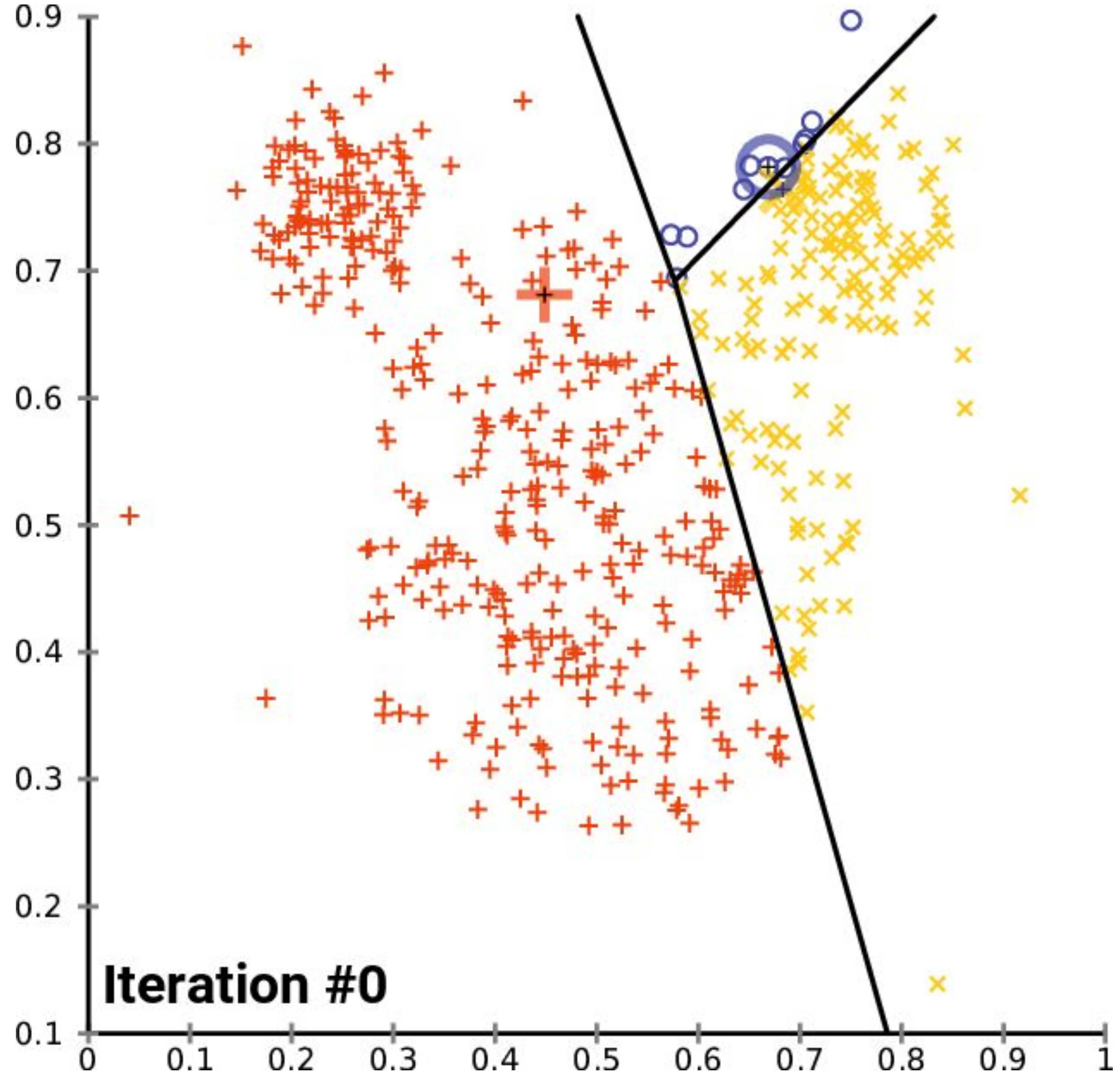
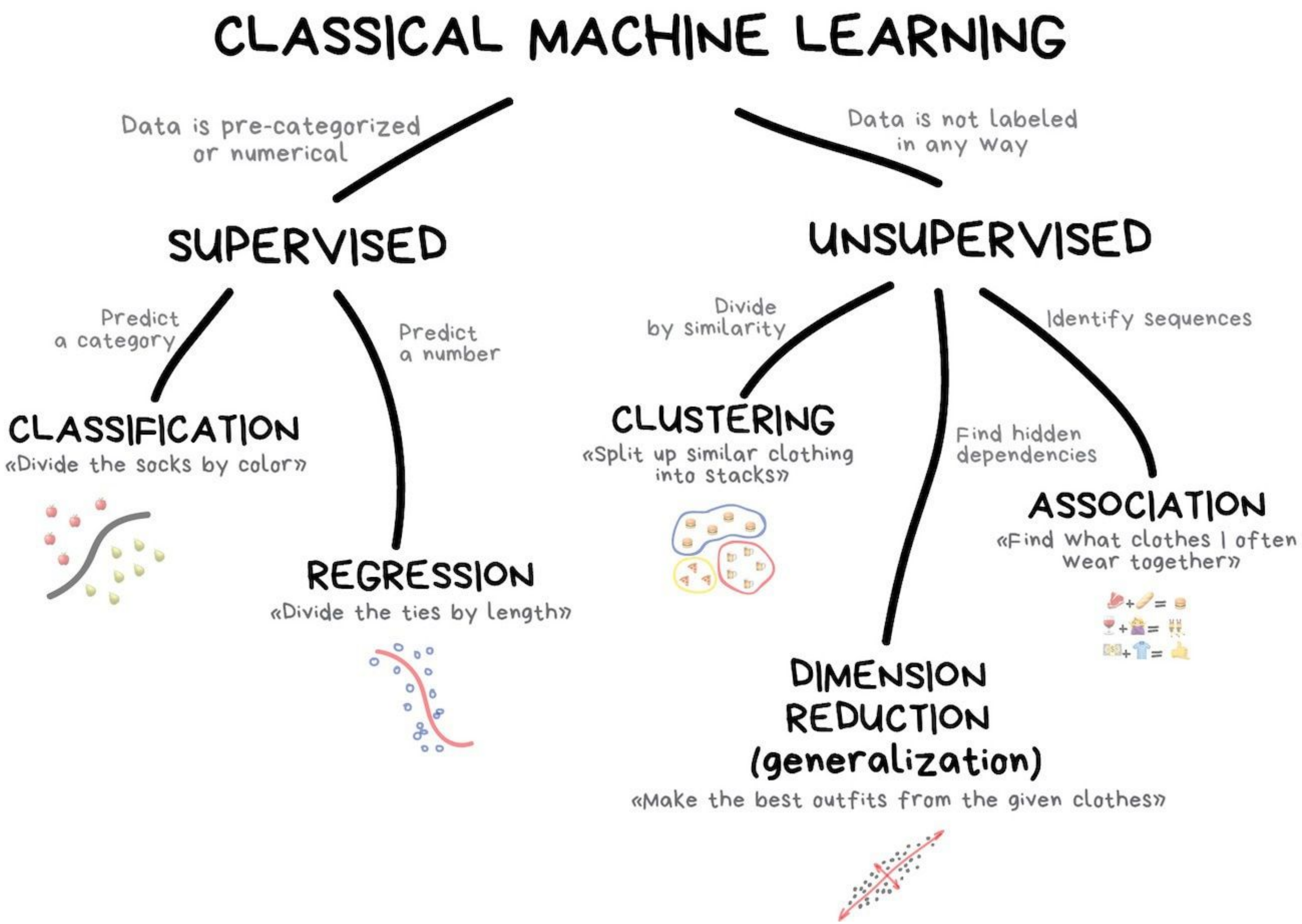


ML#4 Clustering

Providing universal access to AI education and practice

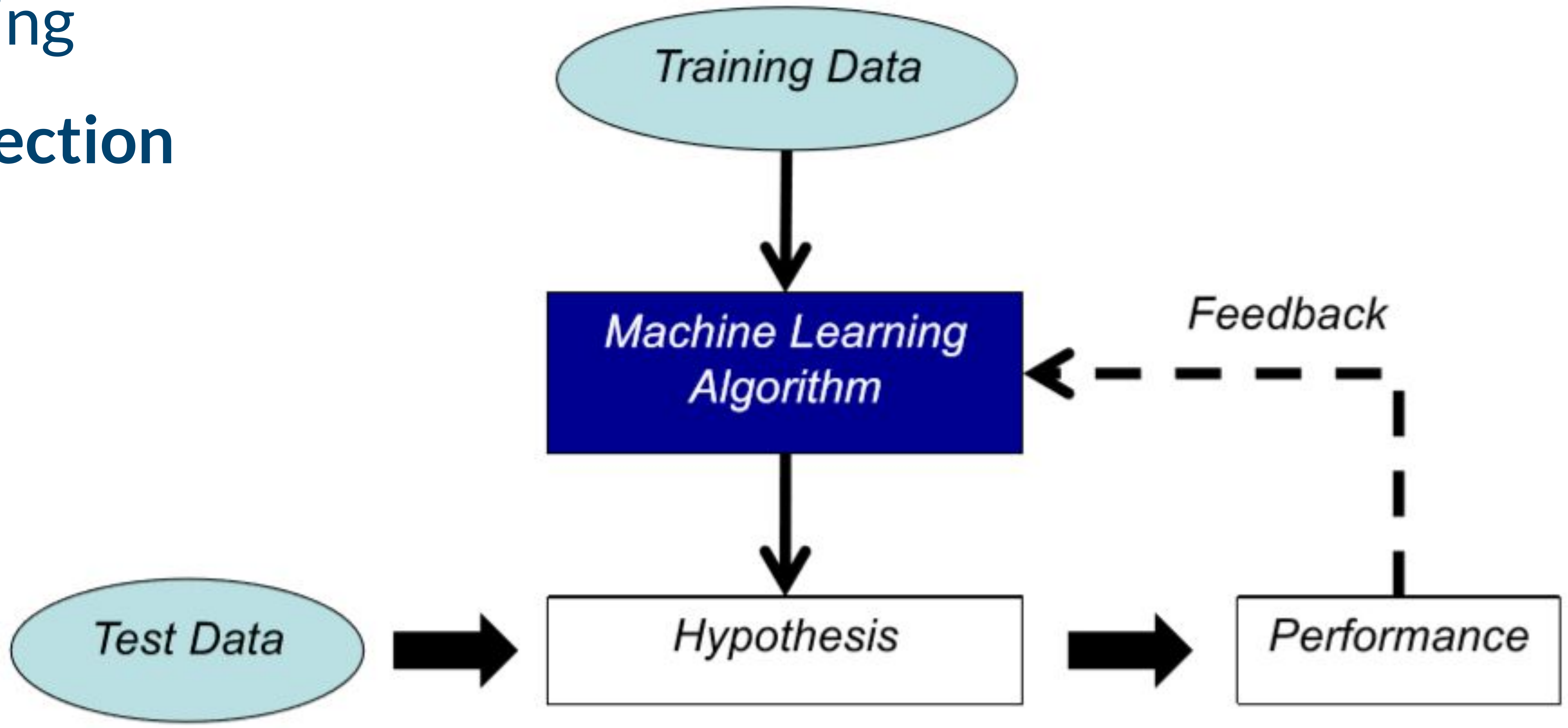
Important things to remember from last week:

Fast Machine Learning **Overview**



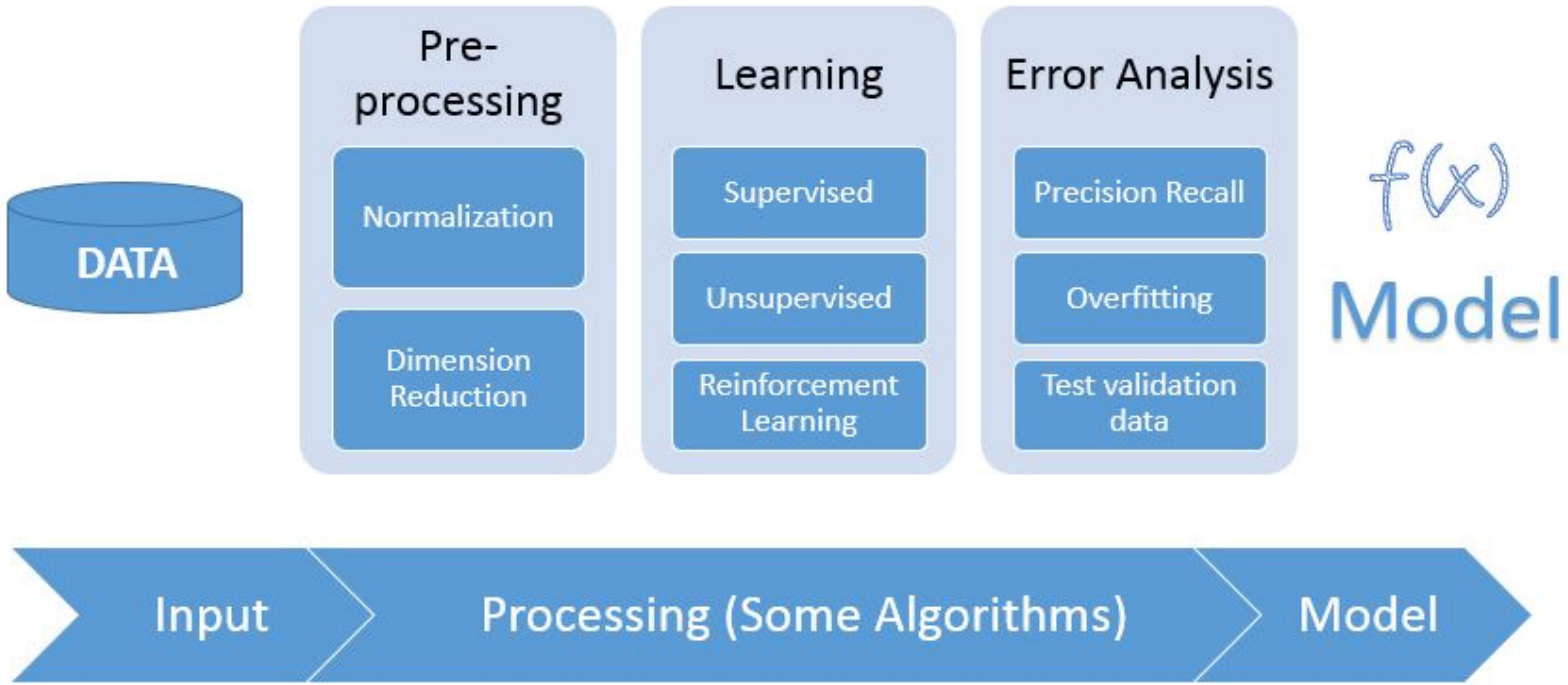
To **review**: ML Process

1. **Data collection** and **Preparation**
2. **Feature Selection** → Ticket price in Titanic
3. **Algorithm** Choice
4. **Split** between Test / Training
5. **Model** and **Parameter selection**
6. **Training Model** with Data
7. Evaluation with **Test set**



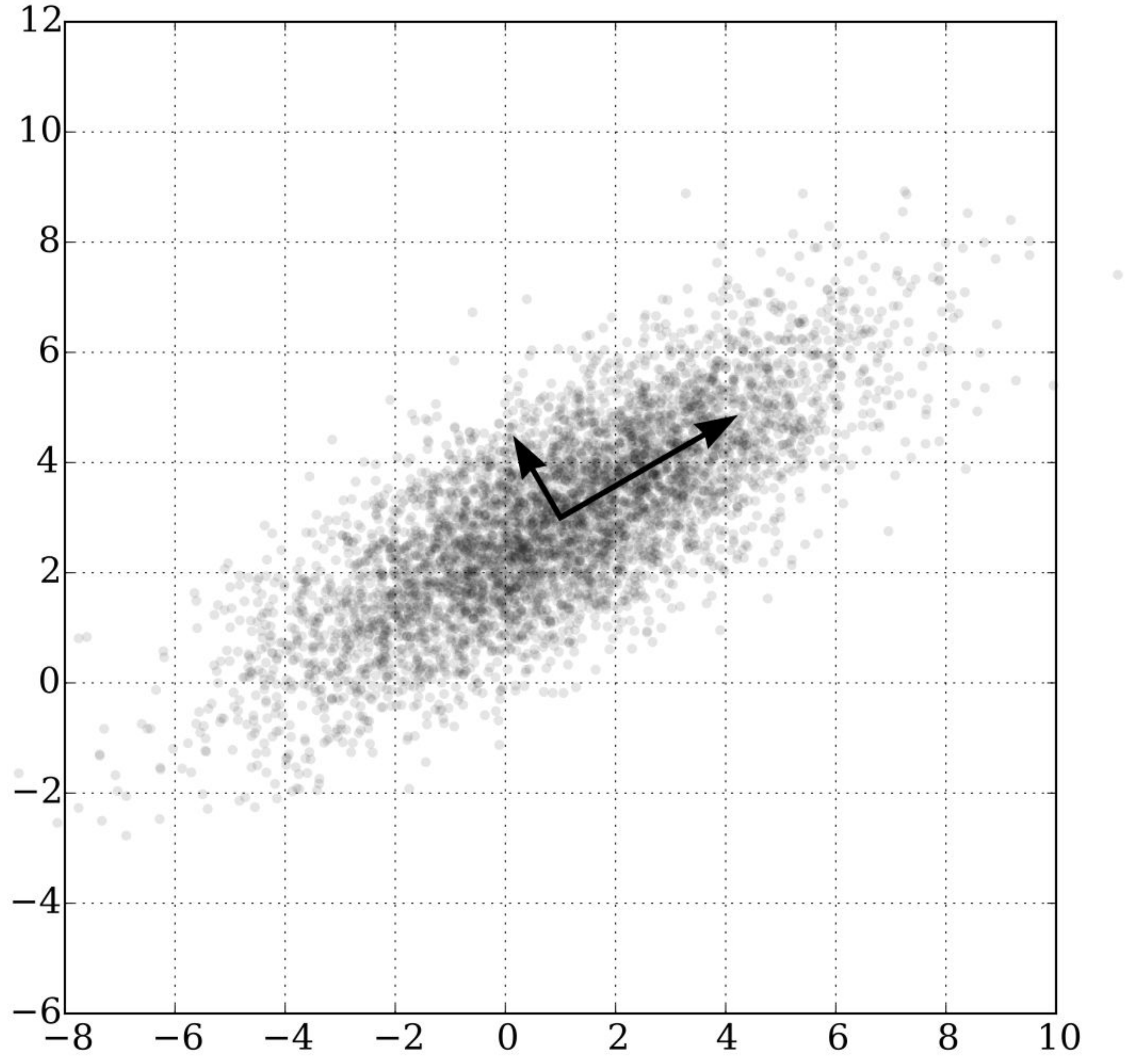
The **Golden Process**

TIP: It never changes. Normally, the more complex the data, the harder it is.



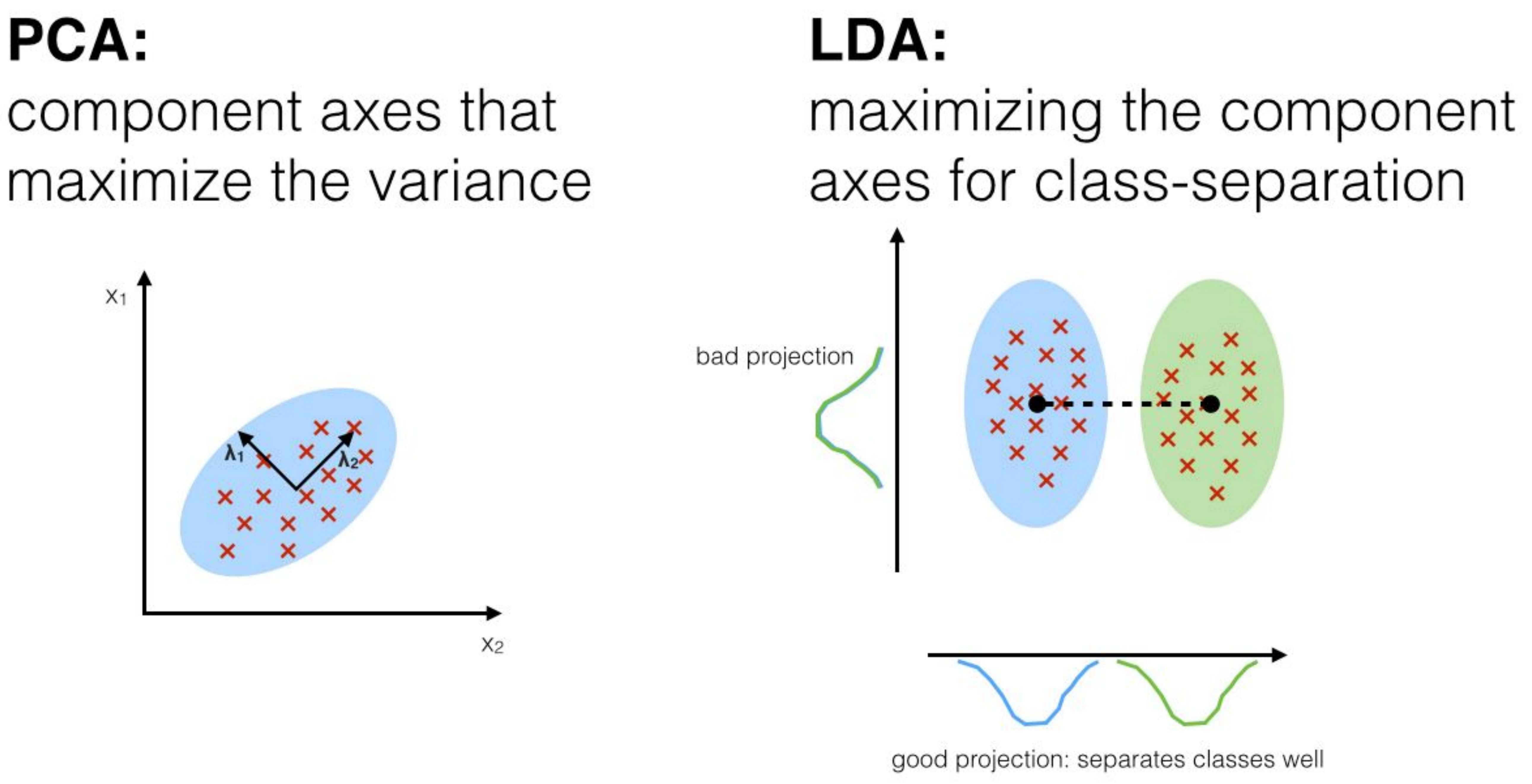
Dimensionality Reduction, Overview

- **Goal:** reducing the number of variables under consideration by obtaining a set of principal variables.
- **How does it work?:** Transforming the data in the high-dimensional space to a space in fewer dimensions.
  - **PCA** (Not considering labels, **unsupervised**)
  - **LDA** (Considering labels, **supervised**)
- **Usage:** Avoiding curse of dimensionality or overfitting due to strong correlations



Dimensionality Reduction, PCA vs LDA

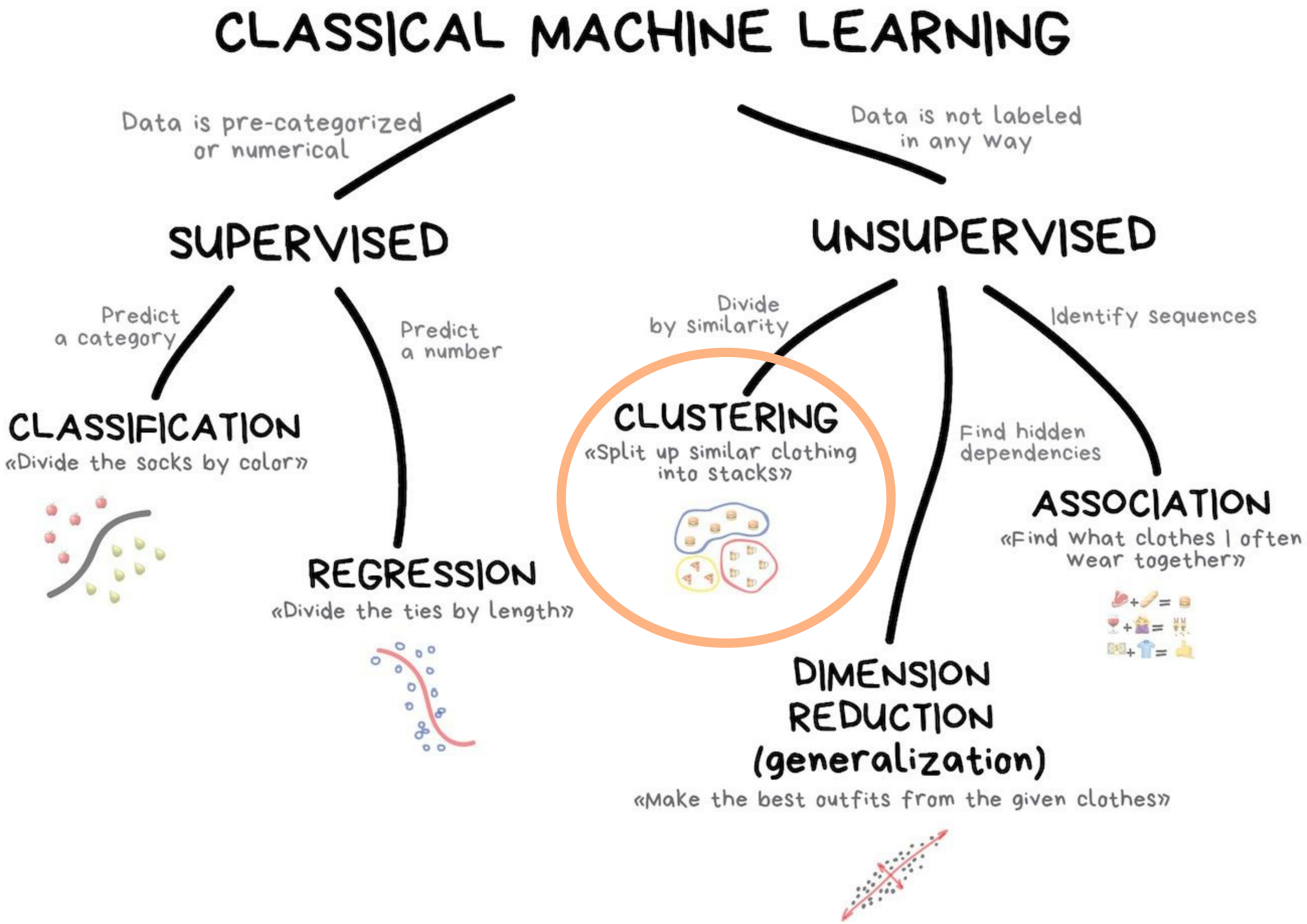
- **Is LDA always better than PCA?**
  - When training set is small, PCA can outperform LDA
  - LDA requires the labels. Can’t do that if you’re dealing with unsupervised data (no target to learn from).
  - LDA > PCA if dataset is large.



What you are doing is not easy. Keep up the hard work

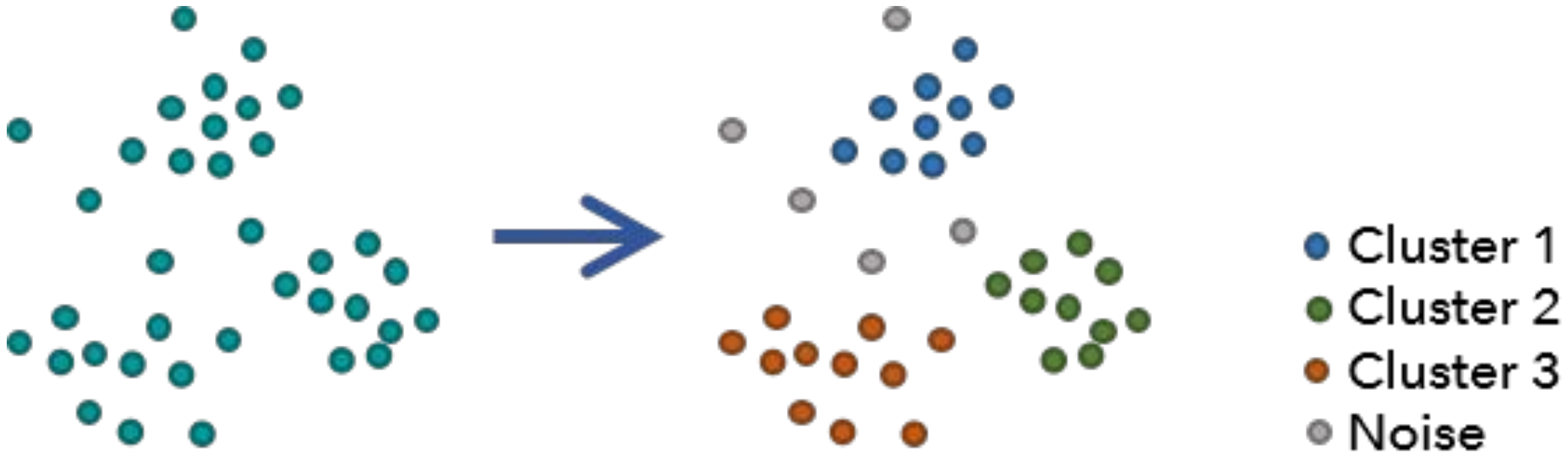
How are you feeling? Ready for some more knowledge? Need help?

What we will do **today**



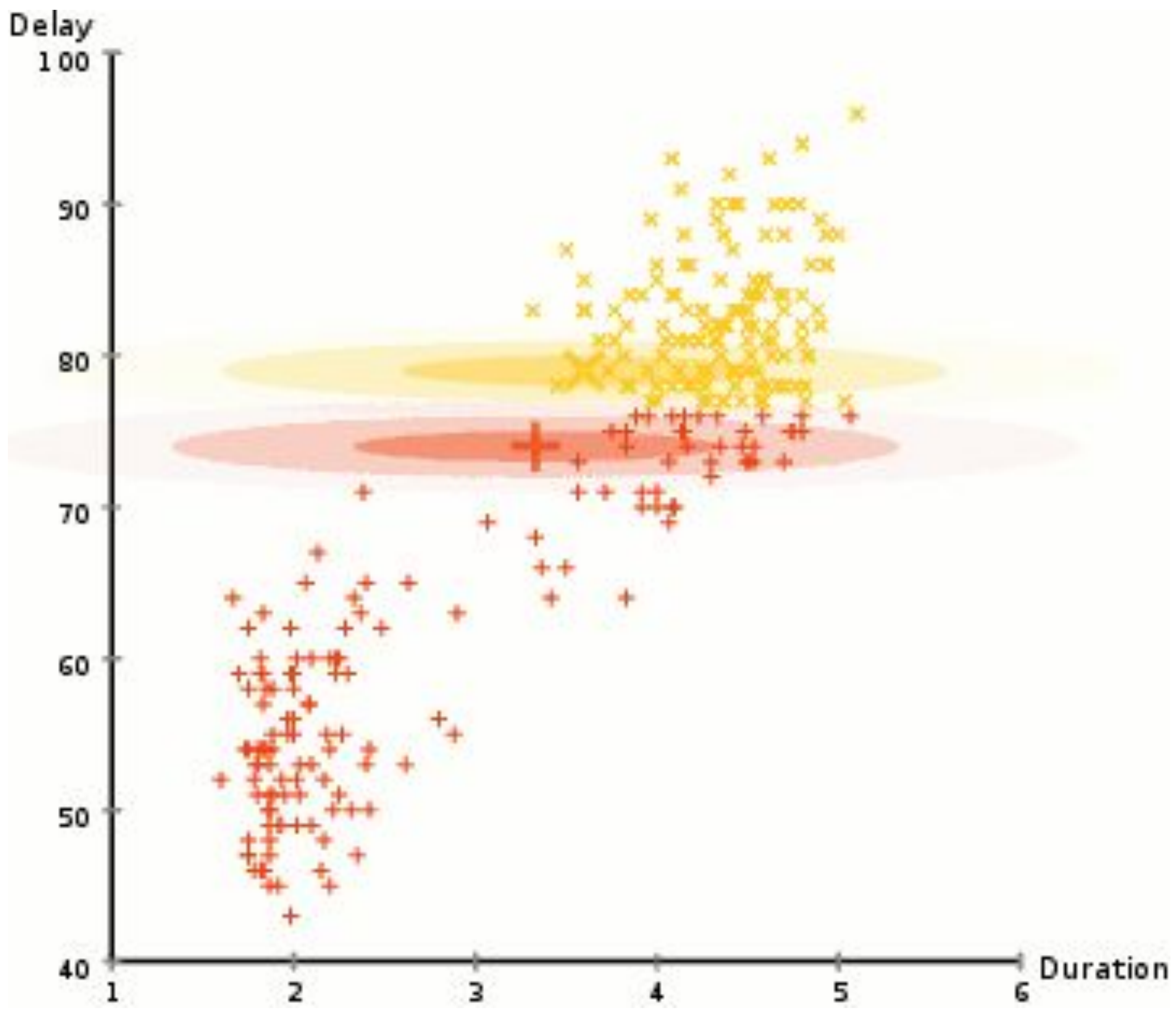
Clustering

- **Formal Definition:** It is the task of grouping a set of objects in such a way that **objects** in the **same group** (called a cluster) are **more similar** (in some sense or another) to each other than to those in **other groups** (clusters).
- Raw Data → **Clustering Algorithm** → Clusters of data



Clustering: Practical Definition

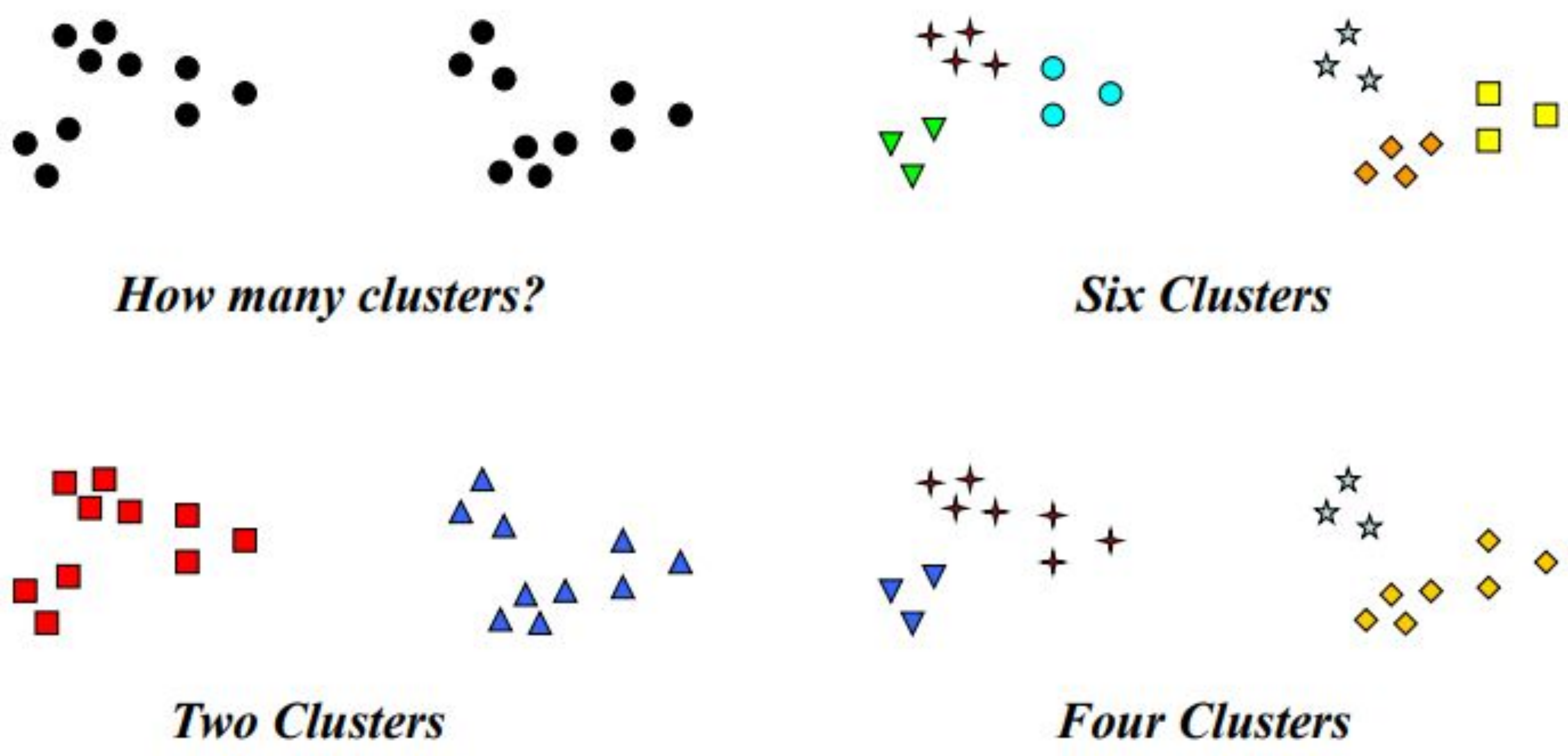
- **Clustering:** method by which large sets of data is grouped into clusters of smaller sets of similar data. There are several ways:
  - Based on **connectivity**: Hierarchical clustering
  - Based on **centroids**: K-means
  - **Distribution-based models:** Mixture models, Expectation-Maximization → estimates for model parameters when your data is incomplete, or has missing data points, or latent variables
  - and a few more...



Clustering: Main Characteristics

- **Self-supervised // Unsupervised Learning:** Discovers the key concepts hidden within the data without guidance.
  - **To summarize:** Obtaining representations that describe a dataset without labels
  - **To know:** Discovering key concepts hidden within our data. *ex: Facebooks algorithm updated on likes data.*
- Different answers may be valid depending on what you seek to discover.
- Hard to evaluate the results yet there are some criteria / methods to do so.

Clustering: Example

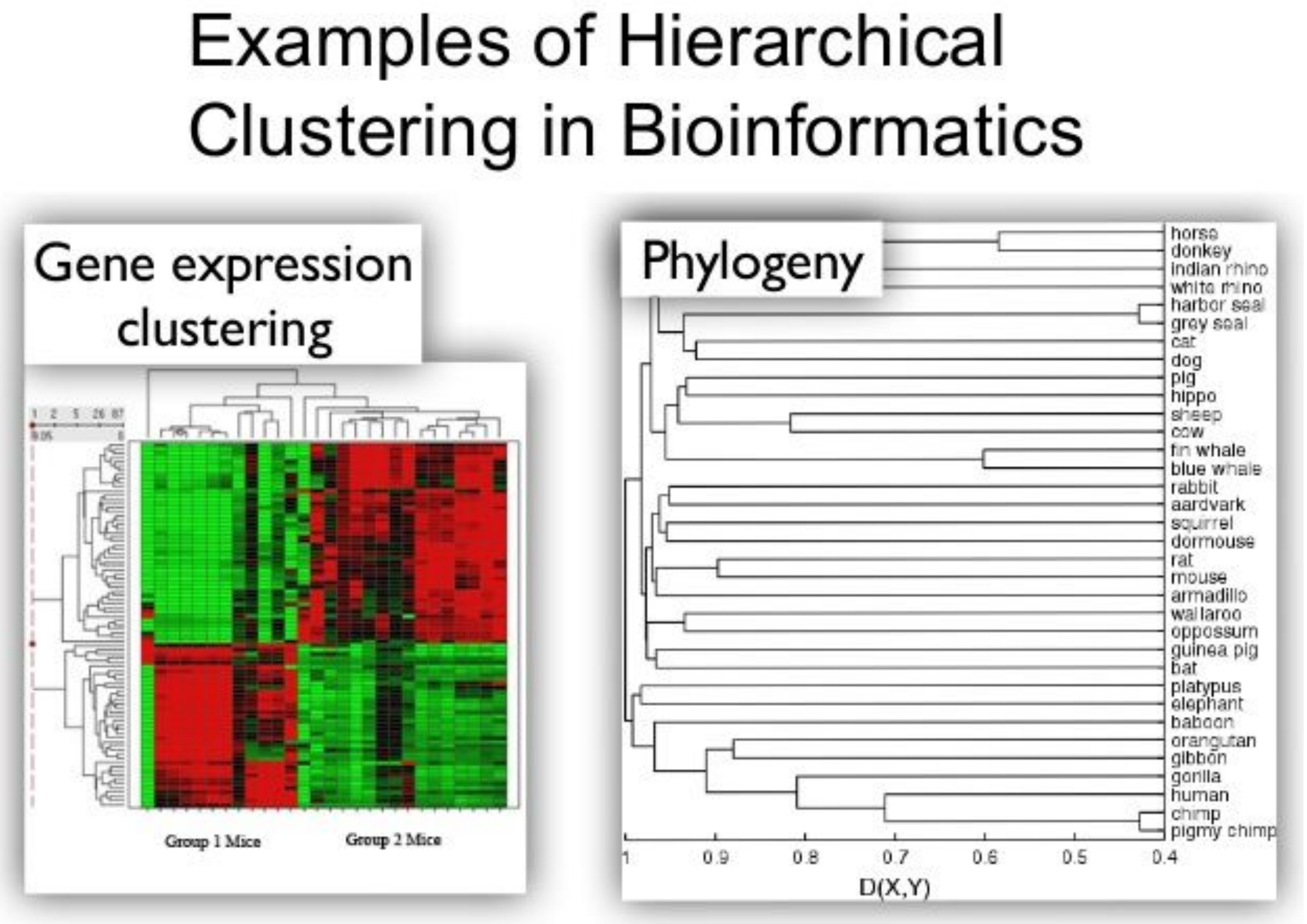


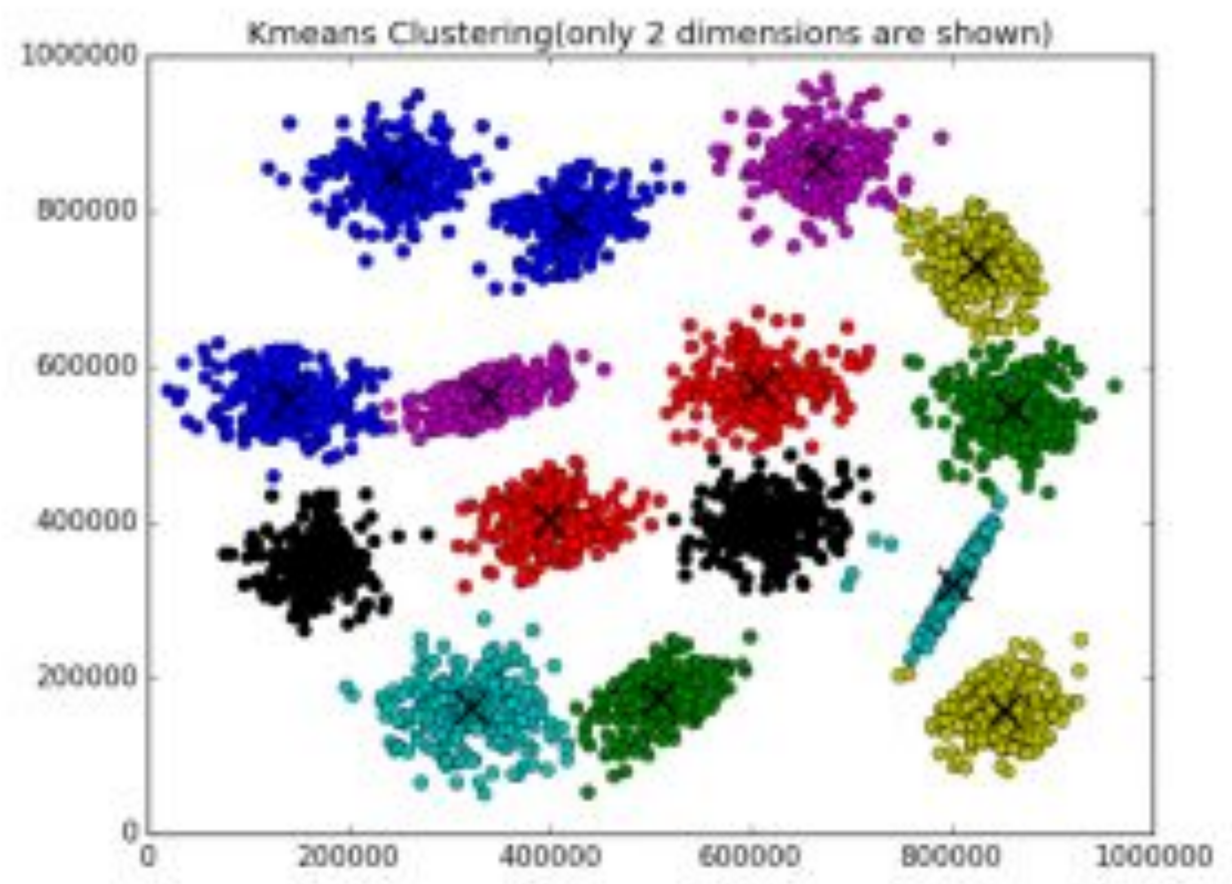
Clustering: Main Issues

- **What is a natural grouping among these objects?**
  - Definition of “groupness”
- **What makes object “related”?** “similarity/distance”
- **Representation of objects** – Vector space? Normalization?
- **How many clusters** – Fixed a priori or data driven?
- **Clustering algorithms**
  - Hierarchical algorithms
  - Partitional algorithms

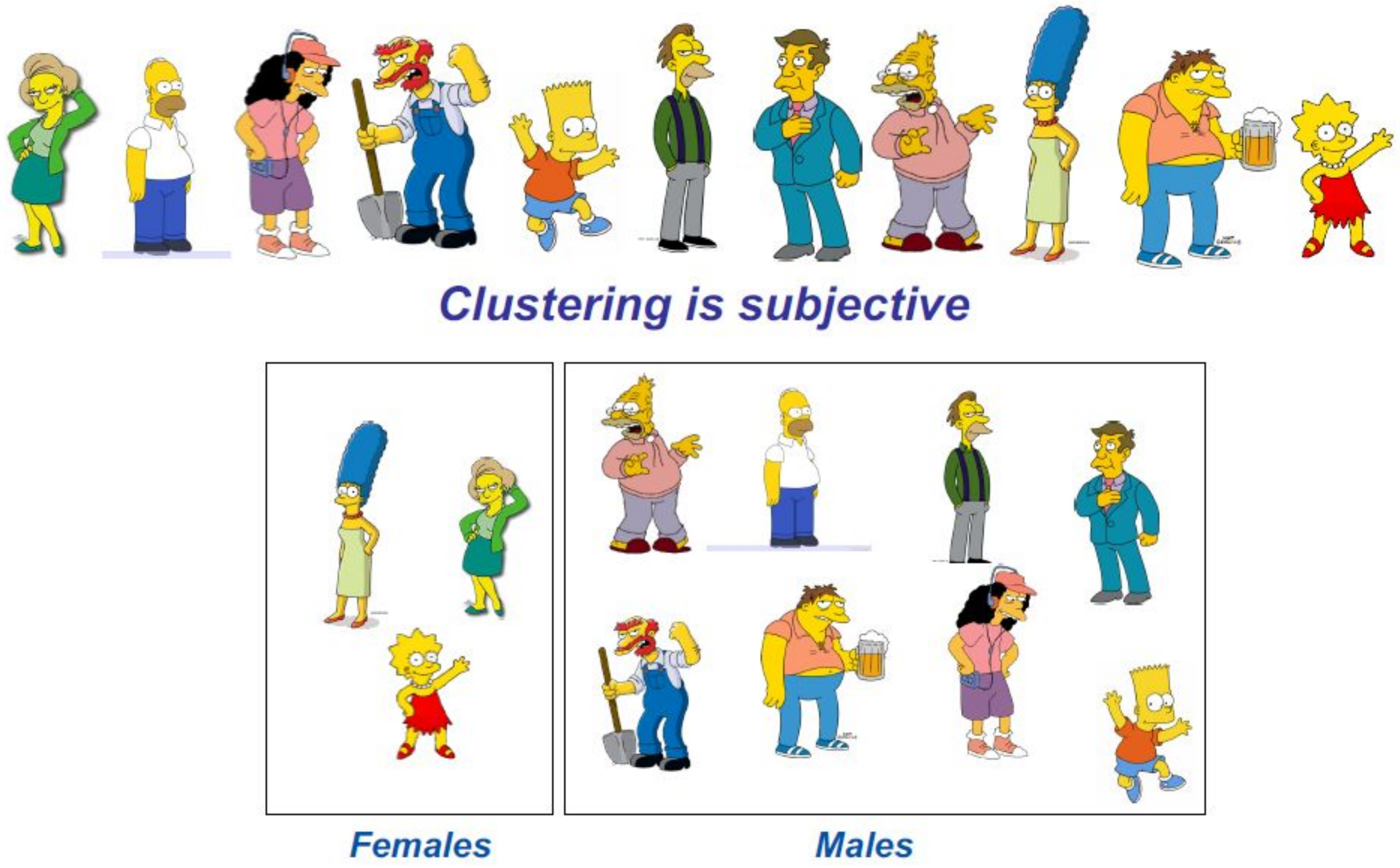
Clustering: Examples of use

- **Bioinformatics**
- Medicine
- Market research
- **Social network** analysis
- **NLP**: clustering de documentos, text mining, concept extraction
- **Image segmentation**
- Climatologia

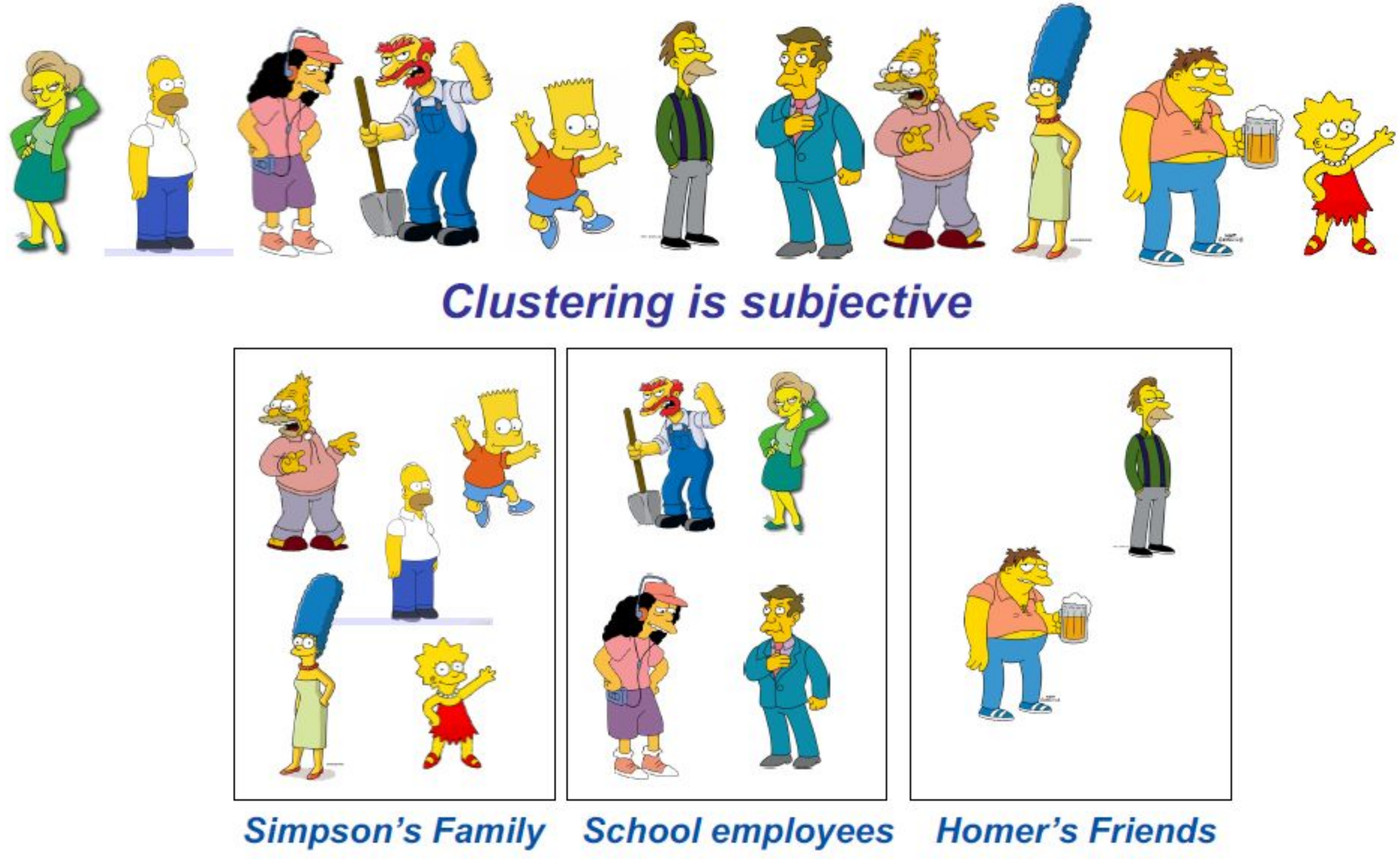




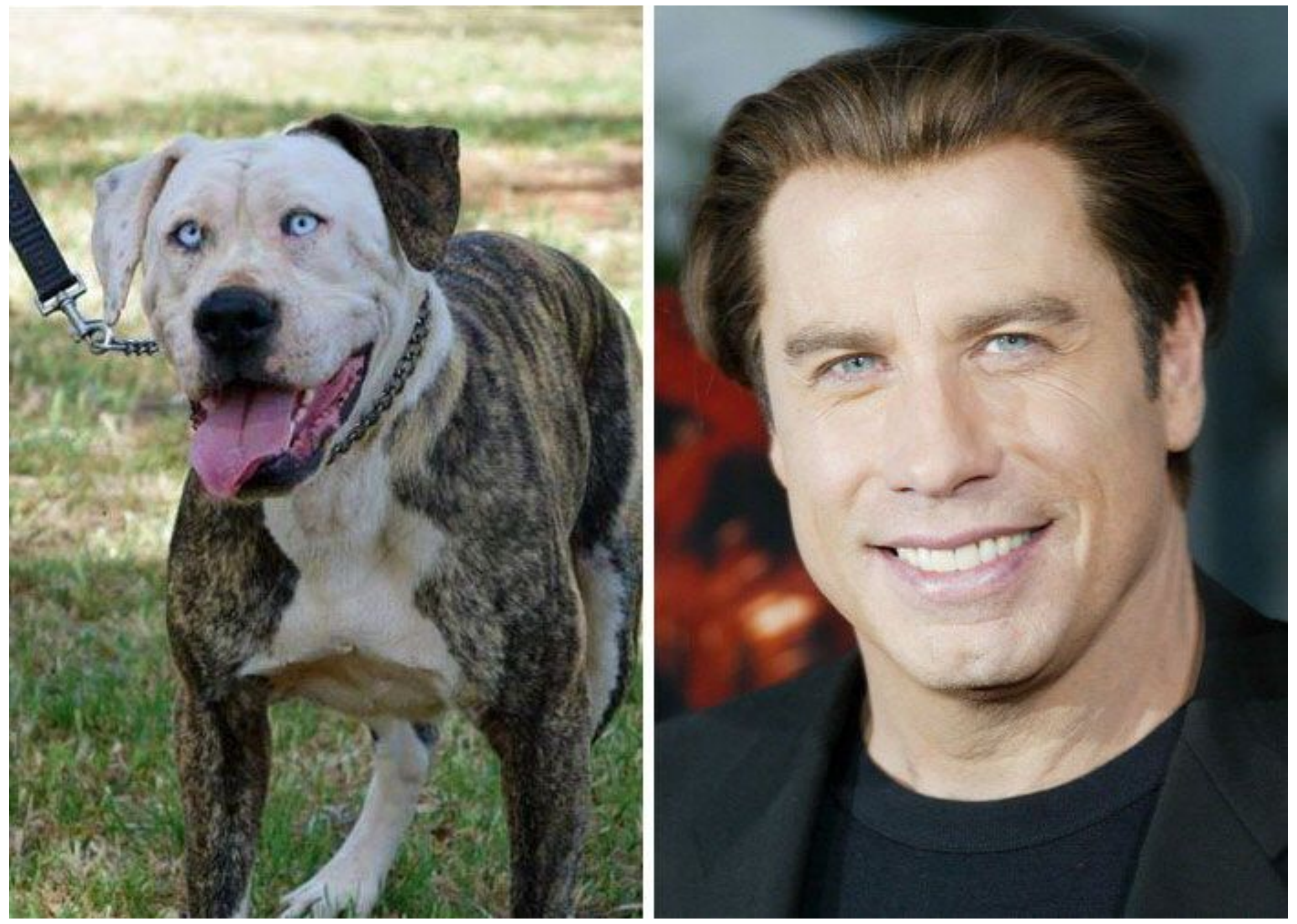
Clustering: What is natural grouping?



Clustering: What is natural grouping?



Clustering: What is similarity?

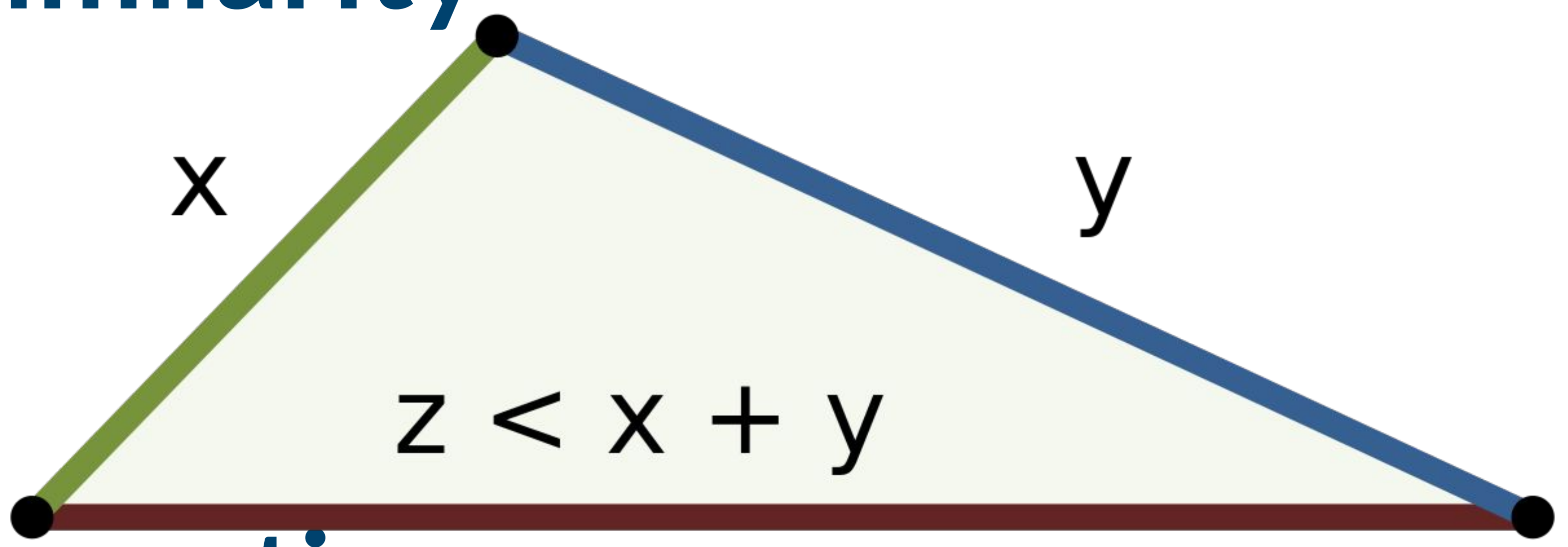


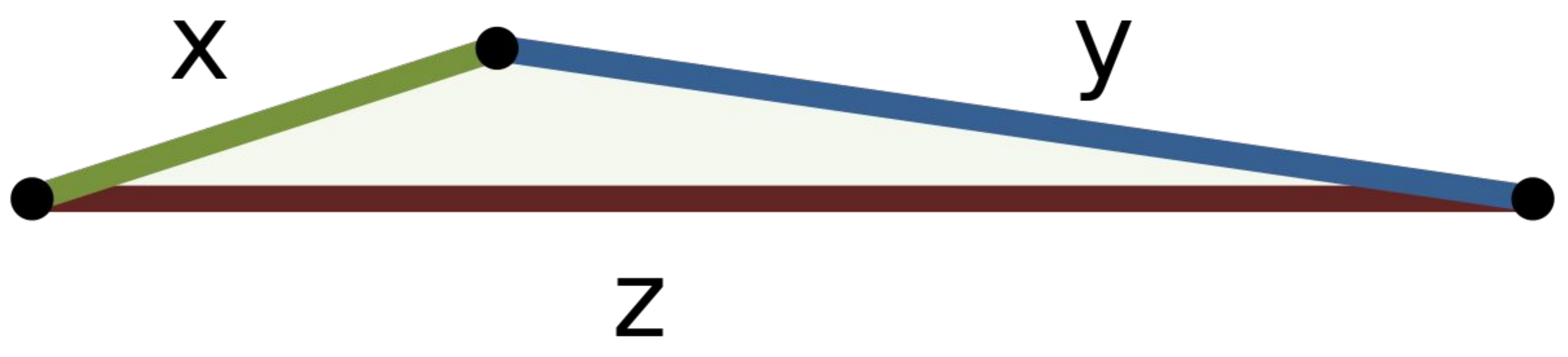
Clustering: What is similarity?

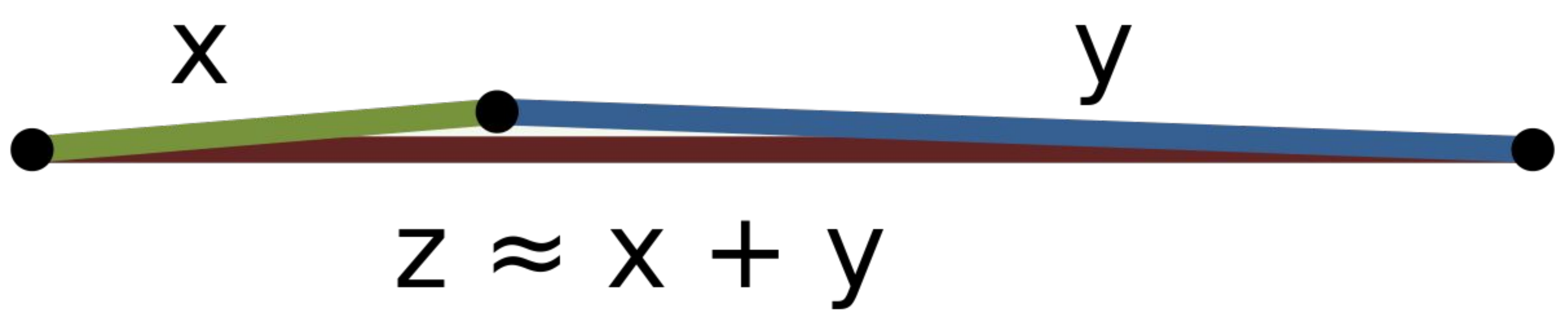
- Hard to define!
- But we know it when we see it.
- We can actually compute the distance between clusters / images… etc

Desirable **properties** of a **distance measure**

- D(A,B) == D(B,A) **Symmetry**
- D(A,A) == 0 Awareness of **Self-Similarity**
- D(A,B) == 0, si A==B **Positive Separation**
- D(A,B) <= D(A,C) + D(B,C) Triangular Inequality





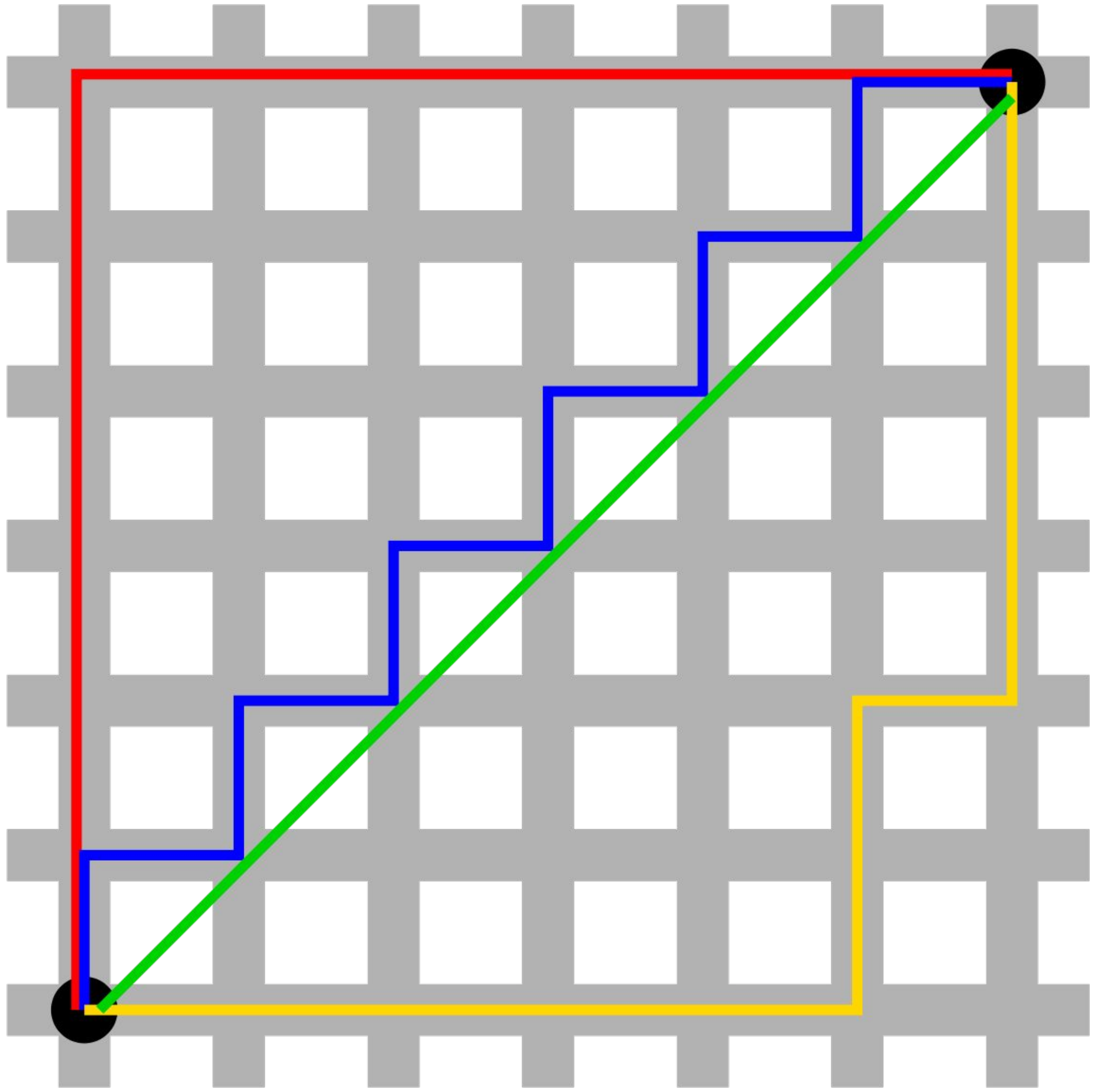


Distance Measure: **Minkowski Metric**

- Suppose two object **x** and **y** both have **p** features

$$x = (x_1, x_2, \cdots, x_p)$$

$$y = (y_1, y_2, \cdots, y_p)$$

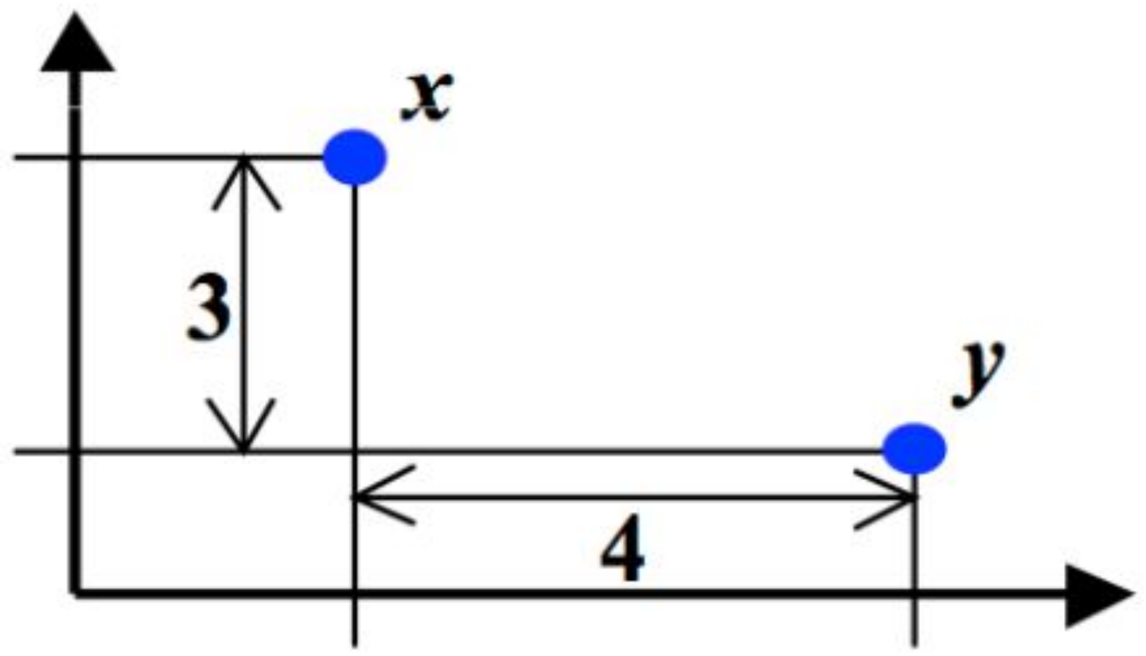


- The Minkowski metric is defined by

$$d(x,y) = \sqrt[r]{\sum_{i=1}^{p} |x_i - y_i|^r}$$

Distance Measure: **Minkowski Metric Example**

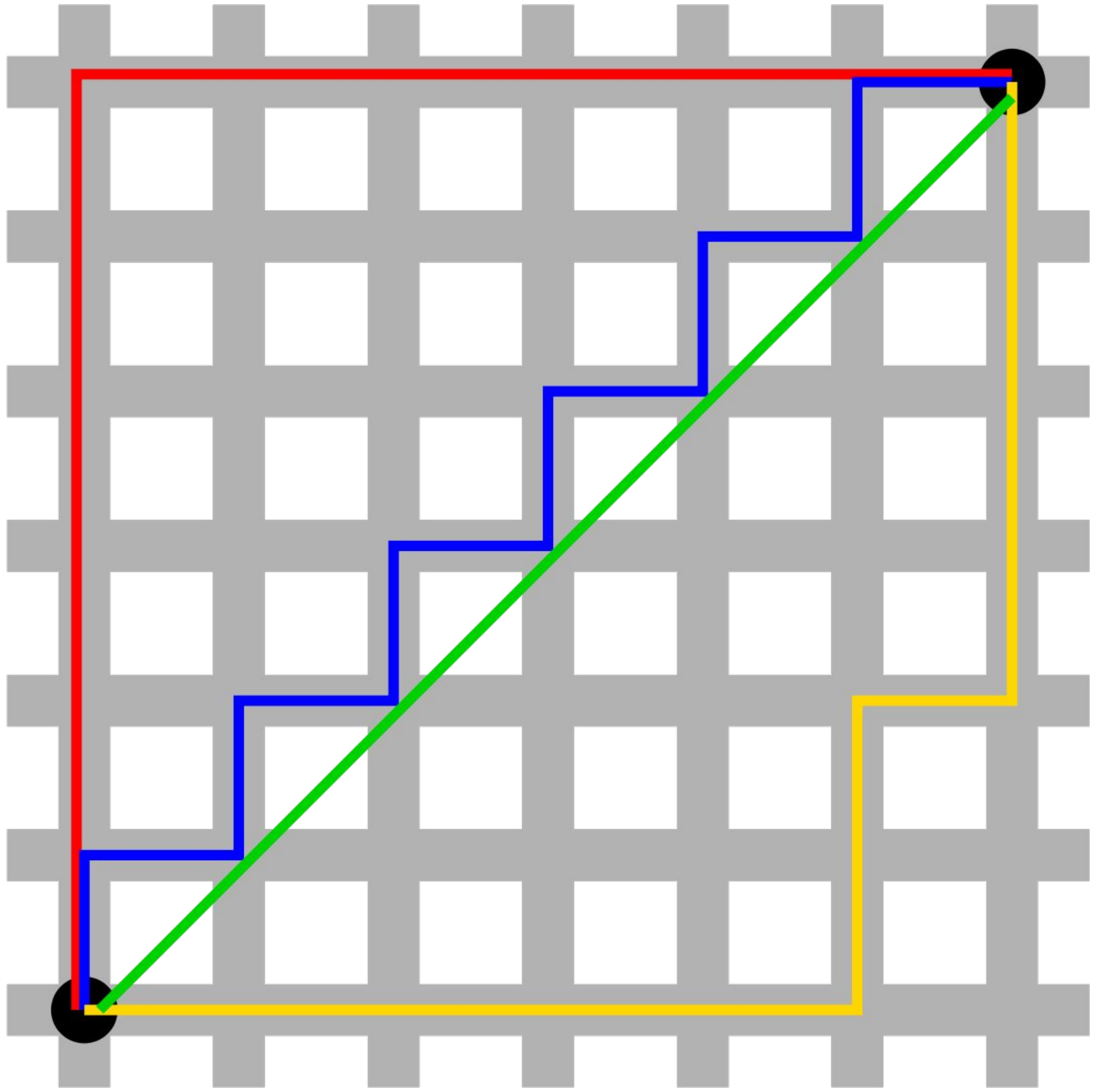
$$d(x,y)=\sqrt[r]{\sum_{i=1}^{p}|x_i-y_i|^r}$$



1: Euclidean distance: $\sqrt[2]{4^2+3^2}=5.$

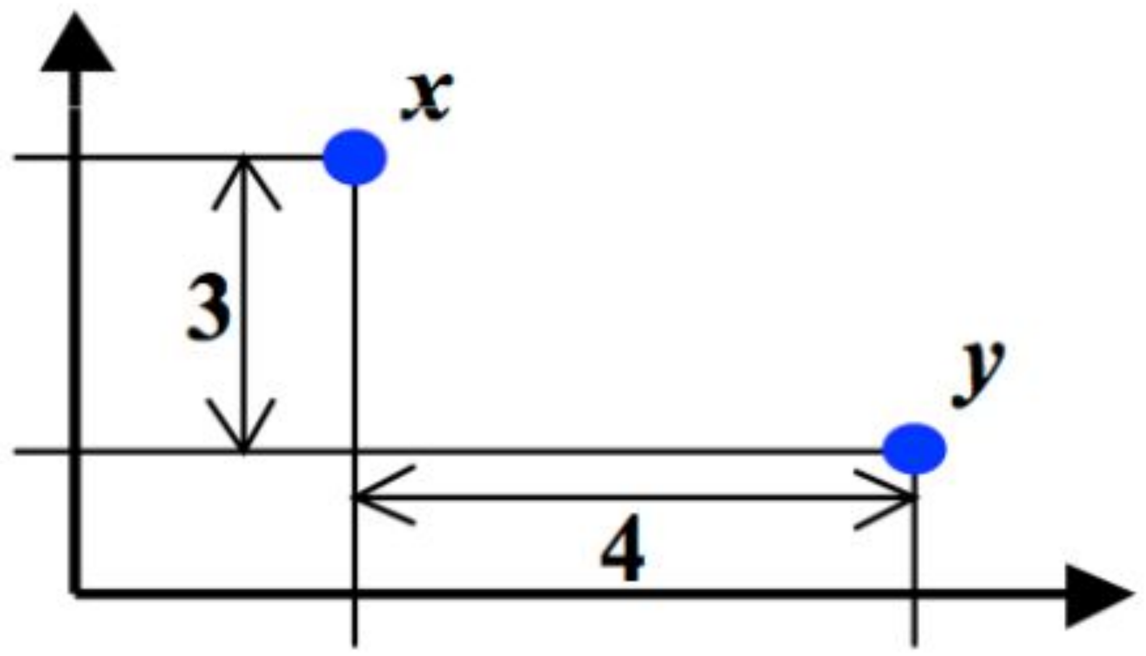
2: Manhattan distance: $4+3=7.$

3: "sup" distance: $\max\{4,3\}=4.$



Distance Measure: **Minkowski Metric Example**

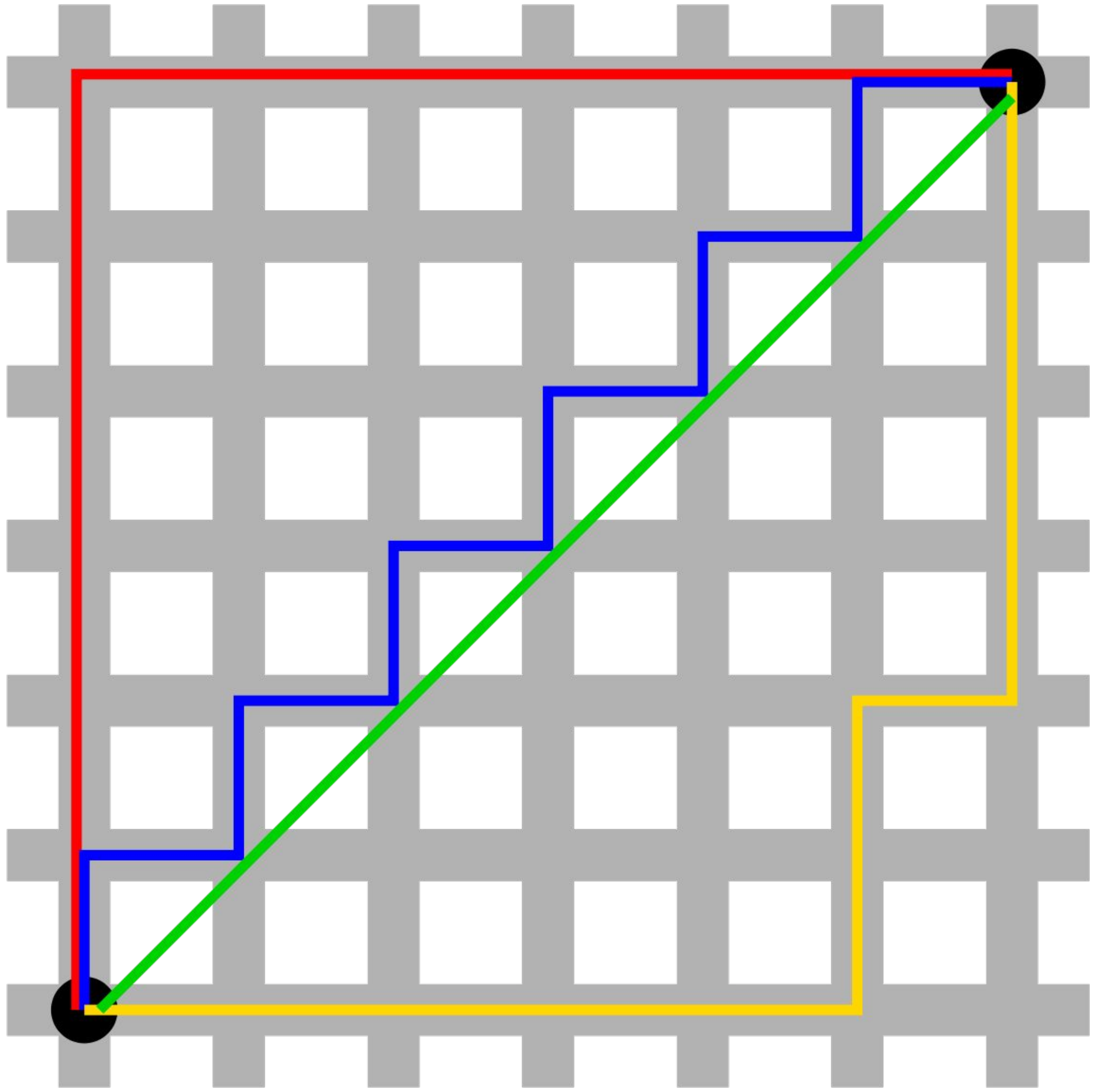
$$d(x,y) = \sqrt[r]{\sum_{i=1}^{p} |x_i - y_i|^r}$$



$1:$ Euclidean distance: $\sqrt[2]{4^2+3^2} = 5.$

$2:$ Manhattan distance: $4+3=7.$

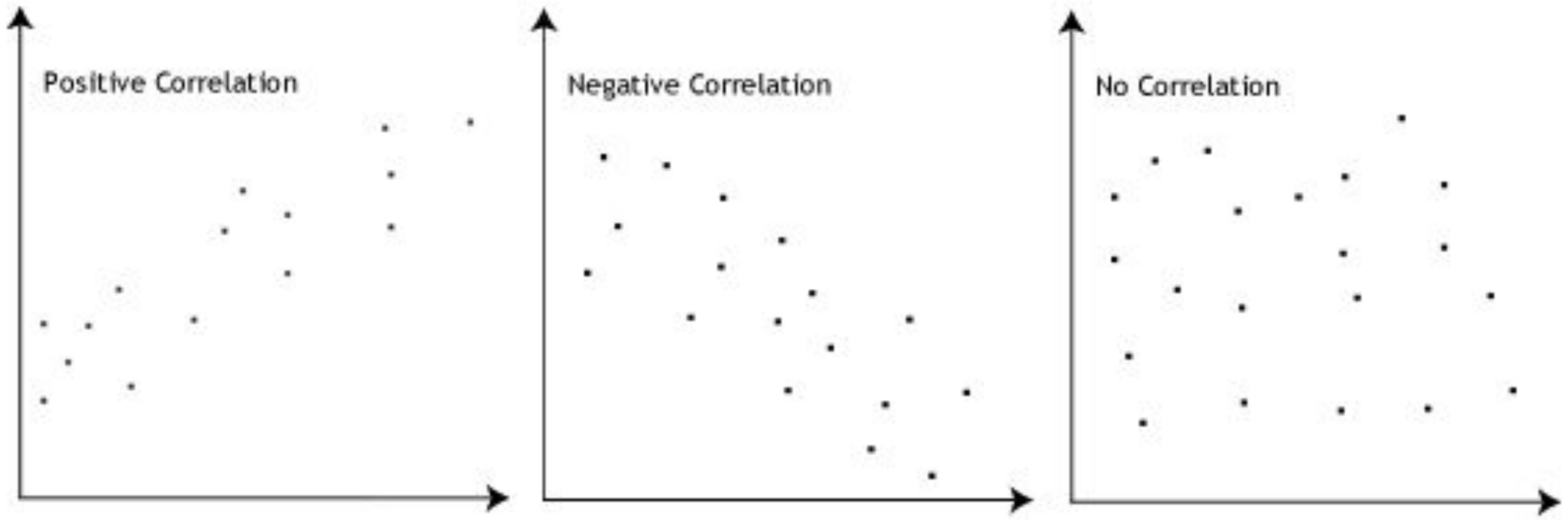
$3:$ "sup" distance: $\max\{4,3\} = 4.$

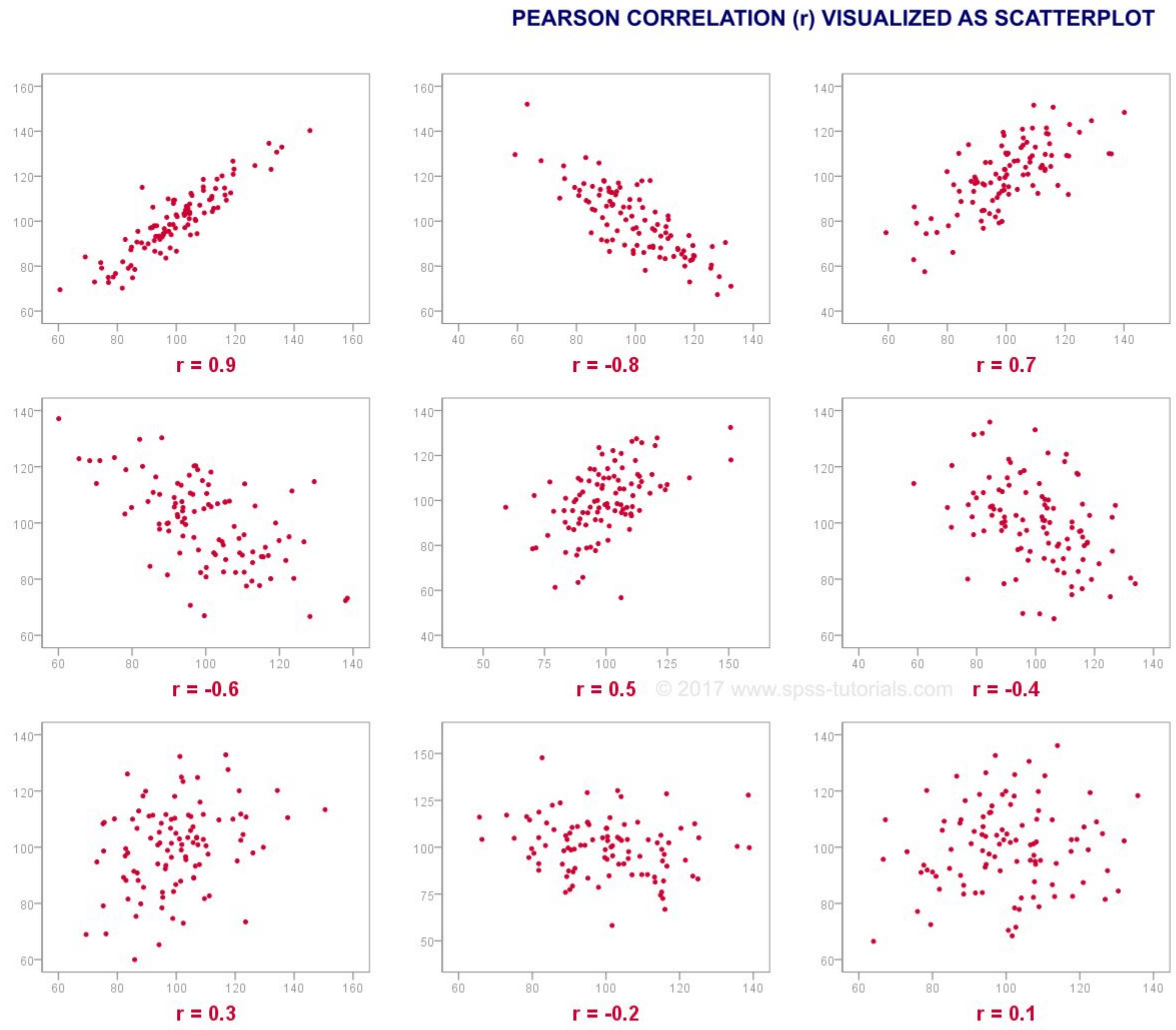


Distance Measure: **Pearson Correlation**

- Pearson correlation measures the degree of a linear relationship between two profiles.

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

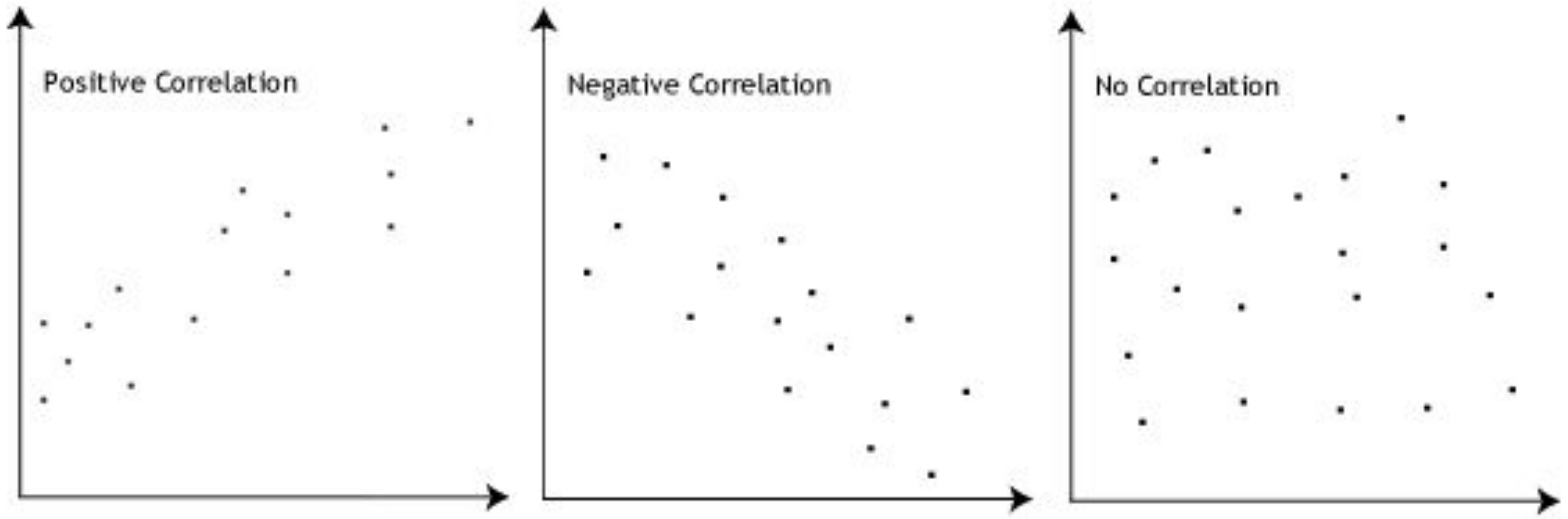


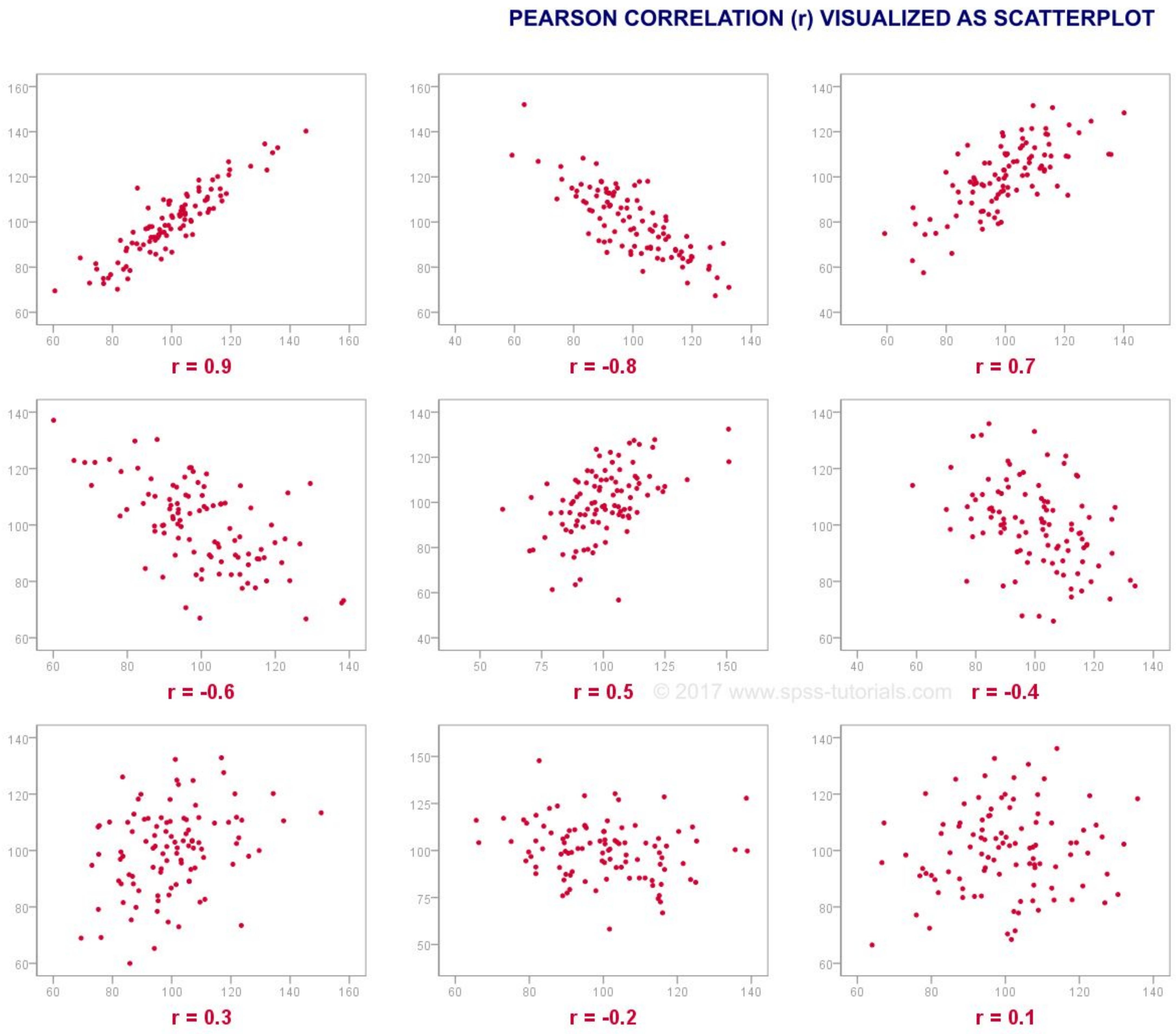


Distance Measure: **Pearson Correlation**

- Pearson correlation measures the degree of a linear relationship between two profiles.

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$





Picking the right **Distance Measure**

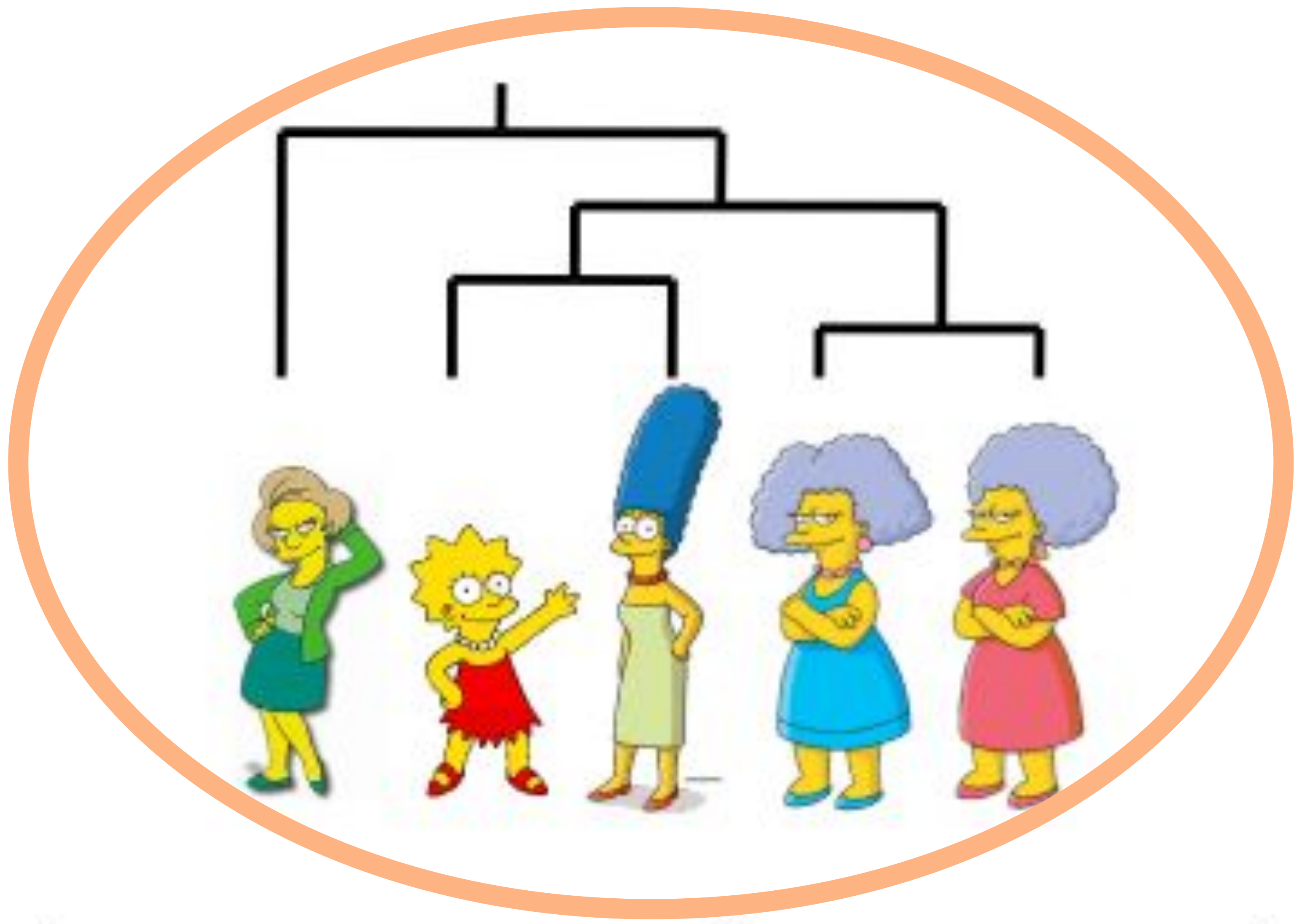
- Actually, there are many more distance measures… In the case of clustering, this parameter is important since it has a **strong influence in the results**. The most common one is the **Euclidean.**

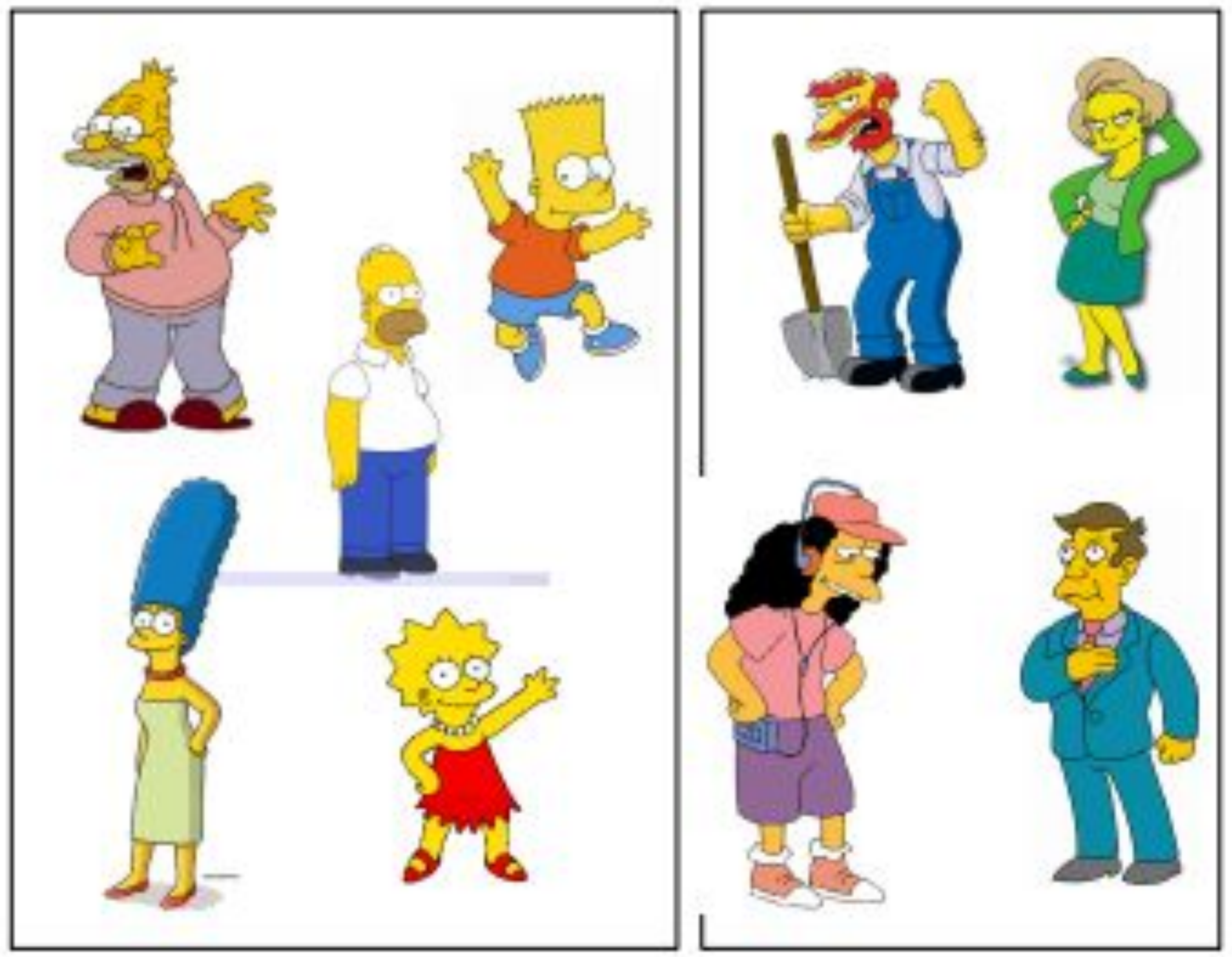
  Correlation based distance considers two objects together if their features are **highly correlated**, even if their values are far apart by Euclidean distance measures.

- Because of this, it is extremely **important to scale the data.** We would like to be able to compare between values.

Types of Clustering **Algorithms**

- **Hierarchical algorithms**
    - Examples are organized as a binary tree
    - No explicit division in groups
        - Bottom-up
        - Top-down
- **Partitional algorithms**
    - Usually start with a random (partial) partitioning
    - Refine it iteratively:
        - K-means clustering
        - Mixture-model based clustering

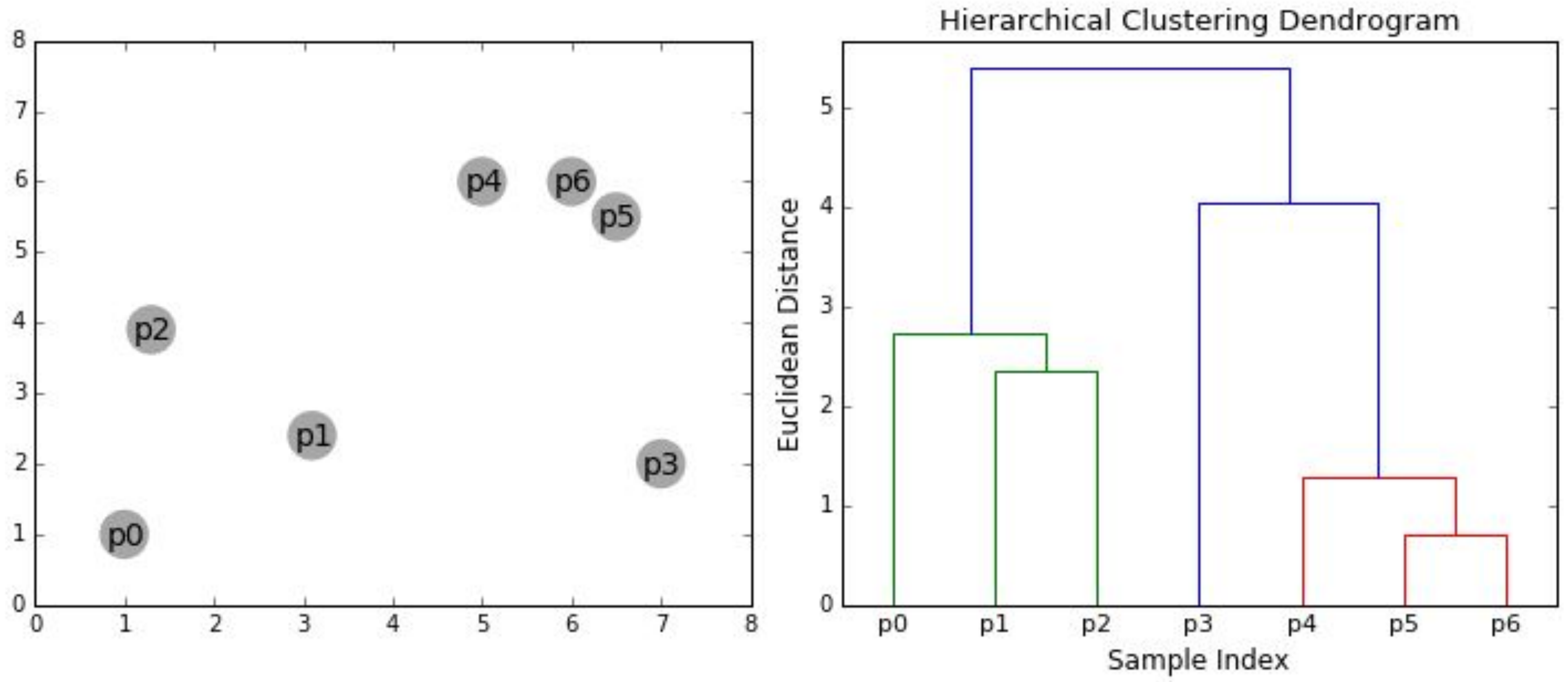


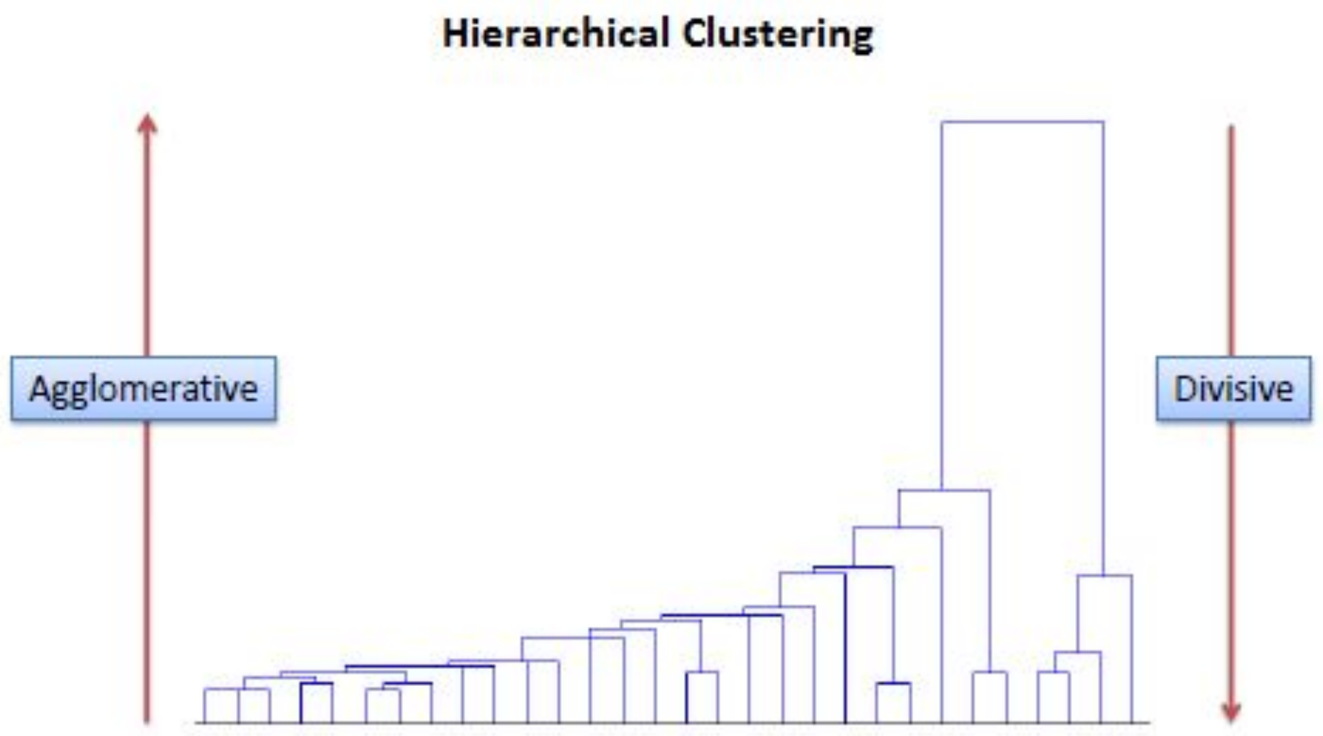


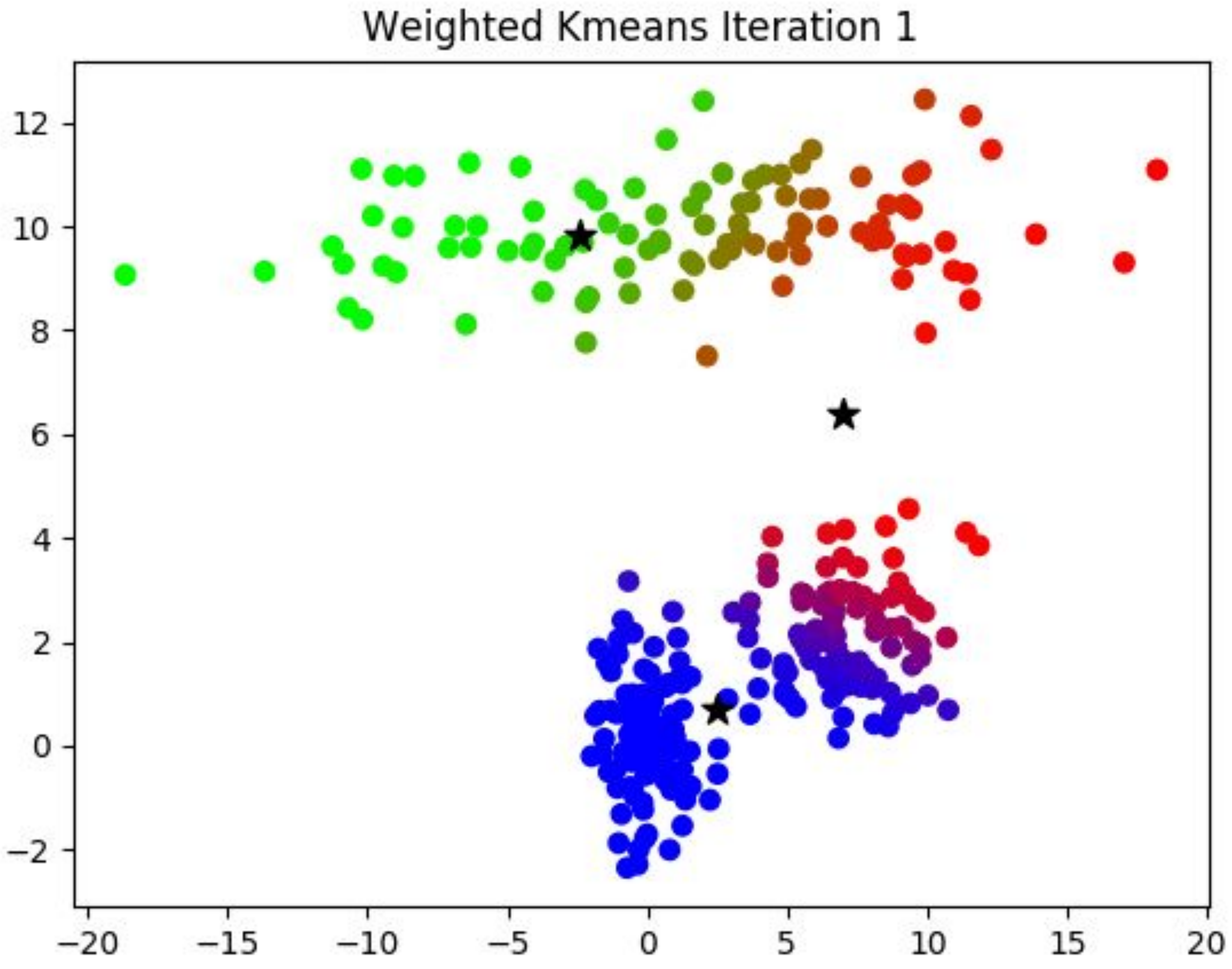
Types of Clustering **Algorithms**

- **Hierarchical algorithms**

Partitional algorithm

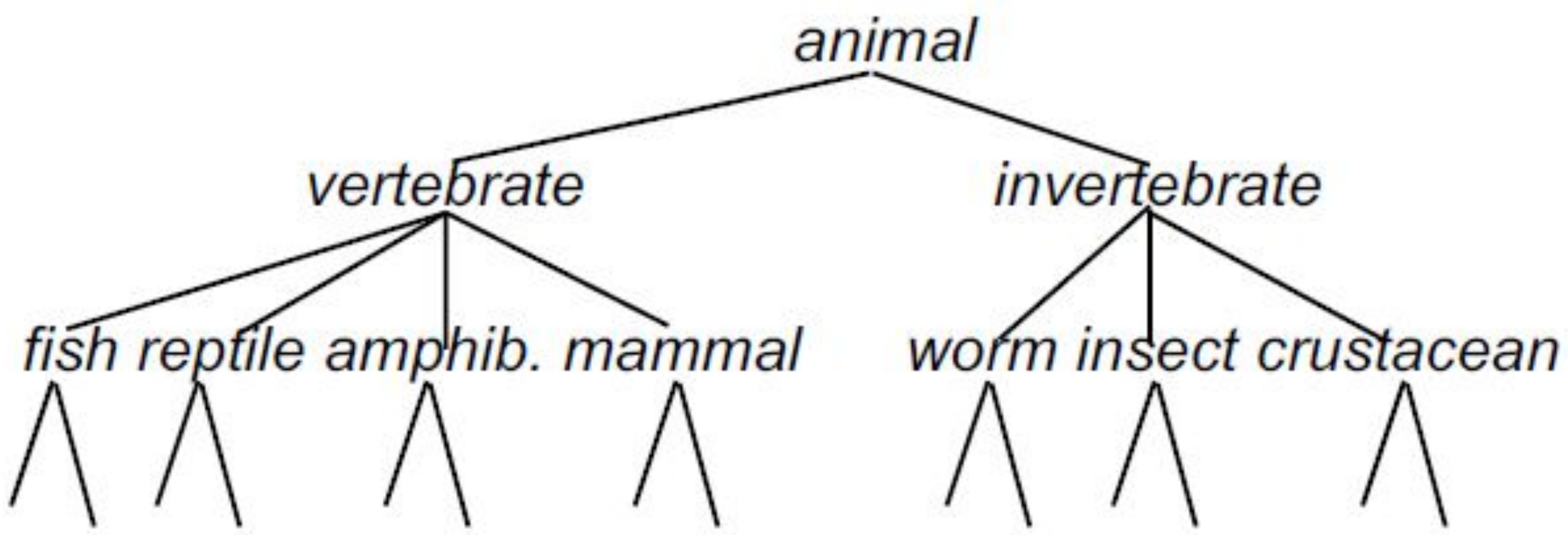


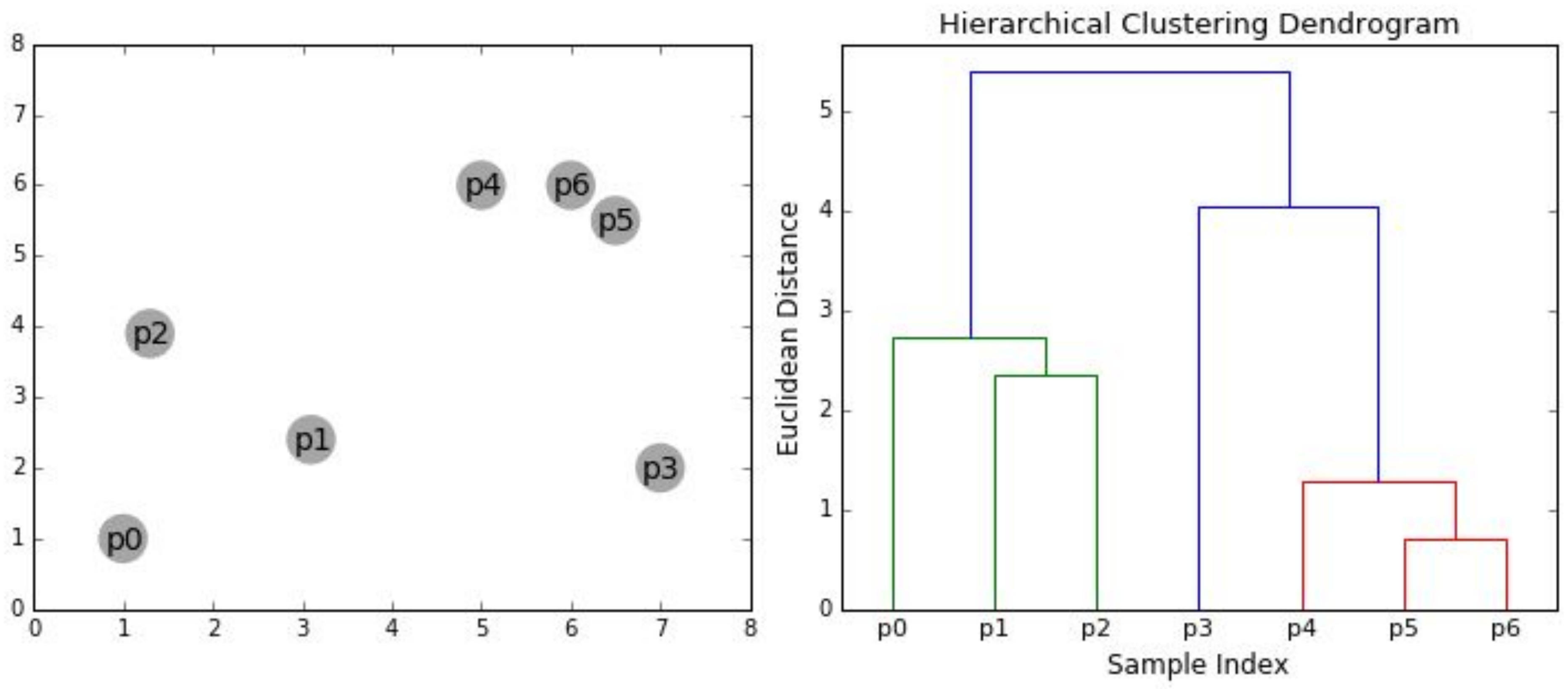




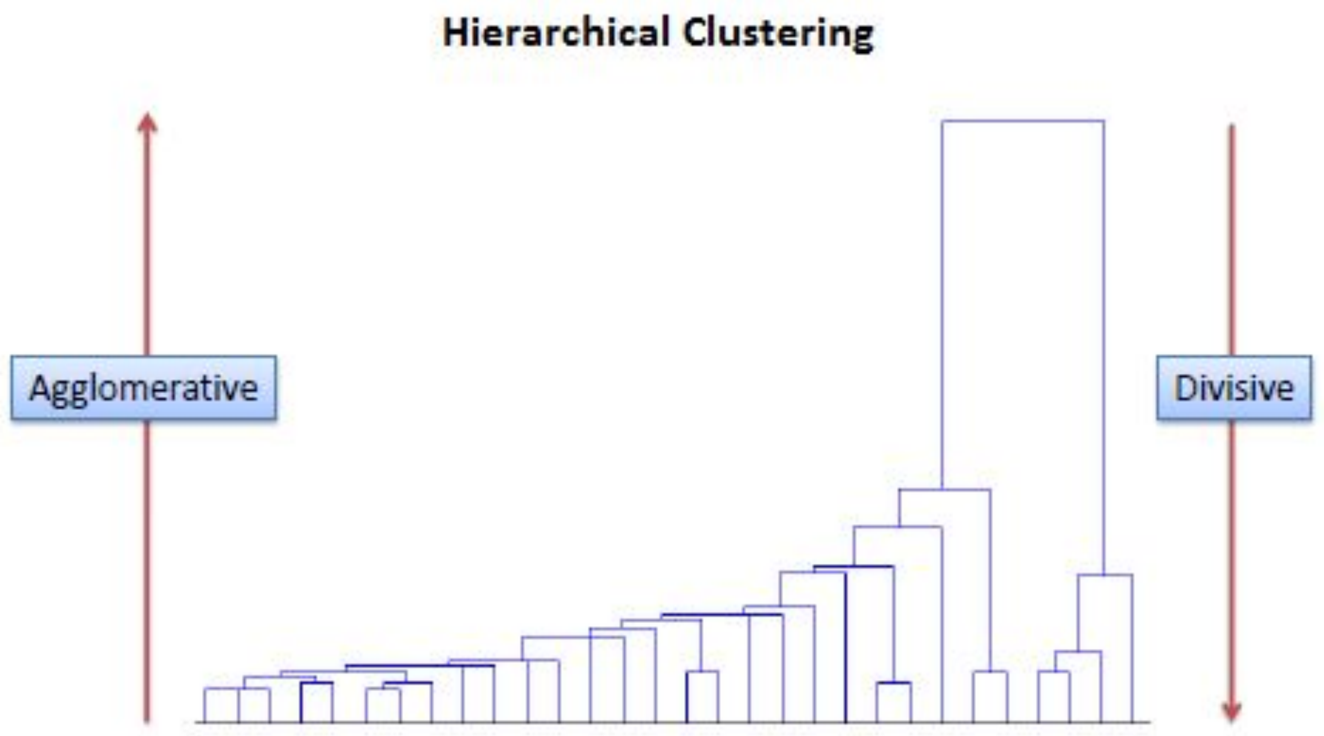
Hierarchical Clustering

- Build a tree-based hierarchical taxonomy (dendrogram) from a set of unlabeled examples



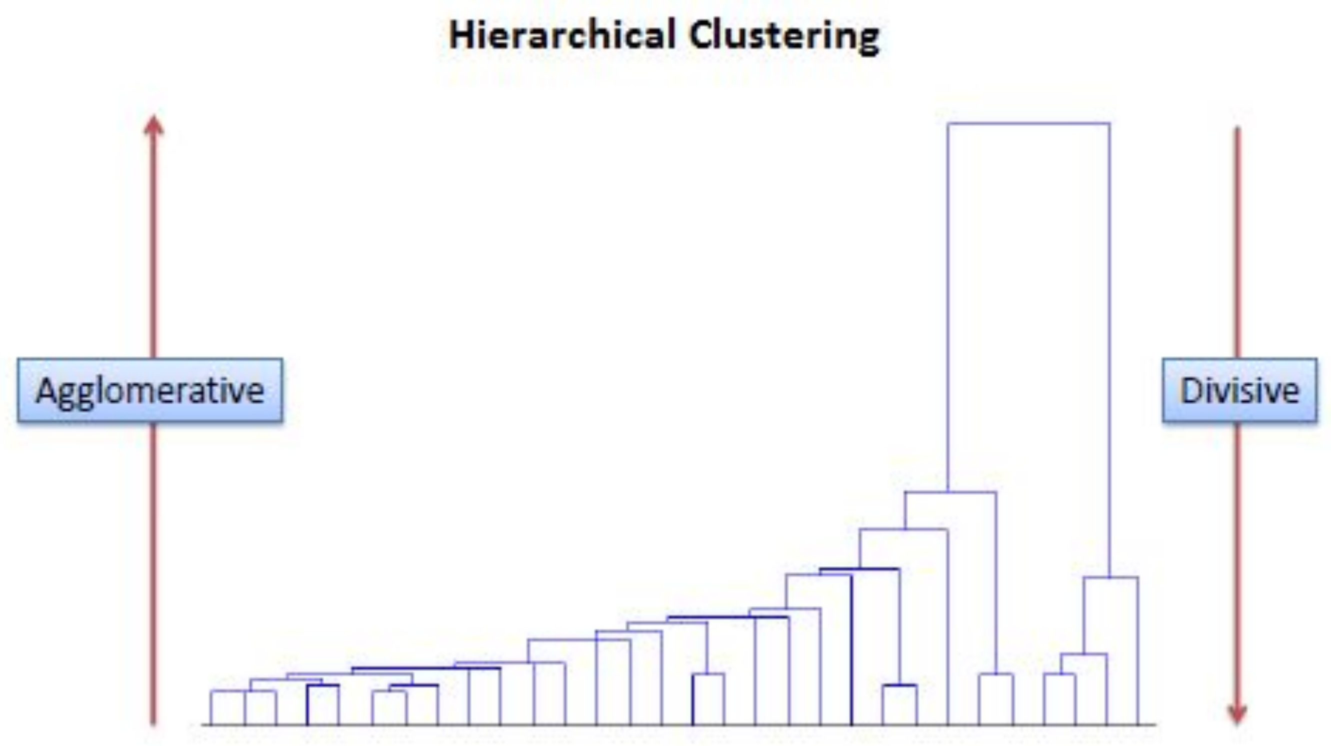


- Recursive application of a standard clustering algorithm can produce a hierarchical clustering



Hierarchical Clustering

- **Agglomerative (bottom-up)**
  - Methods start with each example in its own cluster
  - Iteratively combine them to form larger and larger clusters
- **Divisive (partitional, top-down)**
  - Methods start with all the examples in a single cluster
  - Consider all the possible way to divide the cluster into two. Choose the best division
  - Recursively operate on



Hierarchical Agglomerative Clustering

- **Basic Hierarchical Agglomerative Clustering algorithm:**

```
1. Compute the similarity matrix between the input data points
2. Start with all instances in their own cluster
3. Repeat
4.      Among the current clusters, determine the two
            clusters, ci and cj , that are most similar.
5.      Merge them and replace ci and cj with a single cluster ci U cj
6.      Update the similarity matrix
7. until there is only one single cluster
```

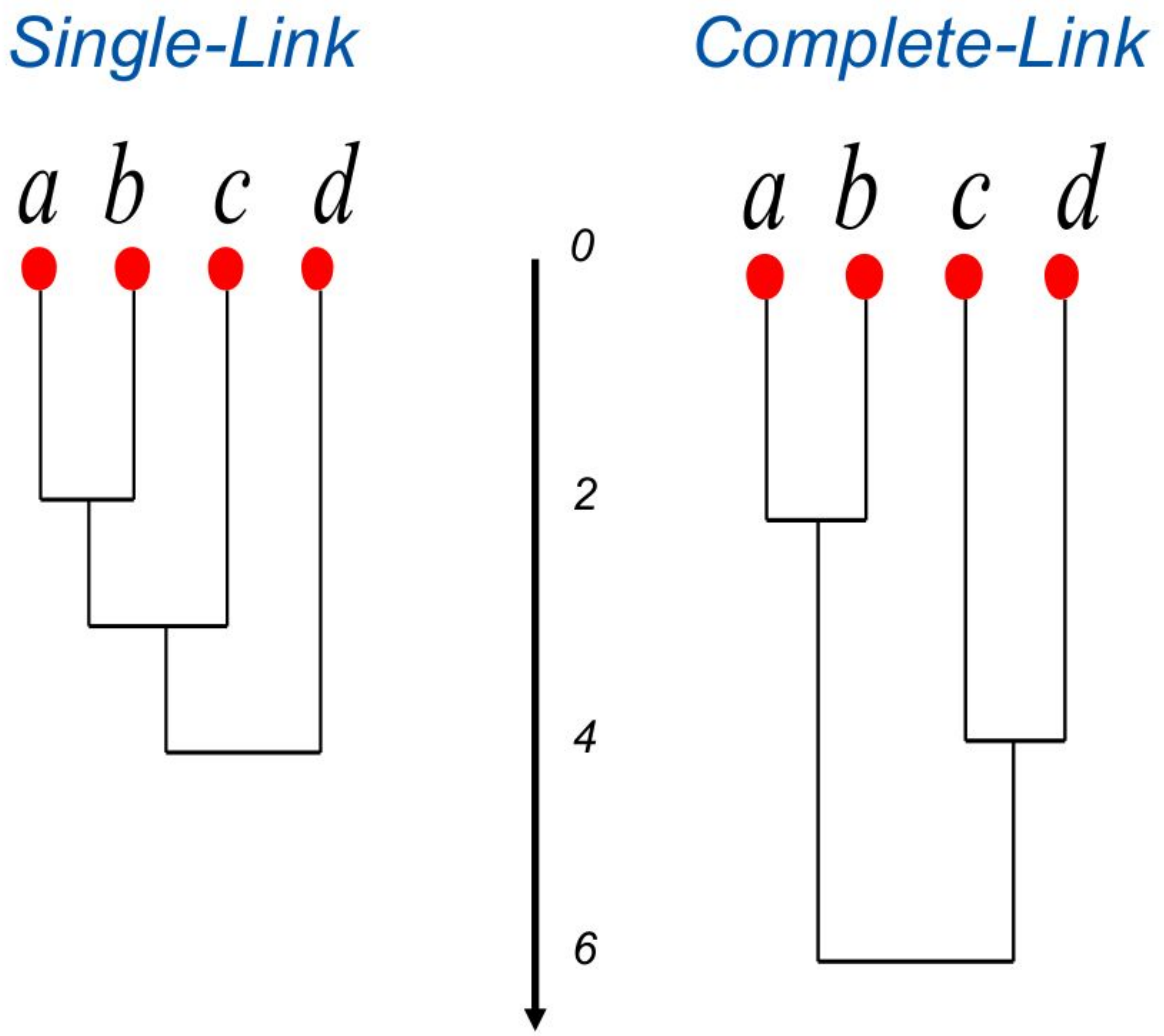
- **Key operation is the computation of the similarity between two clusters**
  - Different definitions of the similarity between clusters lead to different algorithms

Cluster Similarity

- **Assume a similarity function that determines the similarity of two instances: sim(x,y)**
  - For example, Cosine similarity of document vectors
- **How to compute similarity of two clusters each possibly containing multiple instances?**
  - **Single Link:** Similarity of two most similar members
  - **Complete Link:** Similarity of two least similar members
  - **Group Average:** Average similarity between members
  - **Centroid:** clusters whose centroids are the most cosine similar

Cluster Similarity: Simple vs Complete Link

- **Simple Link:** We pay attention solely to the area where the two clusters come closest to each other (Most similar)
- **Complete Link:** Looks for dissimilarity. This merge criterion is non-local; the entire structure of the clustering can influence merge decisions.

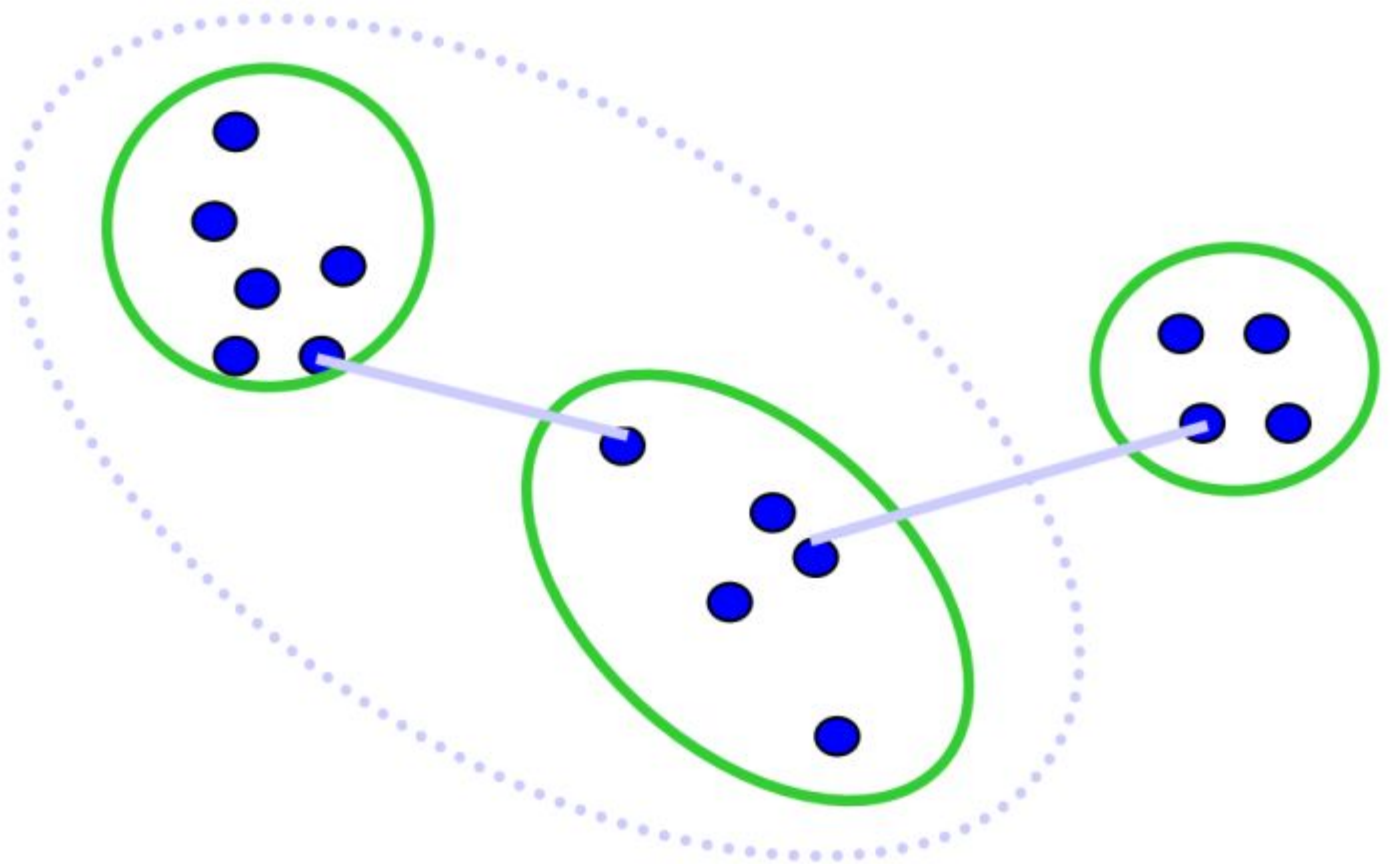


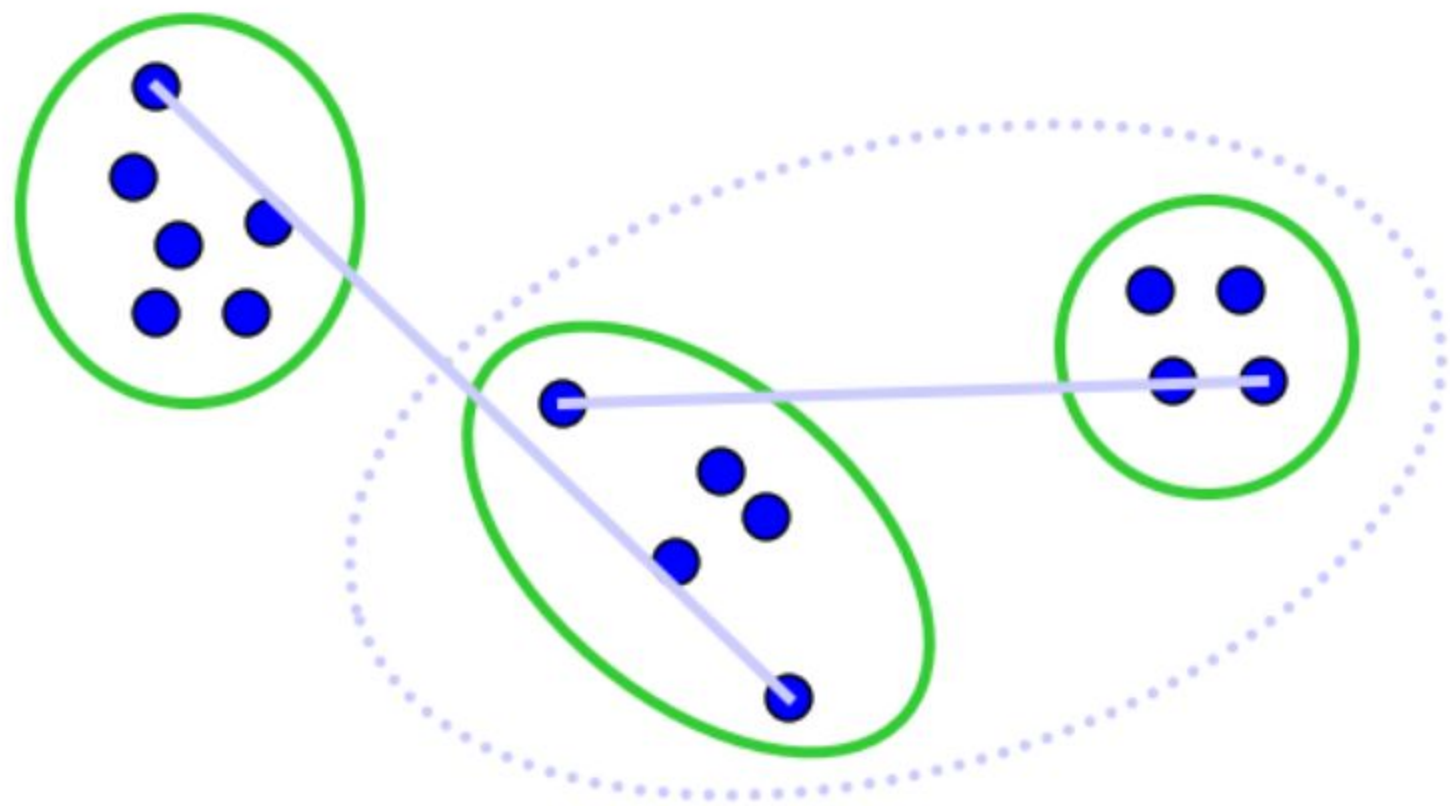
Some things to ponder about

- In the case of Single-Link Agglomerative Clustering:
- Using maximum similarity of pairs vs minimum similarity of pairs:

$$sim(c_i, c_j) = \max_{x \in c_i, y \in c_j} sim(x, y) \qquad sim(c_i, c_j) = \min_{x \in c_i, y \in c_j} sim(x, y)$$

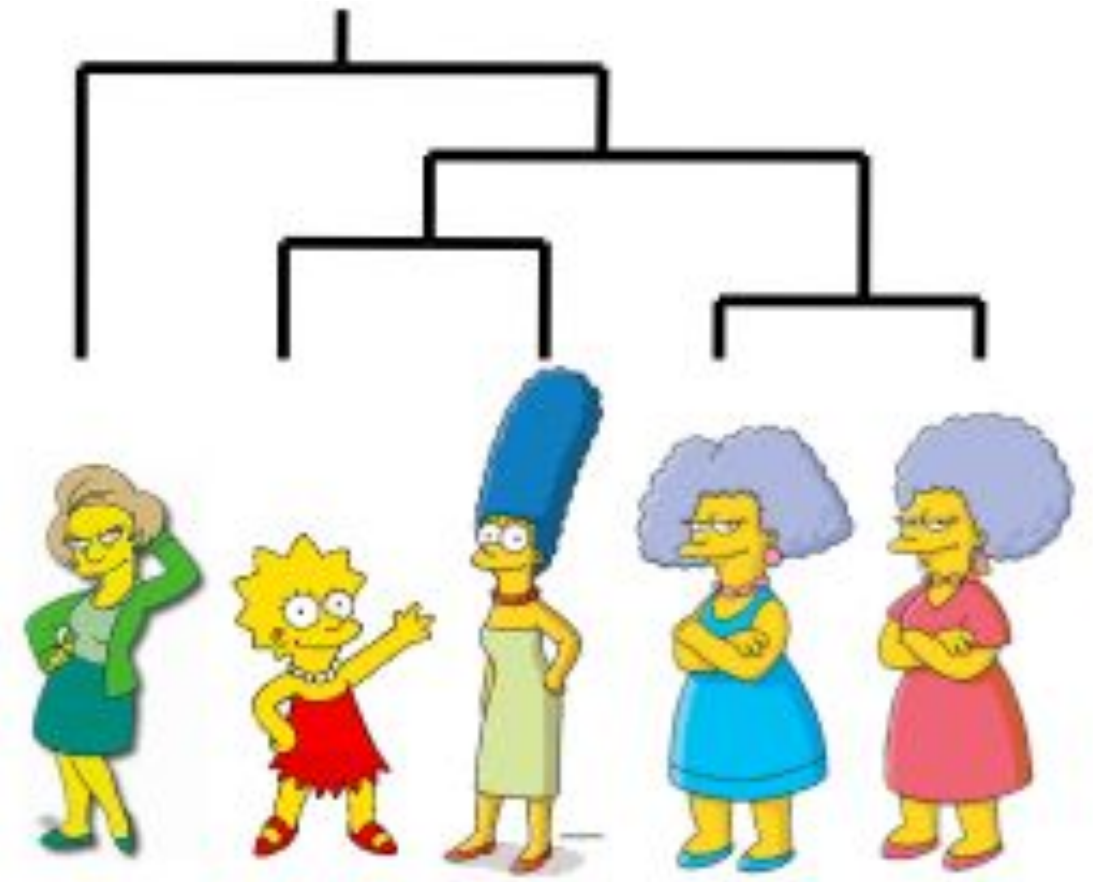
- Maximum can lead to long and thin clusters due to the chaining effect
- Minimum makes more "tight", spherical clusters that are typically desired

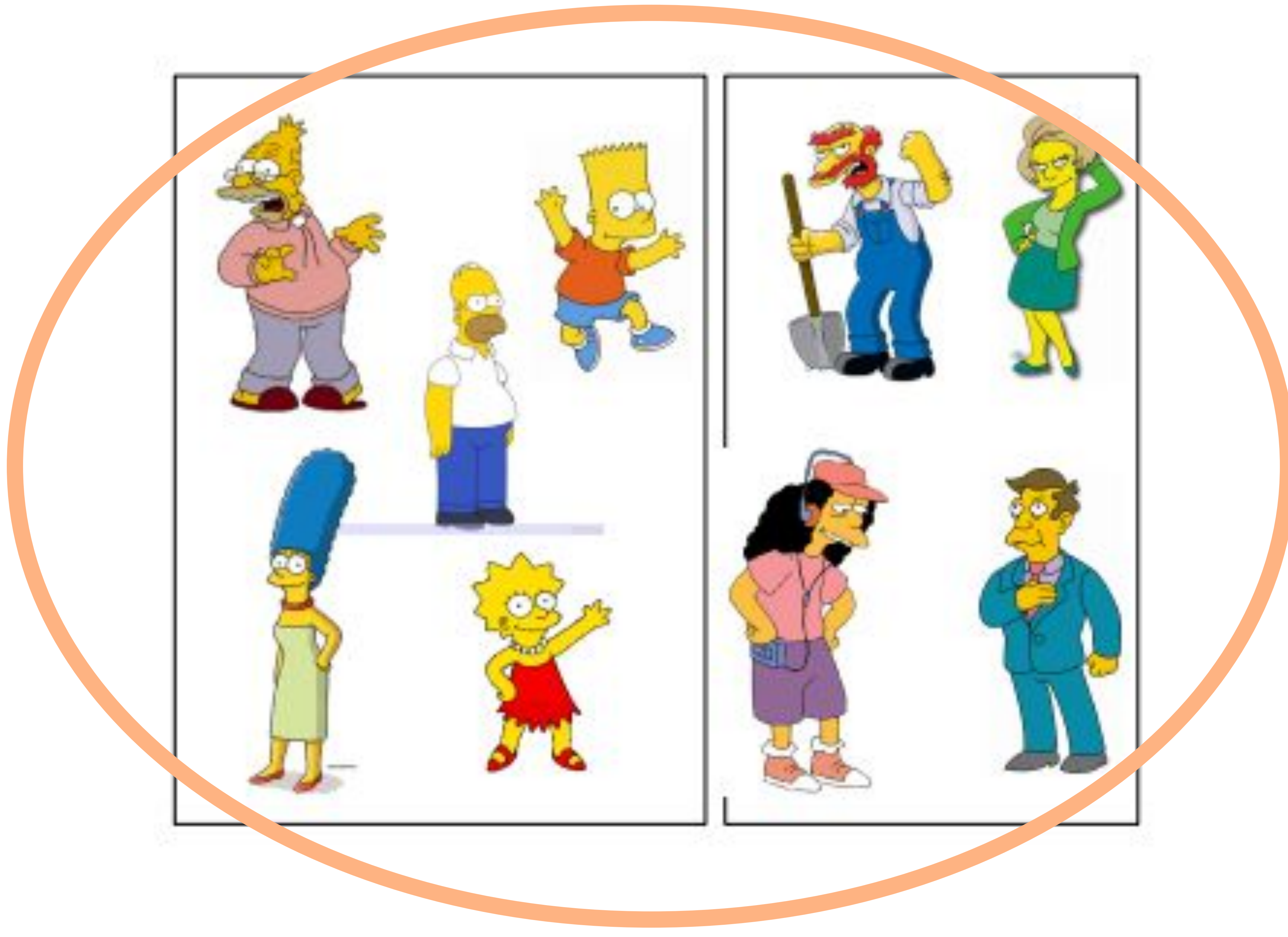




Types of Clustering **Algorithms**

- **Hierarchical algorithms**
  - Examples are organized as a binary tree
  - No explicit division in groups
    - Bottom-up
    - Top-down
- **Partitional algorithms**
  - Usually start with a random (partial) partitioning
  - Refine it iteratively:
    - K-means clustering
    - Mixture-model based clustering





Clustering: Partitional Algorithms

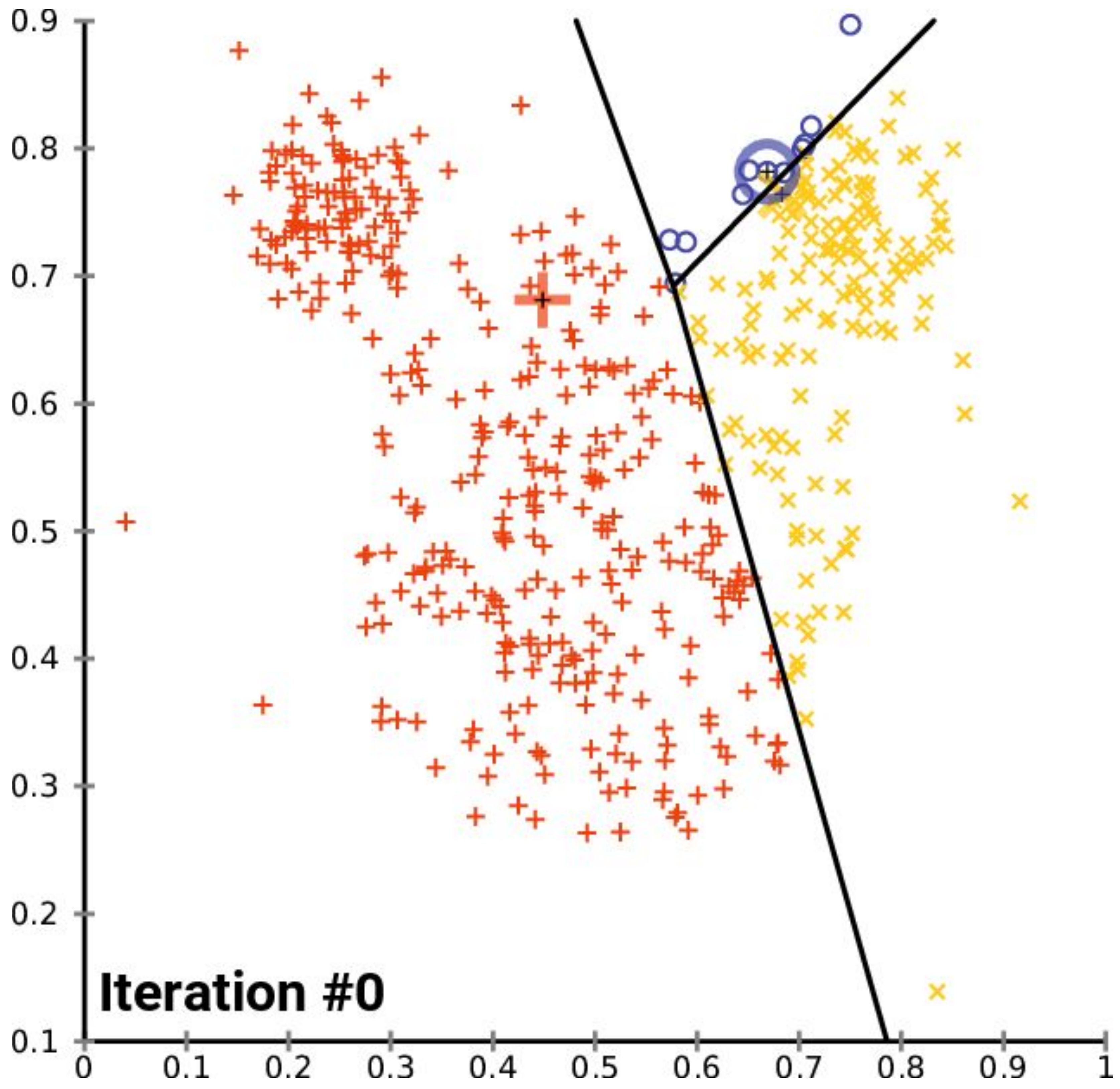
- **Method:** construct a partition of **n** objects into a set of **k** clusters
- **Given:** a set of objects (training set) and typically must provide the number of desired clusters, **K**.
- **Basic process:**
  - Randomly choose K instances as **seeds**, one per cluster
  - Form initial clusters based on these seeds
  - Iterate , repeatedly reallocating instances to different clusters to improve the overall clustering
  - Stop when clustering converges or after a fixed number of iterations

Clustering: Basis for K-Means

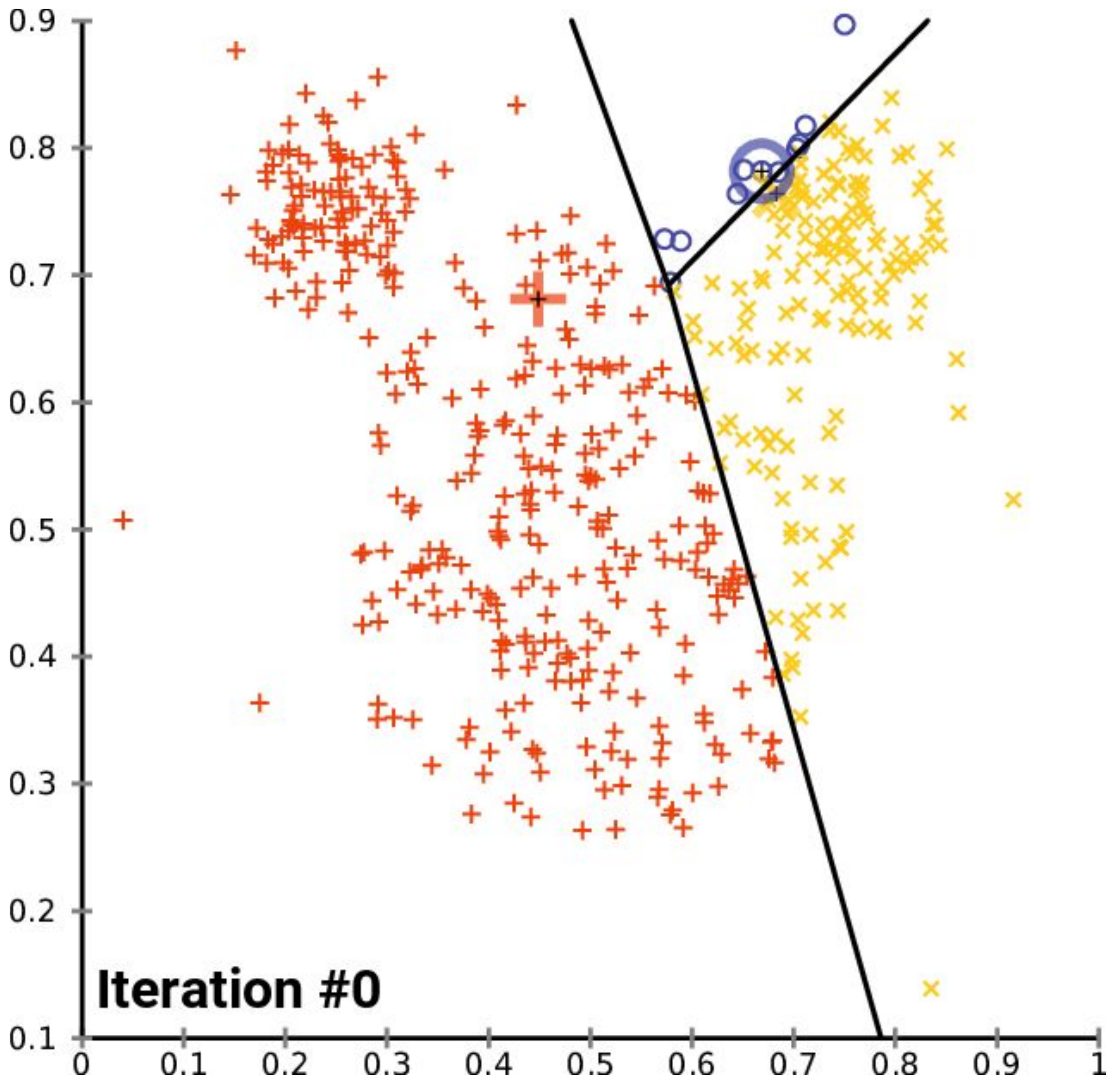
- This is an important Algorithm!
- Assumes instances are real-valued vectors
- Clusters based on centroids, **center of gravity,** or mean of points in a cluster, c:

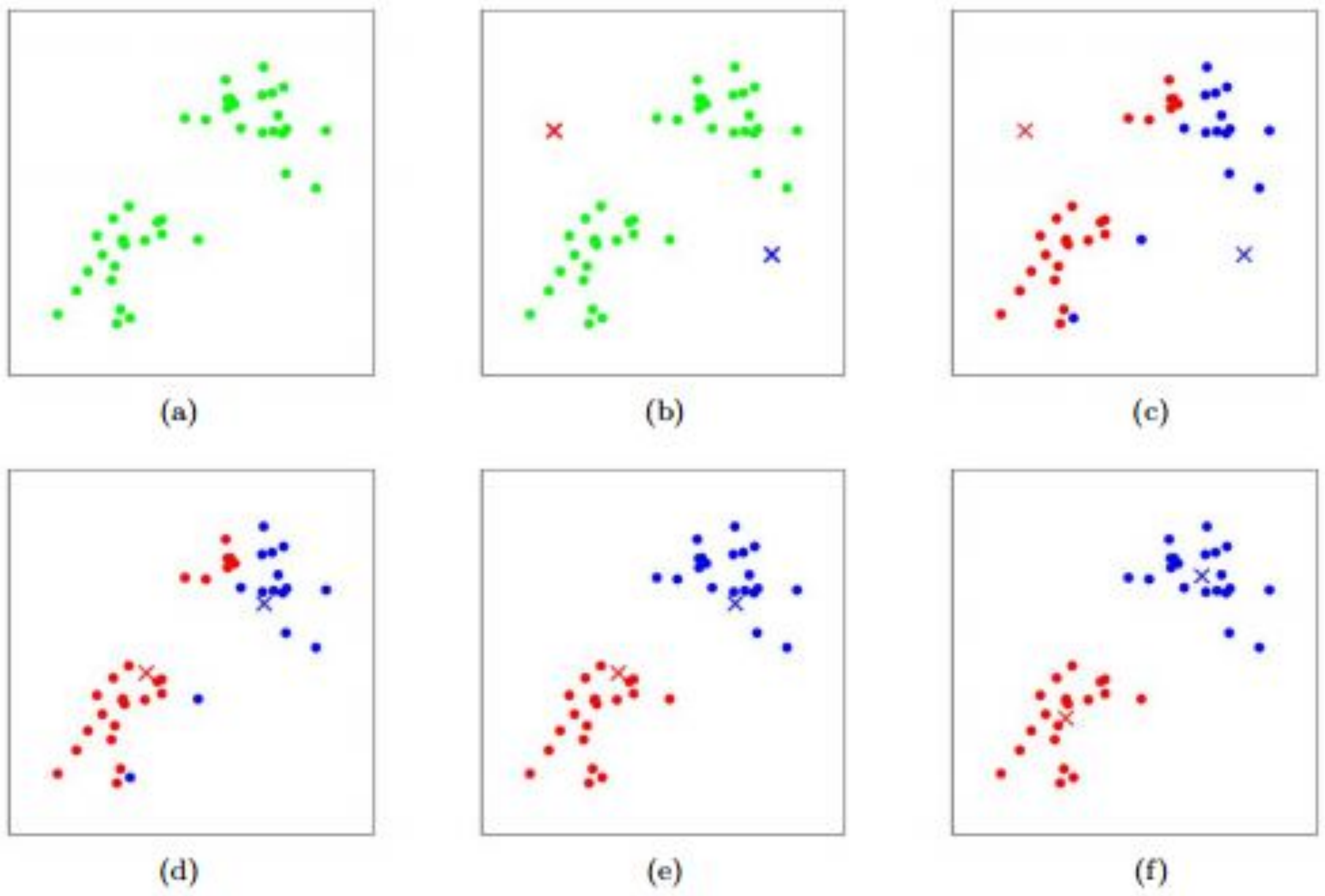
$$\vec{\mu}(c) = \frac{1}{|c|} \sum_{\vec{x} \in c} \vec{x}$$

- Reassignment of instances to clusters is based on distance to the current cluster centroid



Clustering: K-Means Example





K-Means Algorithm

- Let **d** be the **distance measure** between instances.

1. *Decide on a value for k*
2. *Select k random instances {s_1, s_2,… s_k} as seeds.*
3. *(Decide the class membership)*

For each instance x_i:

Assign x_i to the cluster c_j such that $d(x_i, s_j)$ is minimal.

4. *(Update the seeds to the centroid of each cluster)*

For each cluster c_j

$s_j = \mu(c_j)$

$$\vec{\mu}_k = \frac{1}{\mathcal{C}_k} \sum_{i \in \mathcal{C}_k} \vec{x}_i$$

5. *(Until clustering converges or other stopping criterion):*

If none of the N instances changed membership, exit

Otherwise, go to step 3

- More details in the following **highlighted chapter**.

K-Means Objective

- The **objective** of k-means is to **minimize the total sum of the squared distance of every point to its corresponding cluster centroid**.

$$\text{Goodness measure (SD)} = \sum_{l=1}^{K} \sum_{x_i \in X_l} \| x_i - \mu_l \|^2$$

- Finding the global optimum is NP-hard.
- The k-means algorithm is guaranteed to converge a local optimum.

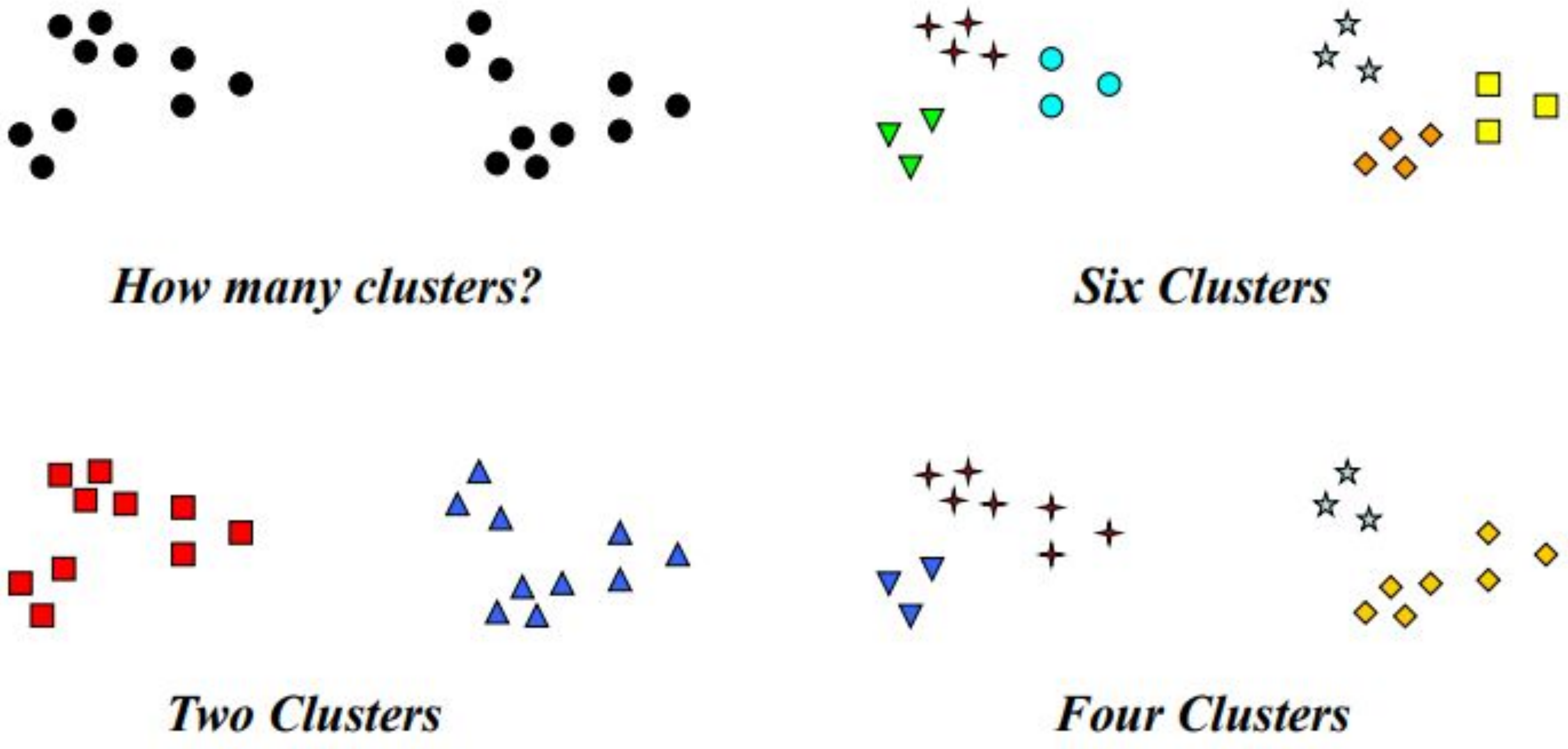
Comments on K-Means Method

- **Strengths:**
  - Relatively Efficient (computationally)
  - Often terminates at a local optimum
- **Weakness:**
  - Applicable only when mean is defined… What about categorical data?
  - Need to specify k, the number of clusters, in advance
  - Unable to handle noisy data and outliers
  - Not suitable is clusters have non-convex shapes
- **Seed Choice:** Results can vary based on Random seed Choice
  - Some seeds can result in poor convergence. Important to **start** several times.

Variations on the K-Means Method

- Because of this, there are several variations that improve on the K-Means method. We cannot cover them here but some ideas are:
    - Bisecting KMeans
    - K-Means ++
    - Fuzzy Clustering
    - Soft Clustering

Clustering Validation



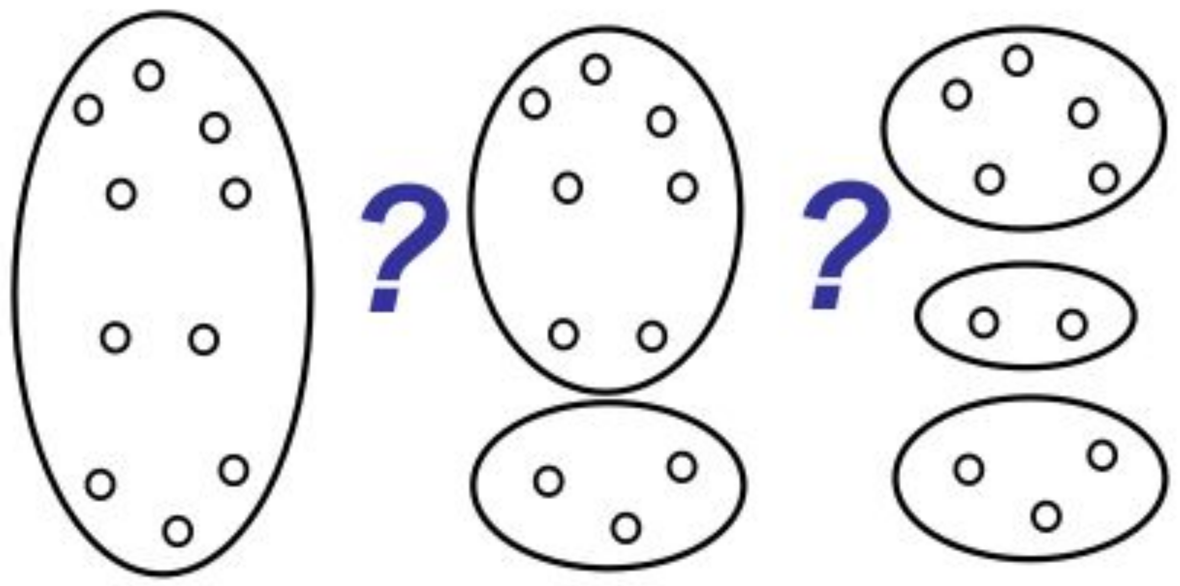
- Which is the best cluster?

What makes a good Cluster?

- **Internal criterion:** A good clustering will produce high quality clusters in which:
  - the intra-class (that is, intra-cluster) similarity is high
  - the inter-class similarity is low
  - The measured quality of a clustering depends on both the example representation and the similarity measure used
- **External criterion:** The quality of a clustering is also measured by its ability to discover some or all of the hidden patterns or latent classes
  - Assessable with gold standard data

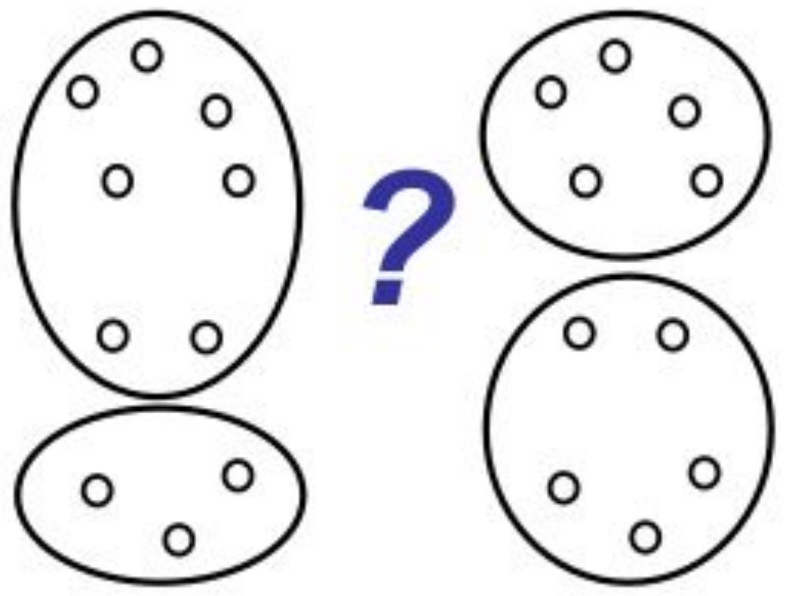
What makes a good Cluster?

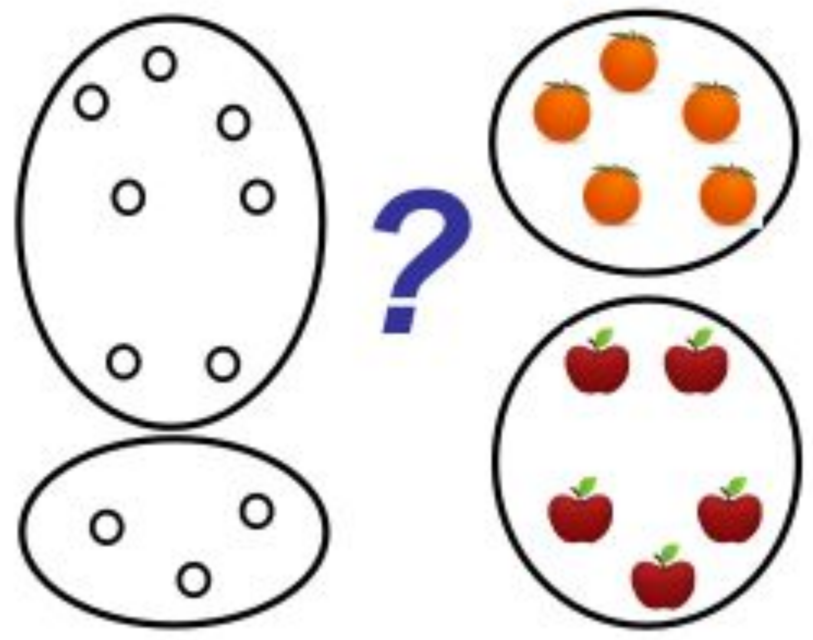
- **Internal Index**
  - Validate without external info
  - With different number of clusters
  - Solve the number of clusters



- **External Index**
  - Validate against ground truth
  - Compare tw :

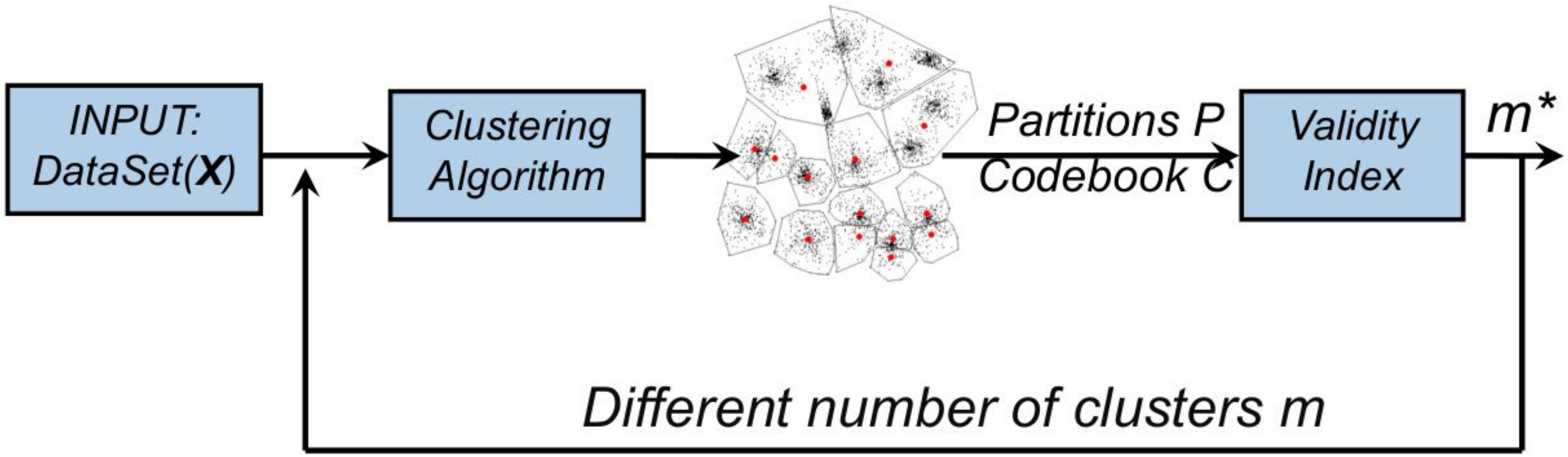
    (how similar)





What makes a good Cluster?

- **Cluster validation** refers to procedures that evaluate the results of clustering in a quantitative and objective fashion
  - How to be "**quantitative**": To employ the measures.
  - How to be "**objective**": To validate the measures!



What makes a good Cluster? Internal Indexes

- **Sklearn** and other platforms will do this for us :)
- Otherwise, we'll have to delve into the math

Name	Formula
SSW	$SSW = \frac{1}{N}\sum_{i=1}^{N} \left\Vert x_i - C_{p_i} \right\Vert^2$
SSB	$SSB = \frac{2}{M(M-1)}\sum_{i=1}^{M}\sum_{j=1, j\neq i}^{M} \left\Vert C_i - C_j \right\Vert^2$
Calinski-Harabasz index	$CH = \frac{SSB/(M-1)}{SSW(N-M)}$
Hartigan	$H_M = (\frac{SSW_M}{SSW_{M+1}} - 1)(N - M - 1)$ $or : H_M = \log\left(SSB_M / SSW_M\right)$
Krzanowski-Lai index	$diff_M = (M-1)^{2/D} SSW_{M-1} - M^{2/D} SSW_M$ $KL_M = \lvert diff_M \rvert \;/\; \lvert diff_{M+1} \rvert$
Ball&Hall	$BH_M = SSW_M / M$
Xu-index	$Xu = D \log\left(\sqrt{SSW_M/(DN^2)}\right) + \log M$
Dunn's index	$Dunn = \sum_{i=1}^{M} \frac{\max\left(\left\Vert x_j - C_i \right\Vert^2\right)_{j \in C_i}}{}$
Davies&Bouldin index	$R_{ij} = \frac{S_i + S_j}{d_{ij}}, i \neq j$ $where : d_{ij} = \left\Vert C_i - C_j \right\Vert^2, S_i = \frac{1}{n_i}\sum_{j=1}^{n_i} \left\Vert x_j - C_i \right\Vert^2$ $and, R_i = \max_{j=1,\ldots,M} R_{ij}, i = 1, \ldots, M$ $DBI = \frac{1}{M}\sum_{i=1}^{M} R_i$

What makes a good Cluster? External Indexes

Table 1: External Cluster Validation Measures.

	Measure	Notation	Definition	Range
1	Entropy	E	$-\sum_i p_i(\sum_j \frac{p_{ij}}{p_i} \log \frac{p_{ij}}{p_i})$	$[0, \log K']$
2	Purity	P	$\sum_i p_i(\max_j \frac{p_{ij}}{p_i})$	(0,1]
3	F-measure	F	$\sum_j p_j \max_i [2\frac{p_{ij}}{p_i}\frac{p_{ij}}{p_j}/(\frac{p_{ij}}{p_i}+\frac{p_{ij}}{p_j})]$	(0,1]
4	Variation of Information	VI	$-\sum_i p_i \log p_i - \sum_j p_j \log p_j - 2\sum_i\sum_j p_{ij} \log \frac{p_{ij}}{p_i p_j}$	$[0, 2\log\max(K, K')]$
5	Mutual Information	MI	$\sum_i\sum_j p_{ij} \log \frac{p_{ij}}{p_i p_j}$	$(0, \log K']$
6	Rand statistic	R	$[\binom{n}{2} - \sum_i \binom{n_{i\cdot}}{2} - \sum_j \binom{n_{\cdot j}}{2} + 2\sum_{ij}\binom{n_{ij}}{2}]/\binom{n}{2}$	(0,1]
7	Jaccard coefficient	J	$\sum_{ij}\binom{n_{ij}}{2}/[\sum_i \binom{n_{i\cdot}}{2} + \sum_j \binom{n_{\cdot j}}{2} - \sum_{ij}\binom{n_{ij}}{2}]$	[0,1]
8	Fowlkes and Mallows index	FM	$\sum_{ij}\binom{n_{ij}}{2}/\sqrt{\sum_i \binom{n_{i\cdot}}{2}\sum_j \binom{n_{\cdot j}}{2}}$	[0,1]
9	Hubert Γ statistic I	Γ	$\frac{\binom{n}{2}\sum_{ij}\binom{n_{ij}}{2} - \sum_i\binom{n_{i\cdot}}{2}\sum_j\binom{n_{\cdot j}}{2}}{\sqrt{\sum_i\binom{n_{i\cdot}}{2}\sum_j\binom{n_{\cdot j}}{2}[\binom{n}{2}-\sum_i\binom{n_{i\cdot}}{2}][\binom{n}{2}-\sum_j\binom{n_{\cdot j}}{2}]}}$	(-1,1]
10	Hubert Γ statistic II	Γ'	$[\binom{n}{2} - 2\sum_i \binom{n_{i\cdot}}{2} - 2\sum_j \binom{n_{\cdot j}}{2} + 4\sum_{ij}\binom{n_{ij}}{2}]/\binom{n}{2}$	[0,1]
11	Minkowski score	MS	$\sqrt{\sum_i \binom{n_{i\cdot}}{2} + \sum_j \binom{n_{\cdot j}}{2} - 2\sum_{ij}\binom{n_{ij}}{2}}/\sqrt{\sum_j \binom{n_{\cdot j}}{2}}$	$[0, +\infty)$
12	classification error	ε	$1 - \frac{1}{n}\max_\sigma \sum_j n_{\sigma(j),j}$	[0,1)
13	van Dongen criterion	VD	$(2n - \sum_i \max_j n_{ij} - \sum_j \max_i n_{ij})/2n$	[0, 1)
14	micro-average precision	MAP	$\sum_i p_i(\max_j \frac{p_{ij}}{p_i})$	(0,1]
15	Goodman-Kruskal coefficient	GK	$\sum_i p_i(1 - \max_j \frac{p_{ij}}{p_i})$	[0,1)
16	Mirkin metric	M	$\sum_i n_{i\cdot}^2 + \sum_j n_{\cdot j}^2 - 2\sum_i\sum_j n_{ij}^2$	$[0, 2\binom{n}{2})$

Note: $p_{ij} = n_{ij}/n$, $p_i = n_{i\cdot}/n$, $p_j = n_{\cdot j}/n$.

- **Purity,** the ratio between the dominant class in the cluster πi and the size of cluster πi

More on Evaluation Methods later...