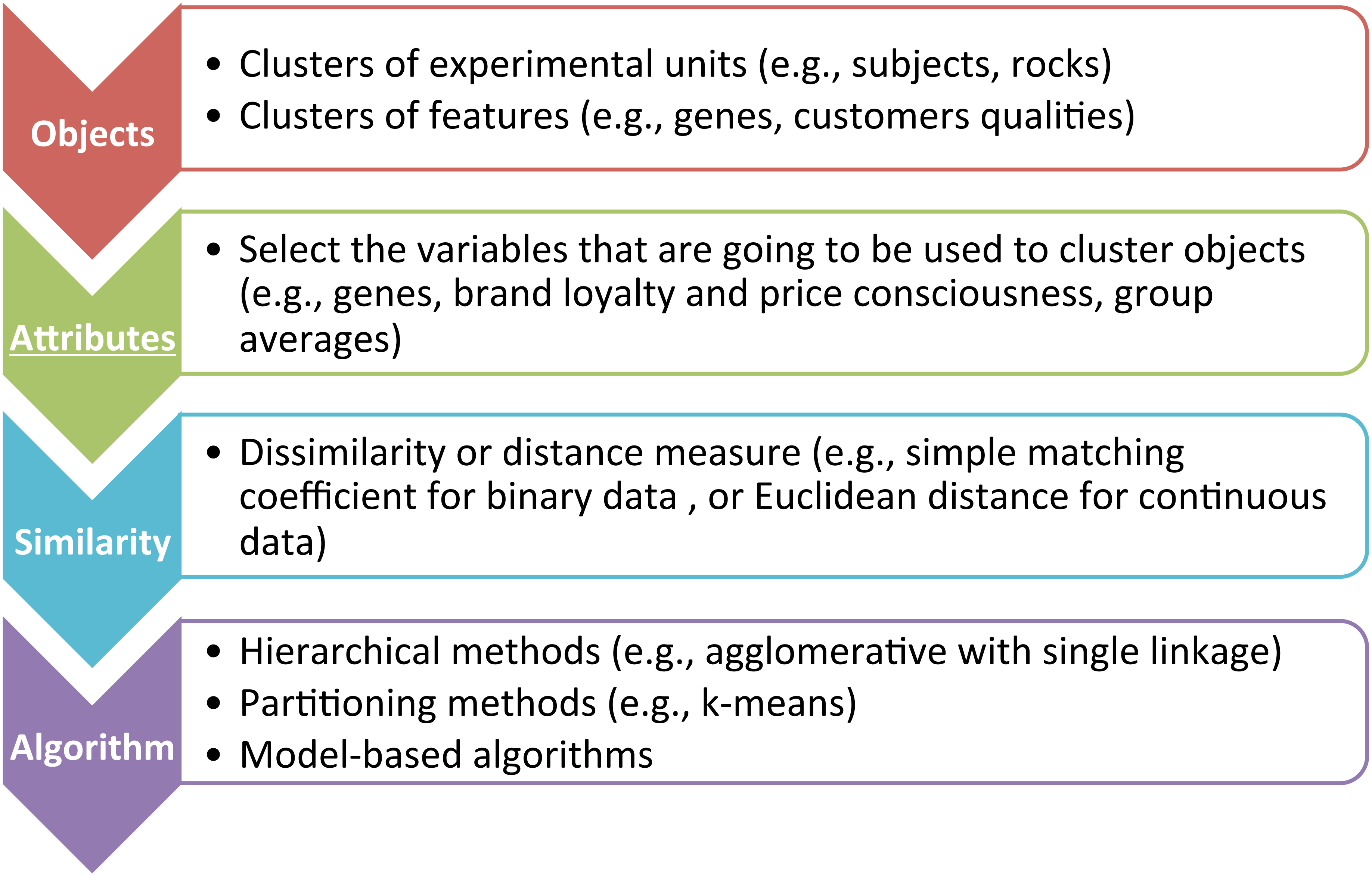
- **GAP Statistic:** Tibshirani, Walther and Hastie, *Journal of the Royal Statistical Society*, **63, 411-423.**
- http://www.jstor.org/stable/2680607
- Slightly modified code: http://stat.rutgers.edu/~rebecka
- Clest Algorithm: Dudoit, Fridlyand, 2002. *Genome Biology* **3**(7): research0036.1 -0036.21

  http://genomebiology.com/2002/3/7/research/0036
- Dudoit, Fridlyand, 2003. *Bioinformatics* **19**, 1090-1099. http://bioinformatics.oxfordjournals.org/cgi/content/abstract/19/9/1090
- Ben-Hur, Elisseeff, Guyon, 2002. *Pacific Symposium on Biocomputing* **7**: 6-17

  http://www.ncbi.nlm.nih.gov/pubmed/11928511

Early conclusions

- Different algorithms, different distance metrics will produce different clusterings
  - which is best?
- Algorithms may or may not produce what you would have picked "by eye"
- Algorithms may or may not actually optimize some relevant objective function



Objects

- Clusters of experimental units (e.g., subjects, rocks)
- Clusters of features (e.g., genes, customers qualities)

Attributes

- Select the variables that are going to be used to cluster objects (e.g., genes, brand loyalty and price consciousness, group averages)

Similarity

- Dissimilarity or distance measure (e.g., simple matching coefficient for binary data , or Euclidean distance for continuous data)

Algorithm

- Hierarchical methods (e.g., agglomerative with single linkage)
- Partitioning methods (e.g., k-means)
- Model-based algorithms

Number of clusters???

Choosing the attributes

- When attributes are noisy, so is the resulting clustering
- photoRec: the attribute for each gene will be:

$$\mathbf{X}_g = (X_{g1,\mathrm{E16}}, X_{g2,\mathrm{E16}}, \ldots, X_{g39,\mathrm{4\_week}})$$

- Model: $X_{gi,\mathrm{DS}} = \mu_{g,\mathrm{DS}} + \varepsilon_{gi,\mathrm{DS}}$
- Parameter-based attribute:

$$(\mu_{g,\mathrm{E16}}, \mu_{g,\mathrm{P2}}, \mu_{g,\mathrm{P6}}, \mu_{g,\mathrm{P10}}, \mu_{g,\mathrm{4w}})$$

- **Estimated attribute**: use the within-DS averages (vectors of size 5, instead of vectors of size 39!)

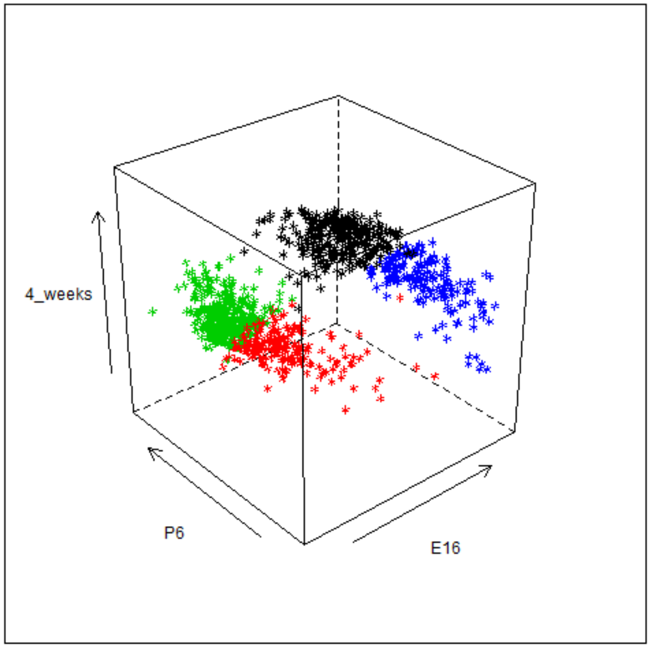
Other model-based alternatives

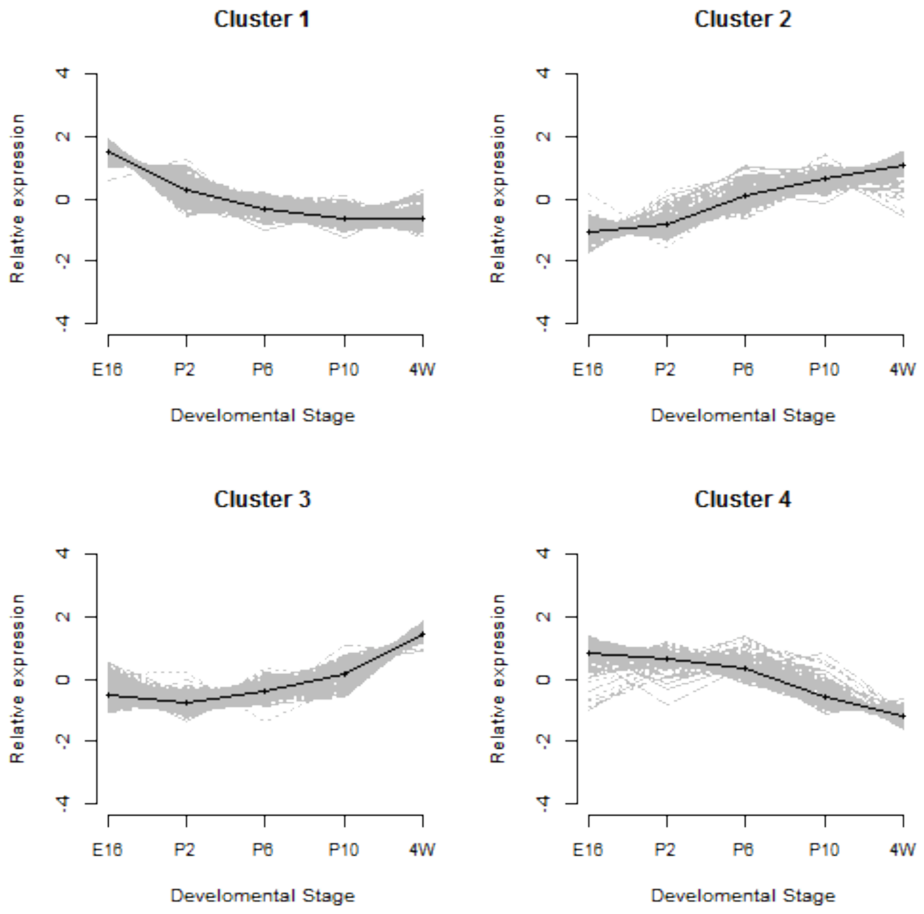
- If ‘time’ is a condition of interest, the attribute could be the time-specific expectations
- For example, we can fit a linear models and use the regression parameters as attributes:

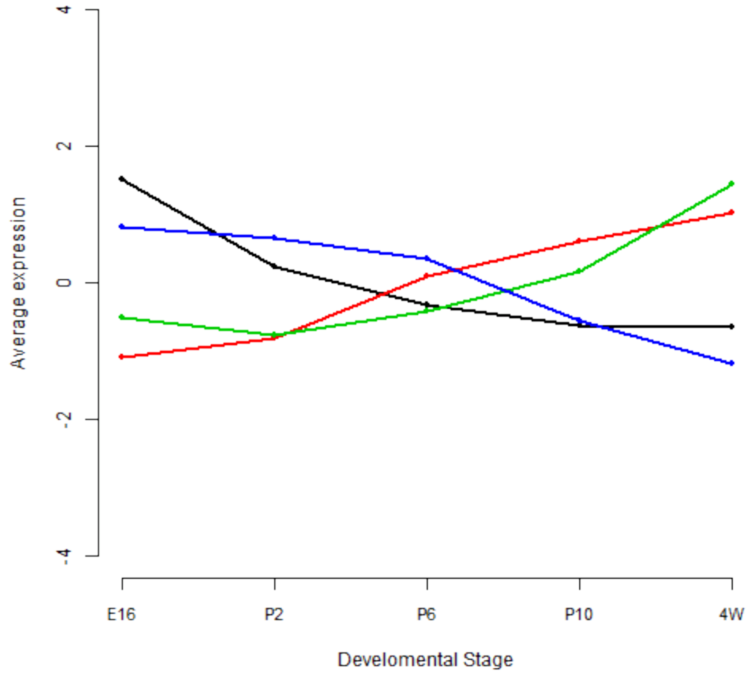
$$X_g(t) = \beta_{0g} + \beta_{1g}t + \beta_{2g}t^2 + \varepsilon_g(t)$$

$$Attr_g = (\beta_{1g}, \beta_{2g})$$

Gene Clustering (k-means, k=4) based on DS attributes



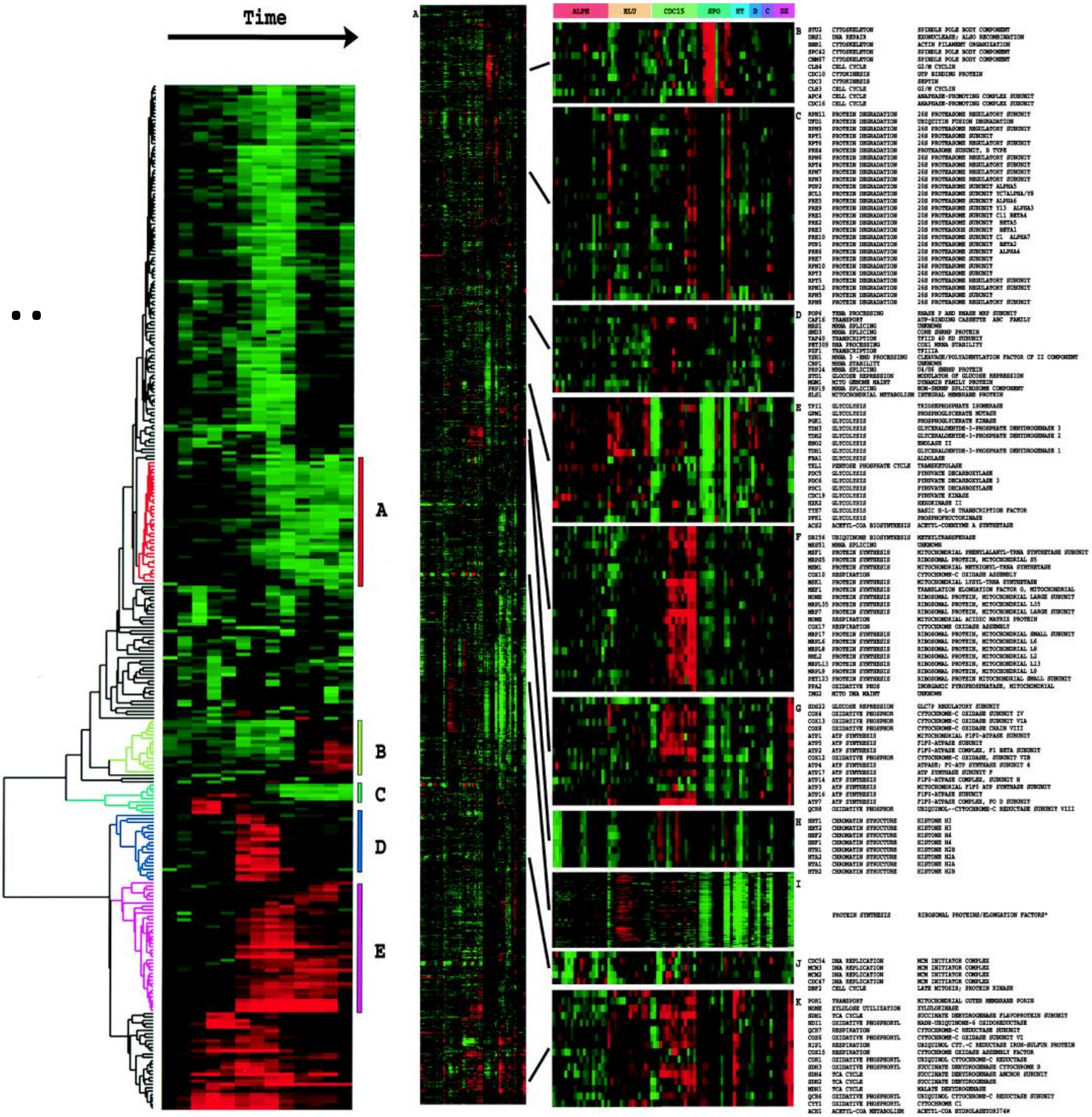




Summary Notes

- Many choices to make when you want to cluster a set of objects:
  - Objective, algorithm, **attributes/features**, distance metric, number of clusters.
- Not possible to say which method is best. It all depends on data and goal.
- Clustering is very powerful, but reckless application leads to misguided conclusions.
- CA is still a good way to explore the data and summarize results

A known analysis in an 'omic' paper ...



Eisen, et al (1998)

References on Cluster Analysis

- “Applied multivariate statistical analysis” by R. A. Johnson and D. W. Wichern. Prentice-Hall.
- “Finding Groups in Data: An Introduction to Cluster Analysis” by Leonard Kaufman, P J Rousseeuw. Wiley Blackwell, 2005.
- The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Trevor Hastie, Robert Tibshirani,and Jerome Friedman.New York: Springer-Verlag, 2001. ISBN 0-387-95284-5. (available online at library.ubc.ca)
- “Problems in gene clustering based on gene expression data” by J. Bryan. Journal of Multivariate Analysis 90 (2004) (special issue on Bioinformatics).

Bonus Slides

Algorithms: k-means

Note that

$$\begin{aligned}
\sum_{i=1}^{n}\sum_{j=1}^{n} d\left(\mathbf{X}_i, \mathbf{X}_j\right) &= \sum_{r=1}^{K}\sum_{i\in\mathcal{C}_r}\sum_{j=1}^{n} d\left(\mathbf{X}_i, \mathbf{X}_j\right) \\
&= \sum_{r=1}^{K}\sum_{i\in\mathcal{C}_r}\left[\sum_{j\in\mathcal{C}_r} d\left(\mathbf{X}_i, \mathbf{X}_j\right) + \sum_{j\notin\mathcal{C}_r} d\left(\mathbf{X}_i, \mathbf{X}_j\right)\right] \\
&= \sum_{r=1}^{K}\sum_{i,j\in\mathcal{C}_r} d\left(\mathbf{X}_i, \mathbf{X}_j\right) + \sum_{r=1}^{K}\sum_{i\in\mathcal{C}_r}\sum_{j\notin\mathcal{C}_r} d\left(\mathbf{X}_i, \mathbf{X}_j\right)
\end{aligned}$$

$$T = W + B$$

When $d(\mathbf{X}_i, \mathbf{X}_j) = \|\mathbf{X}_i - \mathbf{X}_j\|^2$

$$W = \sum_{r=1}^{K} \sum_{i,j \in \mathcal{C}_r} \|\mathbf{X}_i - \mathbf{X}_j\|^2 = \sum_{r=1}^{K} \sum_{i \in \mathcal{C}_r} \|\mathbf{X}_i - \bar{\mathbf{X}}_r\|^2$$

- Given $\bar{\mathbf{X}}_1, \bar{\mathbf{X}}_2, \ldots, \bar{\mathbf{X}}_K$, the minimum of *W* is attained assigning $\mathbf{X}_i$ to the cluster C_r with the closest mean ($\bar{\mathbf{X}}_r$).

- Given $C_1, C_2, \ldots, C_K$, the minimum of *W* is attained estimating the center of the cluster with its sample mean $\bar{\mathbf{X}}_r$.

$$\min_{\hat{\mu}_1, \ldots, \hat{\mu}_K} \sum_{r=1}^{k} \sum_{i \in \mathcal{C}_r} \|\mathbf{x}_i - \hat{\mu}_r\|^2 \longrightarrow \hat{\mu}_r = \bar{\mathbf{X}}_r = \frac{1}{n_r} \sum_{i \in \mathcal{C}_r} \mathbf{x}_i$$

Distance measures for binary data are based on the number of matches and mismatches:

	0	1
0	m	r
1	s	t

$$a_i \in \{0,1\}, b_i \in \{0,1\}, i = 1,\ldots, p$$

$$m = \#\{a_i = b_i = 0, i = 1,\ldots, p\}$$

$$r = \#\{a_i = 0, b_i = 1, i = 1,\ldots, p\}$$

…

– Example:

	Gender	Race	Obese	Smoke
Subject a	0	0	1	1
Subject b	1	0	1	1

$$m = 1, t = 2$$

$$r = 1, s = 0$$

- Typical examples of distance measures for binary data are
  - Simple matching coefficient

$$a_i \in \{0,1\}, b_i \in \{0,1\}, i = 1,\ldots, p : d(a,b) = \frac{m+t}{m+r+s+t}$$

  - Jaccard coefficient

$$a_i \in \{0,1\}, b_i \in \{0,1\}, i = 1,\ldots, p : d(a,b) = \frac{m}{m+r+s}$$