STAT540
Lecture 16: March 9th 2015

Clustering: problem, objectives, and algorithms

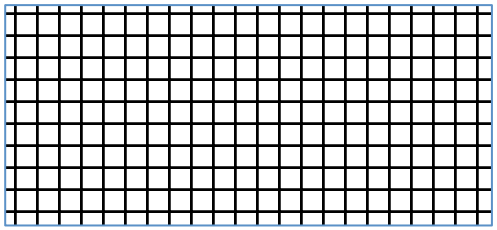
Sara Mostafavi
Department of Statistics
Department of Medical Genetics
Center for Molecular Medicine and Therapeutics

** Many thanks to Drs. Gabriela Cohen-Freue and Jenny Bryan for lecture slides**

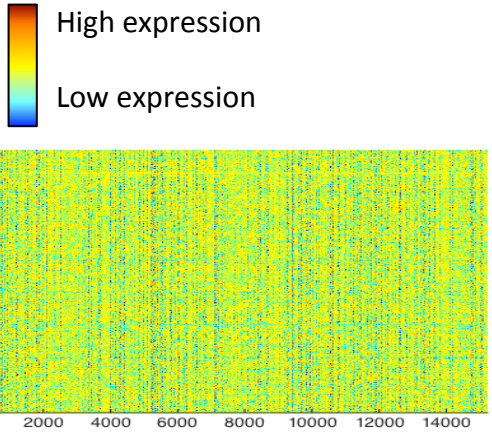
Visualizing “raw” expression data (without clustering) is NOT informative…

Genes (expression levels)

Samples (individuals/animals/ conditions)



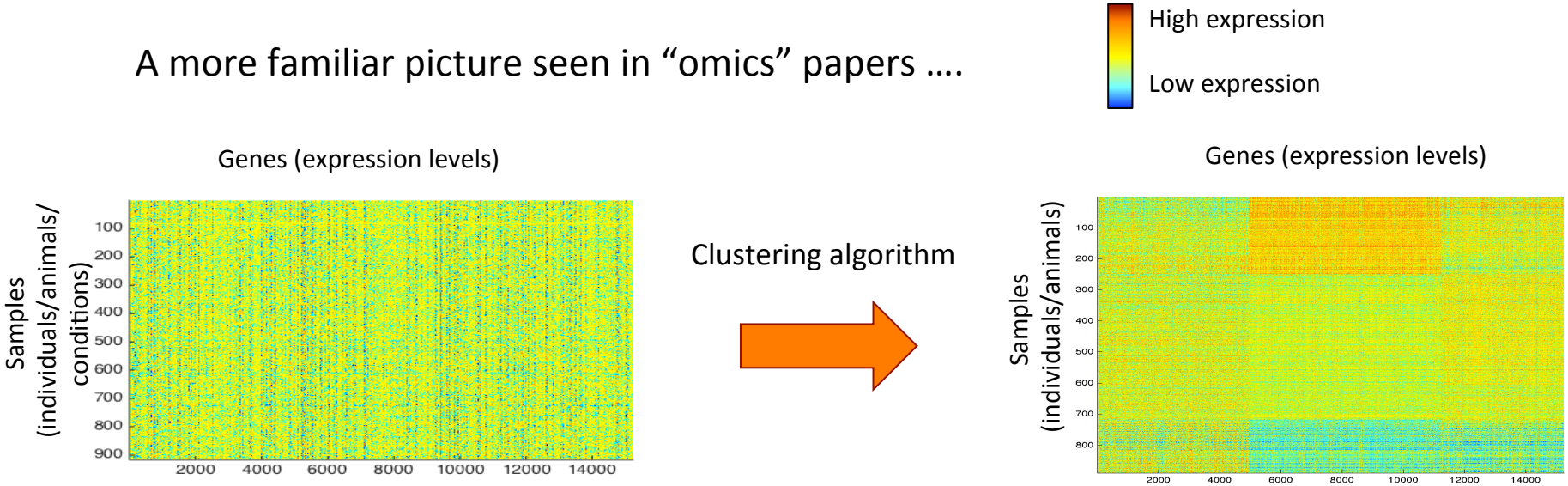
Visualizing your data:
(e.g., heatmap function in R
imagesc in MATLAB)



Matrix X with dimensions n by p (n rows and p columns)

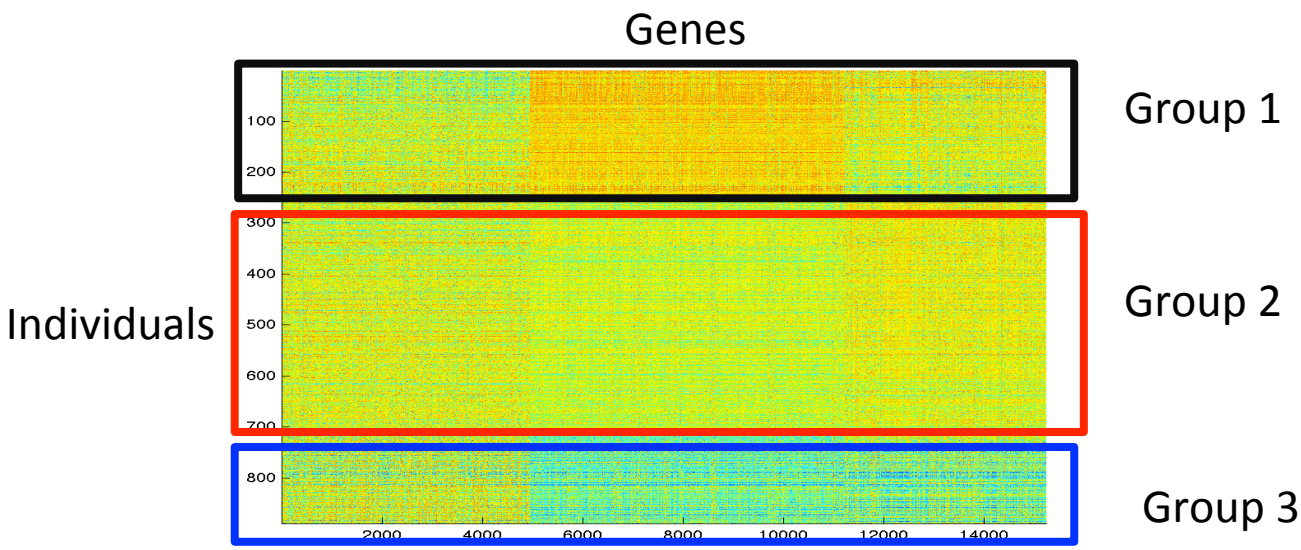
Pervasive application of clustering in analysis of gene expression data

A more familiar picture seen in “omics” papers ….



Two predominant application of clustering in gene expression studies

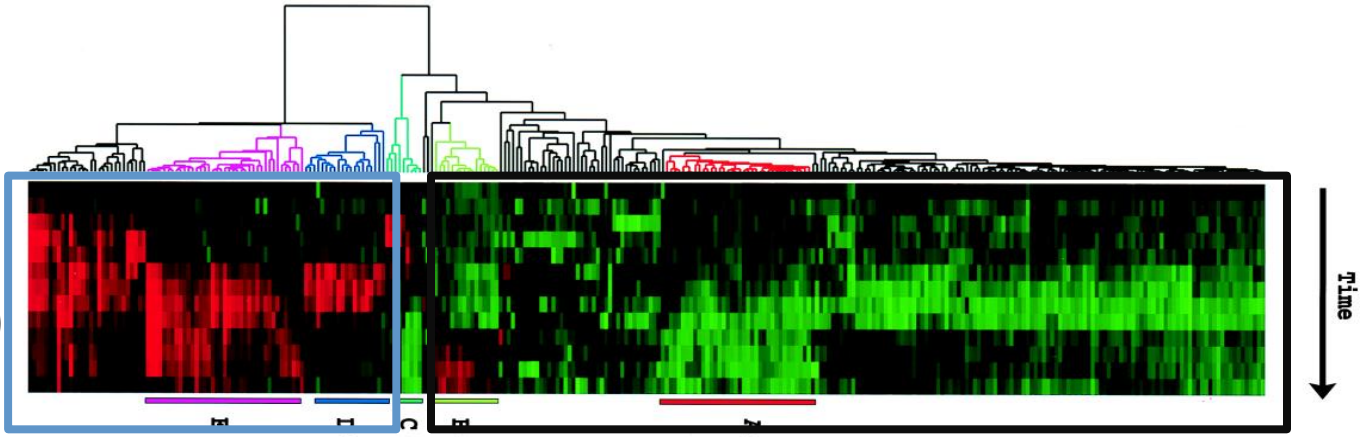
1. Identify groups of individuals that have similar expression profiles:
   - Identifying disease sub-types



Two predominant application of clustering in gene expression studies

2. Identify groups of genes that have correlated expression profiles:
   - Informative of co-functionality (genes in the same “cluster” perform the same function)

Conditions
(Time point)



Genes

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".

- Rigorous application of clustering is very powerful but also hard to do (computational complexity, suitable definition of clustering objective, determining the number of clusters …)

- Clustering in bioinformatics is often used as a tool for visualization, hypothesis generation, selection of genes for further analysis.

  – Keep in mind, with typical use of clustering in bioinformatics: there is no measure of "strength of evidence" or "strength of clustering structure" provided.

Origins of clustering:
Machine learning (sub-field of CS) & Statistics

Computer Science:

- Machine Learning
  - Unsupervised learning
    - Clustering algorithms

Special insights & emphasis:

- Analyzing computational difficulty of the problem.
- Designing **Algorithms** for solving a given clustering problem.

Statistics:

- Data Mining
  - Density estimation/Clustering

Special insights:

- How do we determine the optimal clustering algorithm (model selection/justification for number of parameters).

Three key concepts with distinct definitions:

1) The clustering problem
2) A clustering objective function (model)
3) A clustering algorithm

Clustering problem: Definition

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

Cluster some rocks:

Clustering: Definitions

- Goal: place a set of objects into groups or **clusters**.
- How do we do this?
  - gather a set of attributes for each object.
  - Place objects in clusters so that objects within each cluster are more similar to each other compared to objects that outside their group/cluster.

Rocks were clustered according to their color and texture.

A clustering objective function

- Goal (the clustering problem):place a set of **objects** into groups or **clusters** in a way that **similar** objects are in the same cluster.
- How do we do this?
  - gather a set of attributes for each object.
  - Place objects in clusters so that objects within each cluster are more similar to each other, based on their attributes, compared to objects that outside their group/cluster.

→ Clustering **objective** function: maximize within cluster similarity

- A precise definition of "good/optimal" clustering: precise enough to be translated into an equation.

Defining attribute/feature vector for each object

- We need to numerically define a attribute or feature vector that describes the relevant properties of each object

Set of objects $\{\vec{x}_1, \vec{x}_2, \vec{x}_3, \ldots, \vec{x}_n\}$

Each object is represented by a numerical vector: $\vec{x}_1 \subseteq \Re^p$

Attribute/feature p for object 1

Rock1: $\vec{x}_1 = (x_1^{(1)}, x_2^{(1)}, \ldots, x_p^{(1)})$

Numerical value representing texture

Numerical value for color/shade

Commonly Used Measures of Similarity and Distance

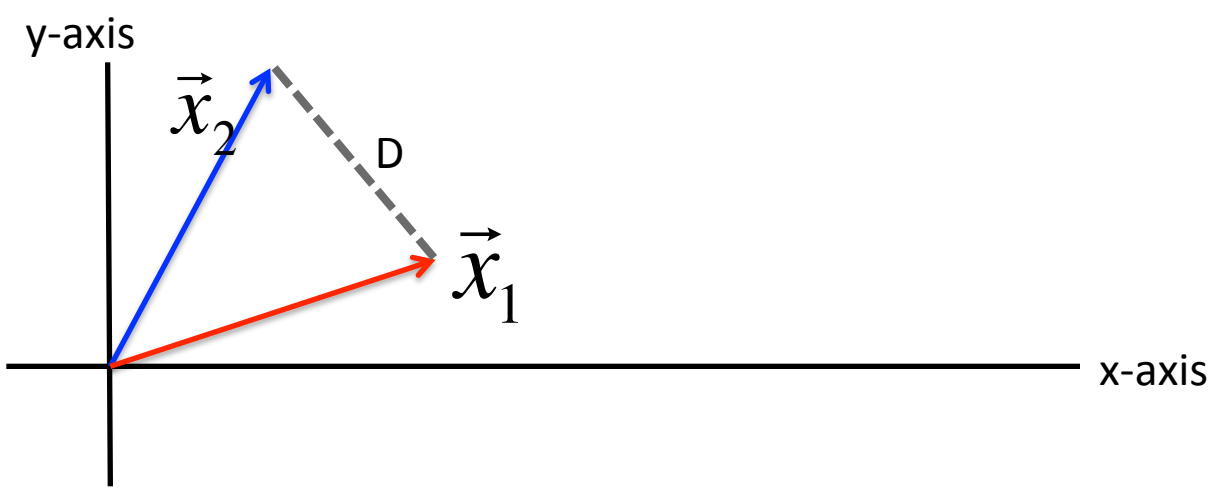
- Every clustering method is based on the measure of distance or similarity.
- We need to compute pairwise similarities between all objects.

- Typical distance/similarity measures:
  - Distance:
    - Euclidean
    - Manhattan
  - Dissimilarity: 1 – Correlation
    - Spearman
    - Pearson

Commonly Used Measures of Similarity and Distance

- Euclidian distance between two feature vectors: $\vec{x}_1$ and $\vec{x}_2$

$$D = \| \vec{x}_1 - \vec{x}_2 \|_2 = \sqrt{\sum_{i=1}^{p} (x_i^{(1)} - x_i^{(2)})^2}$$

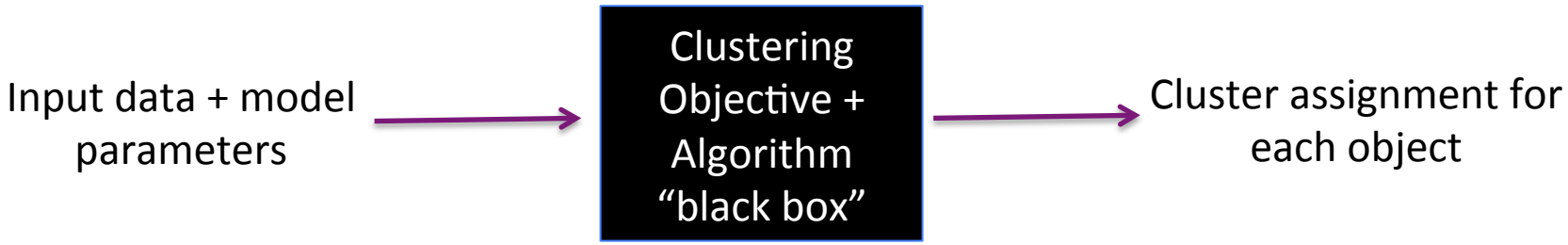


What is an algorithm?

What is an algorithm?

- An algorithm is a self-contained step-by-step set of operations to be performed in order to achieve a given task. Through a set of steps, an algorithm transforms a given input data into the desired output.

Clustering algorithm from a machine learning perspective: what are the inputs and the outputs?



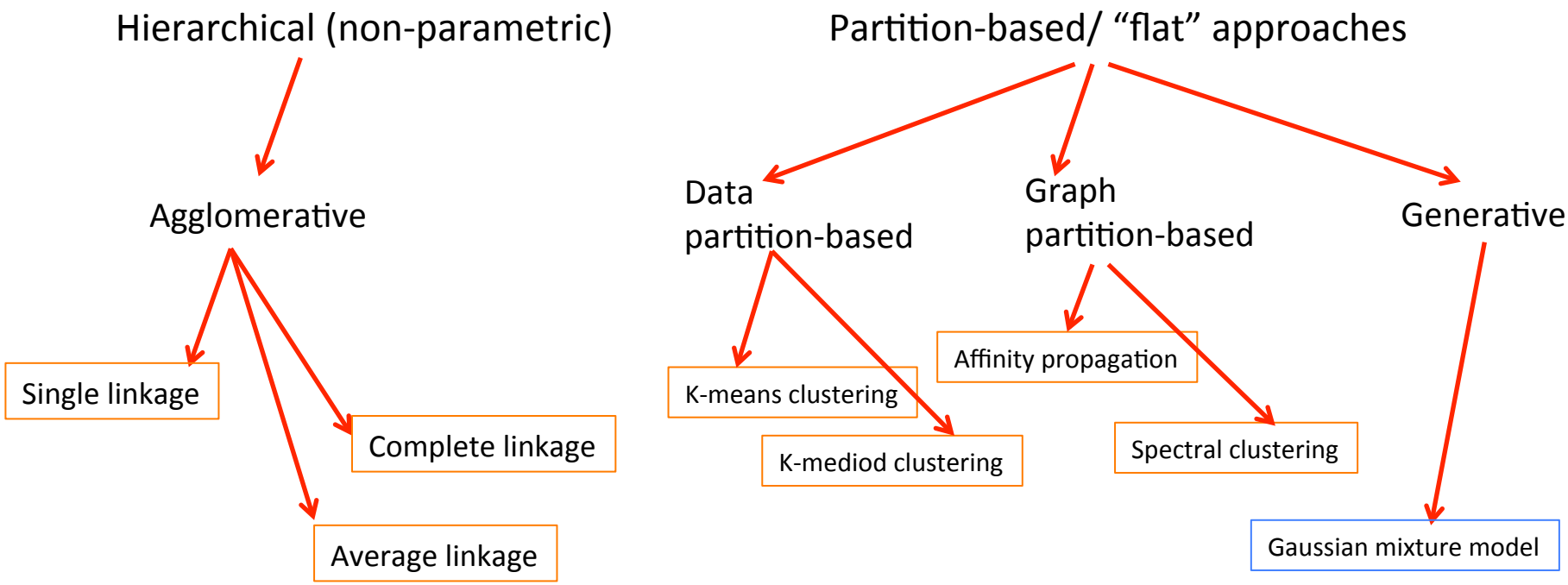
Input: 1) data matrix X_{nxp} (rows are the objects)

2) number of clusters k

Output: an assignment of cluster membership for each object. $C_{nx1} = \{1,k\}^n$, C_i=k if object i is placed in cluster k.

(Note the output vector can also be a probabilistic assignment, we'll ignore this for now.)

Some existing clustering algorithms



Discrete clustering assignment
Probabilistic cluster assignment

Some existing clustering algorithms

Hierarchical (non-parametric)

Partition-based/ “flat” approaches

Single l

Almost all clustering algorithm that partition the objects require user to define the number of clusters.

(there are ways of automatically determining the number clusters.)

Discrete clustering assignment

Probabilistic cluster assignment

K-means clustering objective function

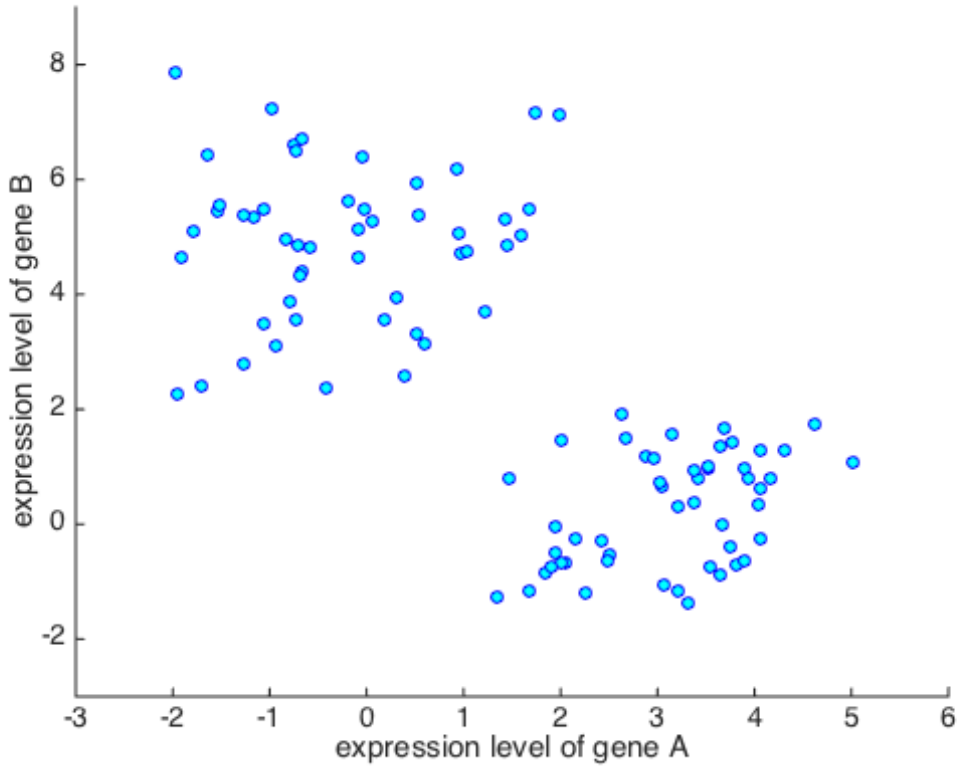
- One of the most widely used partition-based clustering approaches.
- **Objective function**: minimize the average squared Euclidean distance of objects from their assigned cluster centers. A cluster center (or centroid) is defined as the mean of objects in the given cluster.

K-means clustering objective function

- One of the most widely used partition-based clustering approaches.
- **Objective function**: minimize the average squared **Euclidean distance** of objects from their assigned cluster centers. A **cluster center** (or centroid) is defined as the mean of objects in the given cluster.

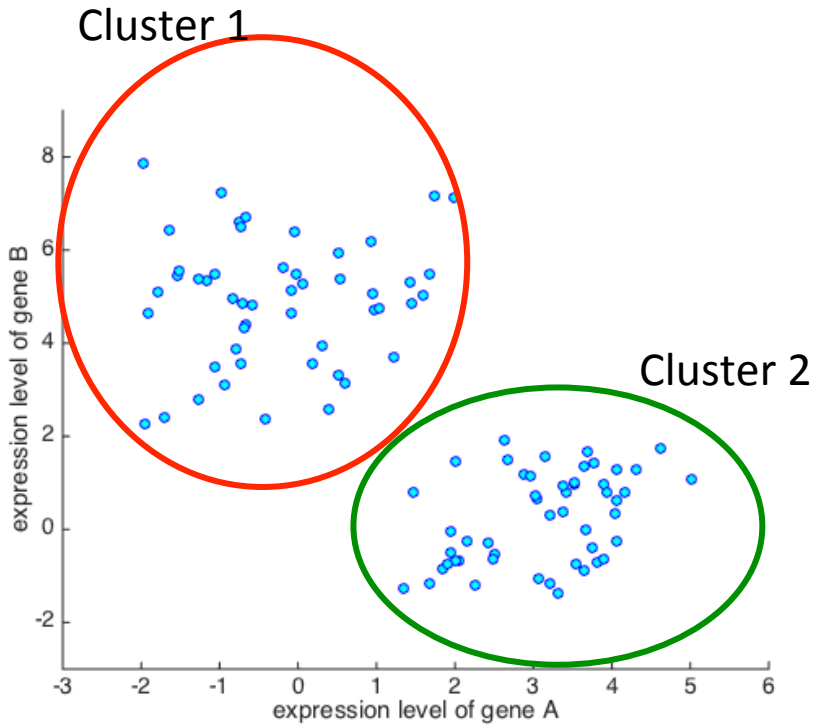
How many clusters are there?

Suppose you measured expression levels for 2 genes (gene A and gene B) for 100 individuals



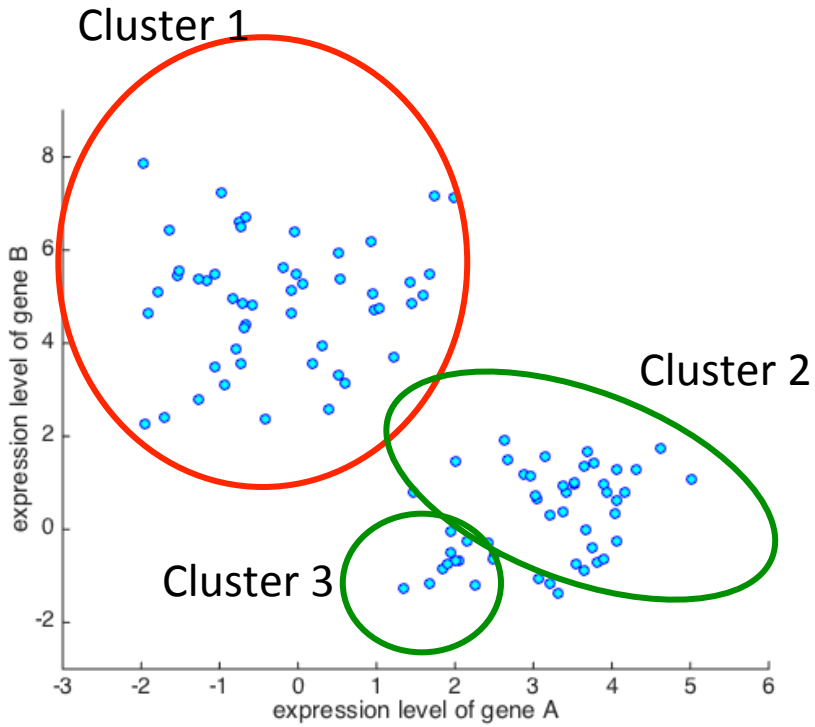
How many clusters are there?

Suppose you measured expression levels for 2 genes (gene A and gene B) for 100 individuals



How many clusters are there?

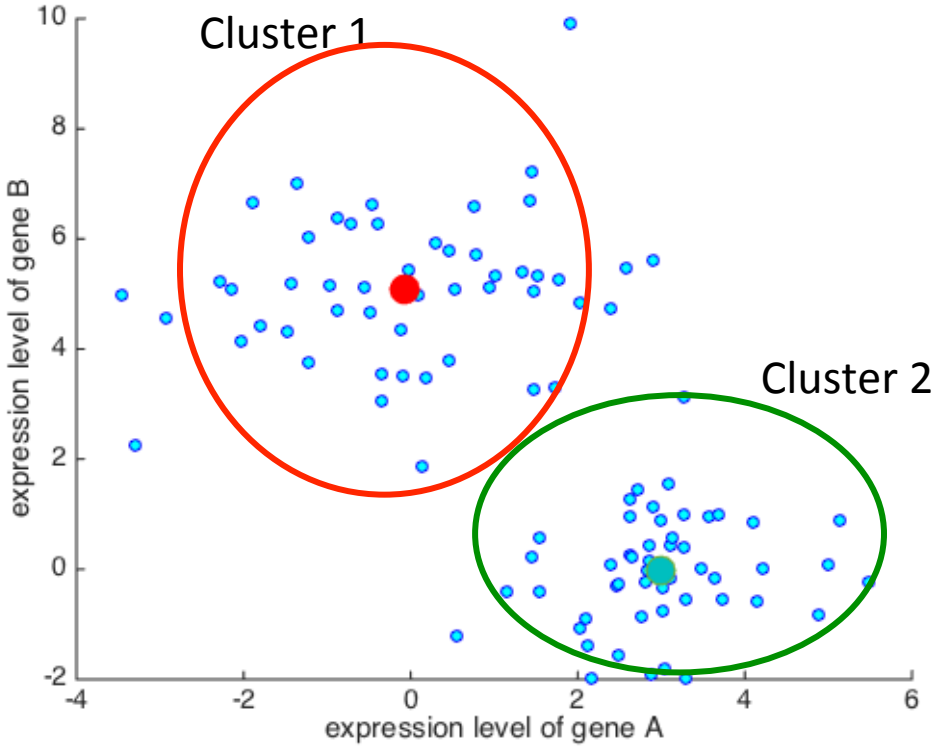
Suppose you measured expression levels for 2 genes (gene A and gene B) for 100 individuals



K-means objective function

Objective function: minimize the average squared **Euclidean distance** of objects from their assigned cluster centers. A **cluster center** (or centroid) is defined as the mean of objects in the given cluster.

Computing the "mean/centroid" for each cluster:



K-means objective function (formula/equation)

Objective function: minimize the average squared **Euclidean distance** of objects from their assigned cluster centers. A **cluster center** (or centroid) is defined as the mean of objects in the given cluster.

* N objects, each have p attributes: $\{\vec{x}_1, \vec{x}_2, \vec{x}_3, \ldots, \vec{x}_n\}$, $\vec{x}_1 \subseteq \Re^p$

* Attribute vector for object 1: $\vec{x}_1 = (x_1^{(1)}, x_2^{(1)}, \ldots, x_p^{(1)})$

* k-means objective function:

$$J = \sum_{i=1}^{n} \sum_{j=1, i \subseteq k}^{k} \underbrace{||\vec{x}_i - \vec{u}_k||_2}$$

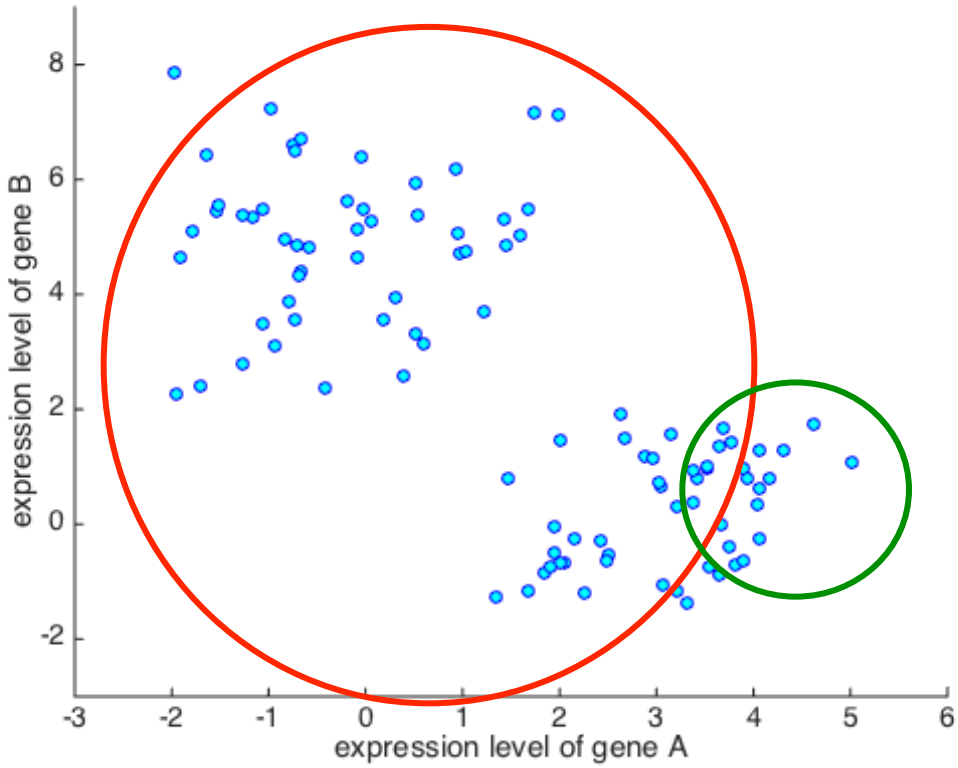
Euclidian distance between x_i and u_k

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

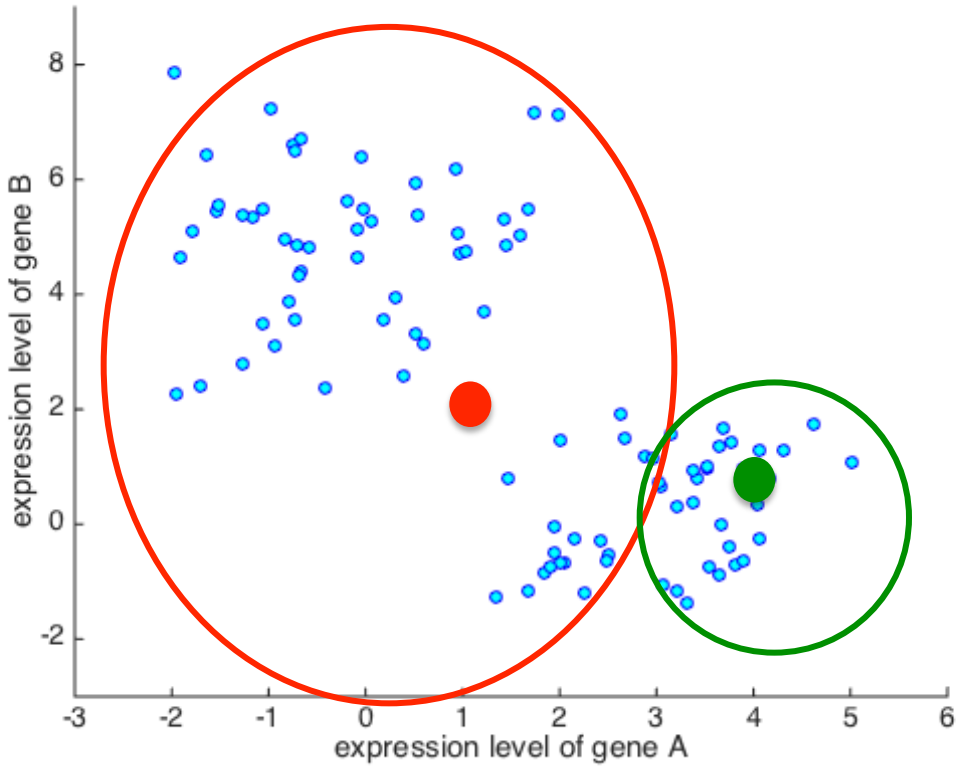
K-means objective function

Step 1: partition space in to two clusters (e.g., randomly assign objects to one of two clusters)



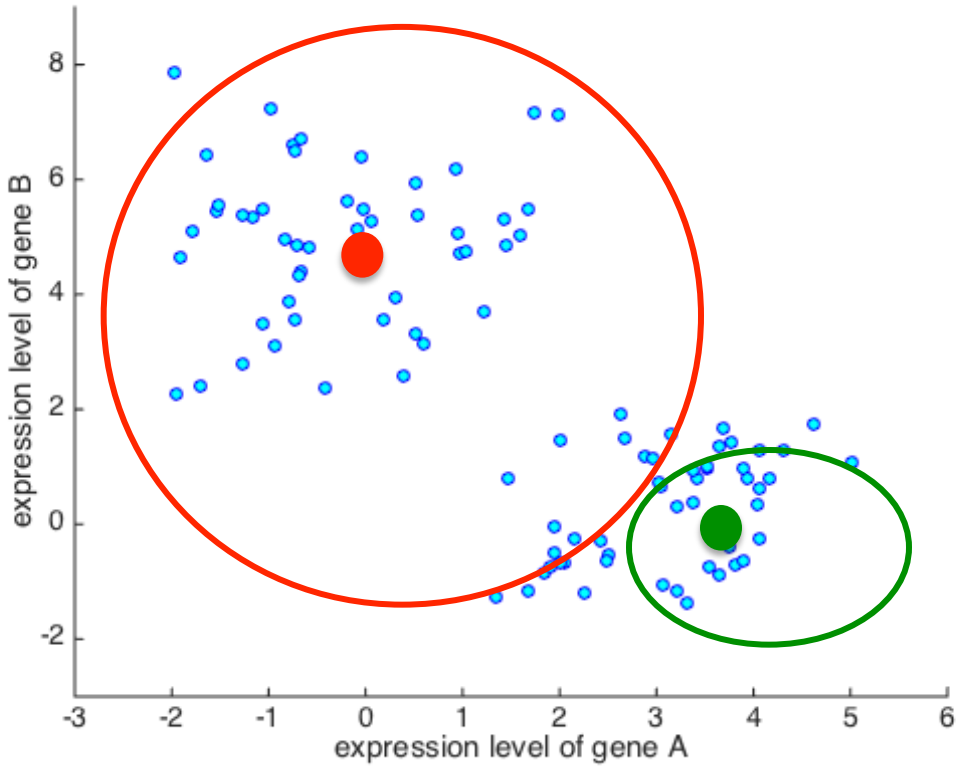
K-means objective function

Step 1: partition space in to two clusters (e.g., randomly assign objects to one of two clusters) – initialize cluster centers.



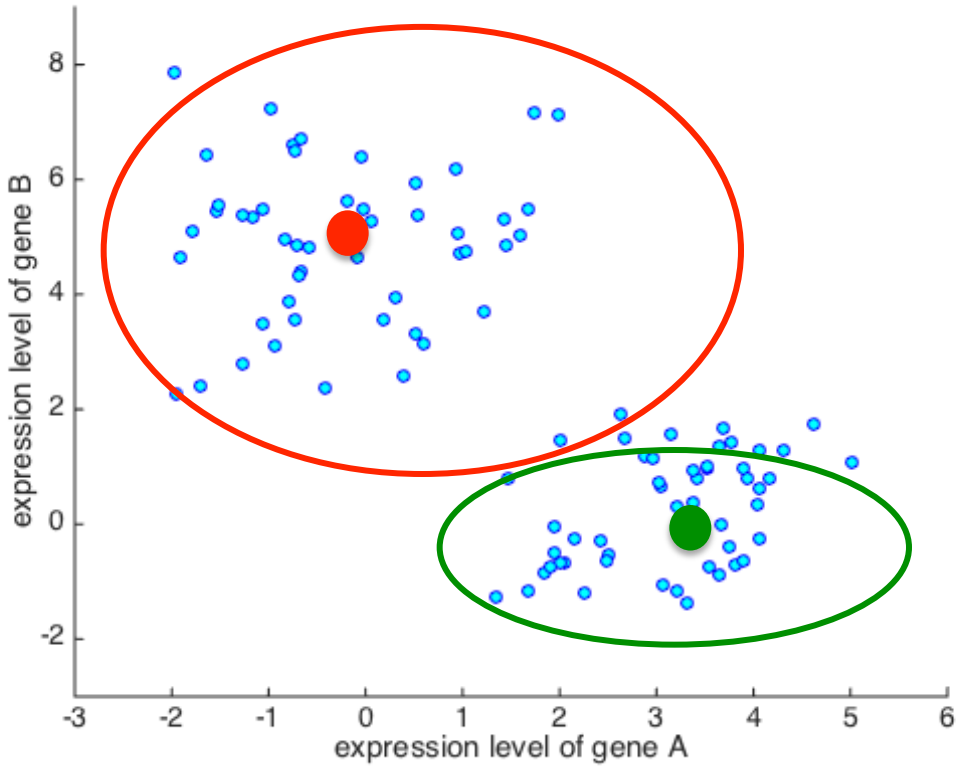
K-means objective function

Step 2: computer distances between each object and all cluster centers, then re-assign each object to its closest cluster.



K-means objective function

Step 3: recalculate the cluster means for each cluster.



Algorithms: k-means

Note that

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

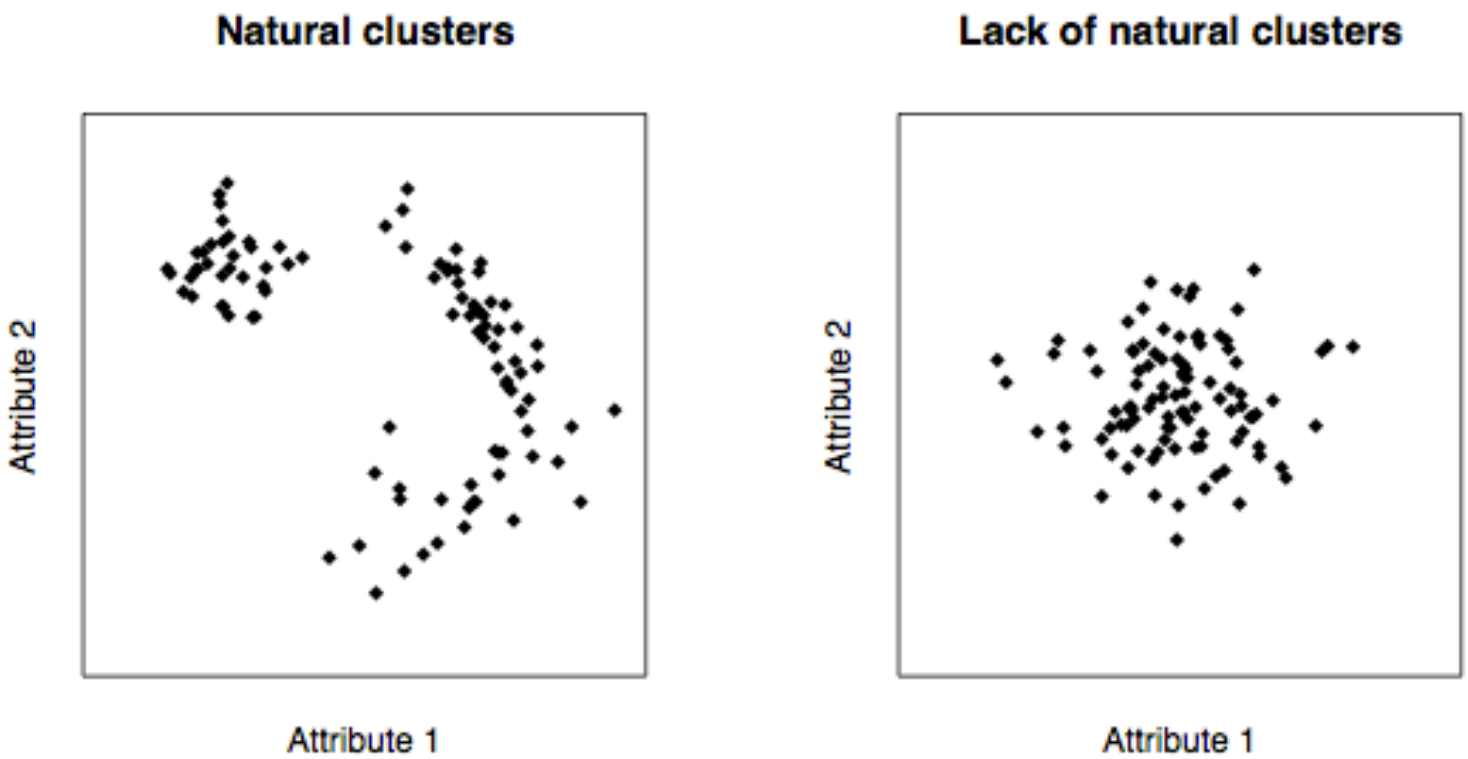
$$
T = W + B
$$

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

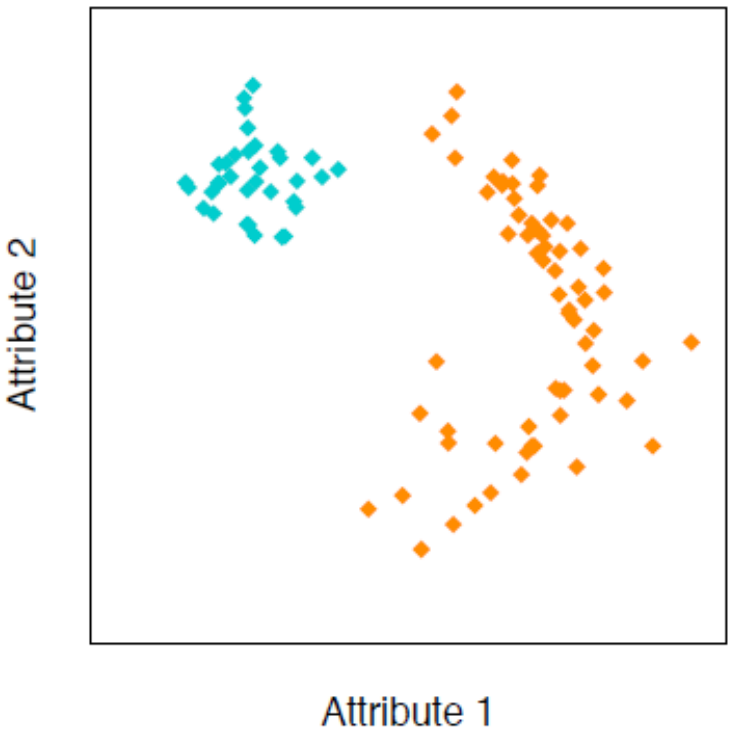
$$W = \sum_{r=1}^{K} \sum_{i,j \in C_r} \|\mathbf{X}_i - \mathbf{X}_j\|^2 = \sum_{r=1}^{K} \sum_{i \in C_r} \|\mathbf{X}_i - \bar{\mathbf{X}}_r\|^2$$

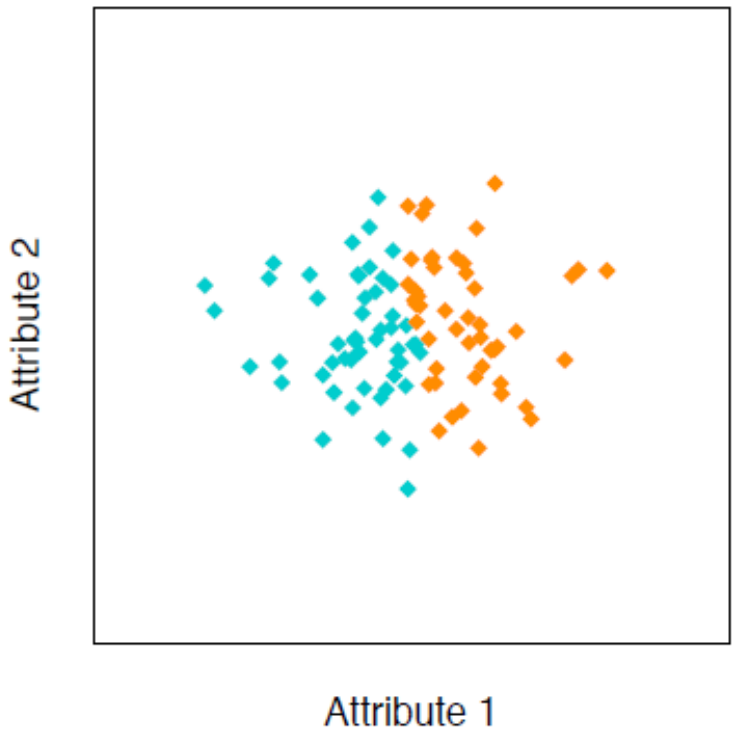
- Given $\overline{\mathbf{X}}_1, \overline{\mathbf{X}}_2, \ldots, \overline{\mathbf{X}}_K$, the minimum of W is attained assigning $\mathbf{X}_i$ to the cluster C_r with the closest mean ($\overline{\mathbf{X}}_r$).
- Given $C_1, C_2, \ldots, C_K$, the minimum of W is attained estimating the center of the cluster with its sample mean $\overline{\mathbf{X}}_r$.

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



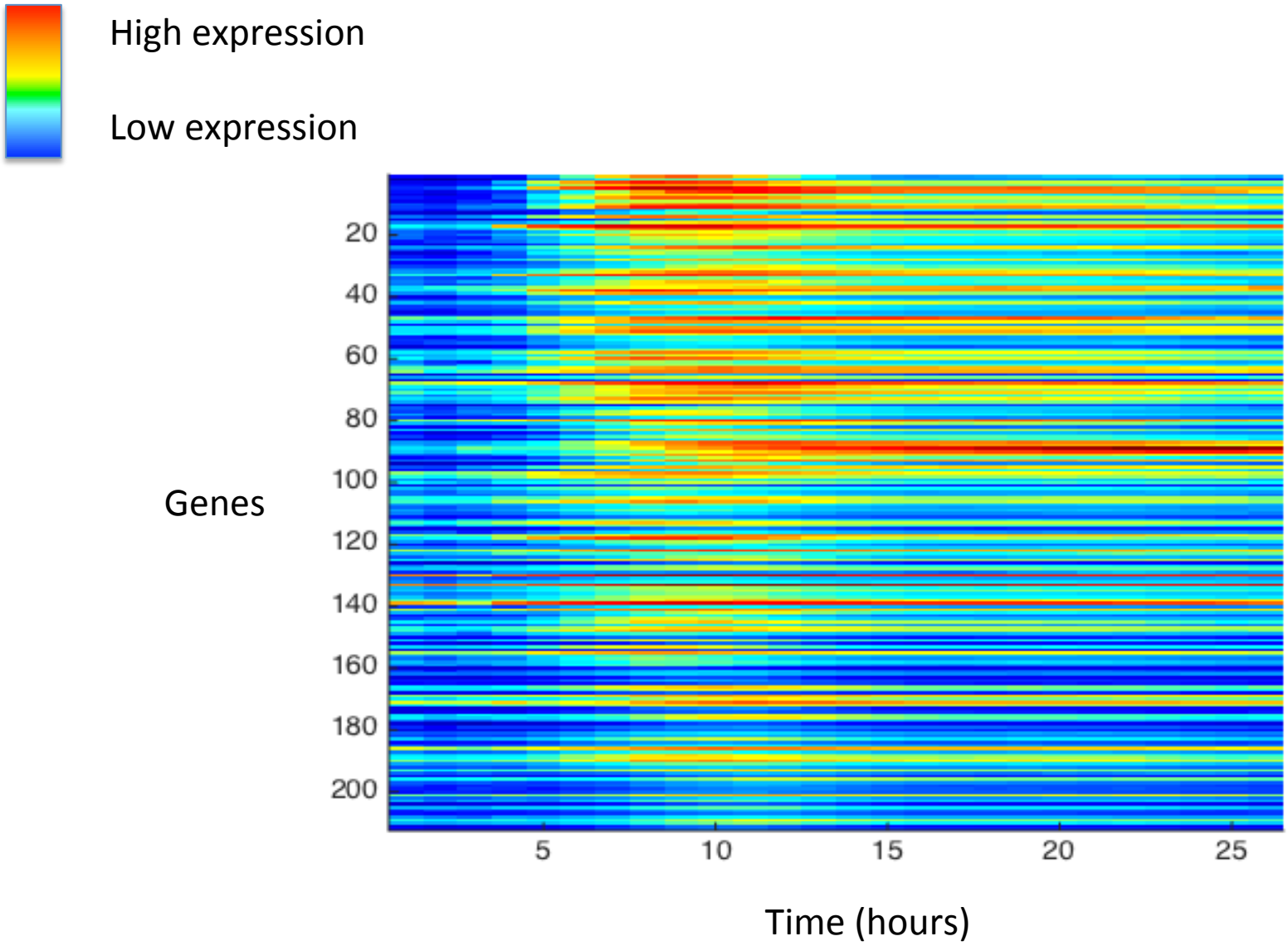
Natural clusters are regions in the attribute space that are densely populated, separated from other such regions by areas that are sparsely populated -- “internal cohesion” and “external isolation”





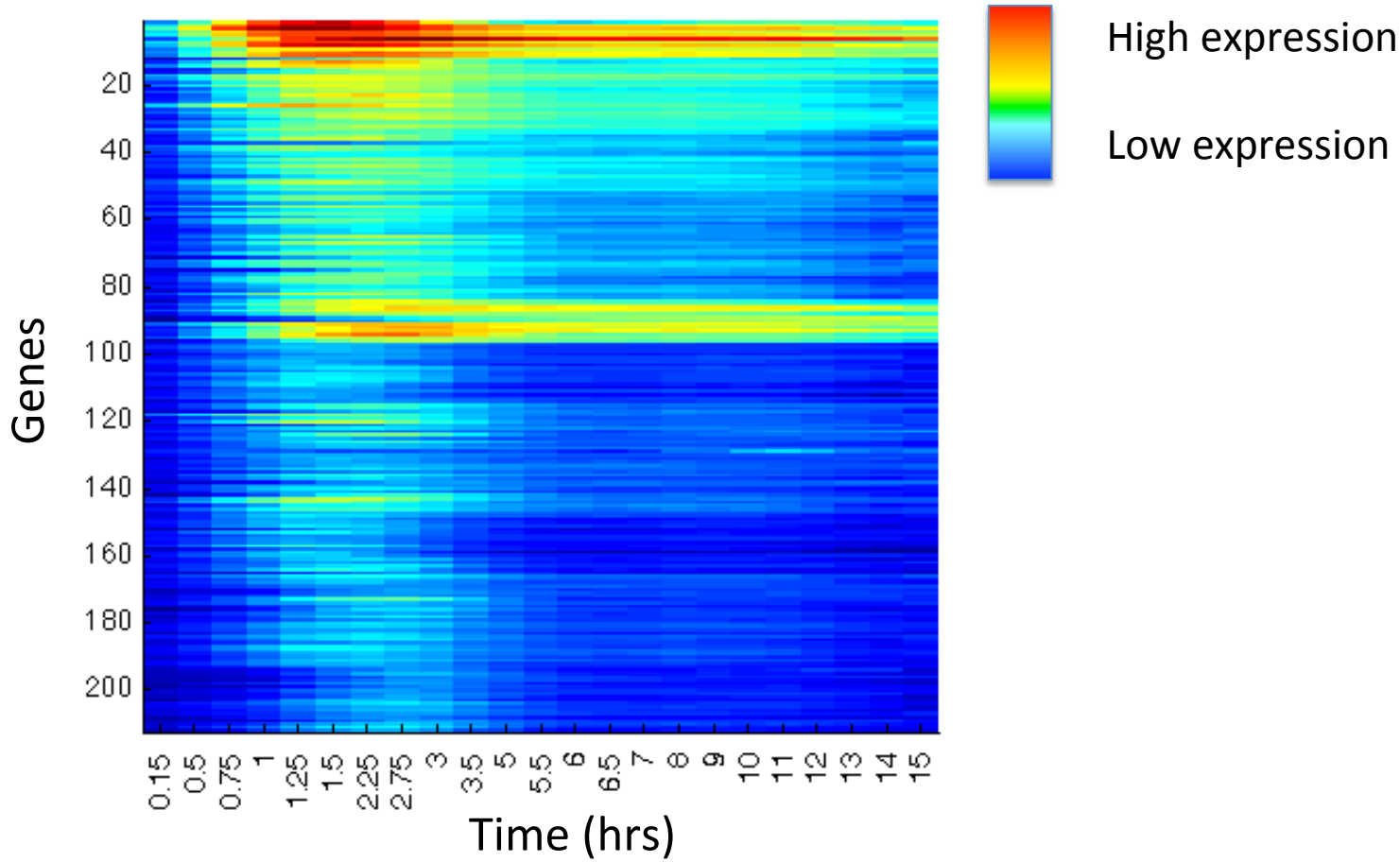
In the absence of natural clusters, grouping is called **data segmentation**. Not in the control of the analyst or the algorithm.

Timing patterns for IFN induced genes in CD19^{+} Bcells



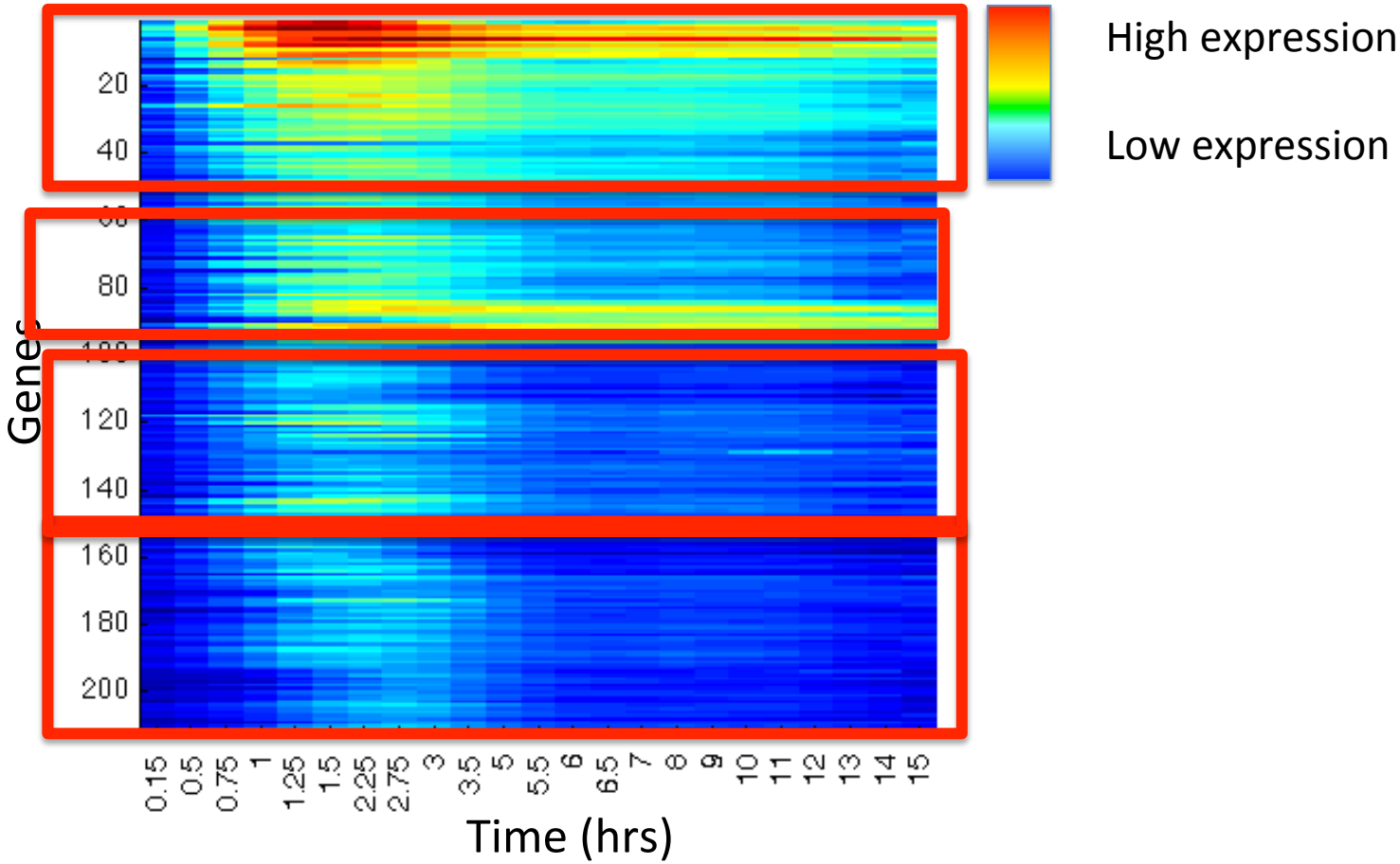
Timing patterns for IFN induced genes in CD19^{+} Bcells

Application of k-means clustering with k=4



Timing patterns for IFN induced genes in CD19^{+} Bcells

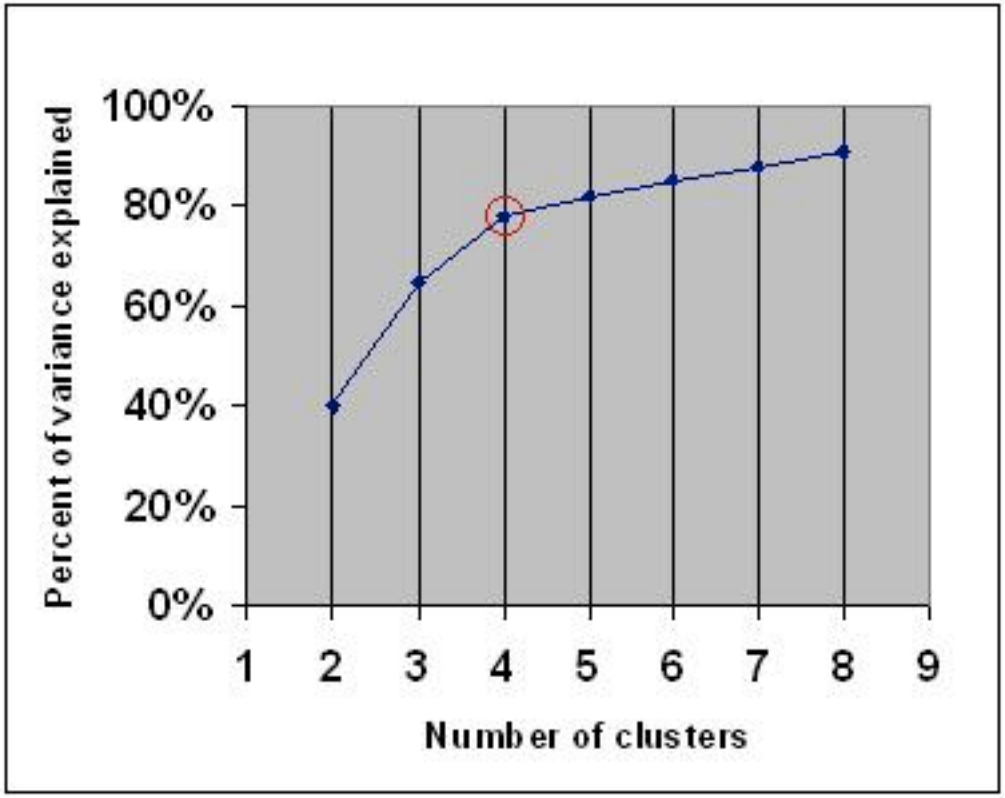
Application of k-means clustering with k=4



How do you determine *k* (number of clusters)?

Note: maximizing the clustering likelihood/objective will not be informative→ each object should be in its own cluster. Therefore, need an algorithm that takes into account the "cost" of additional clusters.

- Prior knowledge
- The "elbow method"



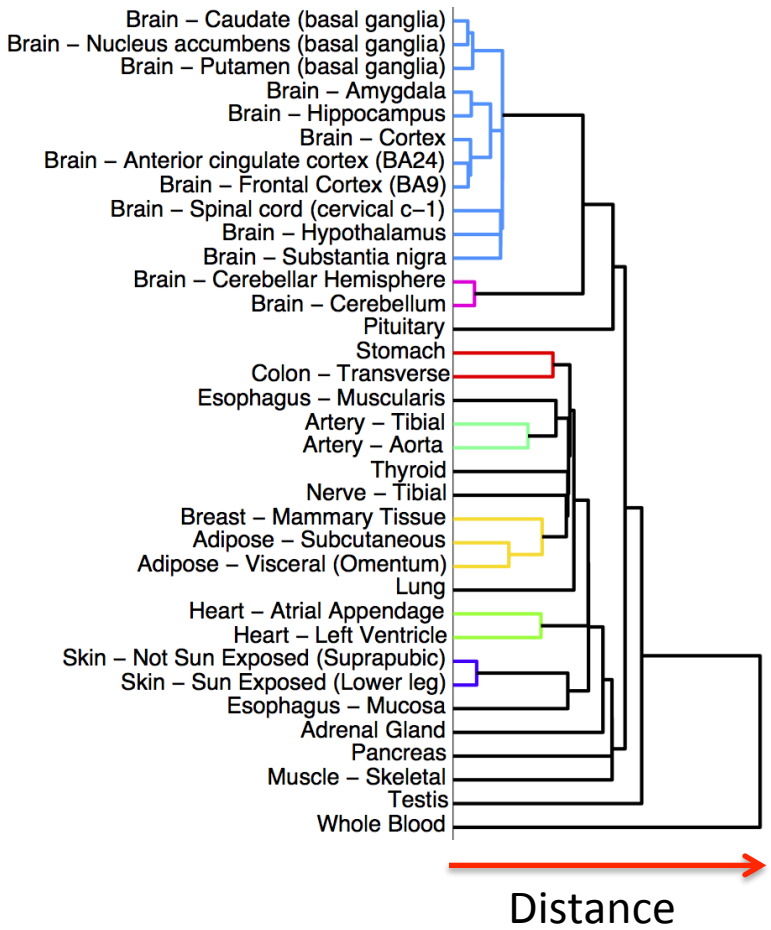
How do you determine *k* (number of clusters)?

Note: maximizing the clustering likelihood/objective will not be informative→ each object should be in its own cluster. Therefore, need an algorithm that takes into account the "cost" of additional clusters.

- Prior knowledge
- The "elbow method"
- Information Criteria Approach: AIC or BIC
- Silhouette method
- The Gap Statistics
- Cross-validation

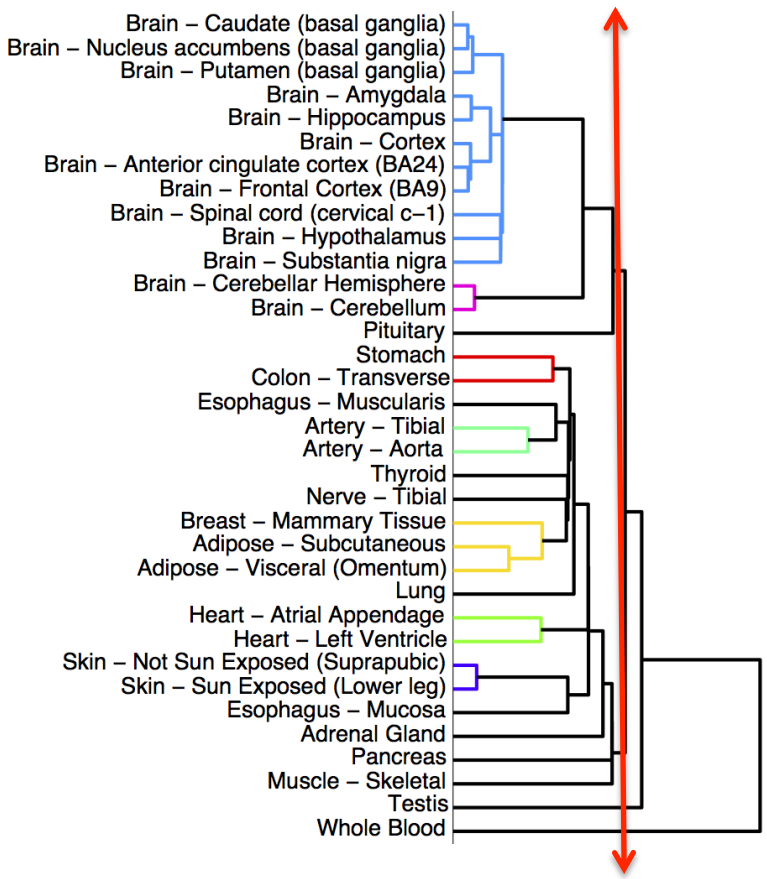
Hierarchical Agglomerative clustering

A clustering approach for revealing hierarchical relationships between objects



Hierarchical Agglomerative clustering

A clustering approach for revealing hierarchical relationships between objects

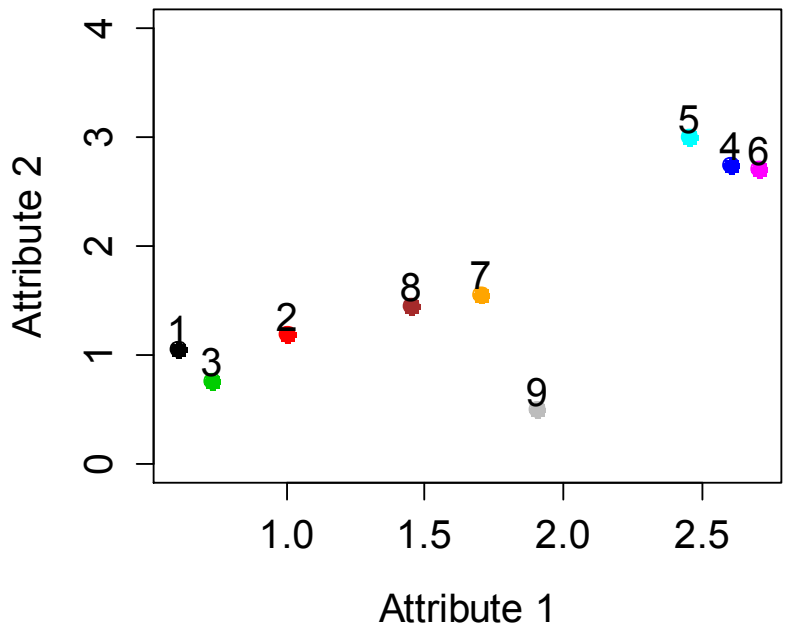


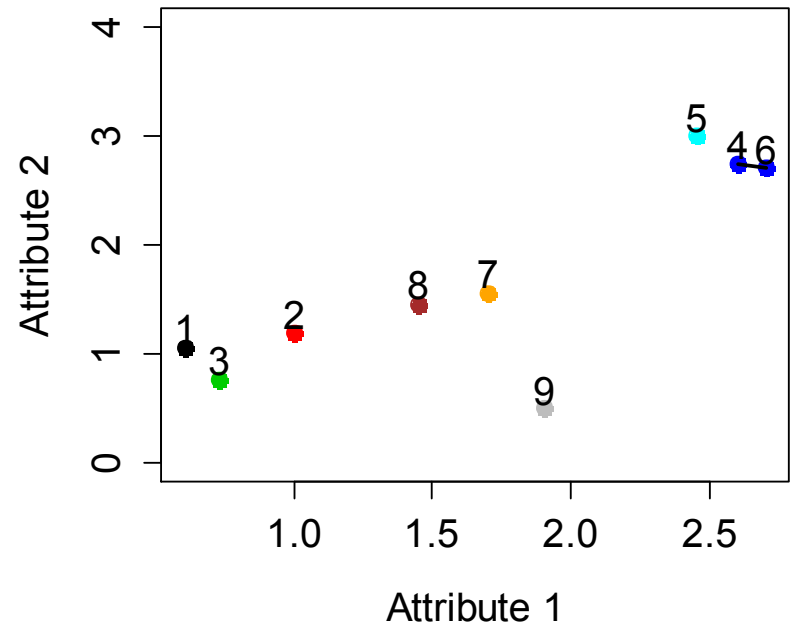
c_i

Algorithms: Hierarchical

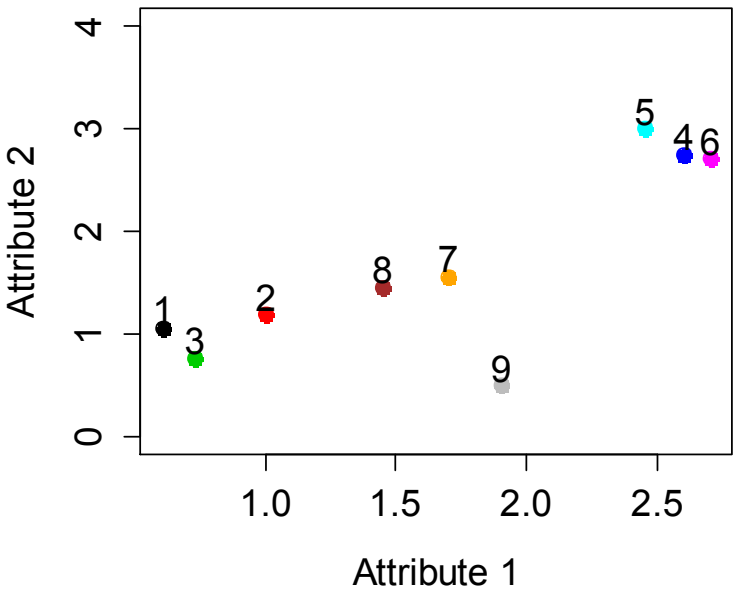
Given *N objects* with *H attributes* and a *distance metric*:

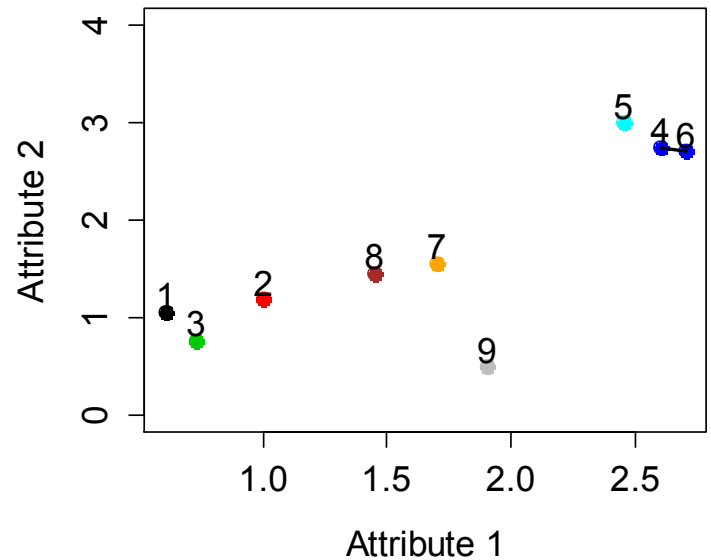
1. Assign each object to a cluster and compute the pairwise distances between all clusters
2. Find the "closest" pair of *clusters* and *merge them* into a single cluster
3. Compute new distances between clusters
4. Repeat steps 2 and 3 until all objects belong to a single cluster.






```
> 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
```

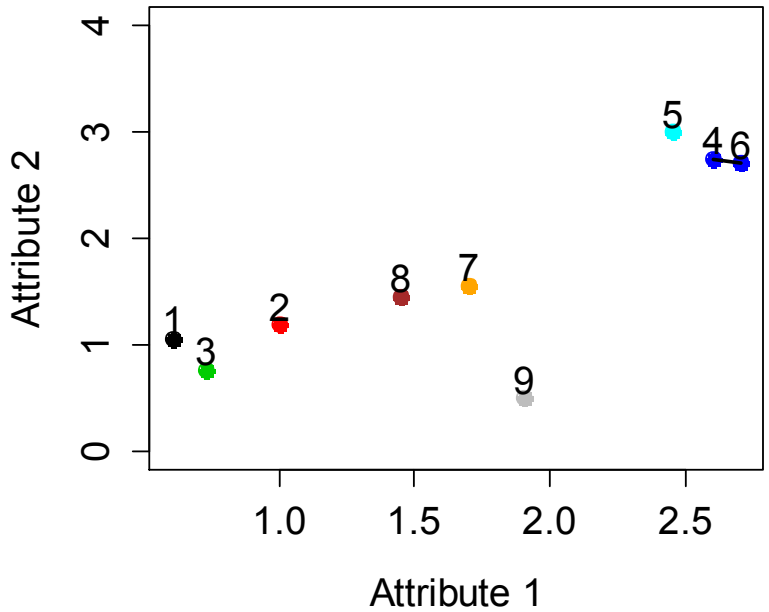




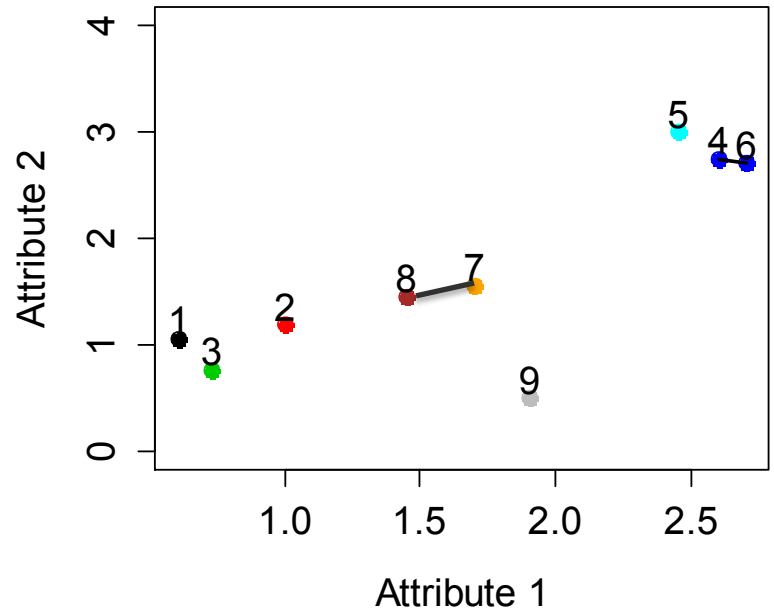

```
> round(dist(a, method='euclidean'),2)
     1    2    3    4    5    6    7    8
2 0.41
3 0.32 0.50
4 2.61 2.23 2.72
5 2.67 2.32 2.81 0.29
6 2.66 2.28 2.76 0.11 0.39
7 1.20 0.79 1.25 1.49 1.62 1.52
8 0.93 0.52 0.99 1.73 1.84 1.77 0.27
9 1.41 1.13 1.20 2.35 2.55 2.34 1.07 1.05
```

→ You can define the cluster "centroids" using:

- Single linkage
- Average linkage
- Complete linkage



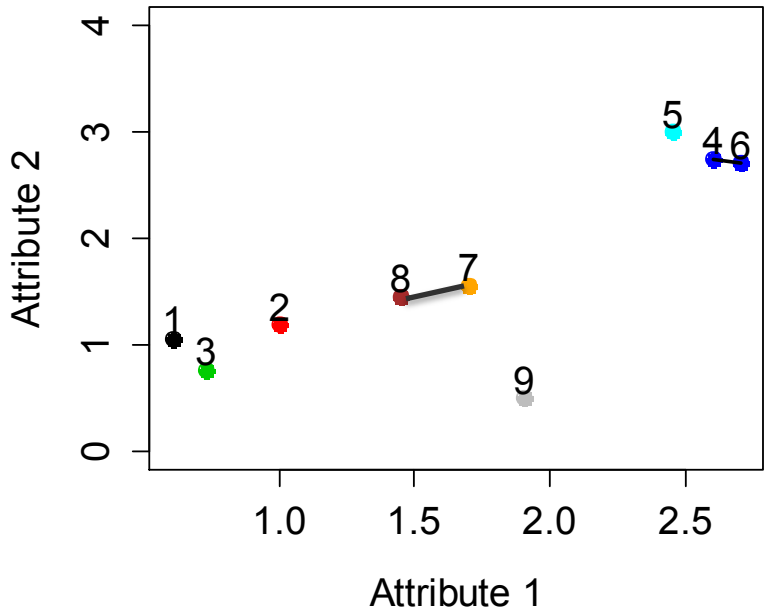
→



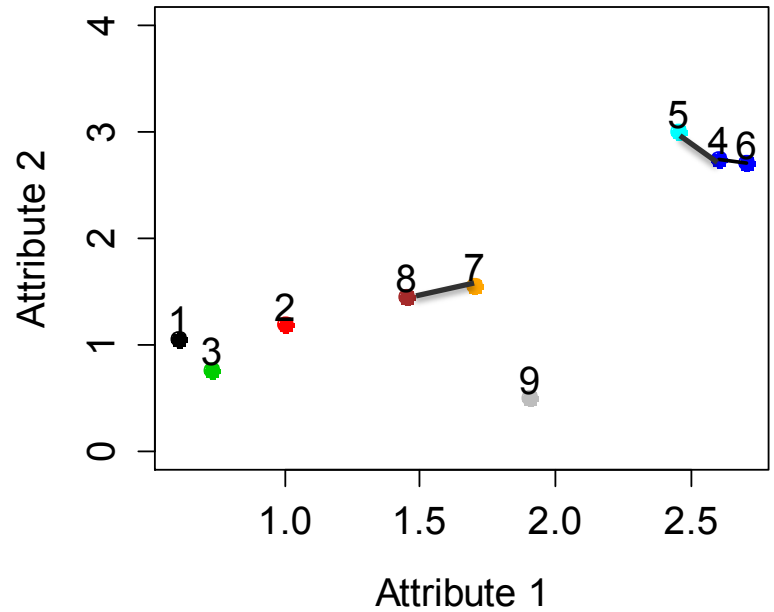

```
> round(dist(a, method='euclidean'),2)
     1    2    3    4    5    6    7    8
2 0.41
3 0.32 0.50
4 2.61 2.23 2.72
5 2.67 2.32 2.81 0.29
6 2.66 2.28 2.76 0.11 0.39
7 1.20 0.79 1.25 1.49 1.62 1.52
8 0.93 0.52 0.99 1.73 1.84 1.77 0.27
9 1.41 1.13 1.20 2.35 2.55 2.34 1.07 1.05
```

→ You can define the cluster "centroids" using:

- Single linkage
- Average linkage
- Complete linkage



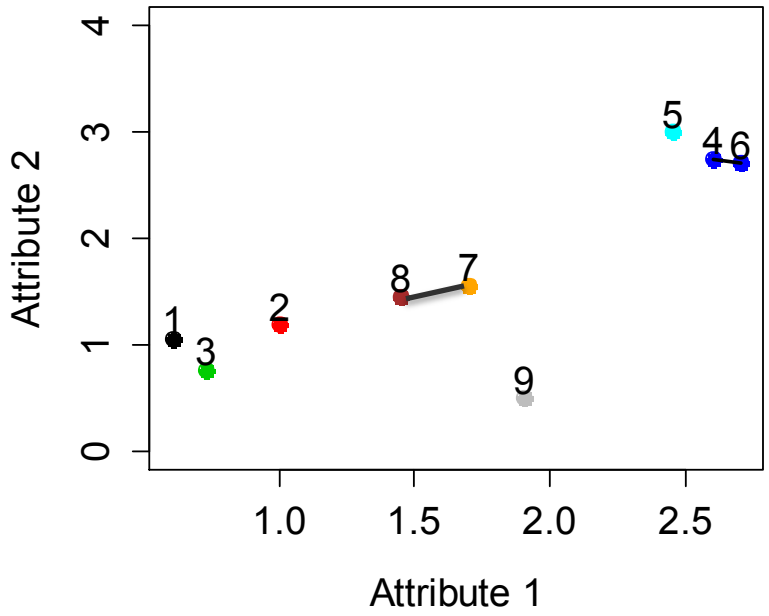
→



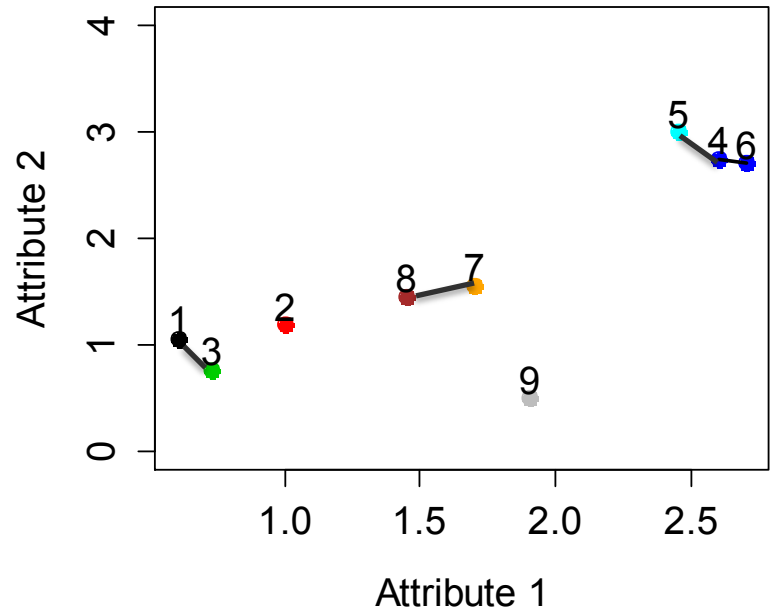

```
> round(dist(a, method='euclidean'),2)
     1    2    3    4    5    6    7    8
2 0.41
3 0.32 0.50
4 2.61 2.23 2.72
5 2.67 2.32 2.81 0.29
6 2.66 2.28 2.76 0.11 0.39
7 1.20 0.79 1.25 1.49 1.62 1.52
8 0.93 0.52 0.99 1.73 1.84 1.77 0.27
9 1.41 1.13 1.20 2.35 2.55 2.34 1.07 1.05
```

→ You can define the cluster "centroids" using:

- Single linkage
- Average linkage
- Complete linkage



→



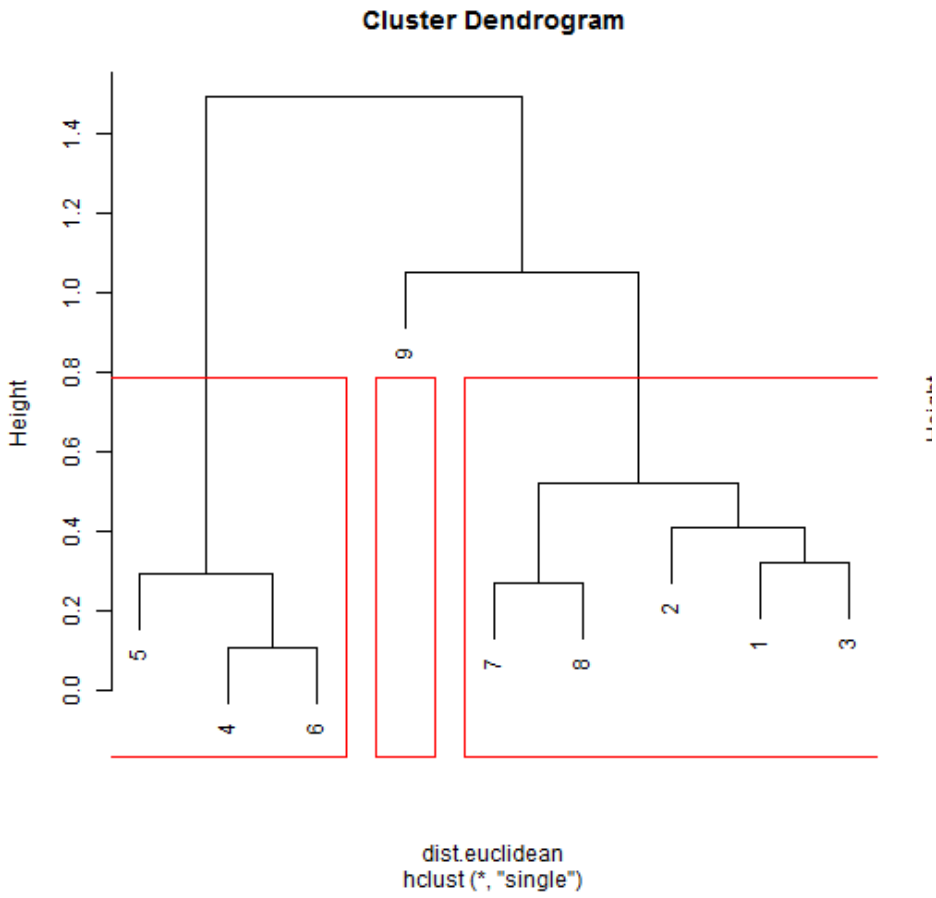

```
> round(dist(a, method='euclidean'),2)
     1    2    3    4    5    6    7    8
2 0.41
3 0.32 0.50
4 2.61 2.23 2.72
5 2.67 2.32 2.81 0.29
6 2.66 2.28 2.76 0.11 0.39
7 1.20 0.79 1.25 1.49 1.62 1.52
8 0.93 0.52 0.99 1.73 1.84 1.77 0.27
9 1.41 1.13 1.20 2.35 2.55 2.34 1.07 1.05
```

→ You can define the cluster "centroids" using:

- Single linkage
- Average linkage
- Complete linkage

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)
```

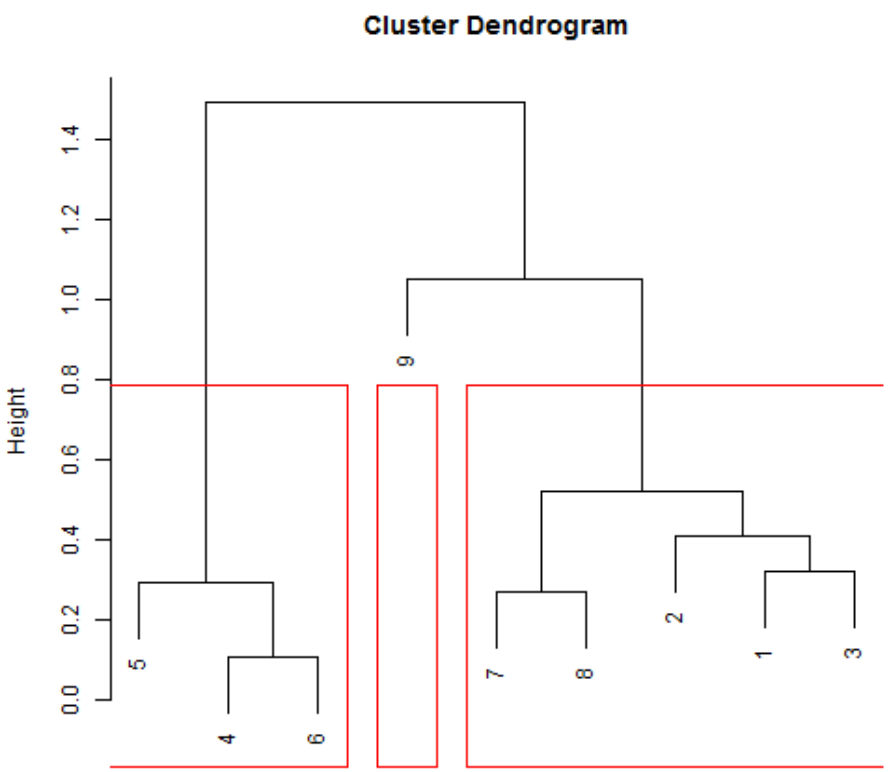


Agglomerative clustering

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

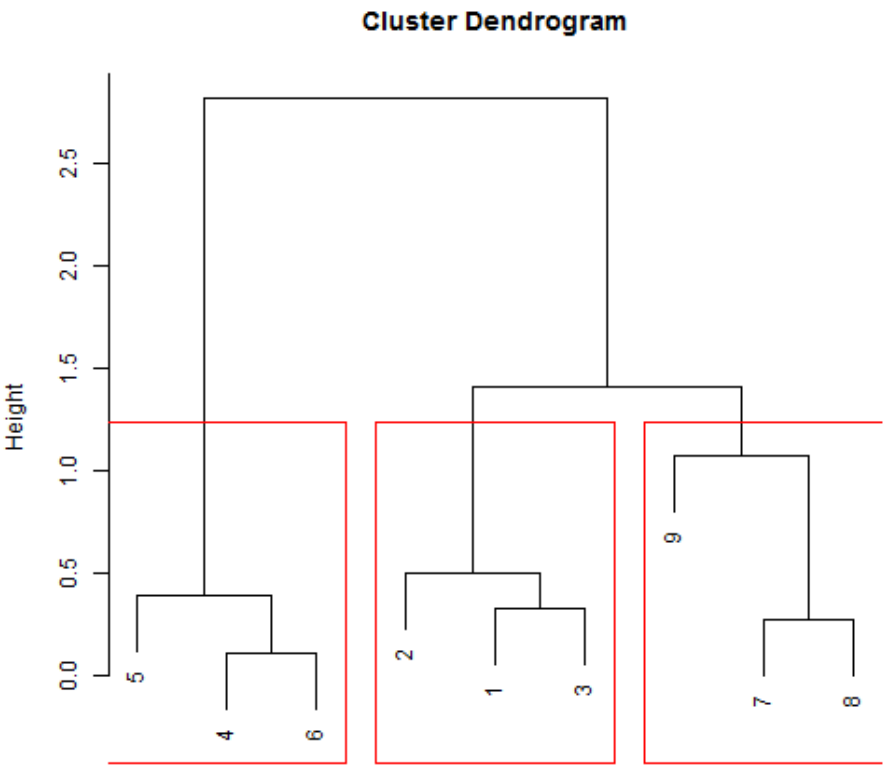
Single Linkage

```
# 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)
```



Summary & conclusions

- 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 the best. It all depends on data and goal.
- Clustering is very powerful, but thoughtless application leads to misguided conclusions.