Lecture 11

# Clustering

Instructor: Ilias Tagkopoulos
iliast@ucdavis.edu

**Show me your friends and I'll tell you who you are.**

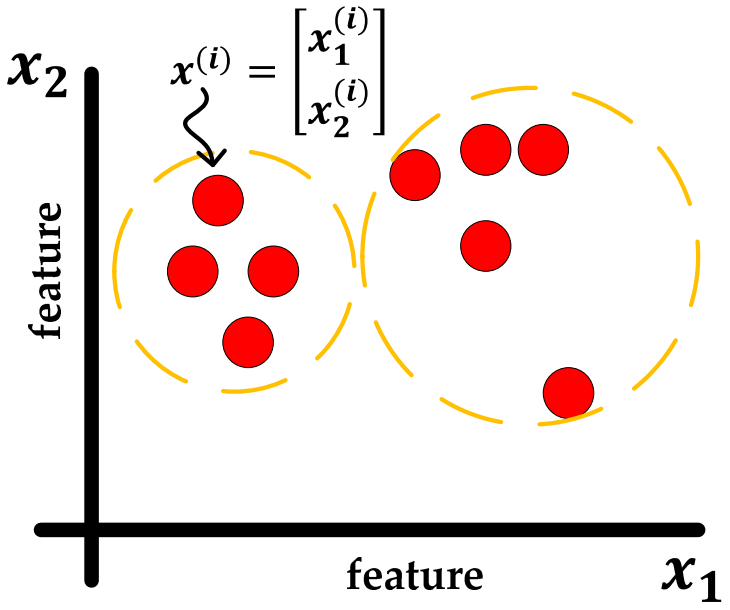
**- proverb**

# Clustering and k-means

- **Clustering:** Group a set of objects together so that objects in the same group are more similar than objects in other groups.
- ***k*-means:** a simple clustering algorithm that aims to partition *M* samples into *K* clusters.
- Assume *m n-dimensional* samples and k non-overlapping clusters. The o**bjective function** of K-means is to minimize the residual sum of squared errors (**RSS**).
- Optimization problem: assign all samples to one of the K clusters $(c_1, c_2, \ldots, c_K)$,

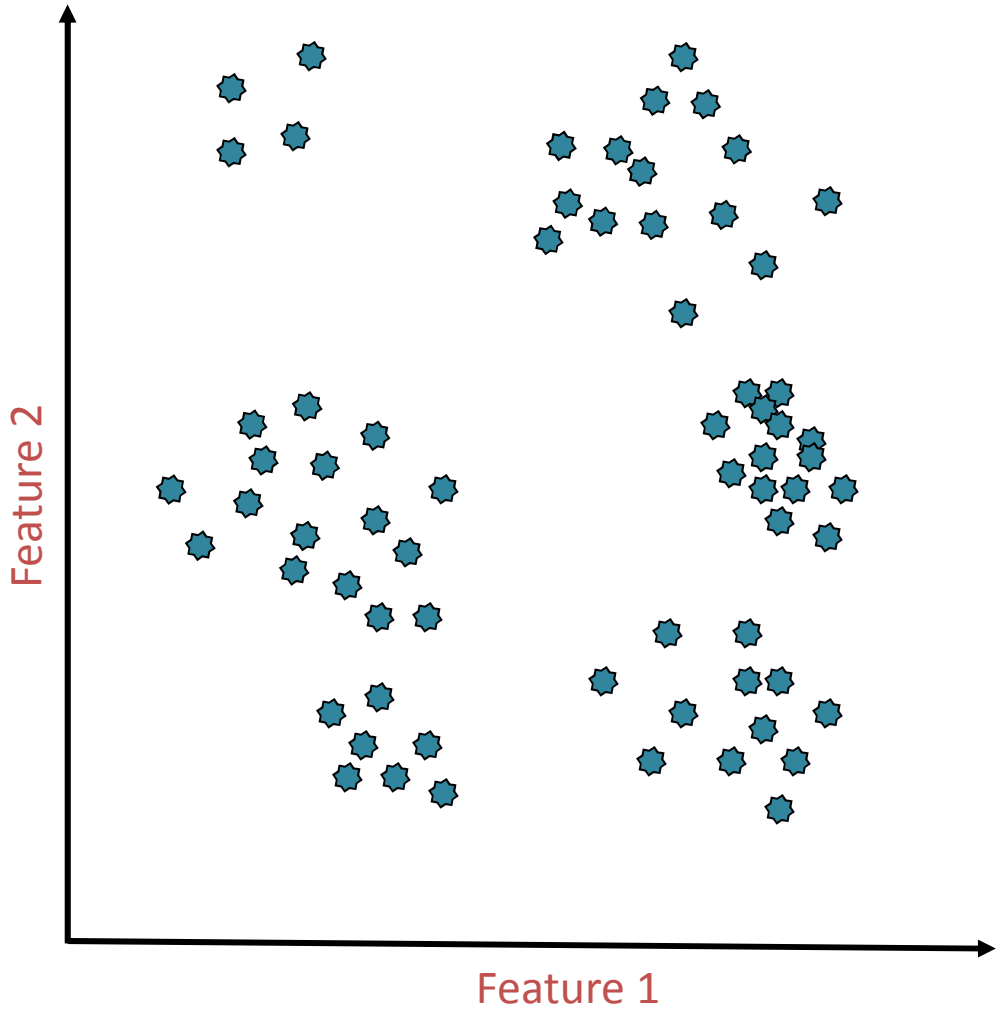
$$\min_{c} \sum_{j=1}^{K} \sum_{x^{(i)} \in c_j} \left(x^{(i)} - \mu_j\right)^2$$

Error here is the distance between a cluster member $x^{(i)}$ and its center $\mu_j$.



# Clustering and k-means

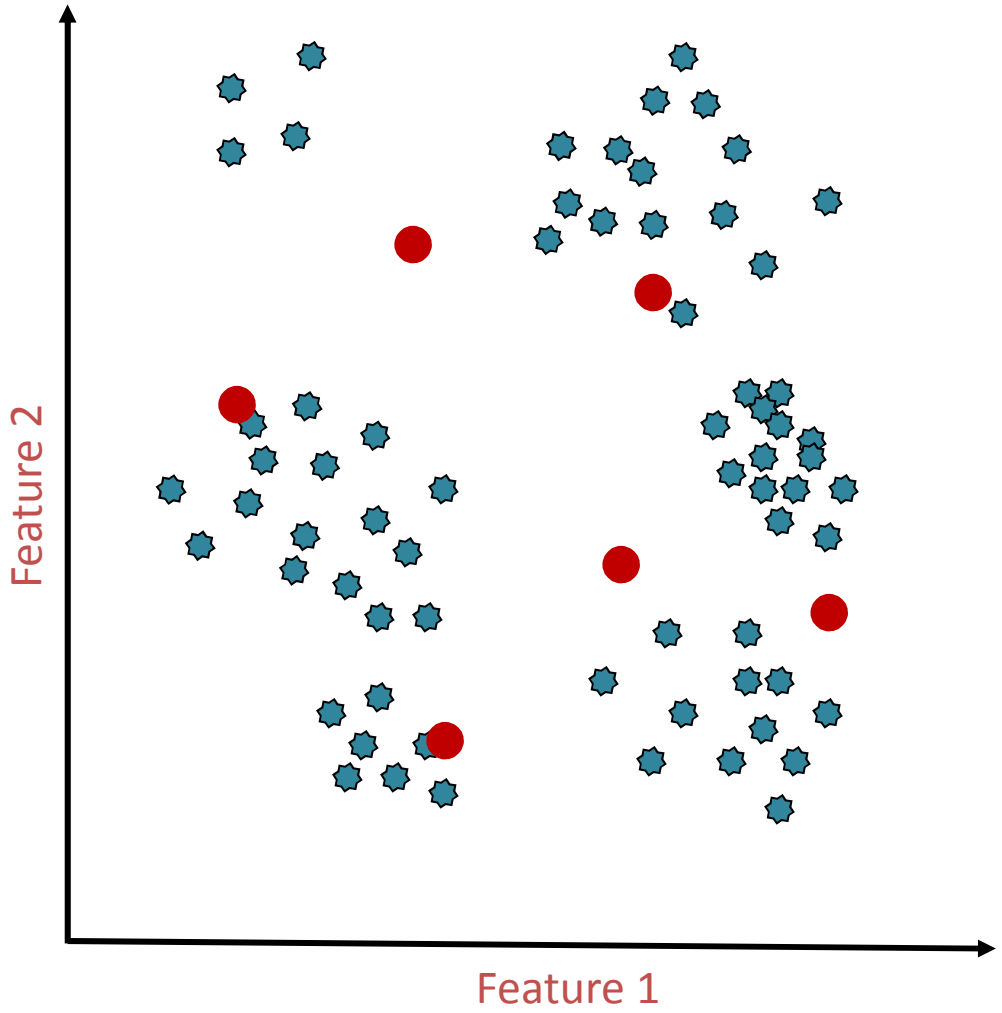
1. Start with a matrix with m rows (samples) and n columns (features)



# Clustering and k-means

1. Start with a matrix with m rows (samples) and n columns (features)

2. Assign a (random) number of *k* clusters
   - Here: $k = 6$

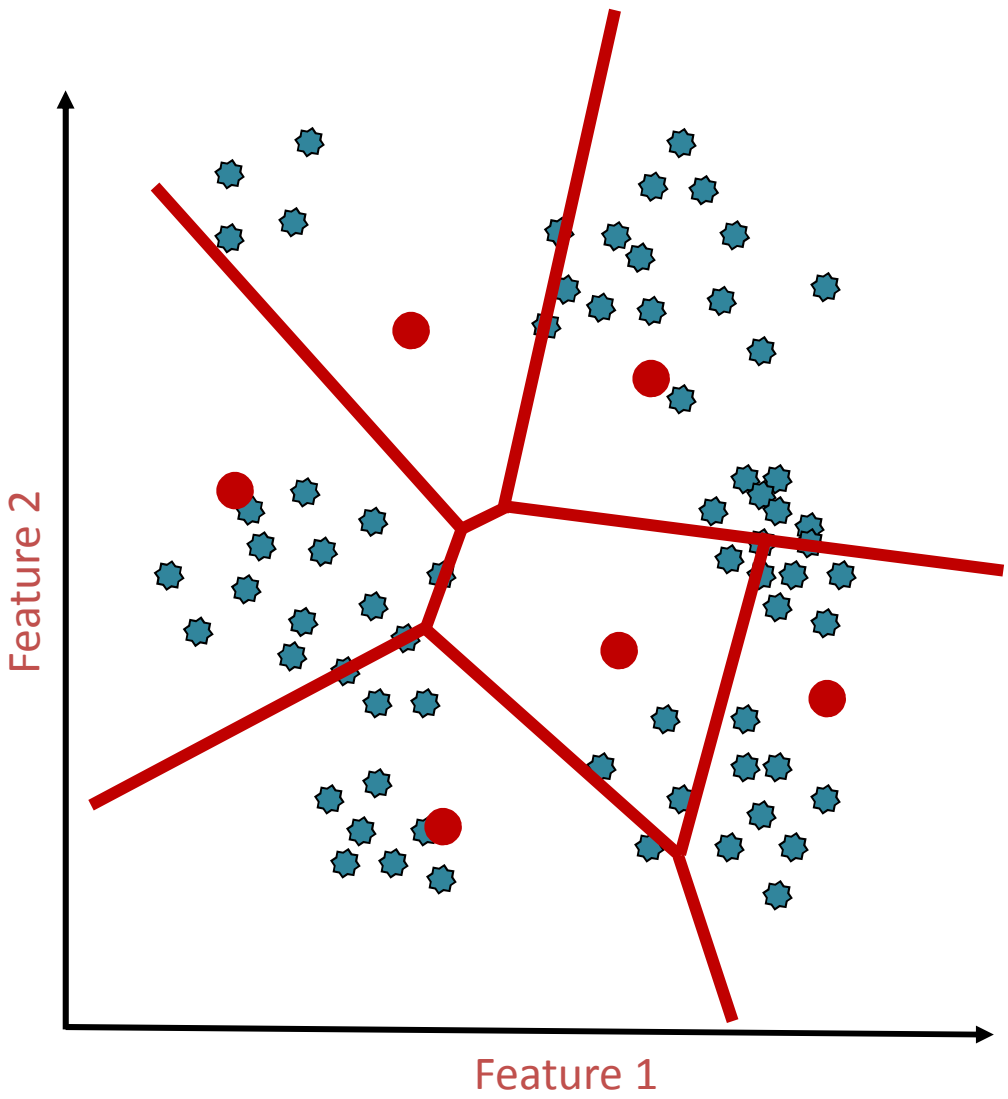


# Clustering and k-means

1. Start with a matrix with m rows (samples) and n columns (features)

2. Assign a (random) number of *k* clusters
   - Here: $k = 6$

3. Assign each sample to the cluster that it is closer to.



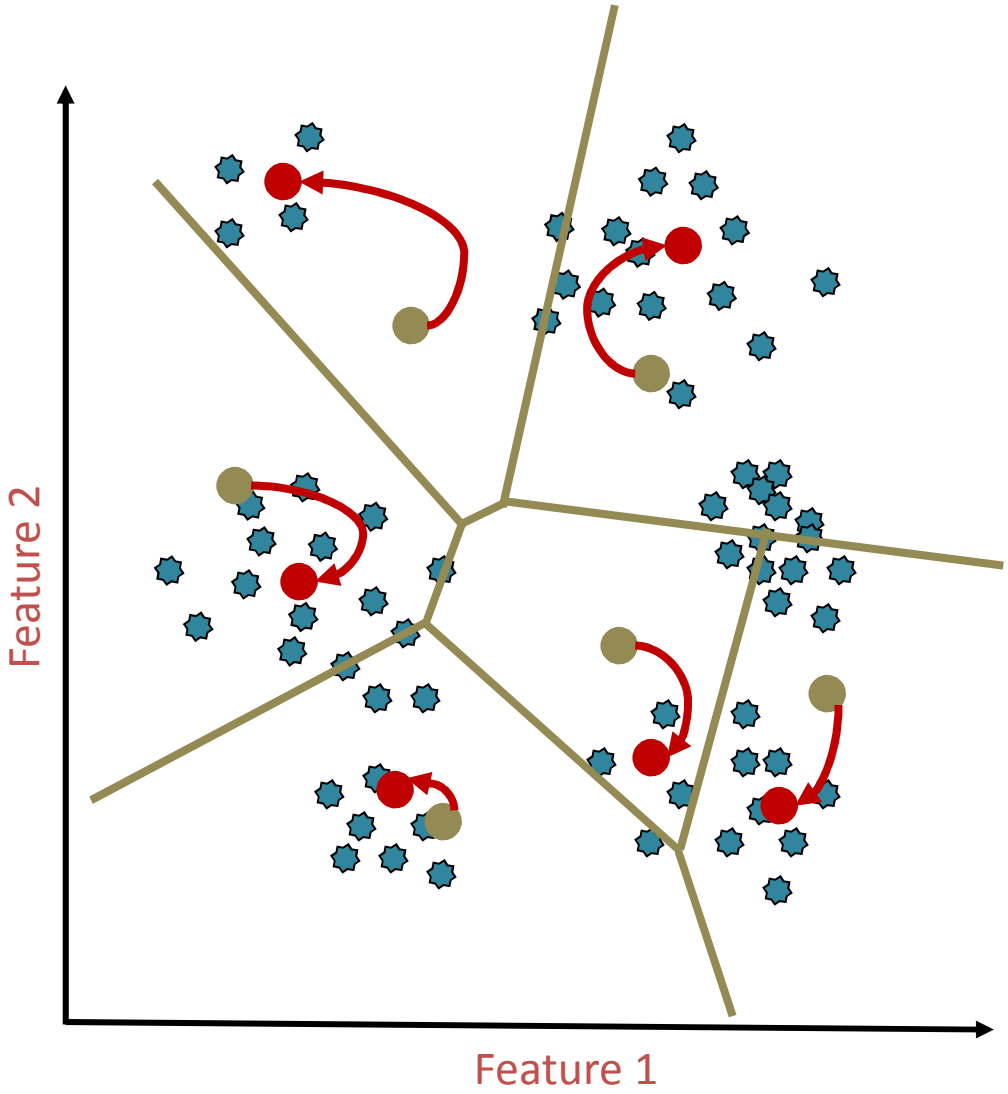
# Clustering and k-means

1. Start with a matrix with m rows (samples) and n columns (features)

2. Assign a (random) number of *k* clusters
   - Here: $k = 6$

3. Assign each sample to the cluster that it is closer to.

4. Recalculate cluster center.



# Clustering and k-means

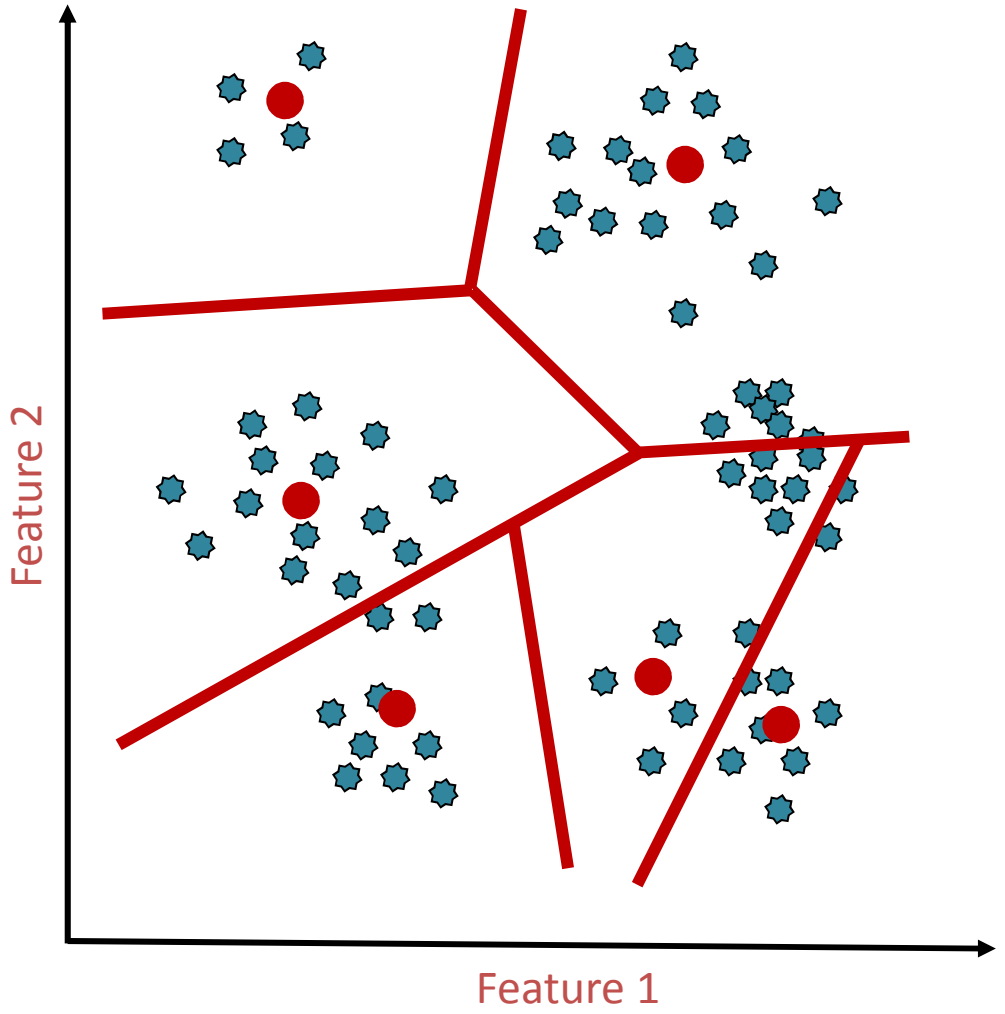
1. Start with a matrix with m rows (samples) and n columns (features)

2. Assign a (random) number of *k* clusters
   - Here: $k = 6$

3. Assign each sample to the cluster that it is closer to.

4. Recalculate cluster center.

5. Repeat from step 3 until convergence



# Clustering and k-means

1. Initialize ***K*** centers
2. Assign each $\mathbf{x}_i$ to the cluster with the nearest center, where the distance between $\mathbf{x}_i$ and cluster $k$ is

$$\left(\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_j\right)^2$$

with $\boldsymbol{\mu}_j$ being the center of cluster $k$

3. Recalculate the center $\boldsymbol{\mu}_j$ of each cluster to be the centroid of its members:

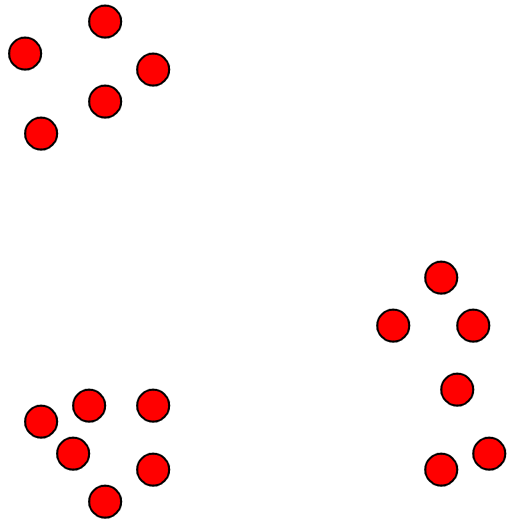
$$\boldsymbol{\mu}_j = \frac{1}{|\boldsymbol{c}_j|} \sum_{\boldsymbol{x}^{(i)} \in \boldsymbol{c}_j} \boldsymbol{x}^{(i)}$$

Where $|c_j|$ is the number of members (samples) that belong to cluster $c_j$.

4. Repeat from step 2, until convergence (e.g. no change in cluster membership).

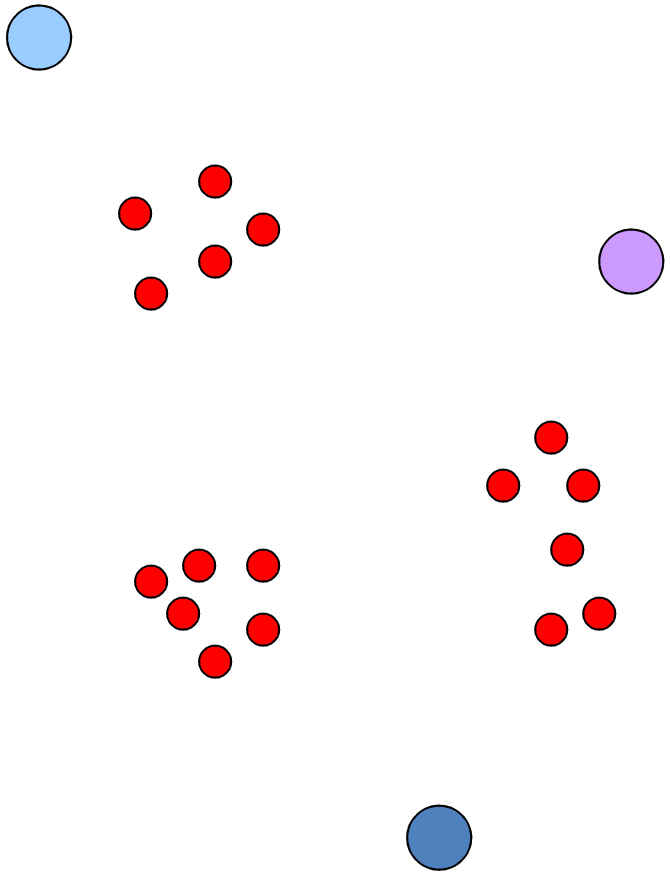
# Another example:

- Assume a fixed number of clusters, K
- Goal: create “compact” clusters (min intra-cluster distance)



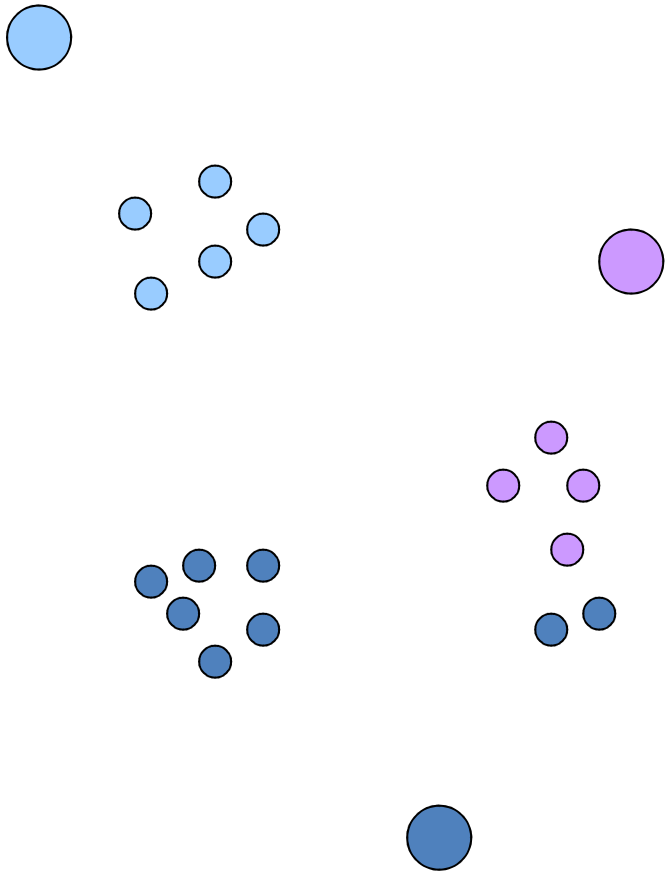
# Another example:

- Randomly initialize clusters



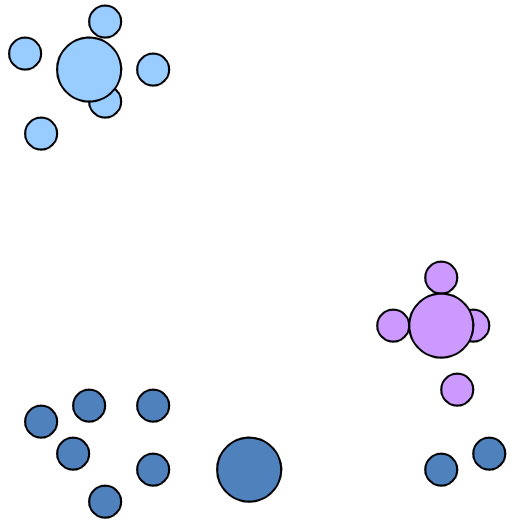
# Another example:

- Randomly initialize clusters
- Assign data points to nearest clusters



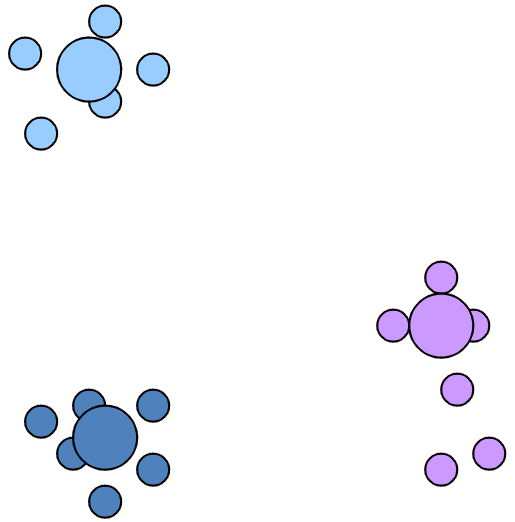
# Another example:

- Randomly initialize clusters
- Assign data points to nearest clusters
- Recalculate cluster centers



# Another example:

- Randomly initialize clusters
- Assign data points to nearest clusters
- Recalculate cluster centers
- Repeat last two steps

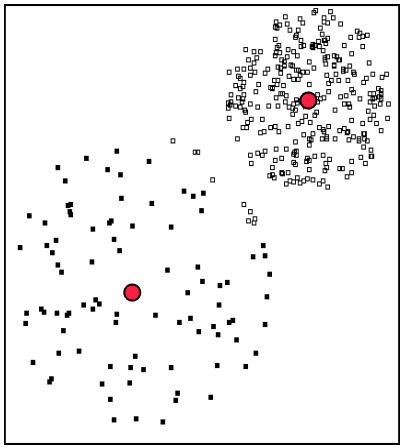


# Fuzzy k-means

- Initialize K clusters

- For each point calculate the probability of membership for each category

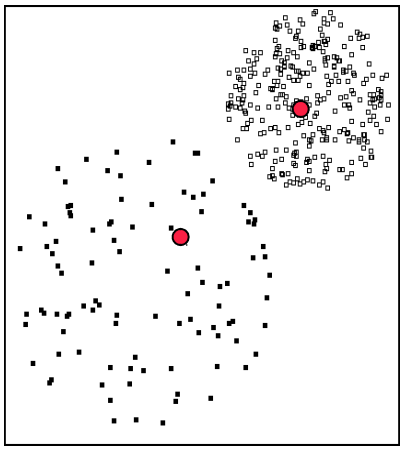
$$P(x_i \in cluster\ k | x_i, \mu_\kappa)$$



K-means

- Move the position of the center of each cluster to the weighted center:

$$\mu_\kappa = \frac{\sum_{i=1}^{N} x_i \cdot P(x_i \in cluster\ k | x_i, \mu_\kappa)^\beta}{\sum_{i=1}^{N} P(x_i \in cluster\ k | x_i, \mu_\kappa)^\beta}$$



Fuzzy K-means

- Iterate

$$P(x_i \in cluster\ k | x_i, \mu_\kappa) = \begin{cases} 1\ if\ k^{th}\ cluster\ closest\ to\ i^{th}\ sample \\ 0 \qquad\qquad\qquad\qquad otherwise \end{cases}$$

## Clustering and k-means

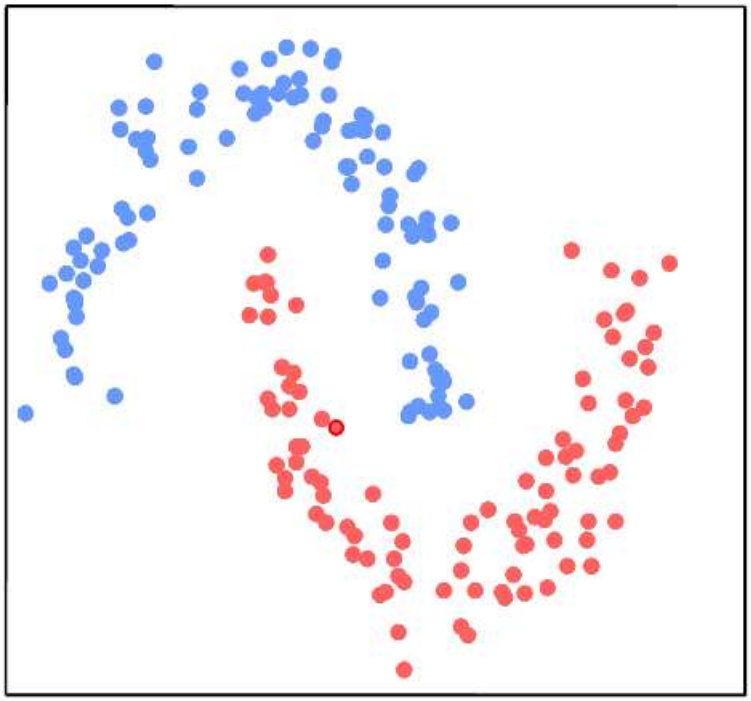
- **Convergence**: How to know when to stop ? Is convergence guarranteed ?
- **Assumptions:** What is the structure of the data? Are all Gaussians equal ?
- **Overfitting**: what is the optimum number of clusters?
- **Similarity metric:** what similarity metric should we used ?

And as with any clustering/classification algorithm, we should be concerned about **irrelevant features**, **curse of dimensionality**, **small sample size**, etc.

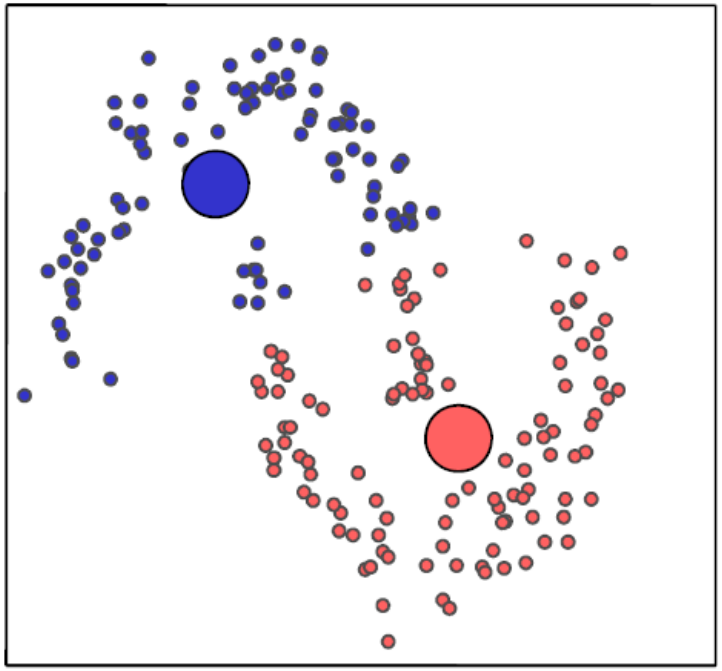
# Convergence & Structure

- **K-means is a Greedy algorithm**
  - *"Greed, for lack of a better word, is good. Greed is right. Greed works. Greed clarifies, cuts through, and captures, the essence of the evolutionary spirit." (Gordon Gekko, Wall Street, 1987)*
- **Convergence is achieved when**
  - Inter-cluster distance is lower than a threshold
  - Number of iterations have reached a certain value
  - A wall-time (cpu cycles, time interval) has been reached
- **K-means is NP-hard** (special case: K,D fixed)
  - Generally the performance of k-means is arbitrary, we do not know how close to the optimal solution we are
- **K-means can be used with various distance metrics**
  - If Euclidean distance is used (as in slides), Gaussian (circular) distribution of data around the cluster center is assumed
  - Poor results when the data have a non-Gaussian structure.

# Convergence & Structure



Actual



K-means

# The perfect cluster

"There is no single best criterion for obtaining a partition because no precise and workable definition of 'cluster' exists. Clusters can be of any arbitrary shapes and sizes in a multidimensional pattern space. Each clustering criterion imposes a certain structure on the data, and if the data happen to conform to the requirements of a particular criterion, the true clusters are recovered."

Jain, A.K. & Dubes, R.C. *Algorithms for Clustering Data.* (Prentice Hall, Englewood Cliffs, New Jersey, 1988).

From: *How does gene expression clustering work by Dhaeseleer, Nature Reviews (2005)*

# Number of Clusters

- How many clusters do we need?
  - **Minimum description length (MDL)**
  - **Bayesian information criterion (BIC)**

$$\mathbf{BIC} = -2 \cdot \ln \mathbf{p}(\mathbf{x}|\mathbf{k}) = -2 \cdot \ln \mathbf{L} + \mathbf{k} \cdot \ln \mathbf{n}$$

where x the observed data, k the parameters of the model, L the max likelihood (fit), n the number of points. The lower the value of BIC, the better.

  - **Akaike's information criterion (AIC)**

$$\mathbf{AIC} = 2\mathbf{k} - 2 \cdot \ln \mathbf{L}$$

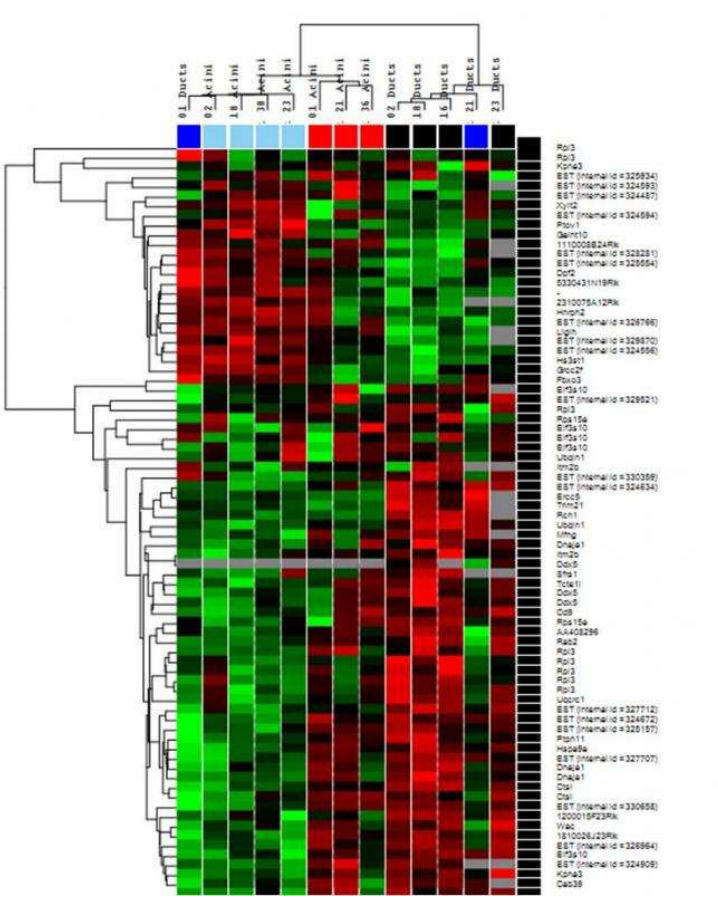
  - **Hierarchical clustering**

# Hierarchical Clustering

- **Goal**: build a hierarchy of clusters based on similarity distance

- Types:
  - **Agglomerative**: Each point/observation is a cluster; merge clusters based on similarity; stop when only one cluster is left. **Bottom-up** approach.
  - **Divisive**: All points/observations are in ONE cluster; split recursively clusters until all points are a cluster themselves. **Top-down** approach.

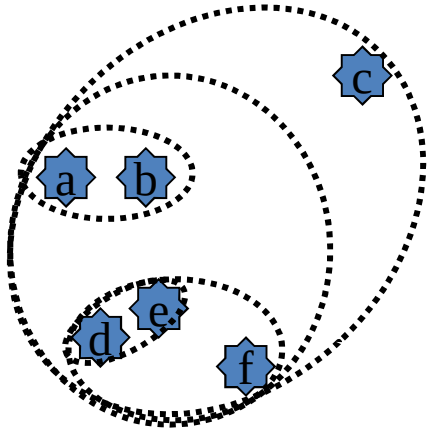
# Hierarchical Clustering

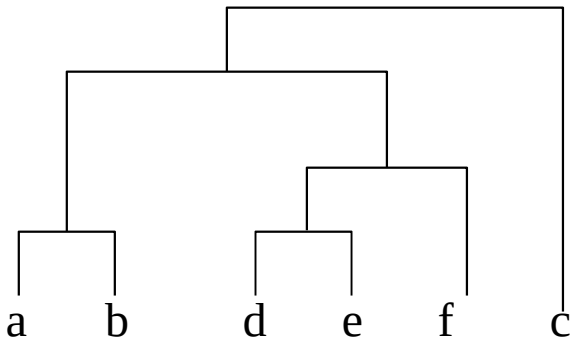


Bottom-up approach (agglomerative):

- Start with each point in a separate cluster
- At each iteration:
  - Choose the pair of closest clusters
  - Merge the pair of clusters to one.

# Hierarchical Clustering




Bottom-up approach:

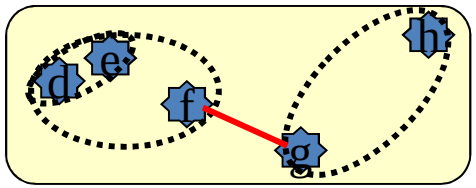
- Start with each point in a separate cluster
- At each iteration:
  - Choose the pair of closest clusters
  - Merge the pair of clusters to one.

UPGMA (Unweighted Pair Group Method with Arithmetic mean)

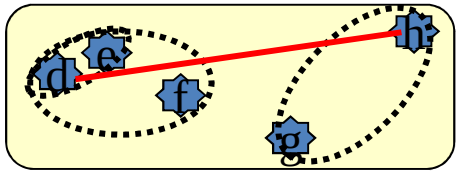
# Distance metric

We may define as “distance” the:

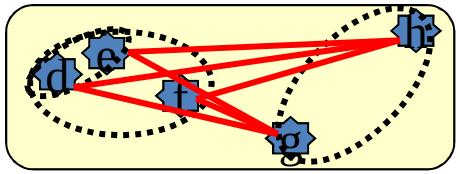
• Minimum distance between elements of each cluster (single-linkage clustering)



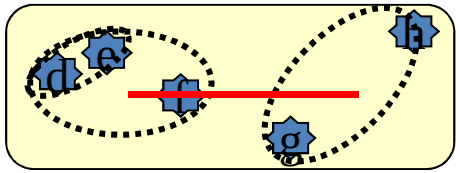

•Maximum distance between elements of each cluster (complete-linkage clustering)




•Mean distance between all elements in each cluster (average-linkage clustering)




•Distance between the centroids of each cluster (centroid-linkage clustering)



# Similarity Measures

**Table 1 Gene expression similarity measures**

| | |
|---|---|
| Manhattan distance (city-block distance, L1 norm) | $d_{fg} = \sum_c \lvert e_{fc} - e_{gc} \rvert$ |
| Euclidean distance (L2 norm) | $d_{fg} = \sqrt{\sum_c (e_{fc} - e_{gc})^2}$ |
| Mahalanobis distance | $d_{fg} = (e_f - e_g)' \Sigma^{-1} (e_f - e_g)$, where $\Sigma$ is the (full or within-cluster) covariance matrix of the data |
| Pearson correlation (centered correlation) | $d_{fg} = 1 - r_{fg}$, with $r_{fg} = \frac{\sum_c (e_{fc} - \bar{e}_f)(e_{gc} - \bar{e}_g)}{\sqrt{\sum_c (e_{fc} - \bar{e}_f)^2 \sum_c (e_{gc} - \bar{e}_g)^2}}$ |
| Uncentered correlation (angular separation, cosine angle) | $d_{fg} = 1 - r_{fg}$, with $r_{fg} = \frac{\sum_c e_{fc} e_{gc}}{\sqrt{\sum_c e_{fc}^2 \sum_c e_{gc}^2}}$ |
| Spellman rank correlation | As Pearson correlation, but replace $e_{gc}$ with the rank of $e_{gc}$ within the expression values of gene $g$ across all conditions $c = 1 \ldots C$ |
| Absolute or squared correlation | $d_{fg} = 1 - \lvert r_{fg} \rvert$ or $d_{fg} = 1 - r_{fg}^2$ |

$d_{fg}$, distance between expression patterns for genes $f$ and $g$. $e_{gc}$, expression level of gene $g$ under condition $c$.

D'haeseleer (2005) Nat Biotech

How to evaluate clusters? Robustness, enrichment, P-value etc

# Metrics

## What is a metric?

Function D with nonnegative real values, defined on Cartesian product V x V of a set V is a metric iff:

$\forall$ x,y,z $\in$ V:

D(x,y) = 0 iff x =y (identity axiom)

D(x,y) + D(y,z) $\geq$ D(x,z) (triangle inequality)

D(x,y) = D(y,x) (symmetry axiom)

Do not confuse with a similarity measure (that doesn't have to meet any constraints)

# Metrics

**$L^p$ norm:**

$$\|x\|_p = \left(|x_1|^p + |x_2|^p + \cdots + |x_n|^p\right)^{1/p}$$

$L^1$ – Manhattan distance / Taxicab geometry

$L^2$ – Euclidian norm

$L^\infty$ – Uniform norm; Chebyshev chessboard distance

$max\{|x_i|\}$

# Pearson correlation

- Measure of the correlation (linear dependence) between two variables
- Formula:

$$\rho_{X,Y} = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y} = \frac{E[(X-\mu_X)(Y-\mu_Y)]}{\sigma_X \sigma_Y},$$

- Or

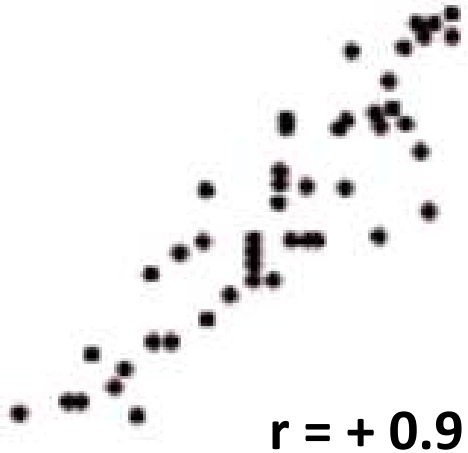
$$r = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}}.$$

- Or

$$r = \frac{1}{n-1}\sum_{i=1}^{n}\left(\frac{X_i - \bar{X}}{s_X}\right)\left(\frac{Y_i - \bar{Y}}{s_Y}\right)$$

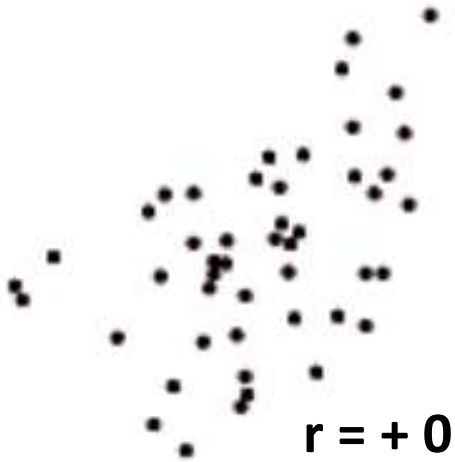
- Values -1 to +1

# Pearson correlation



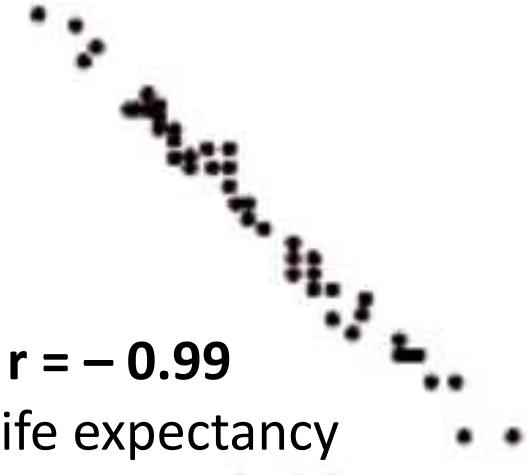
**r = + 0.9**

e.g: lung cancer probability
*vs.* smoking level



**r = + 0.5**

e.g: prostate cancer probability
*vs.* men's age



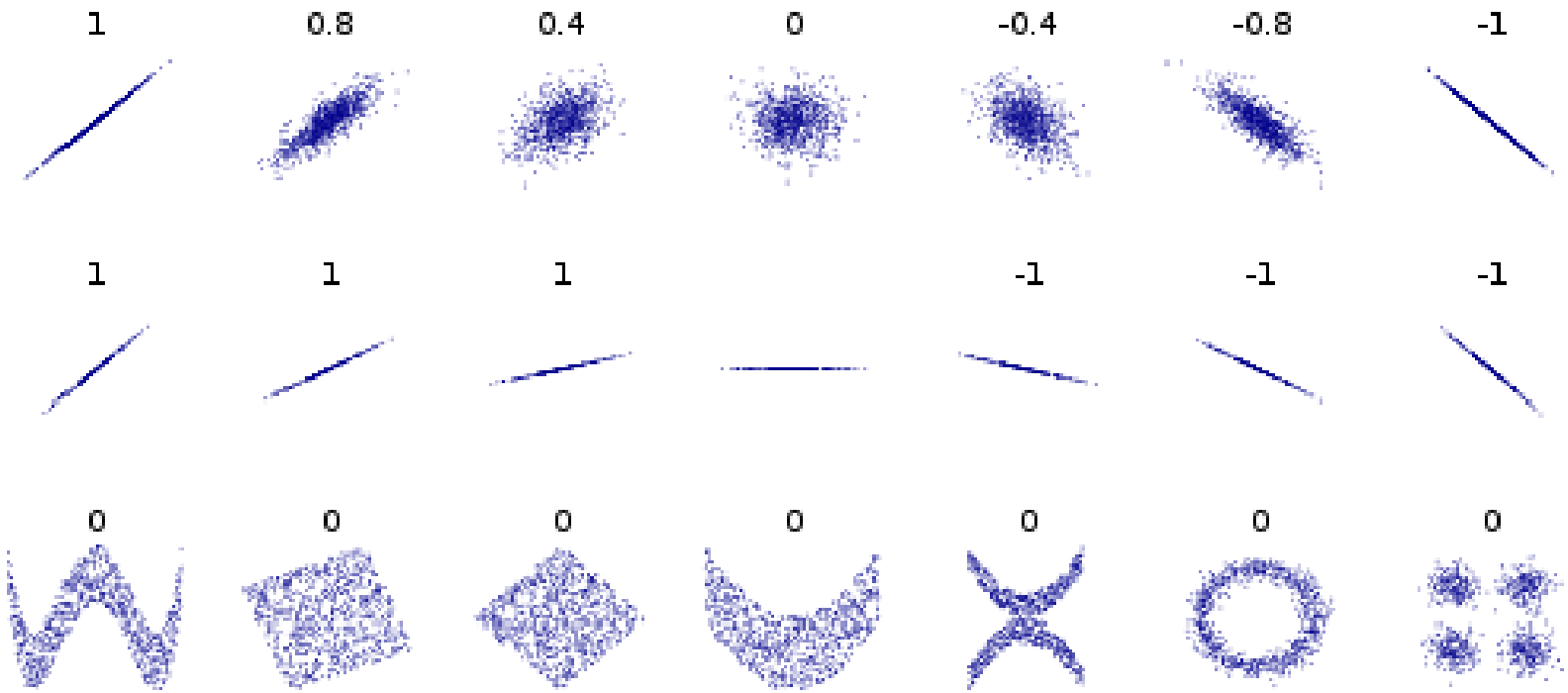
**r = – 0.99**

e.g: life expectancy
*vs.* pancreatic cancer stage

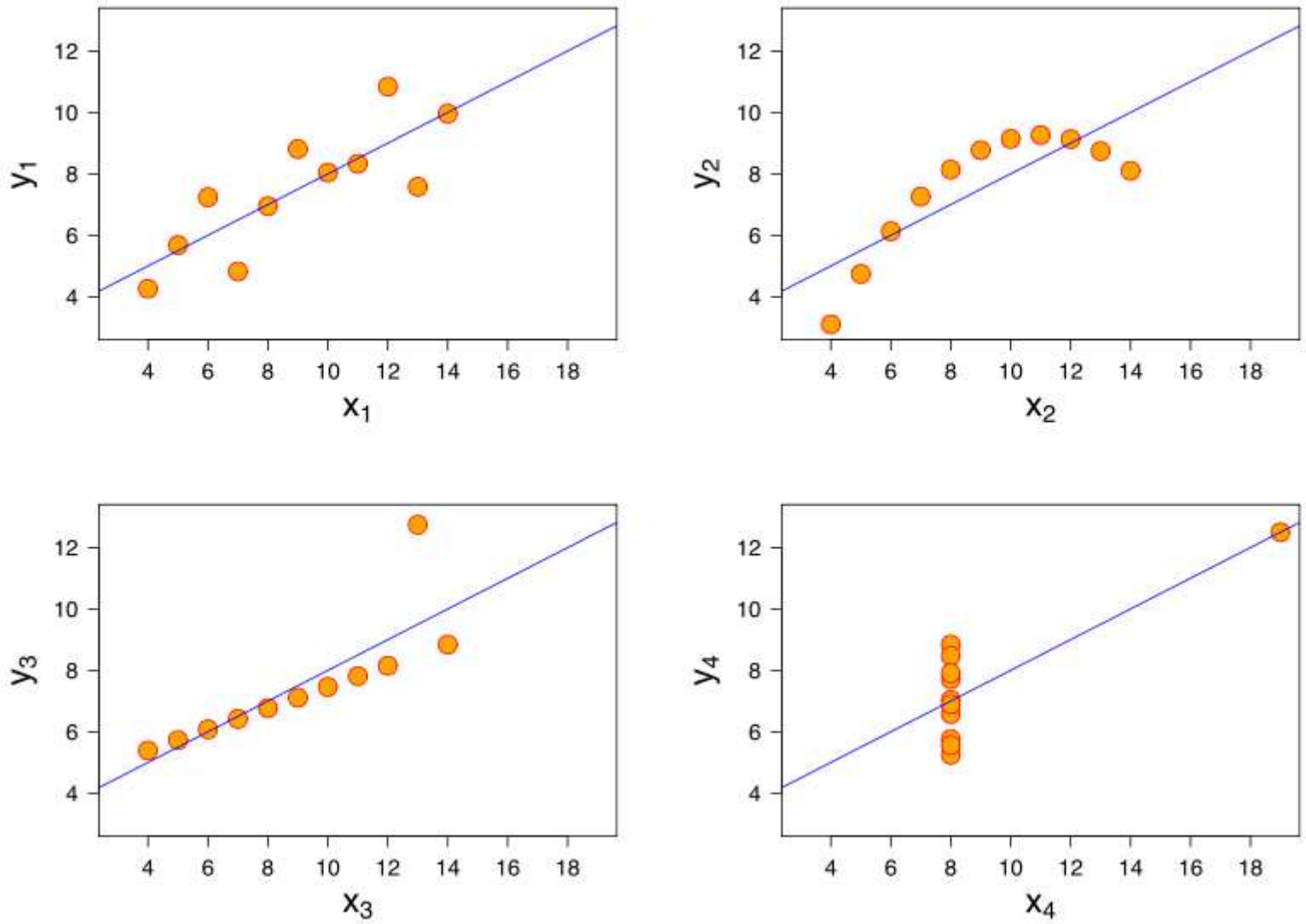
**r = 0.0**

e.g: lung cancer stage
*vs.* eyes color

# Pearson correlation



# Pearson correlation



**End of Lecture 11**