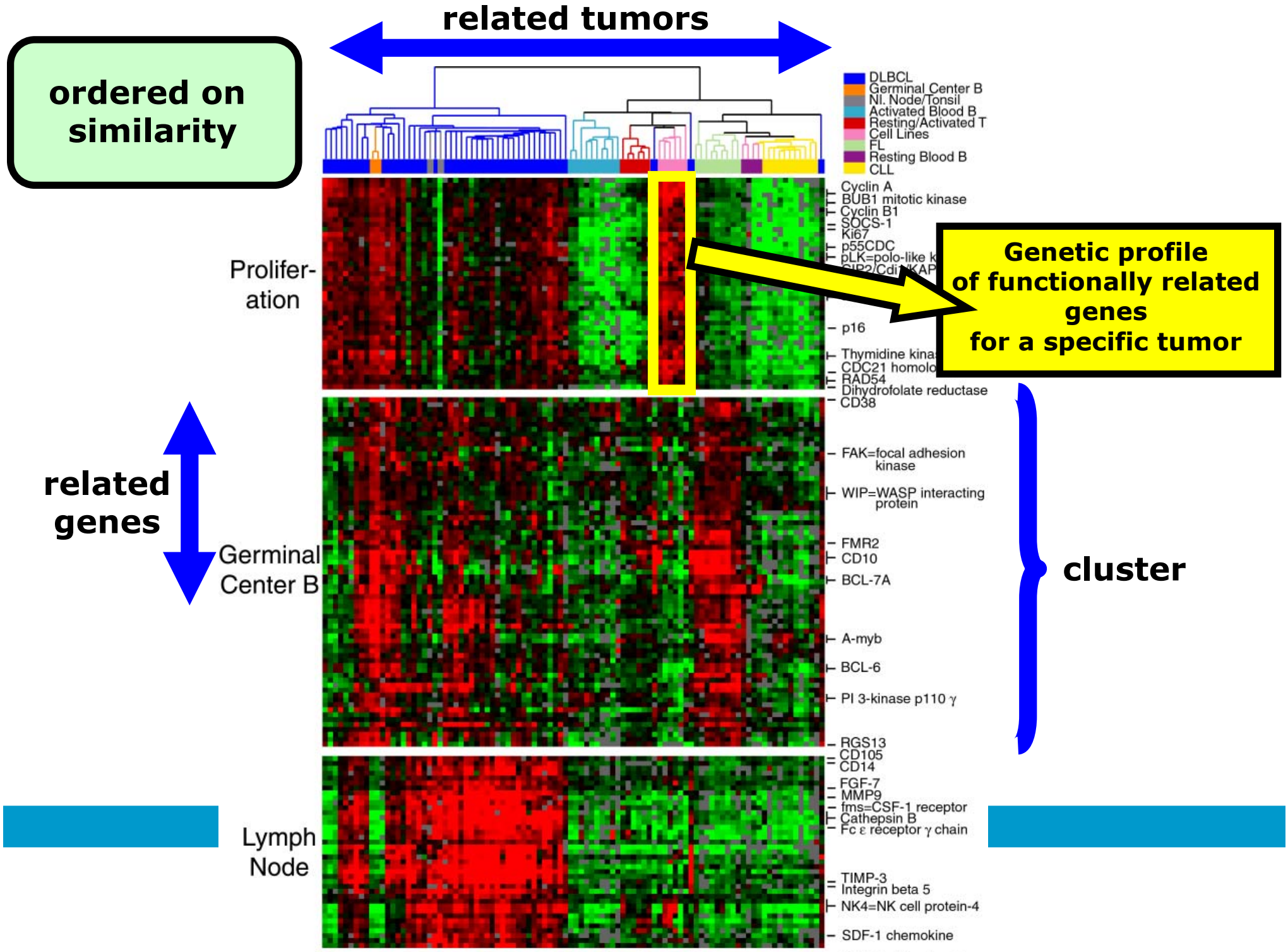
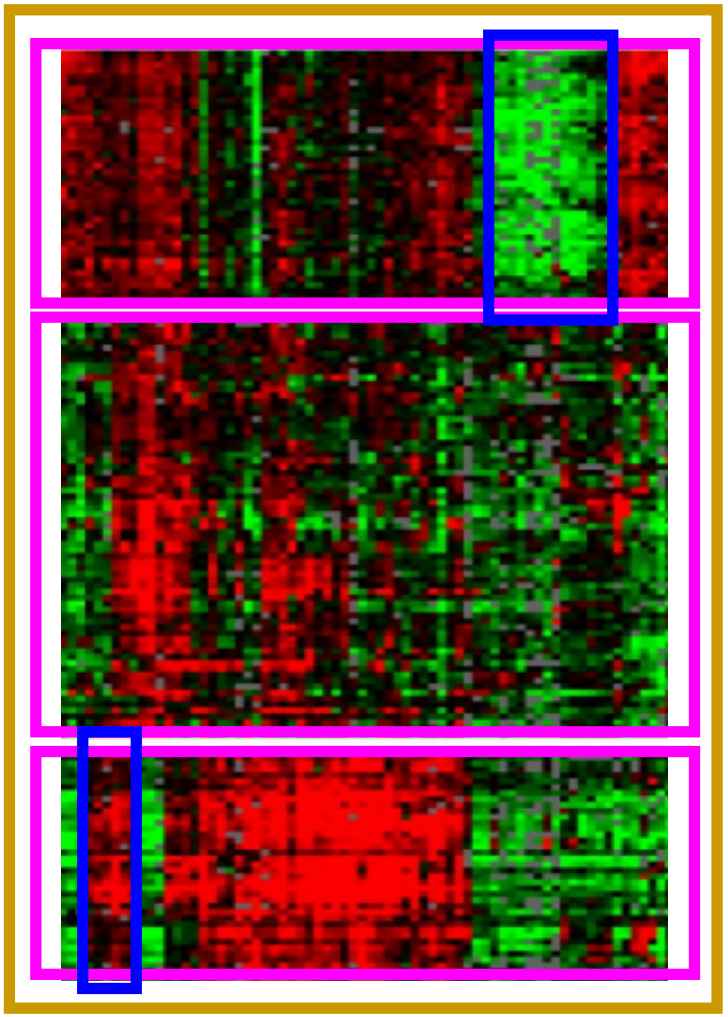
Clustering

Marcel Reinders

TU Delft
Delft Bioinformatics Lab
Faculty of Electrical Engineering, Computer Science and Mathematics
Delft University of Technology

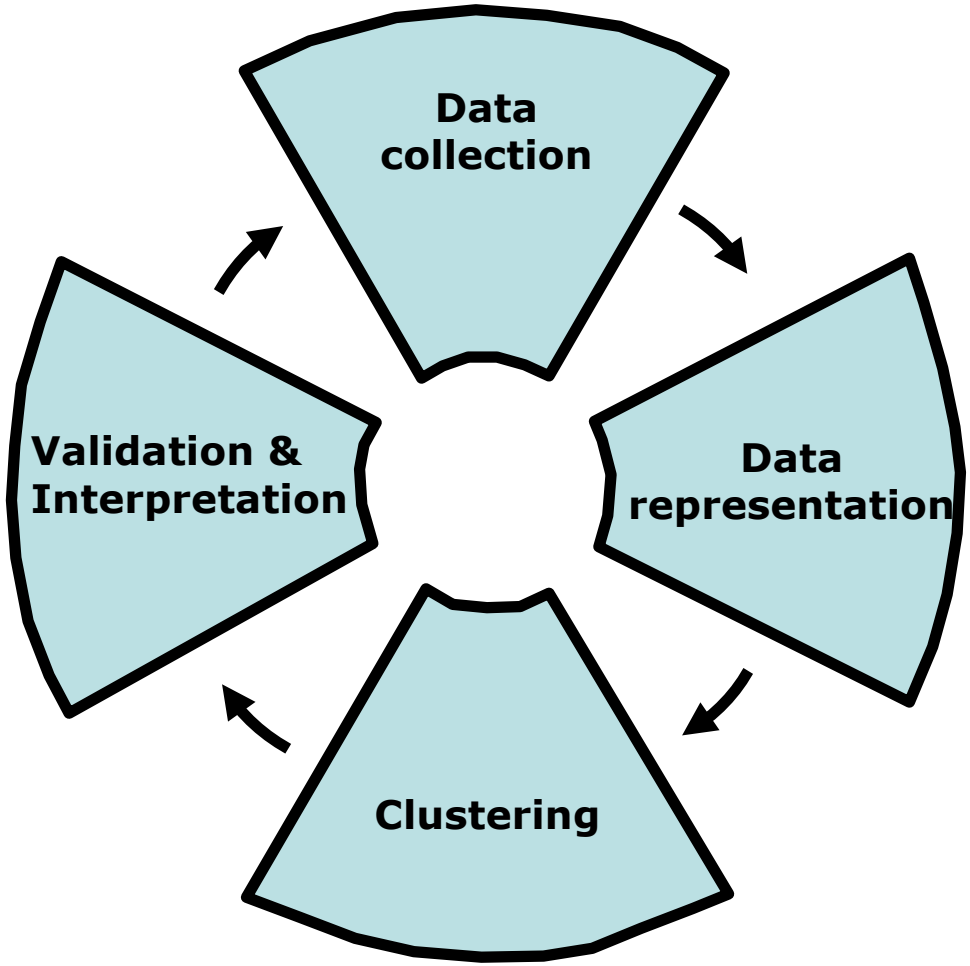

related tumors
ordered on similarity
DLBCL
Germinal Center B
NI. Node/Tonsil
Activated Blood B
Resting/Activated T
Cell Lines
FL
Resting Blood B
CLL
Cyclin A
BUB1 mitotic kinase
Cyclin B1
SOCS-1
Ki67
p55CDC
p16
CDC21 homolo
RAD54
Dihydrofolate reductase
CD38
Prolifer-ation
Genetic profile of functionally related genes for a specific tumor
related genes
FAK=focal adhesion kinase
WIP=WASP interacting protein
FMR2
CD10
BCL-7A
A-myb
BCL-6
PI 3-kinase p110 γ
Germinal Center B
cluster
RGS13
CD105
CD14
FGF-7
MMP9
fms=CSF-1 receptor
Cathepsin B
Fc ε receptor γ chain
TIMP-3
Integrin beta 5
NK4=NK cell protein-4
SDF-1 chemokine
Lymph Node

Cluster analysis

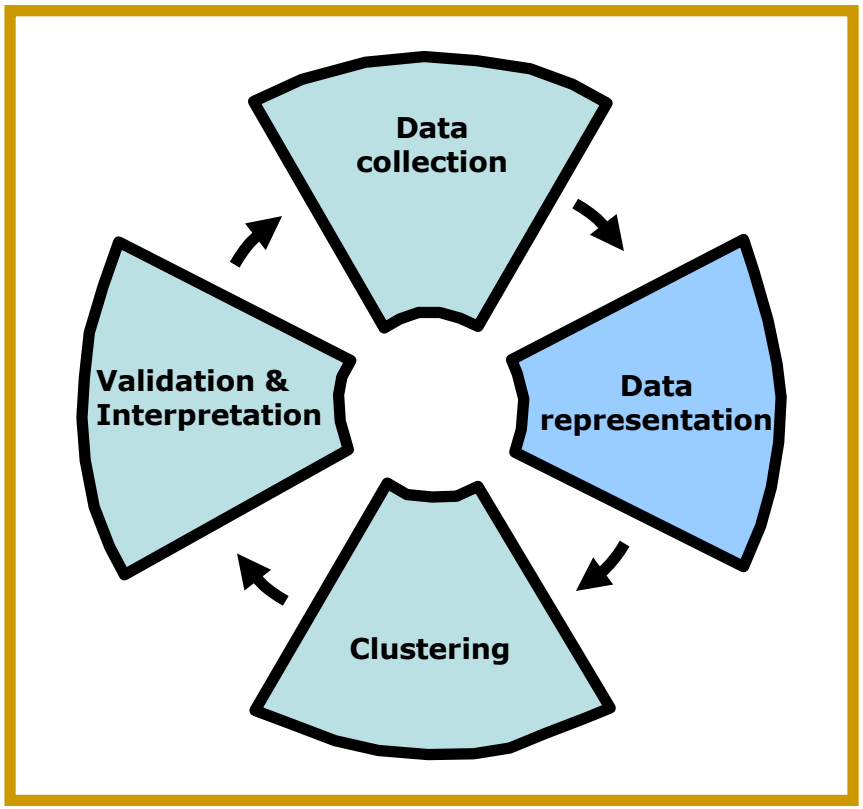


- **Group similar profiles**
- **Finding structure in the data**
- **Requisites:**
  - Measure of similarity
  - Grouping method
- **Subjective measures:**
  - Validation
  - Clustering is a process

Cluster analysis: The process



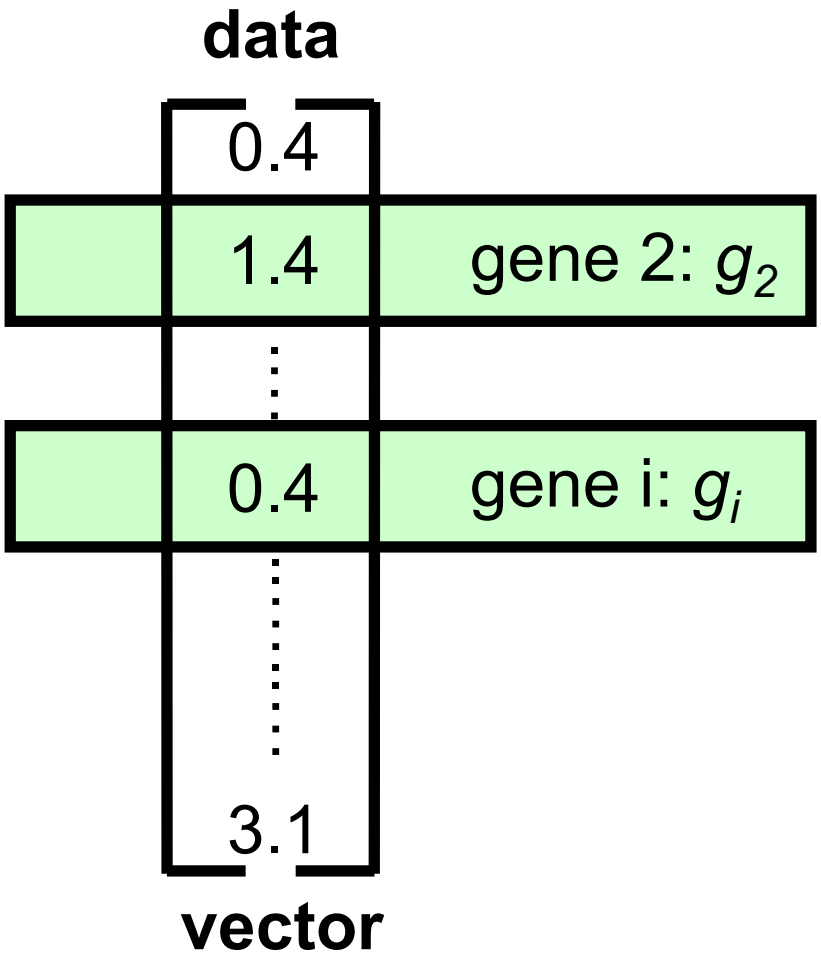
Data representation



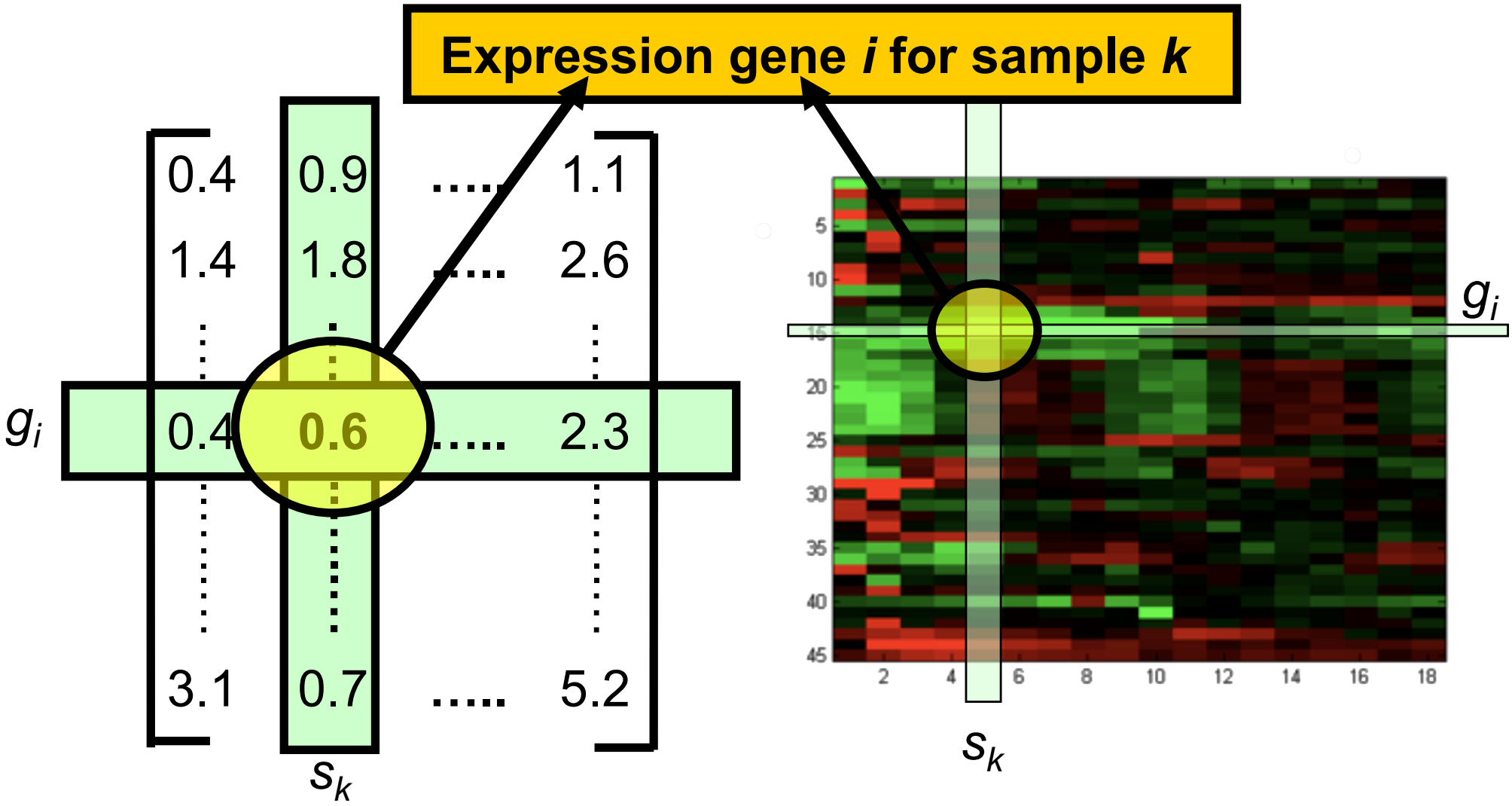
TOPICS

- **Data representation**
- **Data spaces:**
  - Sample-space
  - Gene-space
- **Clustering**

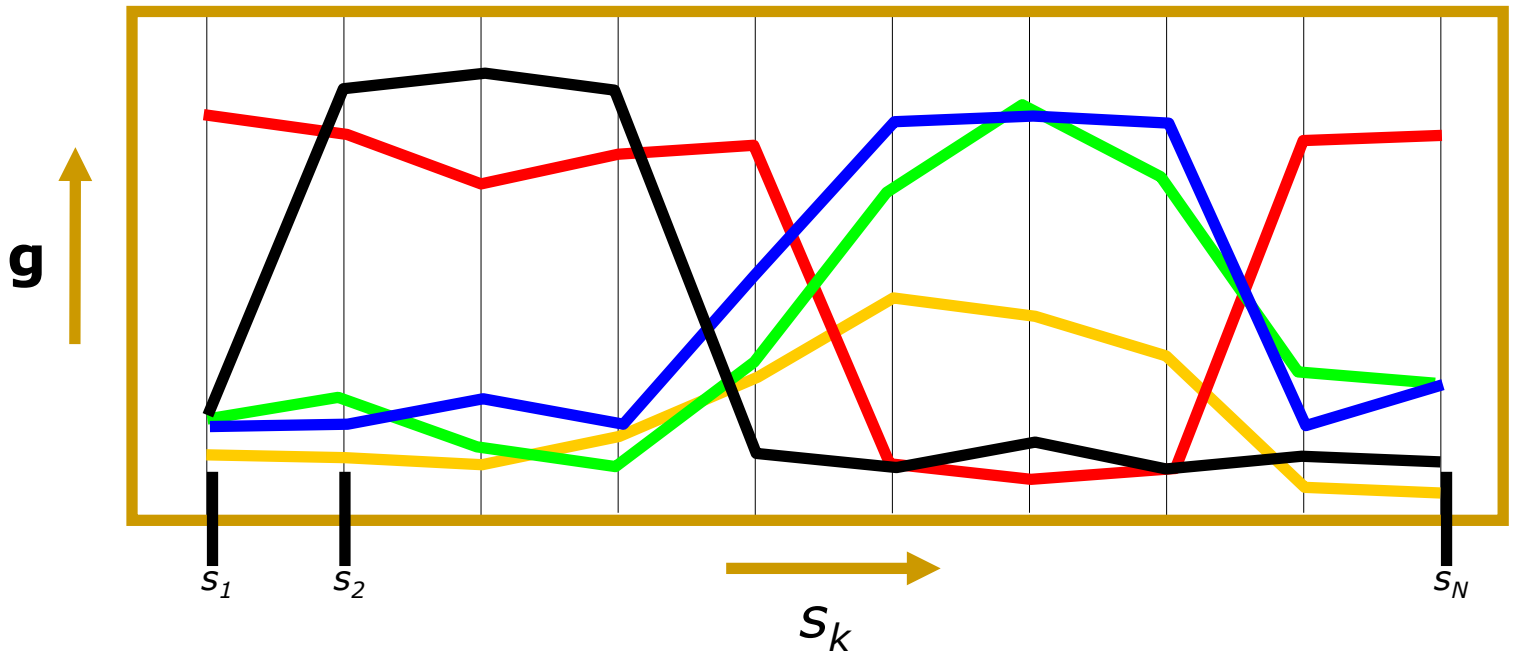
Data representation, one sample



Data representation, multiple samples



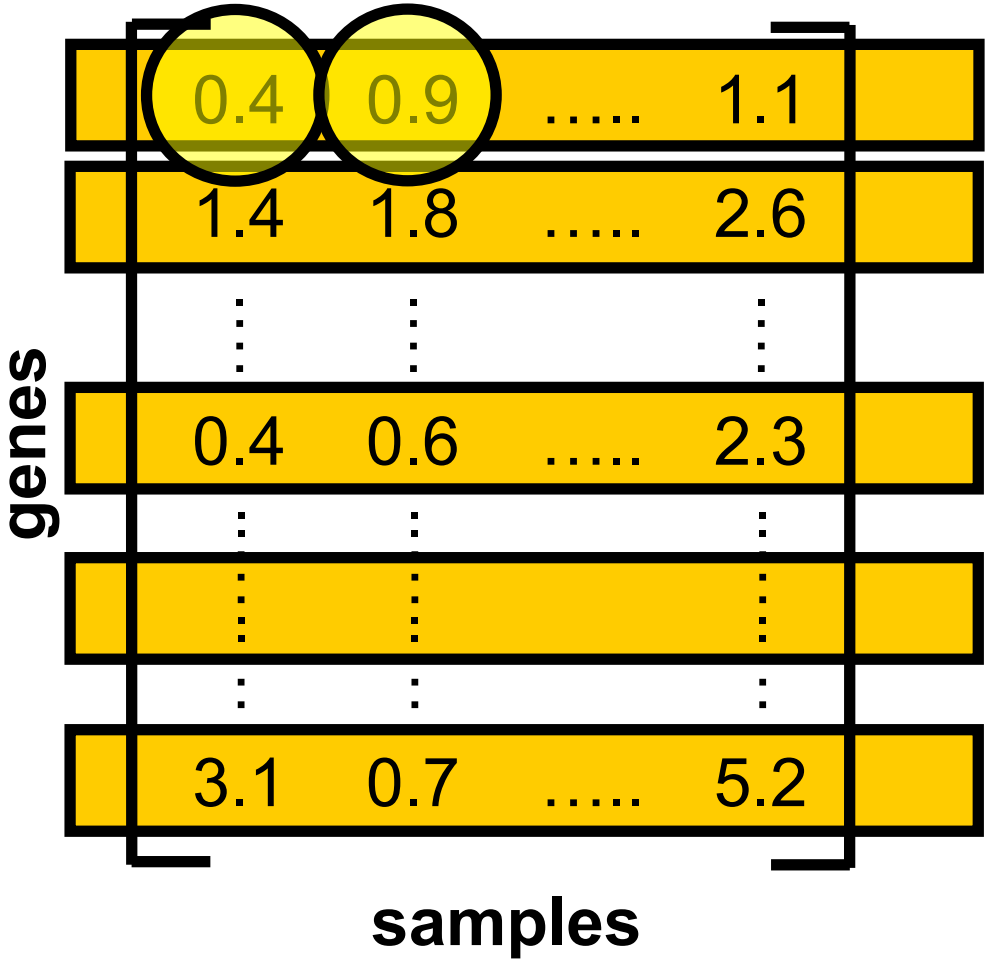
View as profiles

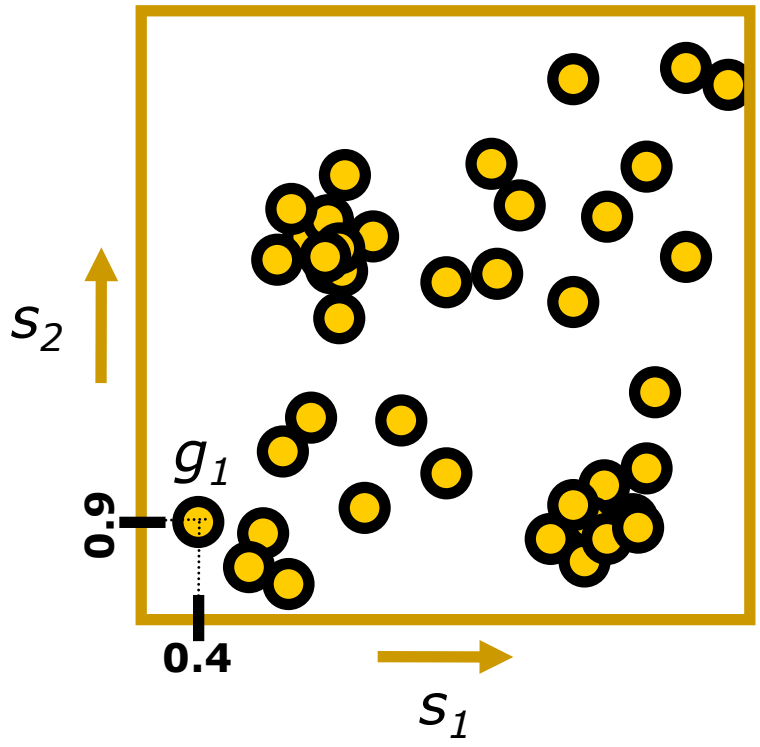


Profile for each gene

Represents gene activity across all samples

Sample-space (1/2)

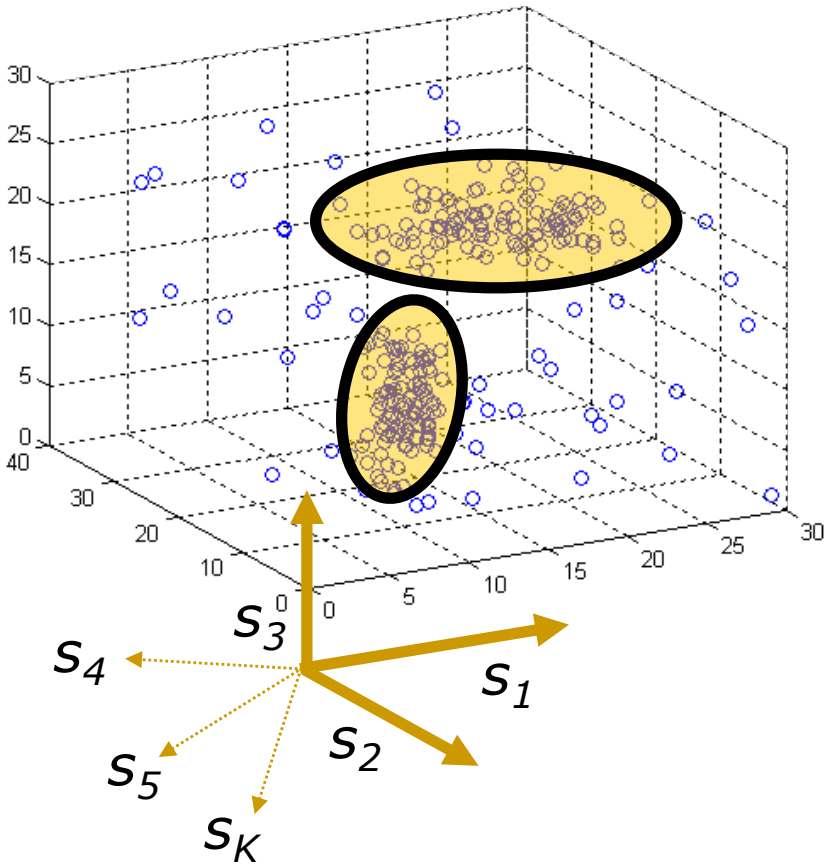




Sample-space

Genes are data points

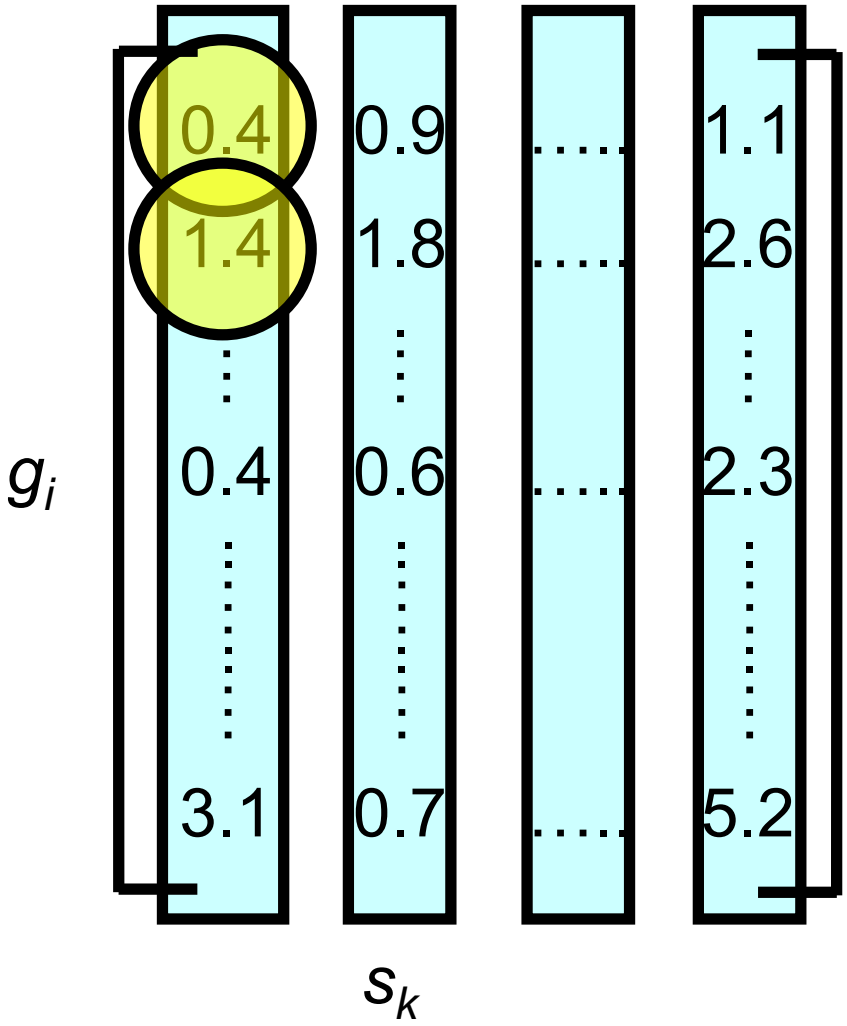
Sample-space (2/2)

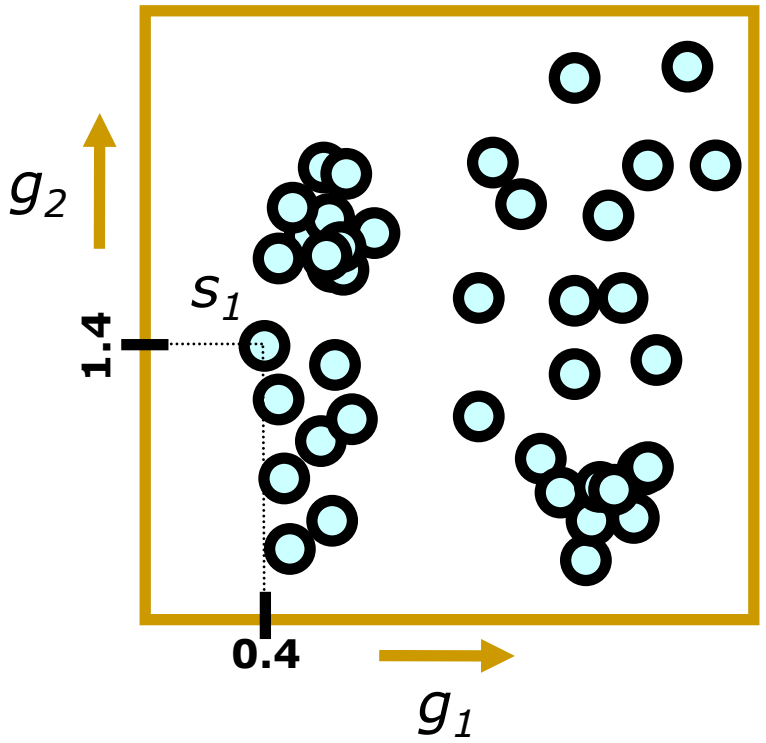


Note!

Sample-space is high dimensional

Gene-space

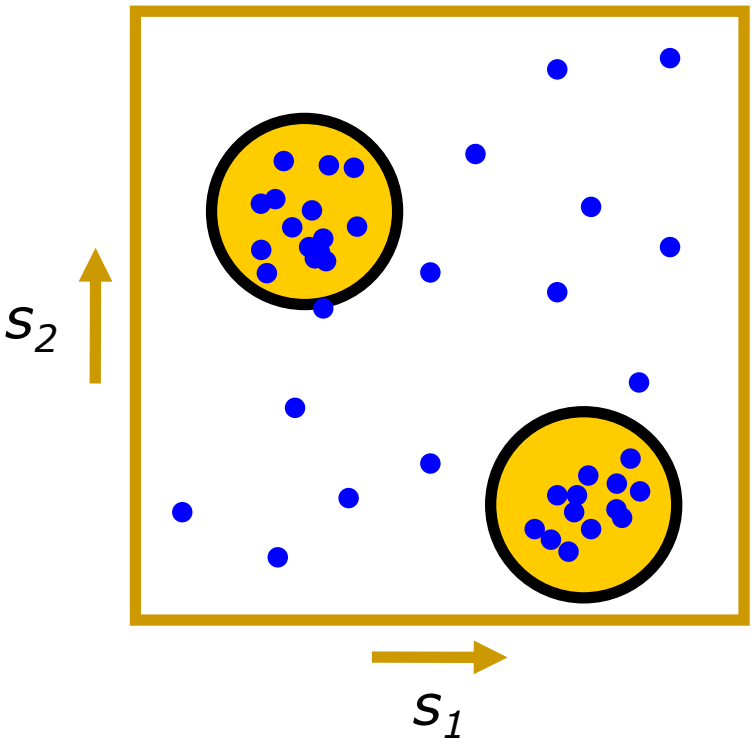


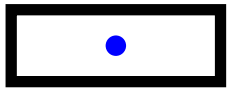


Gene-space

Samples are data points

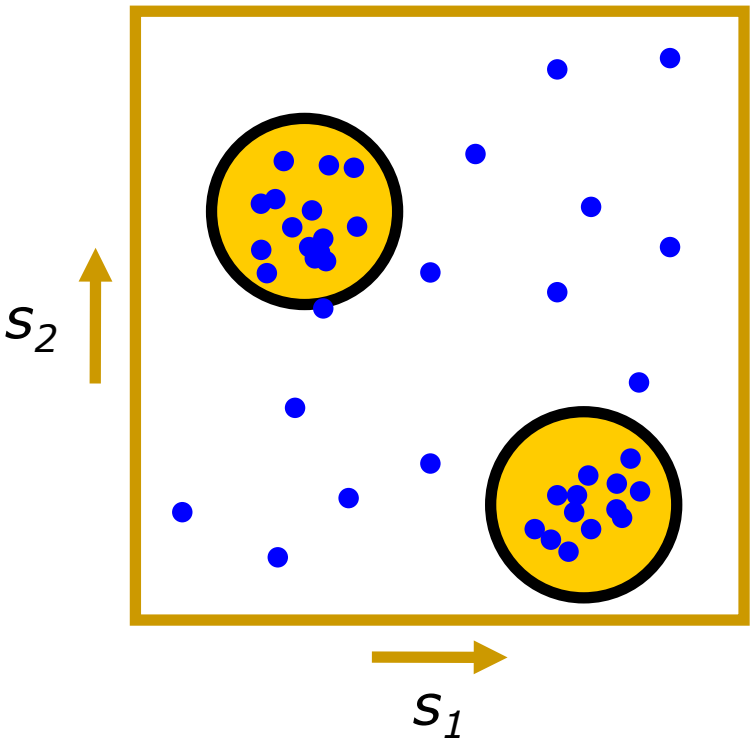
Genes in *sample-space*

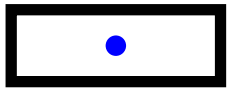




Gene g_i: activity level in Samples s_1 and s_2
(each point represents one gene)

Genes in *experiment-space*



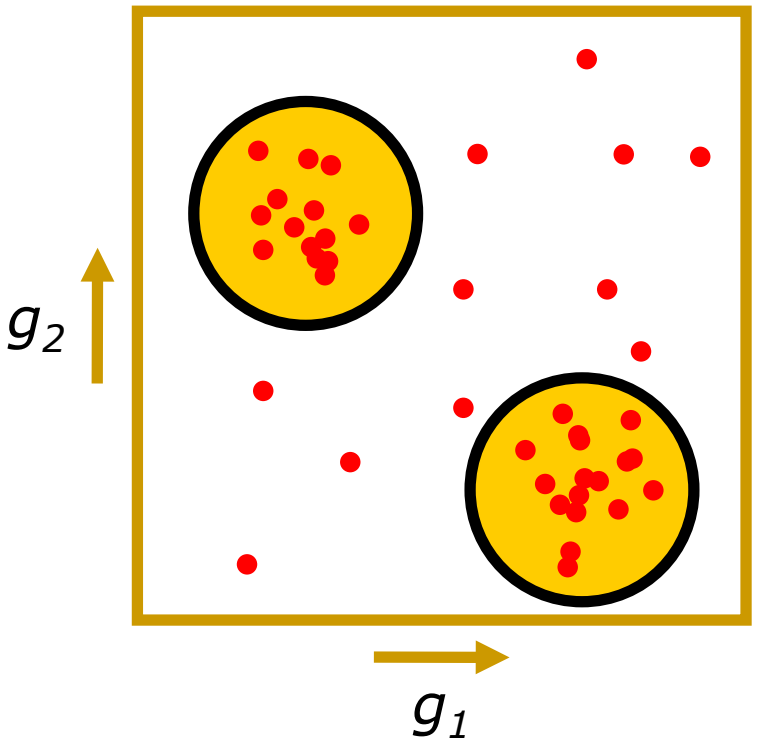


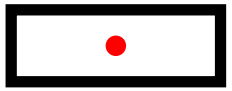
Gene g_i: activity level in samples s_1 and s_2
(each point represents one gene)

Group of genes that have the same activity levels across"all" samples

Cluster: Genes functionally related
Characterization of unknown genes
Hypothesis testing!

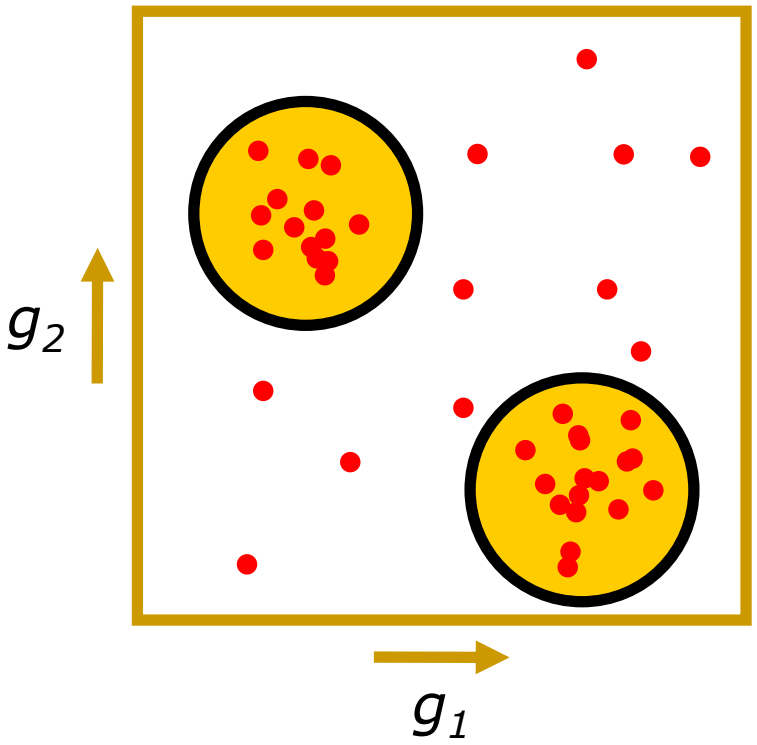
Samples in *gene-space*



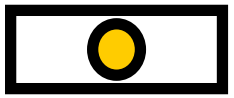


Sample s_k: activity of genes g_1 and g_2
(each point represents one sample)

Samples in *gene-space*



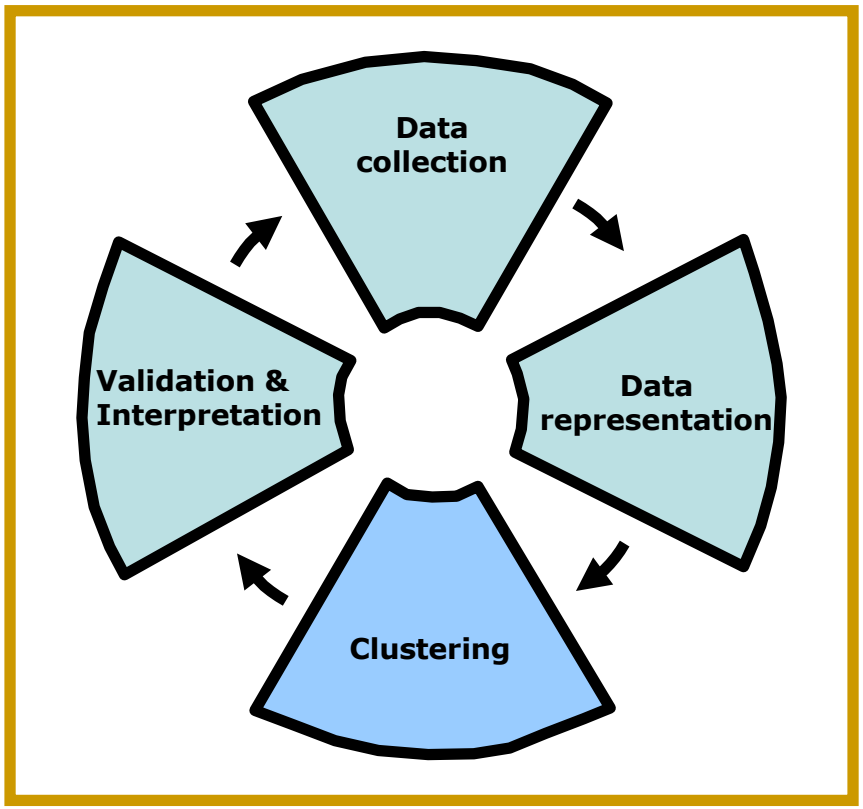
Sample s_k: activity of genes g_1 and g_2
(each point represents one sample)



Group of samples in which genes have the same activity levels across samples

Cluster: Genetic profile for related samples
Characterization of related diseases

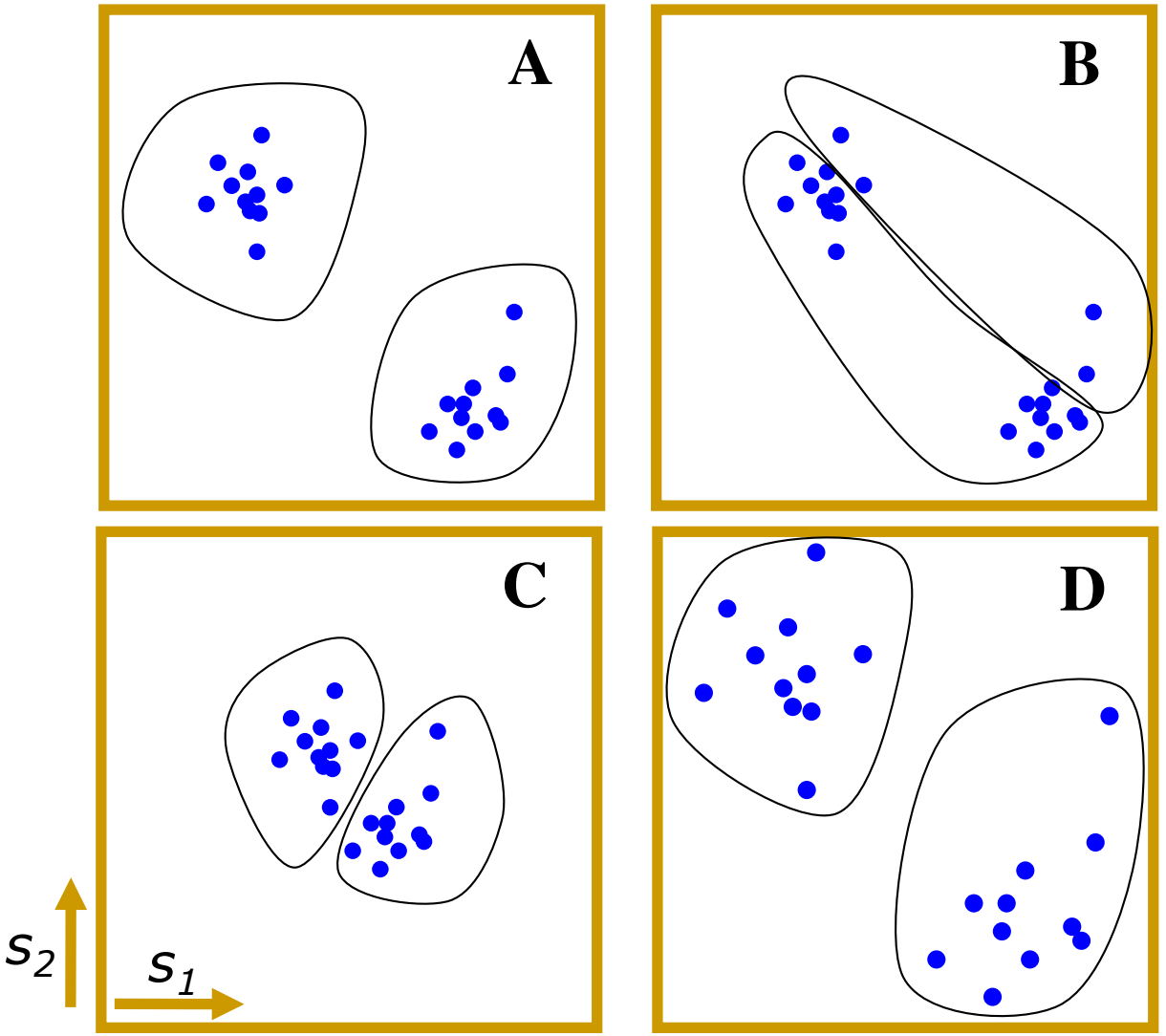
Clustering methods



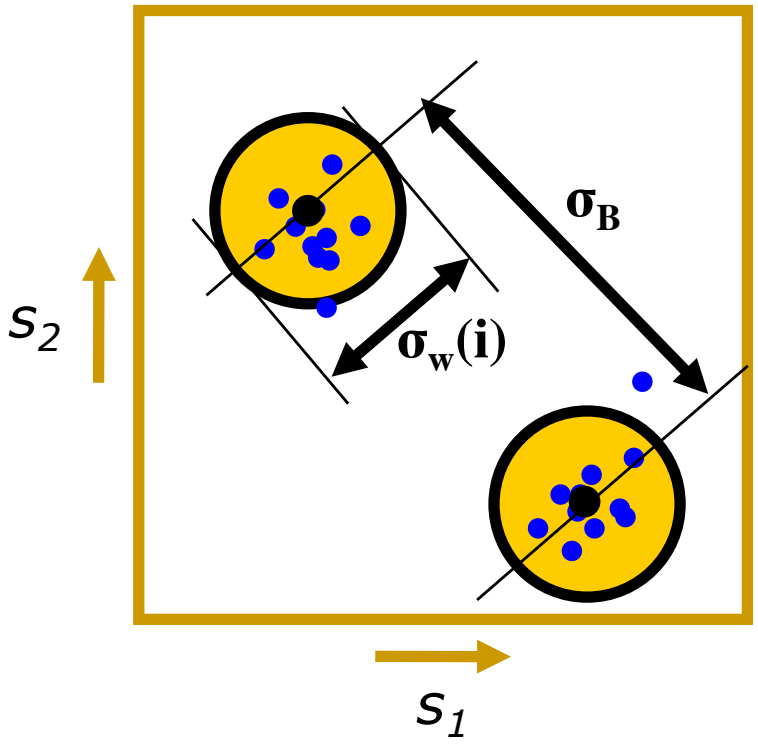
TOPICS

- Finding structure
- Hierarchical clustering
- Similarity & Linkage
- K-means clustering

Finding structure: better grouping?



Finding structure: better grouping?



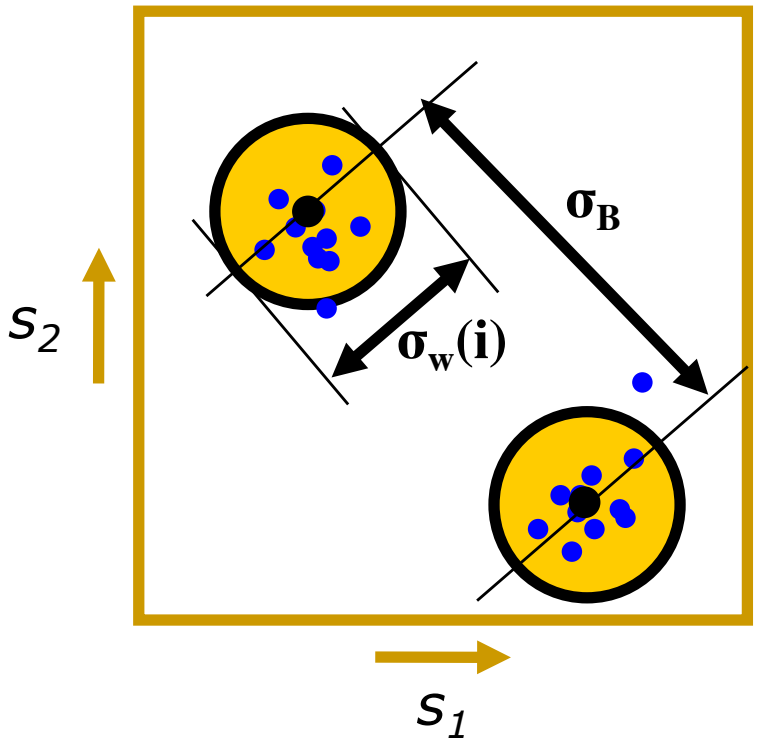
- **Structure when:**
  1) **Points within cluster resemble each other**

     (within variance, $\sigma_W(i)$)
  2) **Clusters deviate from each other *(between variance, σ_B)***

- **Group points such that**

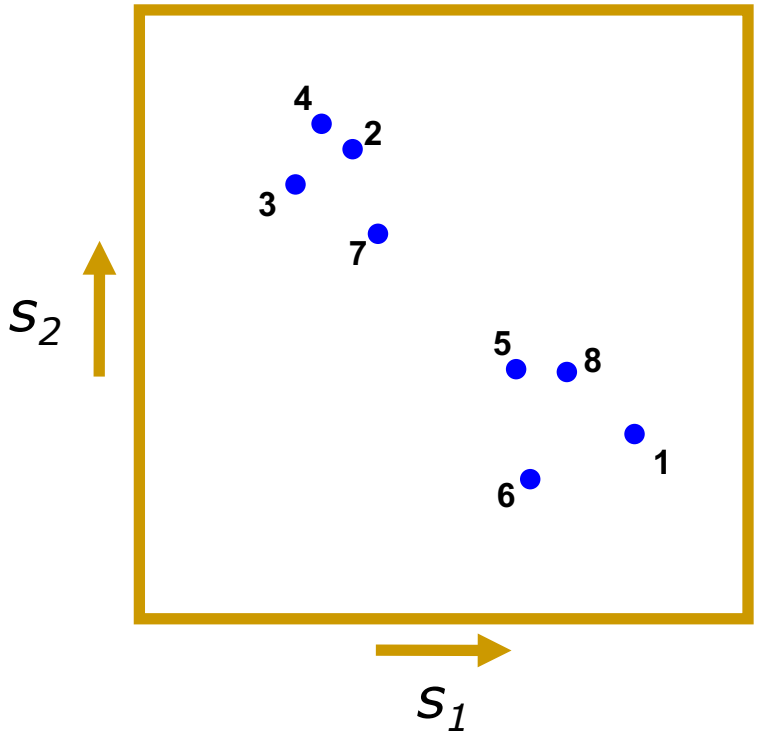
$$\textbf{MIN}\left[\frac{\sum \textbf{within variance}}{\textbf{between variance}}\right] \rightarrow \sigma_W\textbf{: small \&}\ \sigma_B\textbf{ : large}$$

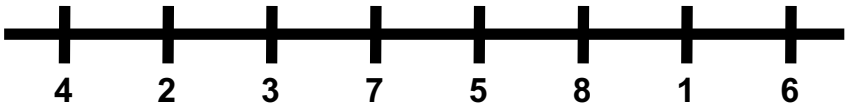
General approaches



- **Agglomerative**
  (building trees)
  hierarchical clustering
  (Mike Eisen's Cluster)
- **Partitional**
  (finding prototypes)
  k-means, som-mapping
  gene shaving

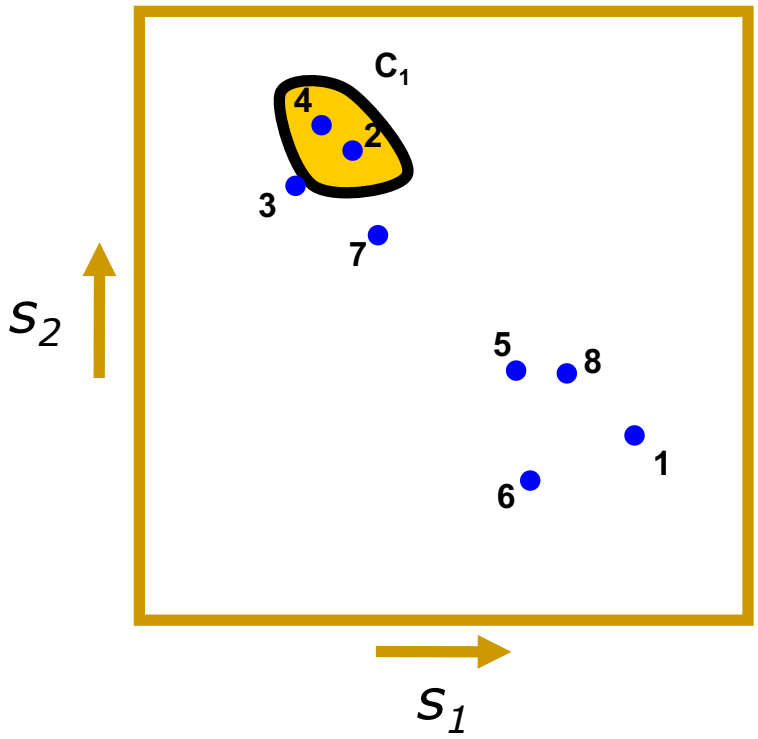
Hierarchical clustering



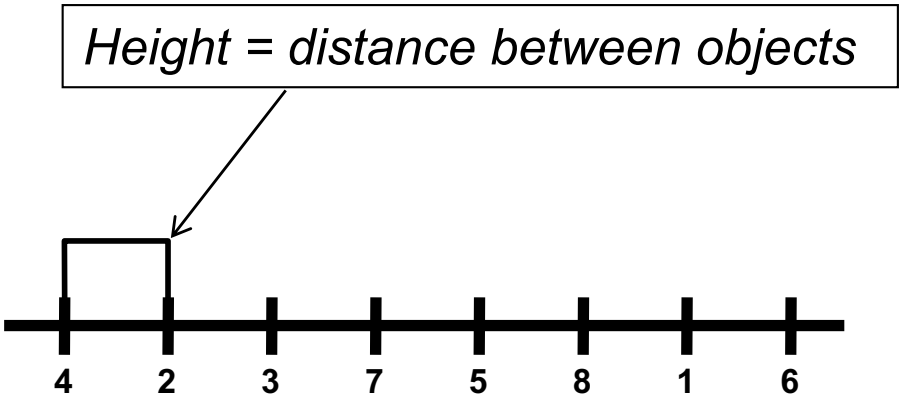


Find most similar objects (points/genes) and group them

Hierarchical clustering

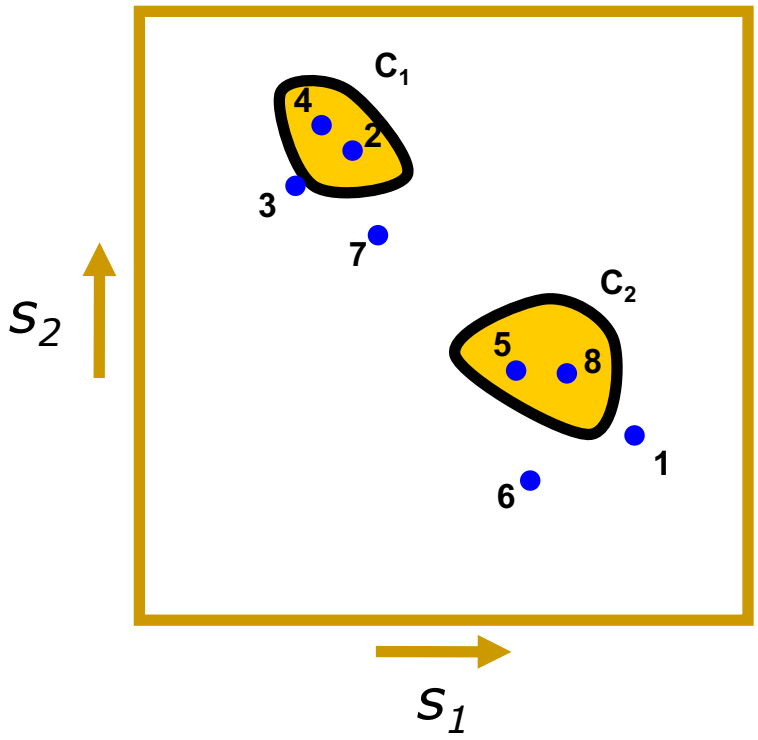


dendrogram

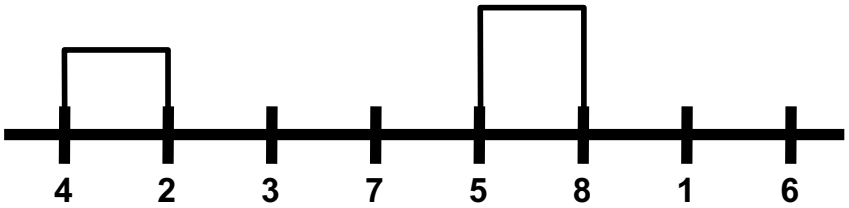


These are: objects 4 and 2
Again, find most similar objects (genes or clusters) and group them

Hierarchical clustering



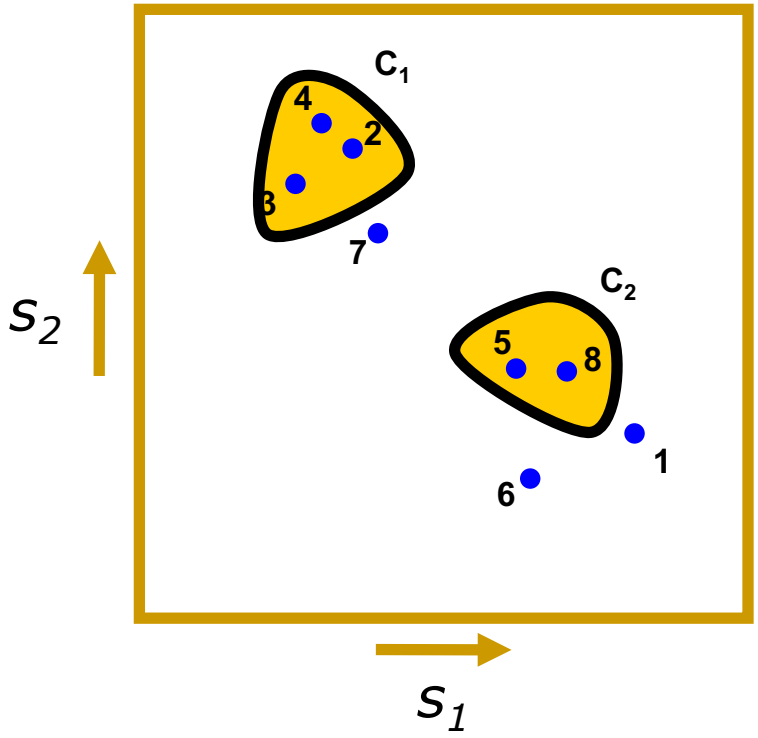
dendrogram

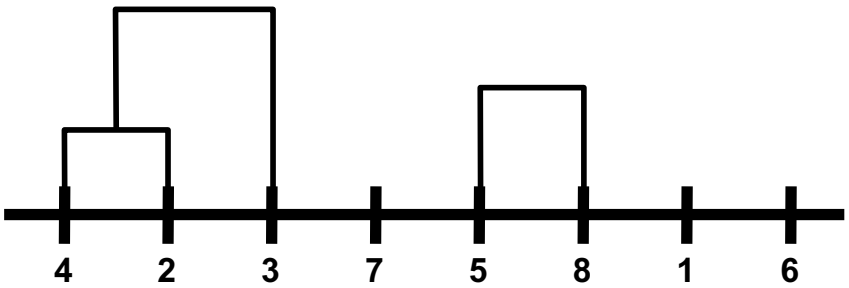


These are: objects 5 and 8
Repeat finding most similar objects (genes or clusters) and grouping them

Hierarchical clustering

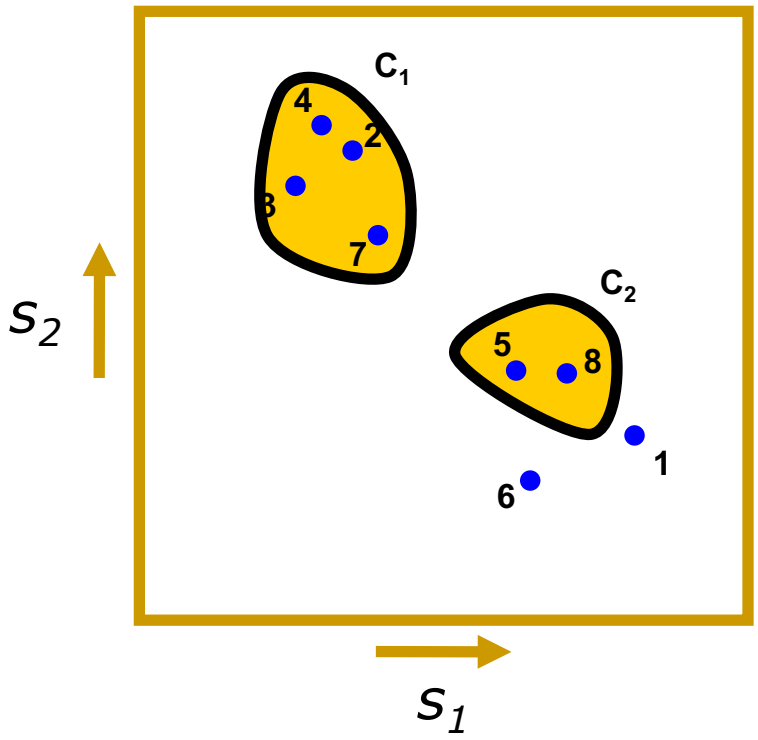
dendrogram



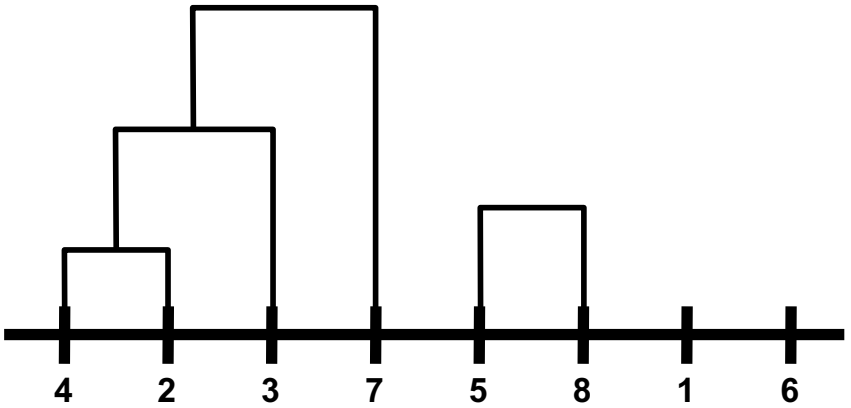


Join object 3 and cluster 1
Repeat process

Hierarchical clustering

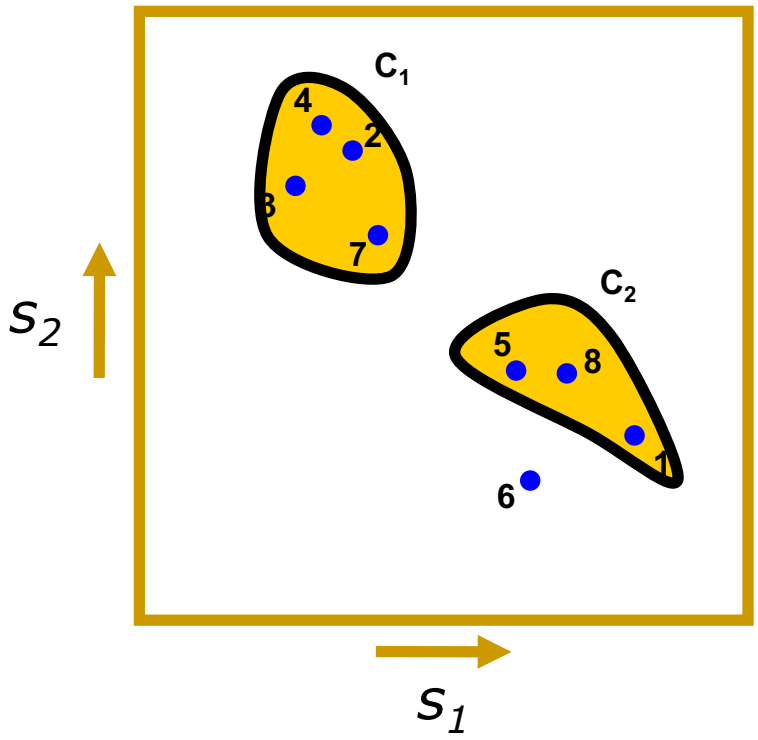


dendrogram

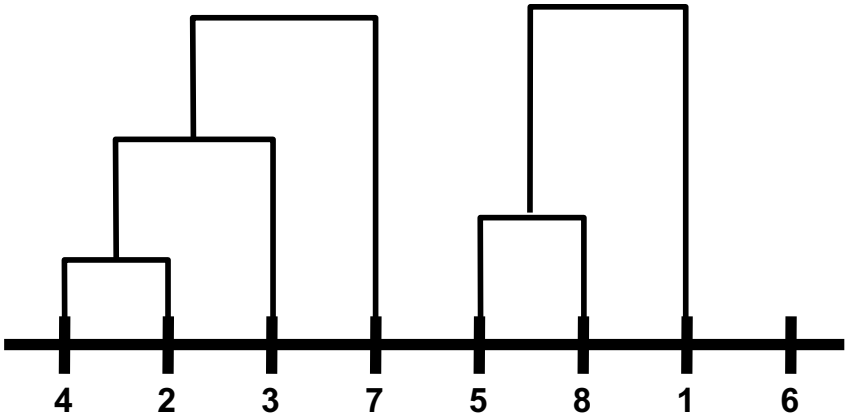


Join [object 7 and cluster 1] into [cluster 1]
Repeat process

Hierarchical clustering

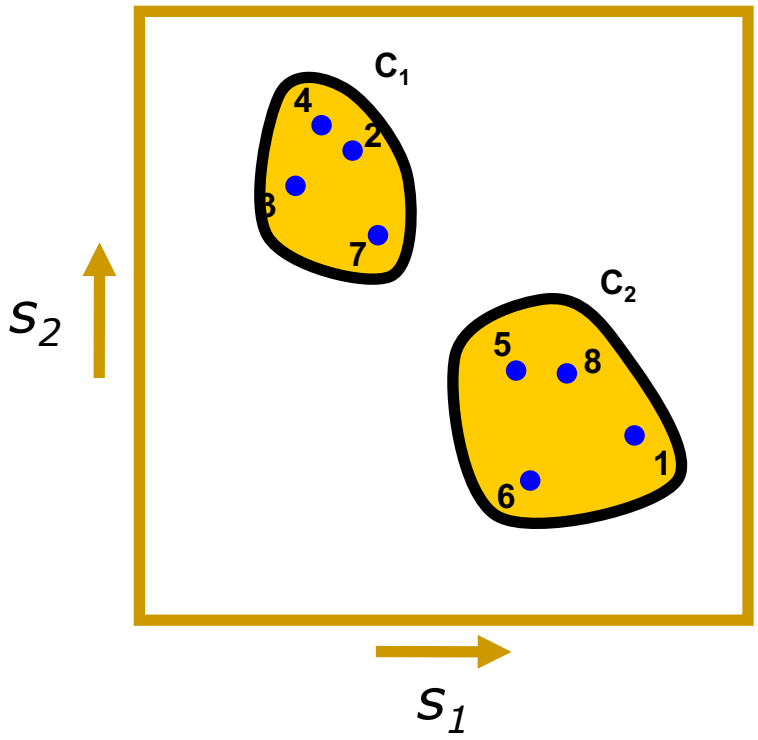


dendrogram

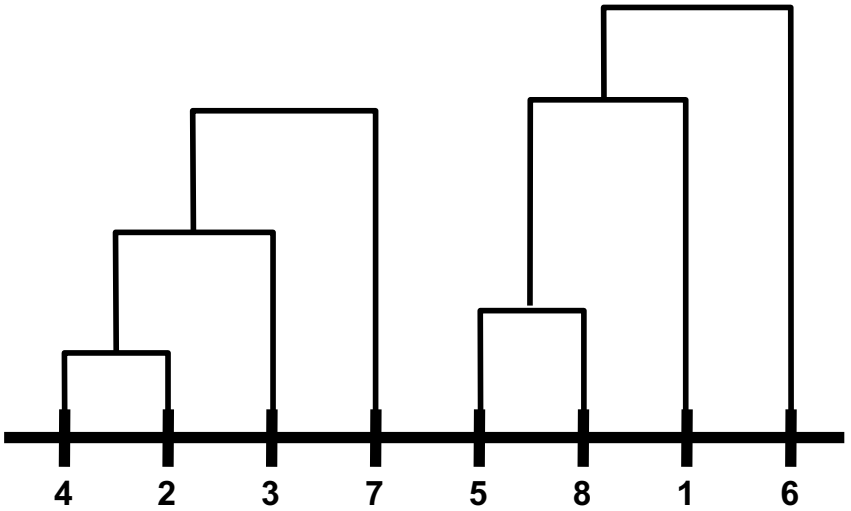


Join [object 1 and cluster 2] ➔ [cluster 2]
Repeat process

Hierarchical clustering

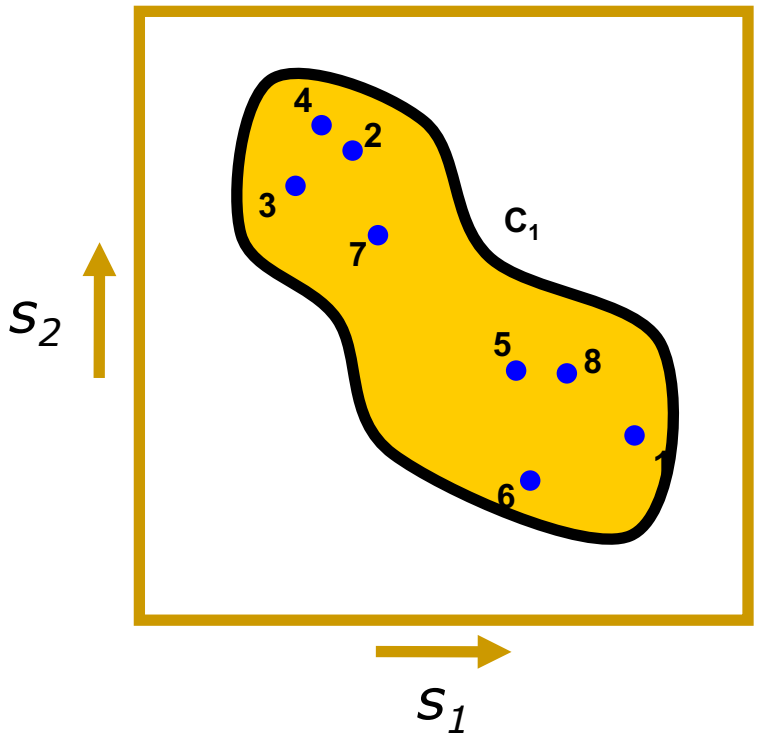


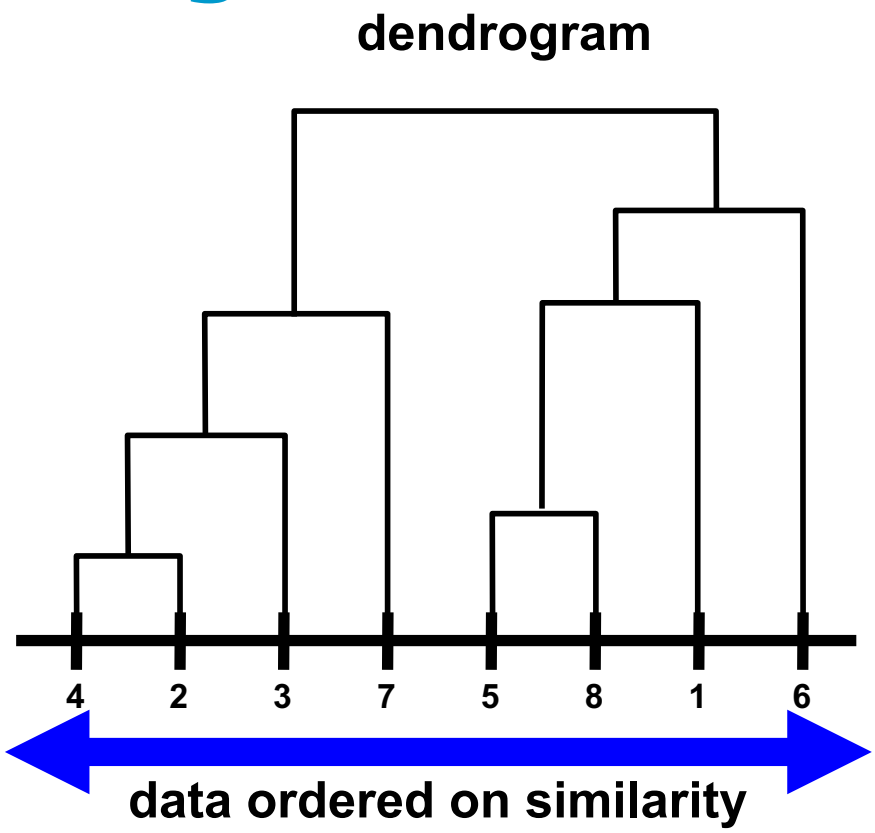
dendrogram



Join [object 6 and cluster 2] ➔ [cluster 2]
Repeat process

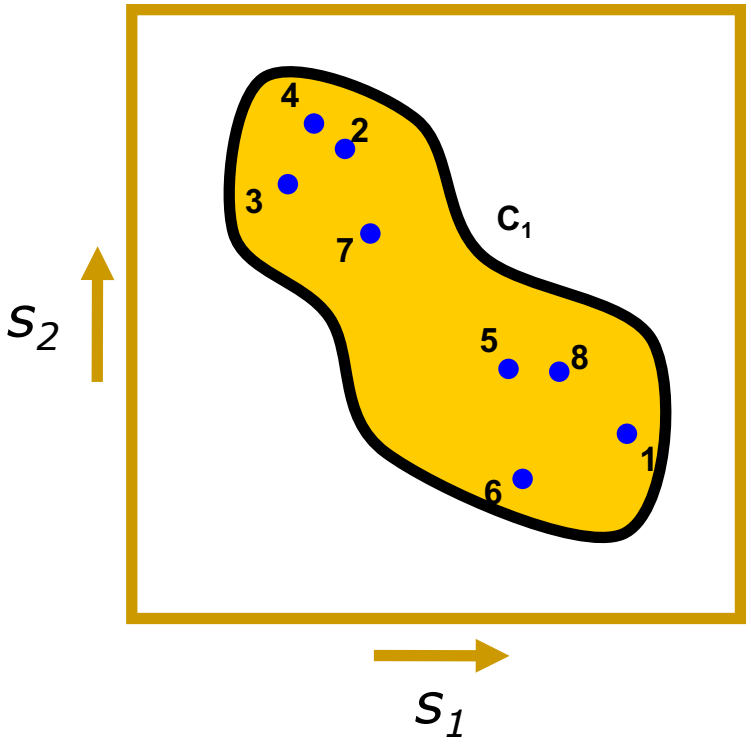
Hierarchical clustering





Join [cluster 1 and cluster 2] ➔ [cluster 1]
All in one cluster: FINISHED!

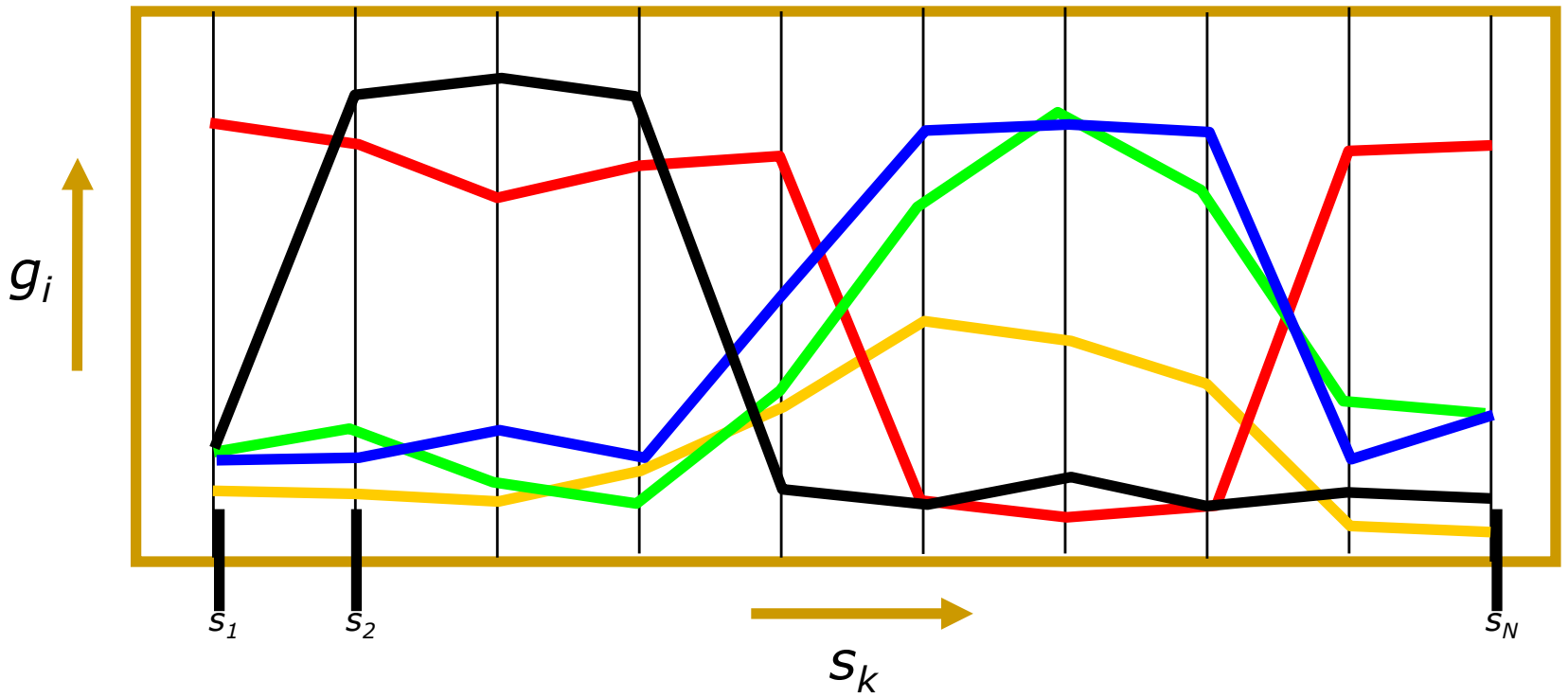
Hierarchical clustering: Choices

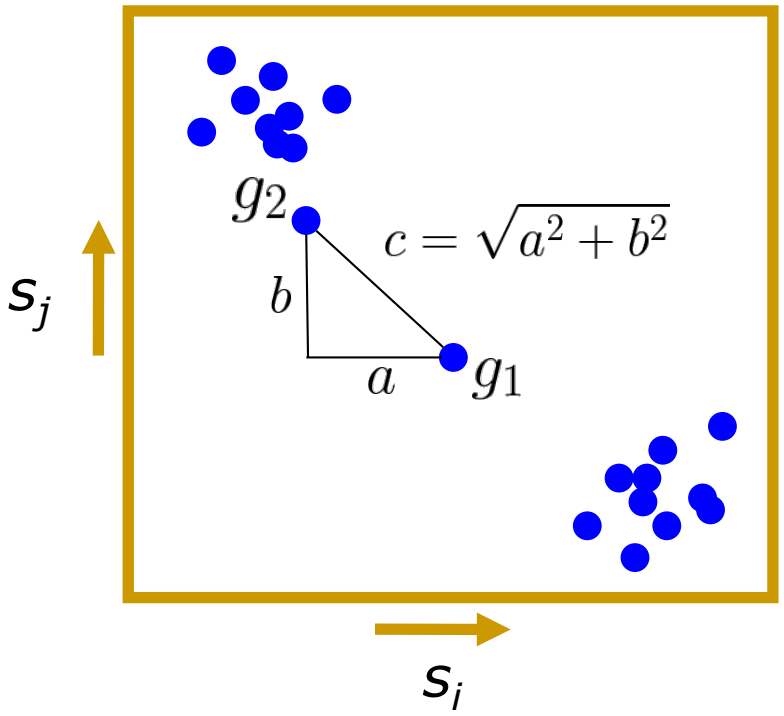


Need to know:

- **Similarity between objects**
- **Similarity between clusters**

Hierarchical clustering: Similarity between objects





Euclidean distance

$$a = g_1(s_i) - g_2(s_i)$$

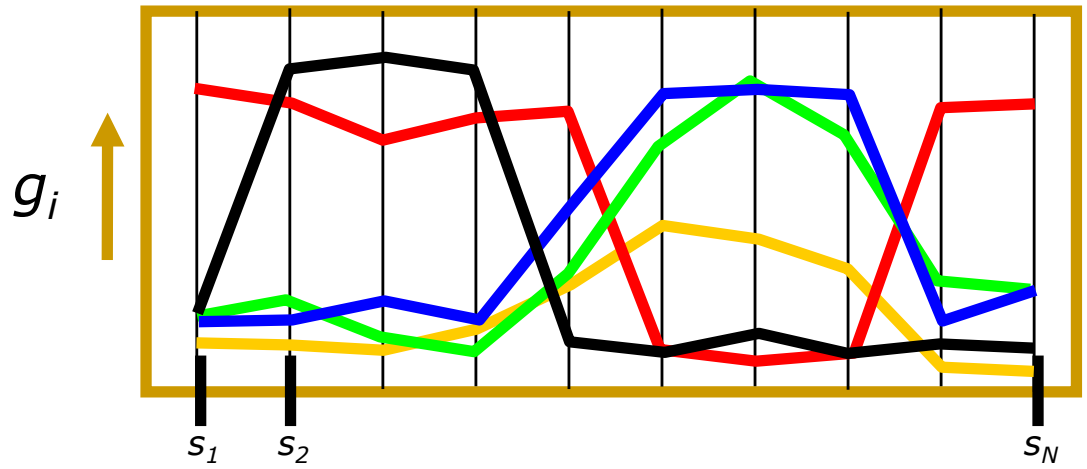
$$b = g_1(s_j) - g_2(s_j)$$

$$c = \sqrt{a^2 + b^2}$$

$$c = \sqrt{(g_1(s_i)) - g_2(s_i))^2 + (g_1(s_j) - g_2(s_j))^2}$$

$$d(g_1, g_2) = c = \sqrt{\sum_{k=1}^{K} (g_1(s_k) - g_2(s_k))^2}$$

Similarity between objects



Euclidean distance

$$d(g_1, g_2) = \sqrt{\sum_{k=1}^{K} (g_1(s_k) - g_2(s_k))^2}$$

d(●,●) < d(●,●)
d(●,●) << d(●,●)
d(●,●) << d(●,●)

Pearson correlation

$$1 - \rho_{g_1,g_2}$$

$$\rho_{g_1,g_2} = \frac{\sum_{k=1}^{K} g_1(s_k) * g_2(s_k)) - \mu_1 * \mu_2)}{(\sigma_1 * \sigma_2)}$$

d(●,●) ≈ d(●,●)
d(●,●) << d(●,●)
d(●,●) << d(●,●)

Mixed Pearson correlation

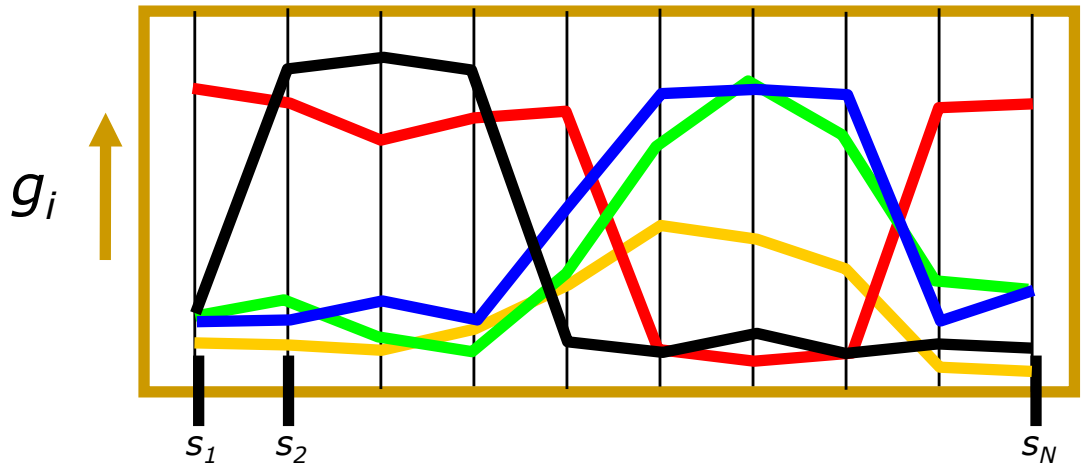
$$1 - |\rho_{g_1,g_2}|$$

d(●,●) ≈ d(●,●)
d(●,●) ≈ d(●,●)
d(●,●) << d(●,●)

The Delft Bioinformatics Lab

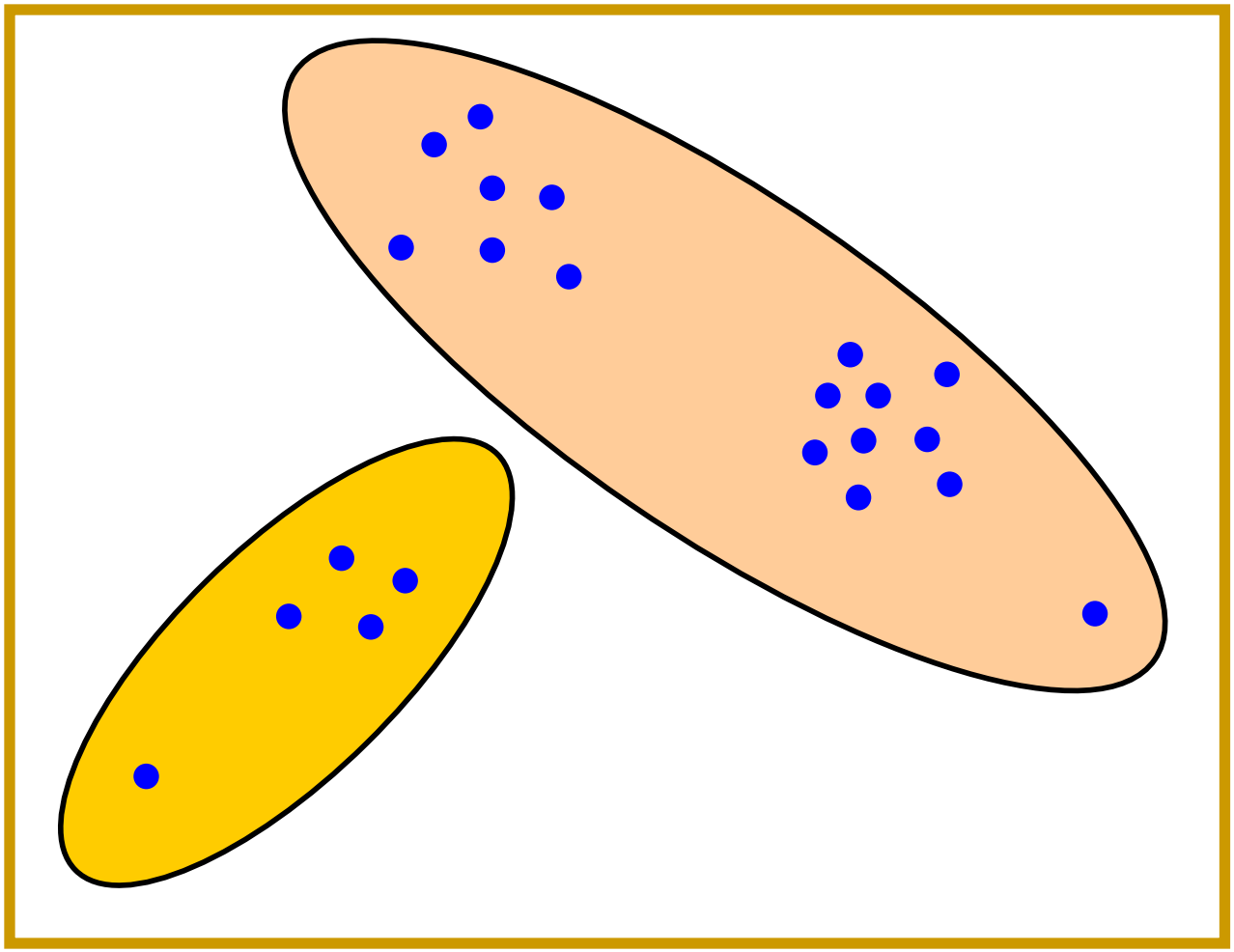
TU Delft

Similarity between objects

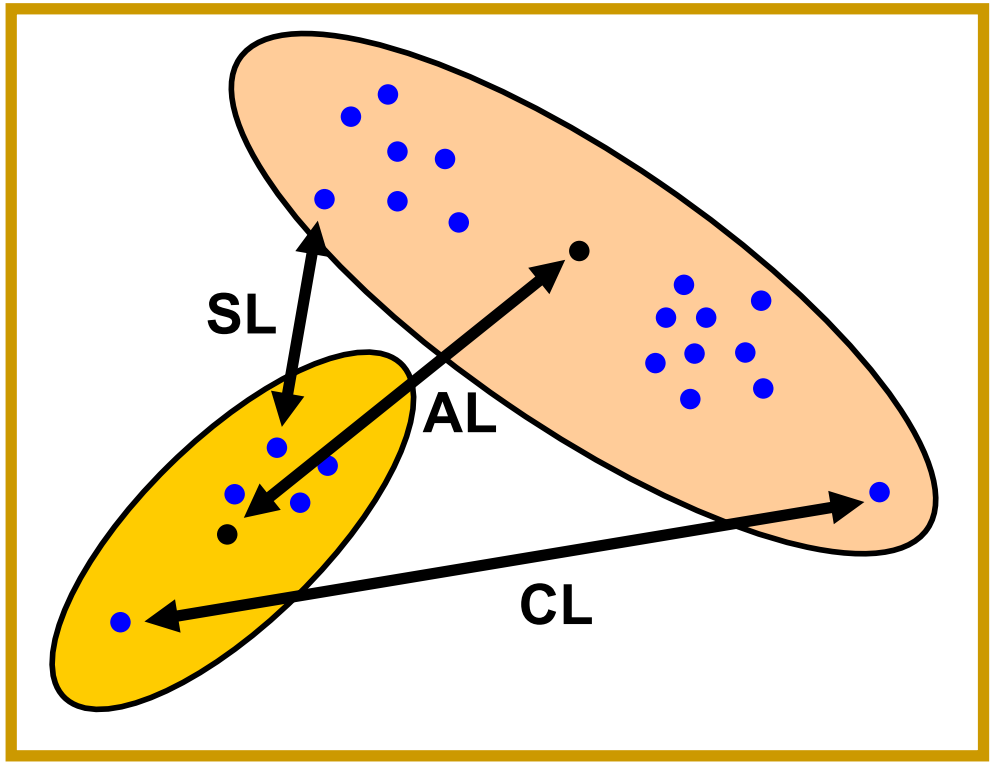


Euclidean distance	Pearson correlation	Mixed Pearson correlation
match exact shape	ignore amplitude	ignore amplitude & sign
d(●,●) < d(●,●)	d(●,●) ≈ d(●,●)	d(●,●) ≈ d(●,●)
d(●,●) << d(●,●)	d(●,●) << d(●,●)	d(●,●) ≈ d(●,●)
d(●,●) << d(●,●)	d(●,●) << d(●,●)	d(●,●) << d(●,●)

Similarity between clusters?

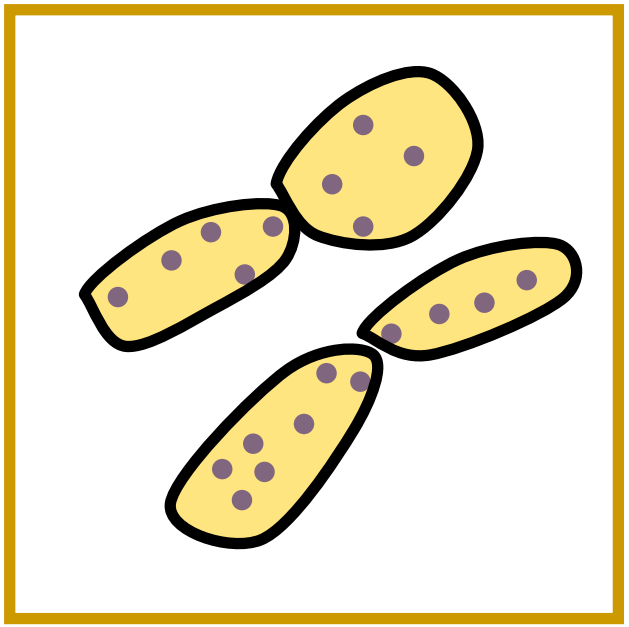


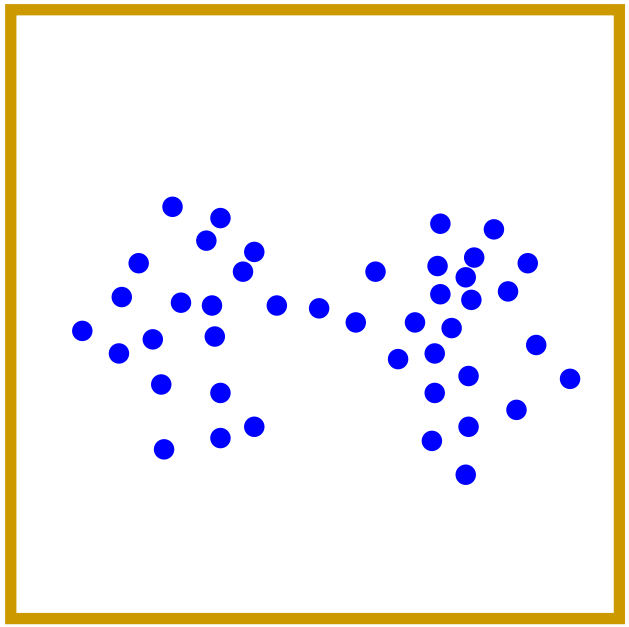
Similarity between clusters

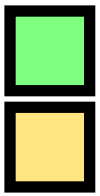


- **Single linkage:** Closest objects
- **Complete linkage:** Furthest objects
- **Average linkage:** Average dissimilarity

SL vs CL: Shape influences



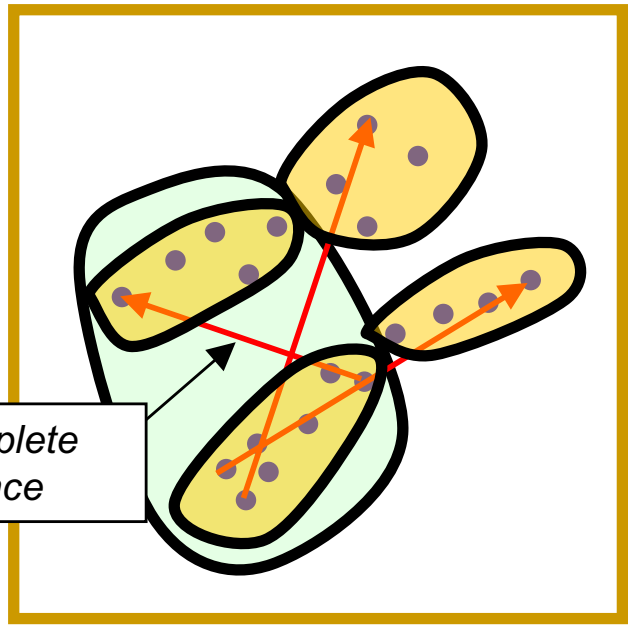




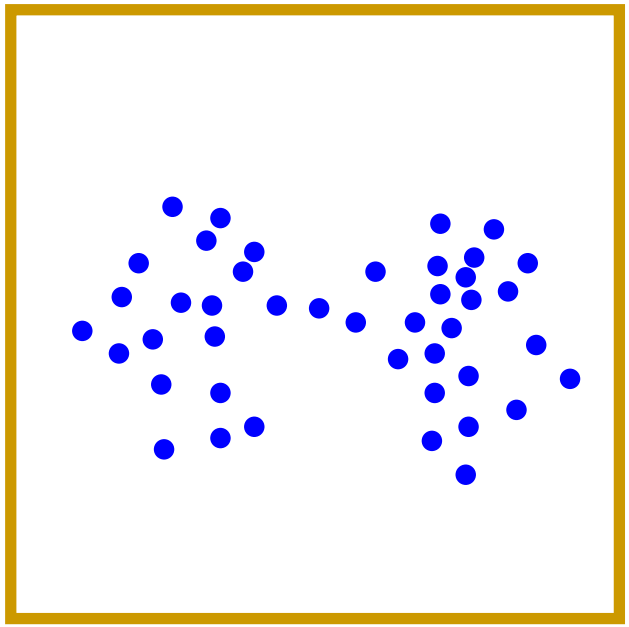
complete linkage ?

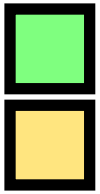
single linkage ?

SL vs CL: Shape influences



Shortest complete linkage distance

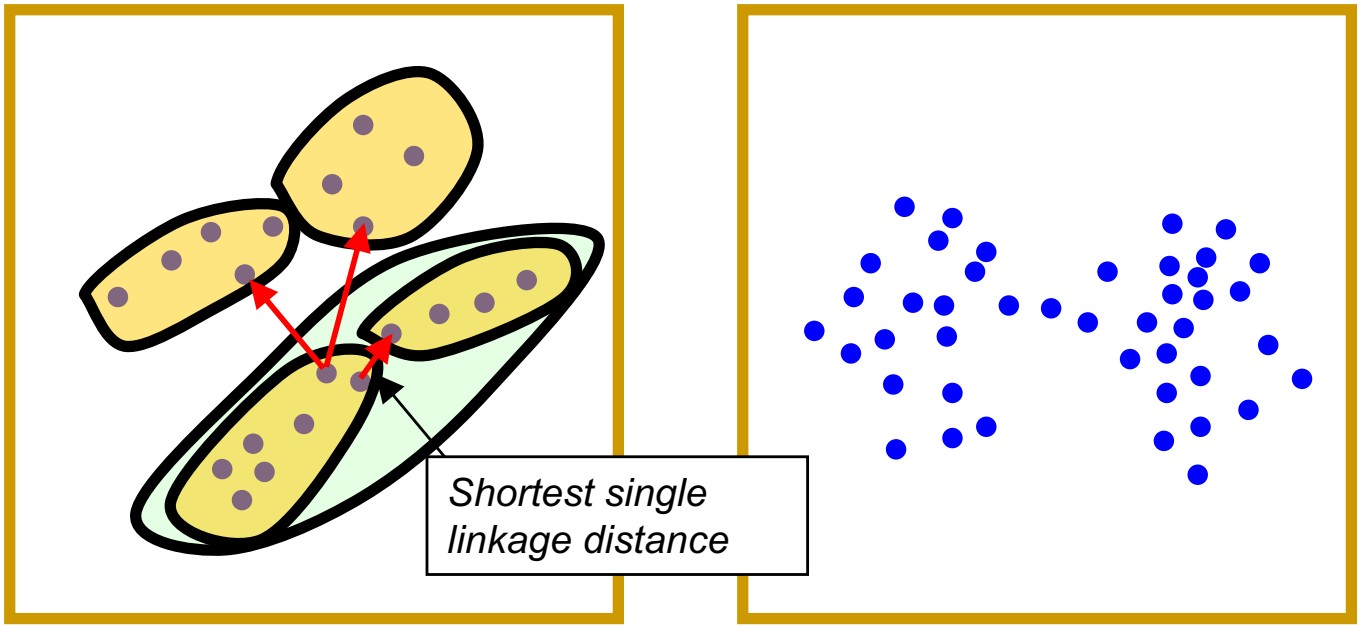


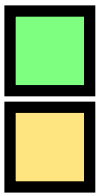


complete linkage ?

single linkage ?

SL vs CL: Shape influences

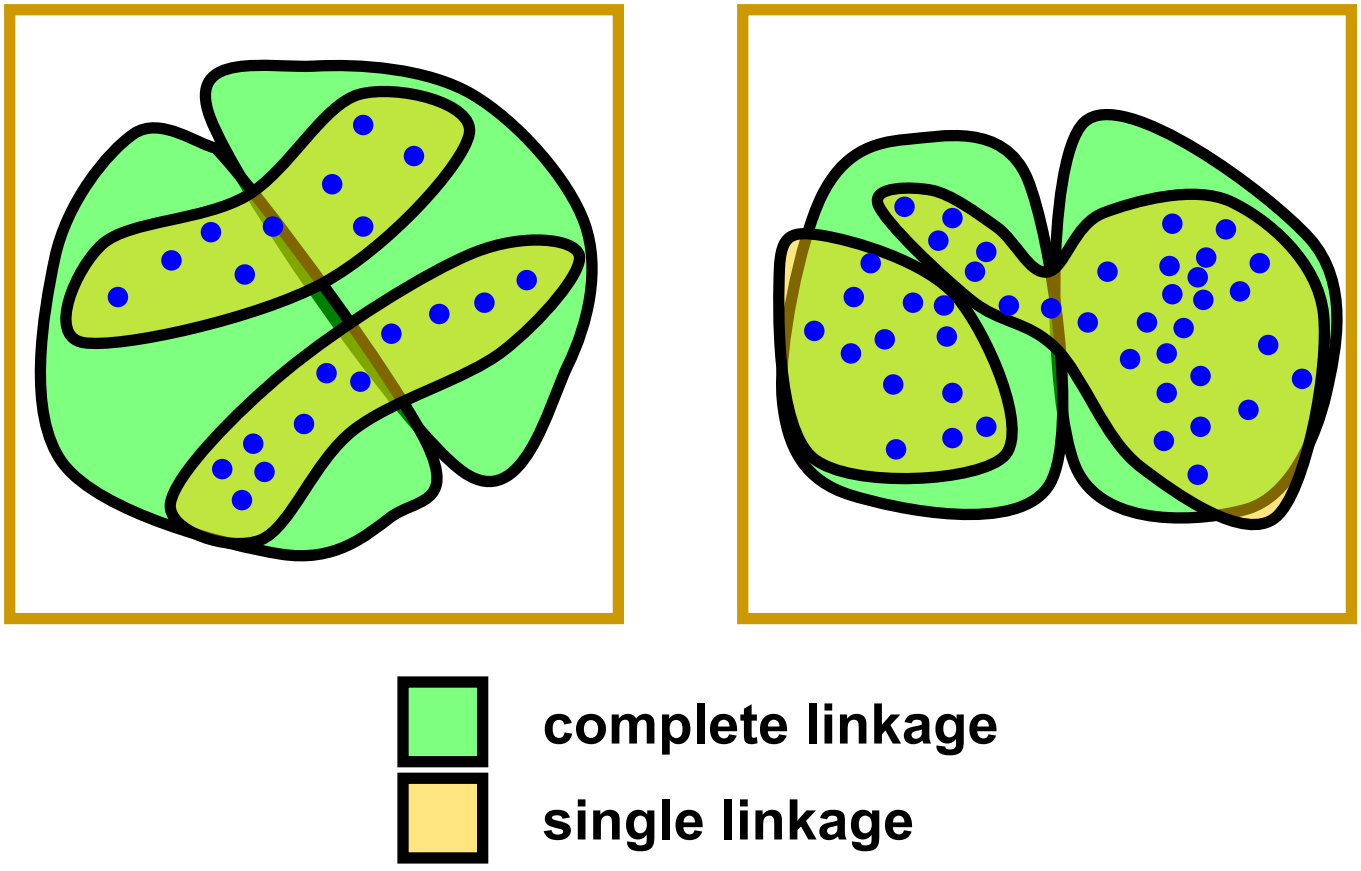




complete linkage ?

single linkage ?

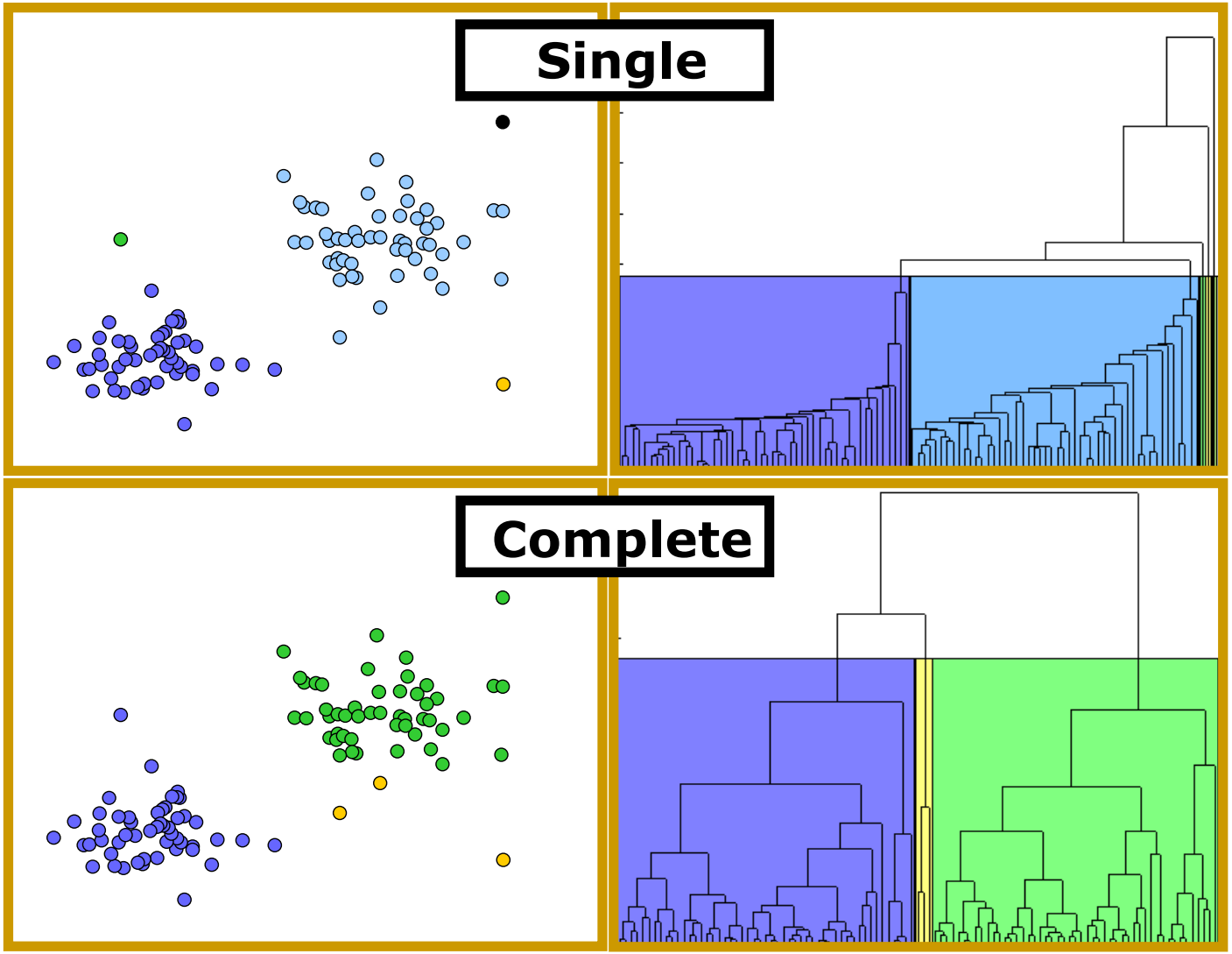
SL vs CL: Shape influences

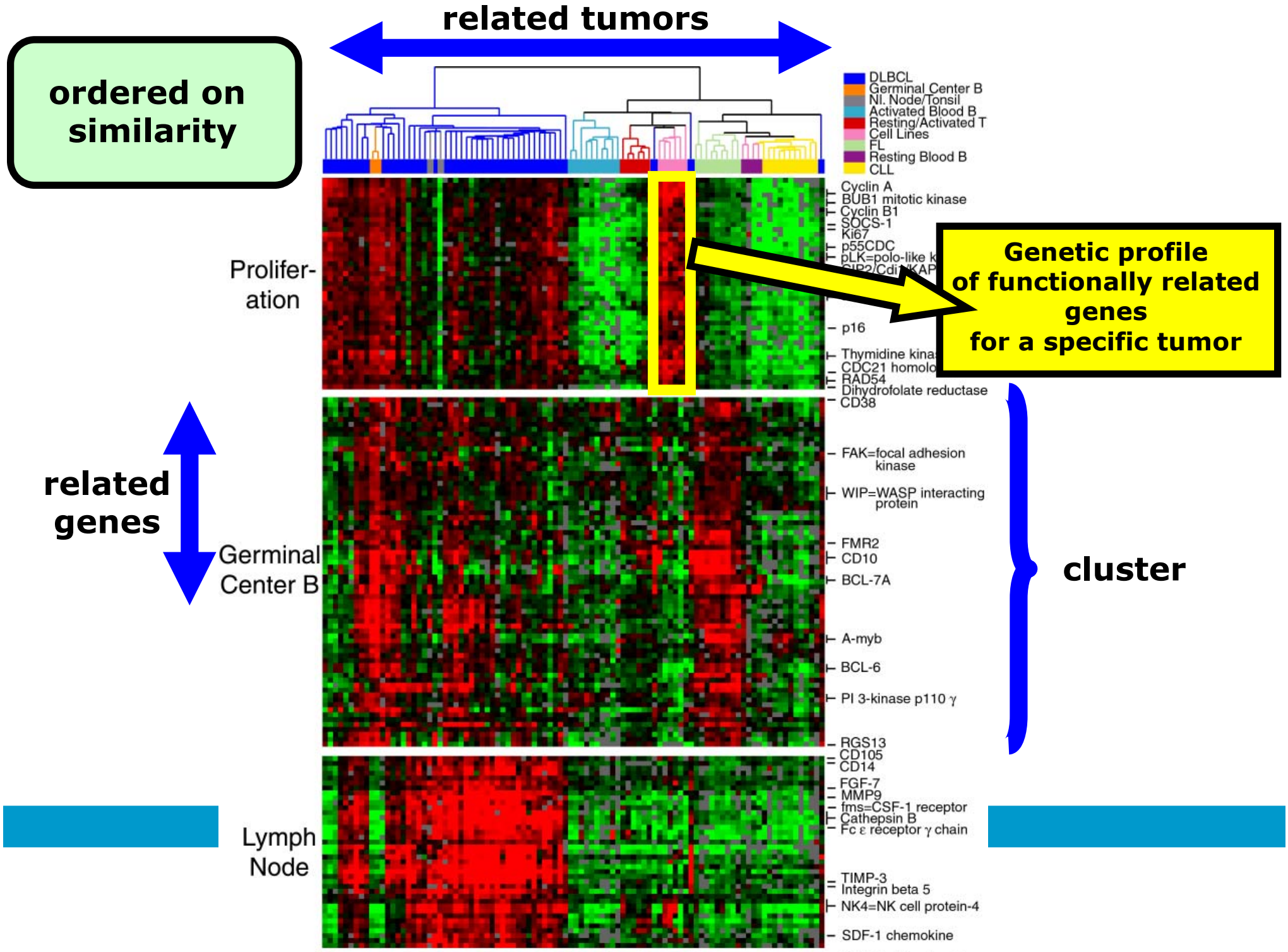


complete linkage

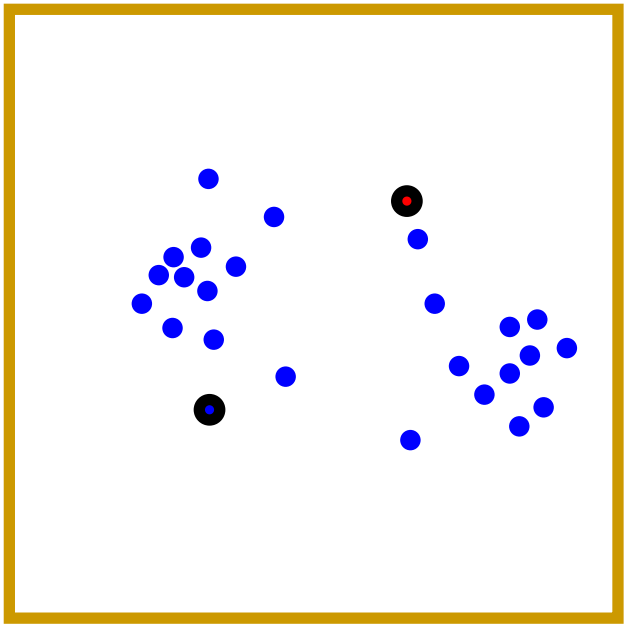
single linkage

SL vs CL: Outlier influences



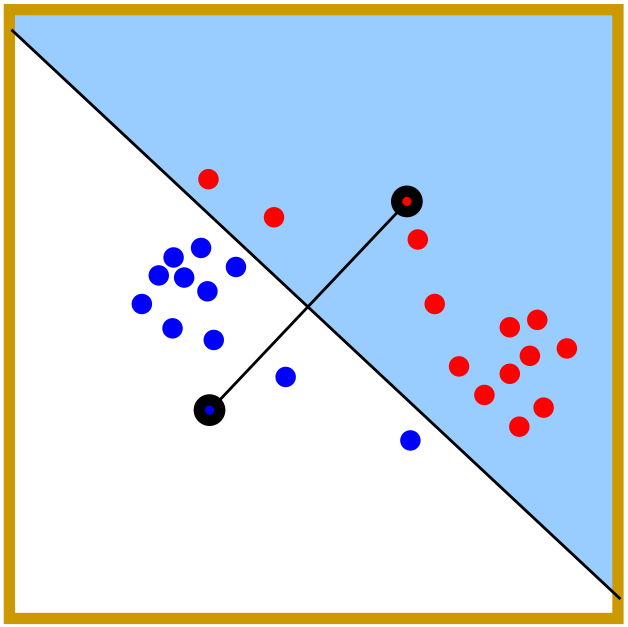

related tumors
ordered on similarity
DLBCL
Germinal Center B
NI. Node/Tonsil
Activated Blood B
Resting/Activated T
Cell Lines
FL
Resting Blood B
CLL
Cyclin A
BUB1 mitotic kinase
Cyclin B1
SOCS-1
Ki67
p55CDC
p16
CDC21 homolo
RAD54
Dihydrofolate reductase
CD38
FAK=focal adhesion kinase
WIP=WASP interacting protein
FMR2
CD10
BCL-7A
A-myb
BCL-6
PI 3-kinase p110 γ
RGS13
CD105
CD14
FGF-7
MMP9
fms=CSF-1 receptor
Cathepsin B
Fc ε receptor γ chain
TIMP-3
Integrin beta 5
NK4=NK cell protein-4
SDF-1 chemokine
Prolifer-ation
Germinal Center B
Lymph Node
related genes
cluster
Genetic profile of functionally related genes for a specific tumor

K-means clustering: Explanation by example



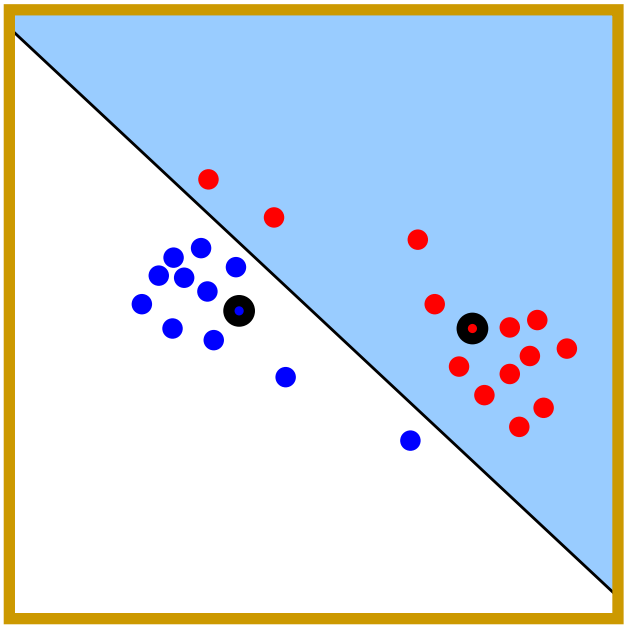
Choose randomly 2 prototypes

K-means clustering: Explanation by example



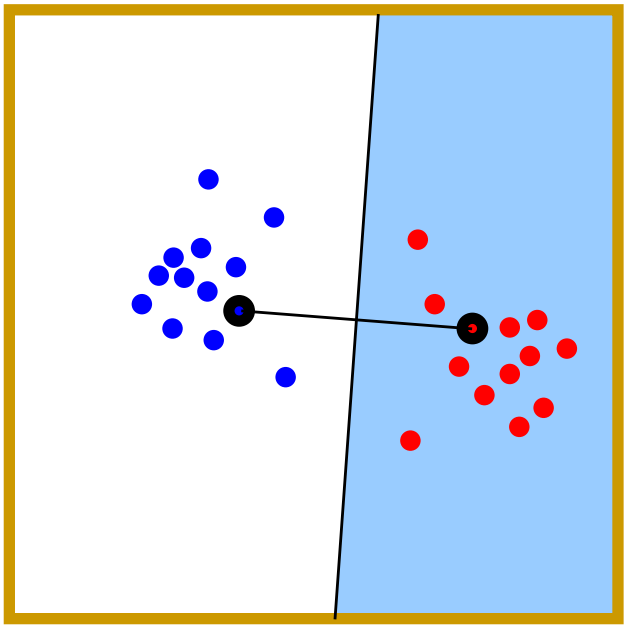
Assign objects to closest prototype
Blue area: cluster 1
White area: cluster 2

K-means clustering: Explanation by example



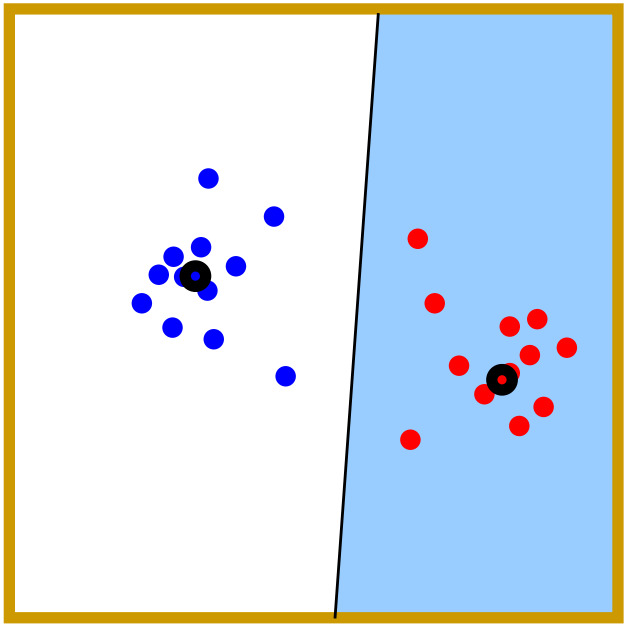
Calculate new cluster prototypes
By averaging objects

K-means clustering: Explanation by example



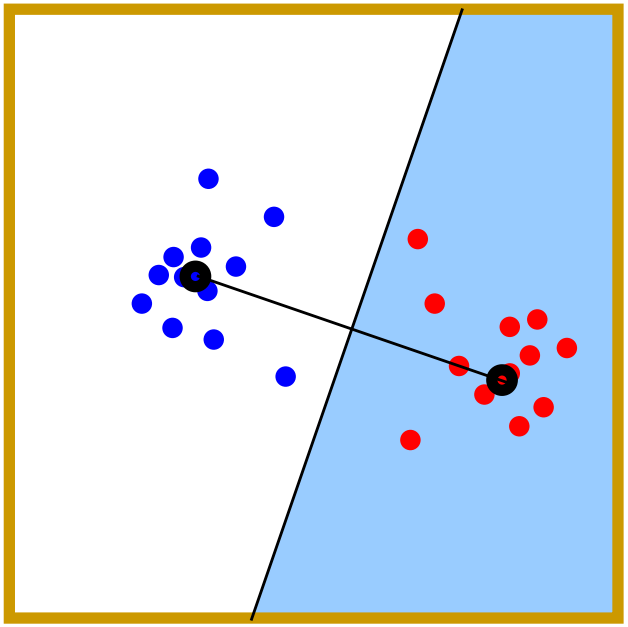
Re-assign objects to closest prototype
Blue area: cluster 1
White area: cluster 2

K-means clustering: Explanation by example



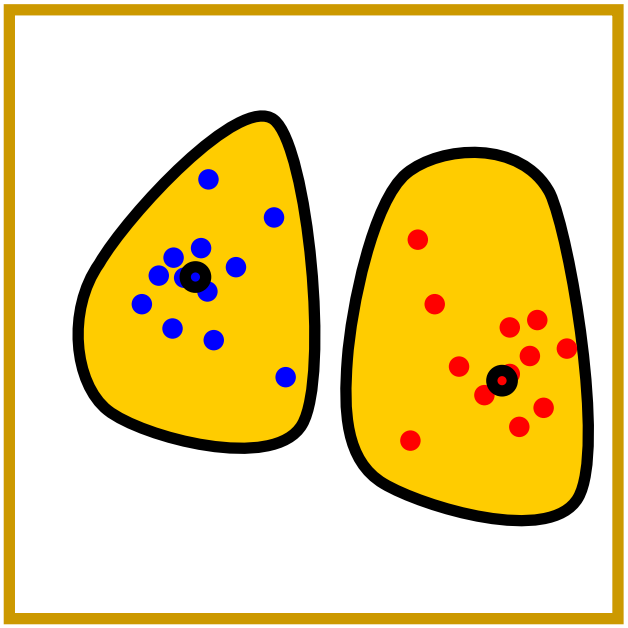
Re-calculate new cluster prototypes

K-means clustering: Explanation by example



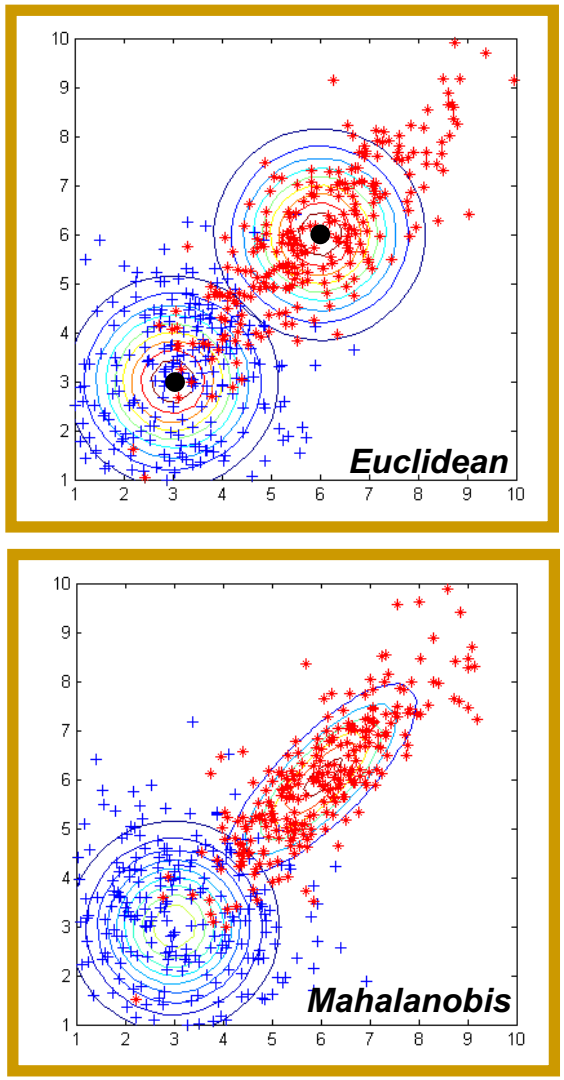
Re-assign objects to closest prototype
If no objects change cluster then finished

K-means clustering: Explanation by example



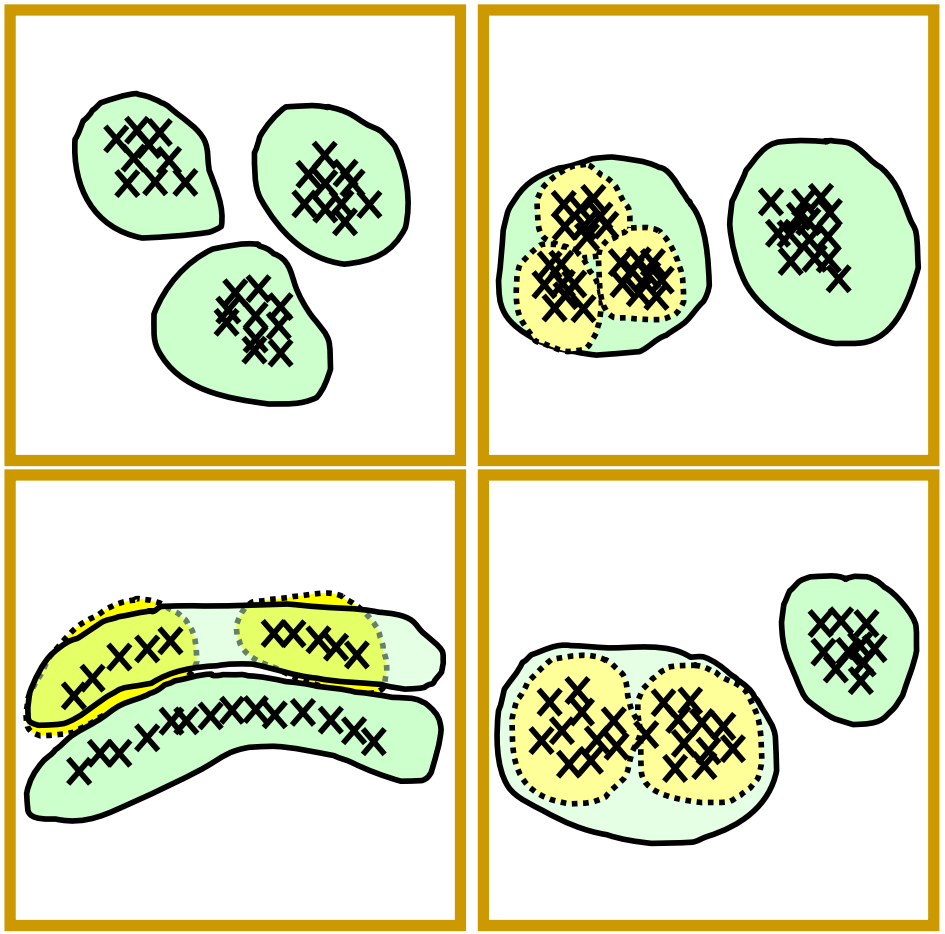
Establish clusters

K-means clustering: Parameters



- **K-means**
  - Fixed number of clusters (need to know a priori)
  - Choice of distance measure
  - Prototype choice
- **Distance measure**
  - Euclidean: Round clusters
  - Mahalanobis: Elongated clusters
- **Prototype choice**
  - Point
  - Line etc.
- **Number of clusters**
  - Validate clustering!

Subjectivity

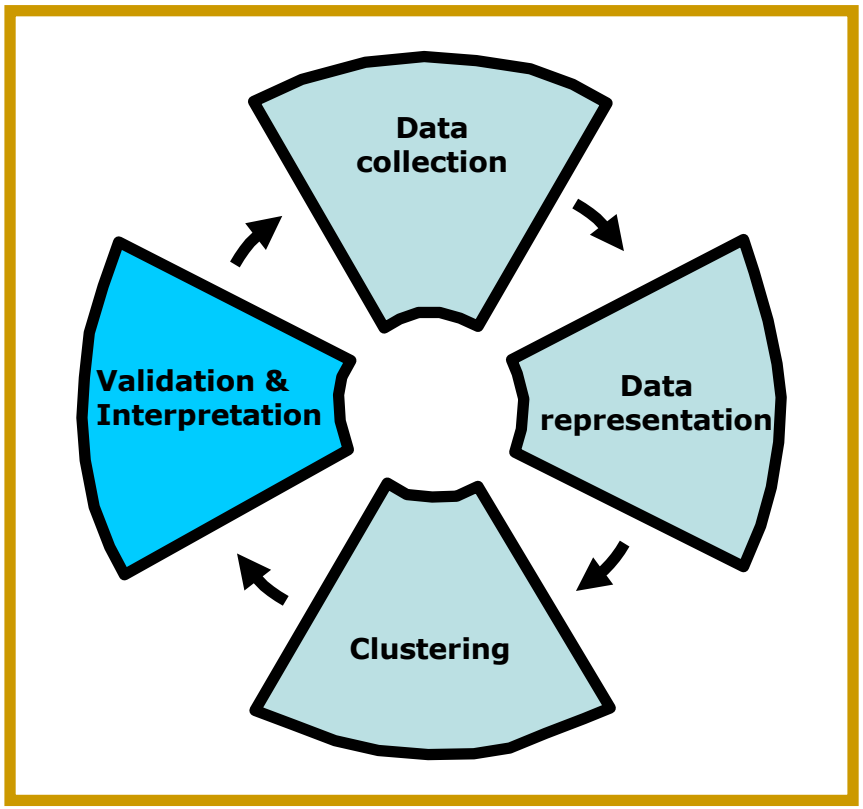


- **Principle choices:**
  - **similarity**
  - **algorithm**
- **Different choice leads to different results**

  Subjectivity becomes reality
- **Cluster process**

  Validate, interpret (generate hypothesis), repeat steps

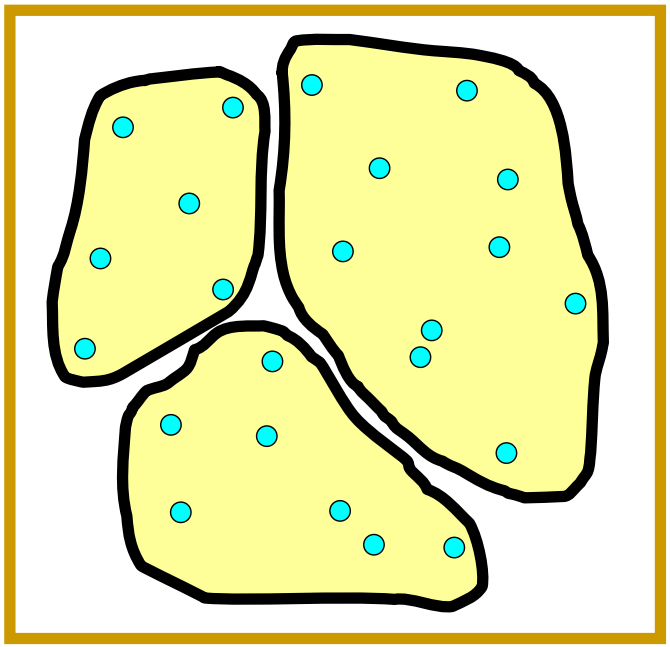
Validation



TOPICS

- **Cluster tendency**
- **Cluster validity**

Cluster Validation



- **Cluster tendency**

  Clustering **IMPOSES** structure even though data may not posses it

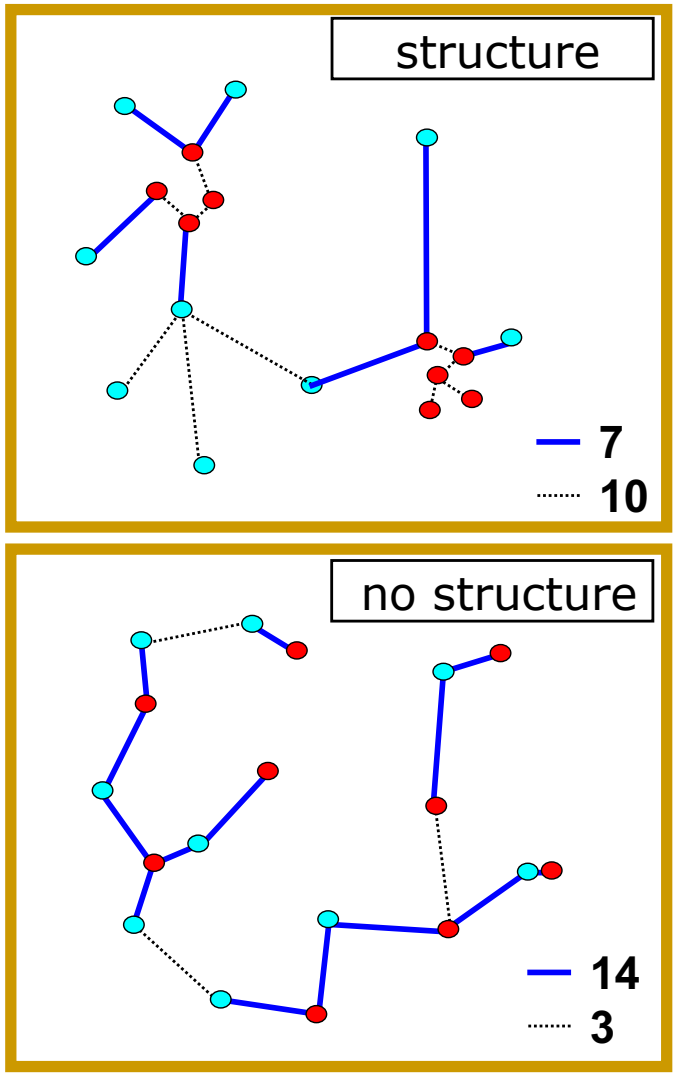
  Aim: Test whether data possesses structure

- **Cluster validity**

  Choices impose restrictions on for example shape

  Aim: Quantitative evaluation of the clustering results

Test for spatial randomness



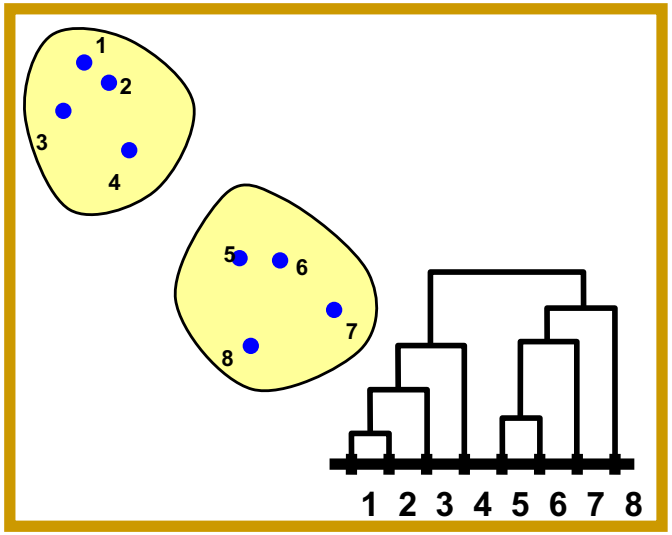
- **Test**

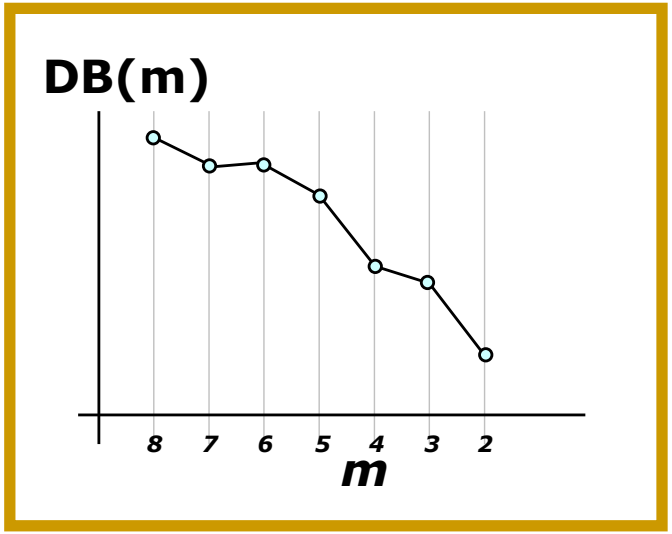
  If data (●) clusters frequently with random data (○) then data structureless

- **Approach**
  - Generate random vectors (**Y**) uniformly over observed region of data (**X**)
  - Find MST (single linkage HC) of **X** ∨ **Y**
  - Determine number of edges *q* that connect vectors of **X** with **Y**
  - If **X** contains clusters *q* should be small!

  (multiple random vs random measurements gives likelihood for *q*)

Davis-Bouldin index





- **Test**

  Select specific clustering according to a criteria

  For example: Davis-Bouldin index

- **DB index**

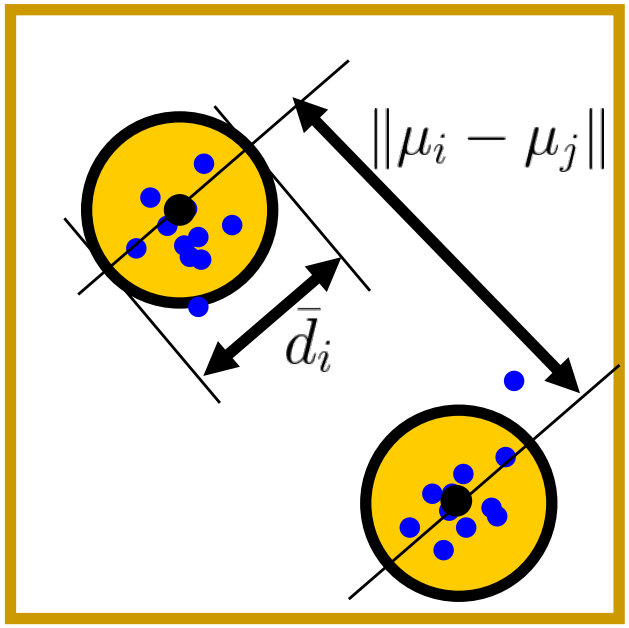
  For a specific clustering m, **DB(m)** : Average similarity of a cluster with its most similar cluster

- **Approach**

  Goal: Clusters to have minimal similarity

  Seek: Clustering that minimize *DB*(*m*) wrt *m*

Davis-Bouldin index



- **Similarity cluster C_i and C_j**

$$D_{i,j} = \frac{(\bar{d}_i + \bar{d}_j)}{\|\mu_i - \mu_j\|}$$

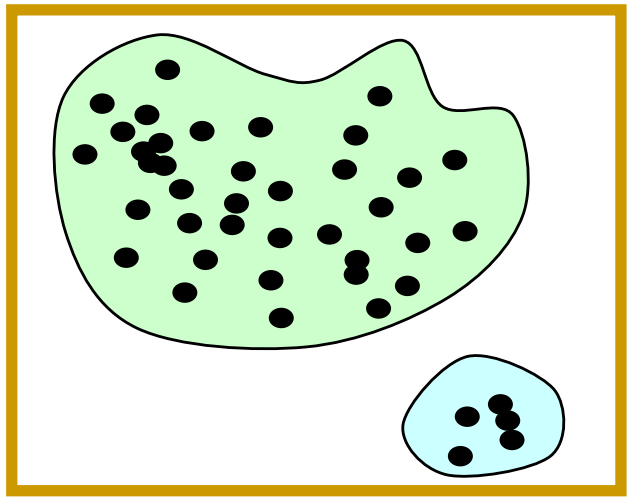
- $\bar{d}_i$:average distance within cluster *i*, μ_i:centroid of cluster *i*
- **Most similar cluster to C_i**

$$R_{i,j} = \max_{j \neq i} \{D_{i,j}\}$$

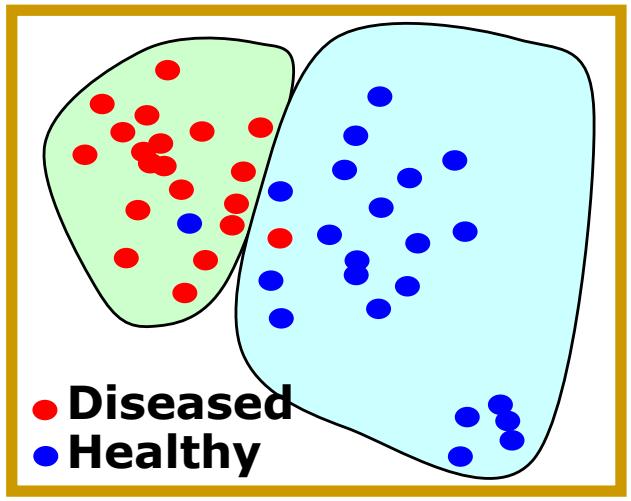
- **DB index**

$$DB = \frac{1}{k} \sum_{k=1}^{k} R_{i,j}$$

Clustering vs classification



- **Machine learning**
- **Clustering**
  - **unsupervised** learning
  - discovering structure/relations



- **Classification**
  - **Supervised** learning
  - Learning certain behavior
  - Prior information available about different groups

Clustering: Summary

- Clustering is an unsupervised, iterative process of interpreting data – not proof!
- Cluster results highly depend on the choice of cluster algorithm and dissimilarity measure
- Clustering and classification serve different purposes

Dimension Reduction

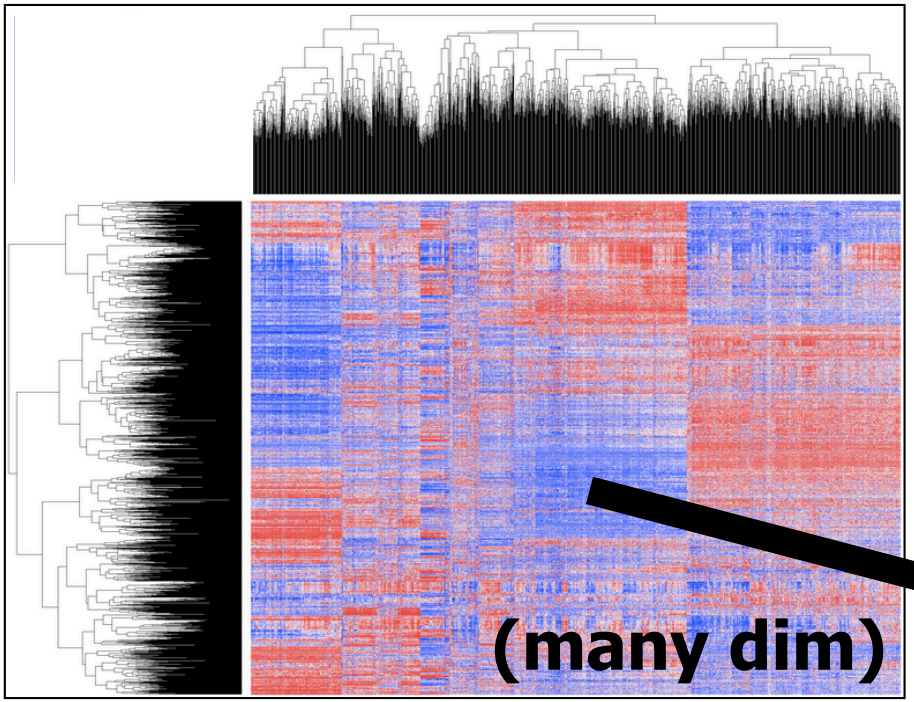
Dimensionality reduction (1)

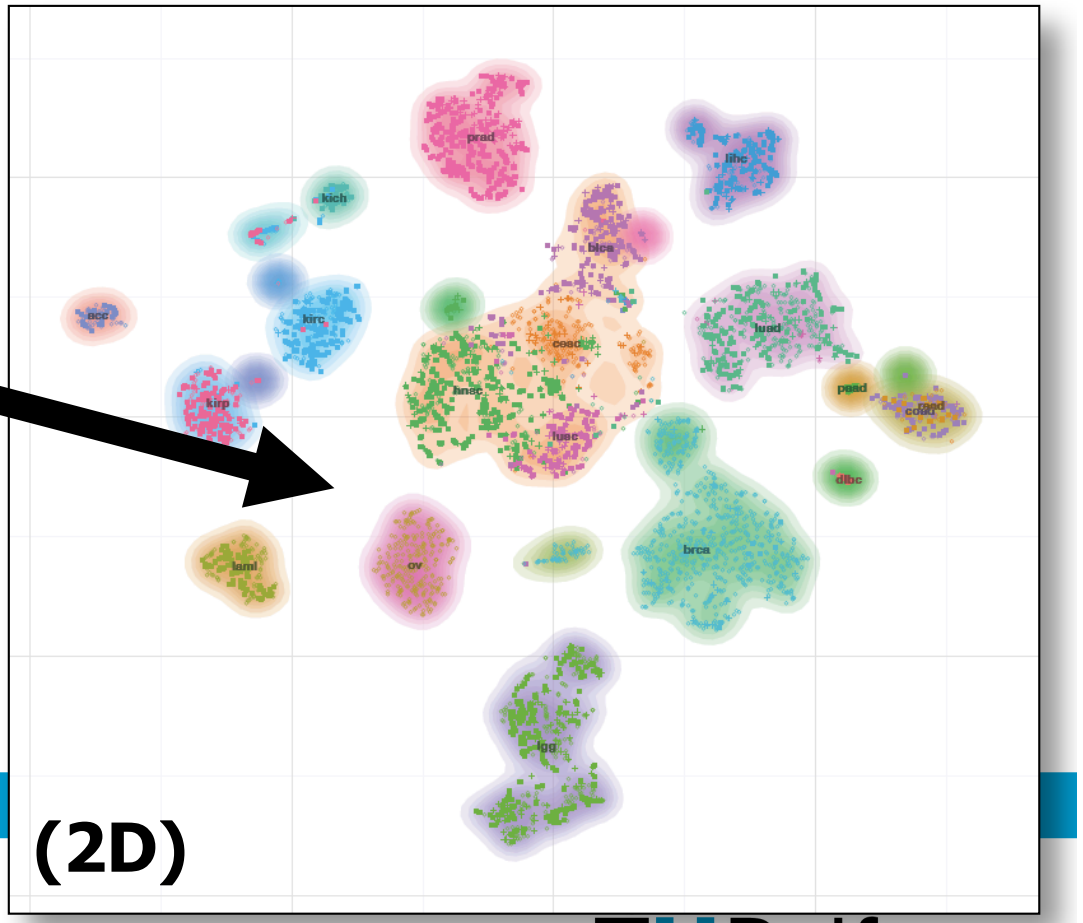
- **Many data sets are *high-dimensional*: each instance contains many features**

- **Why do we want to reduce data dimensionality?**
  - Make storage or processing of data easier
  - (Visual) discovery of hidden structure in the data
  - Intrinsic dimensionality might be smaller
  - Remove redundant and noisy features

Dimensionality reduction (2)

Visual discovery of data structure

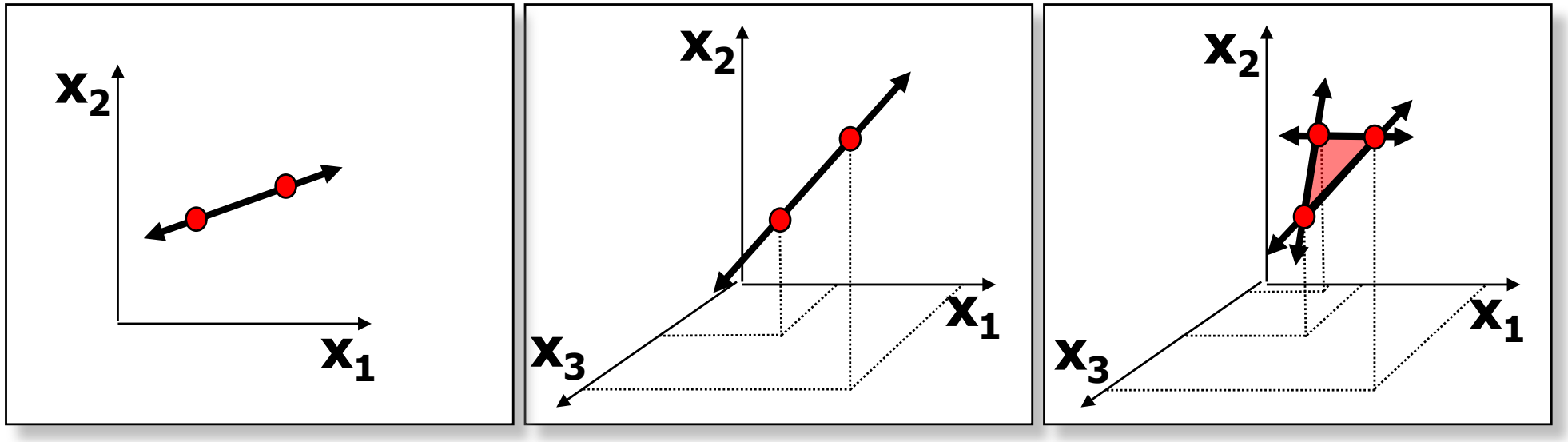




The Delft Bioinformatics Lab

TU Delft

Dimensionality reduction (3)
Intrinsic dimensionality



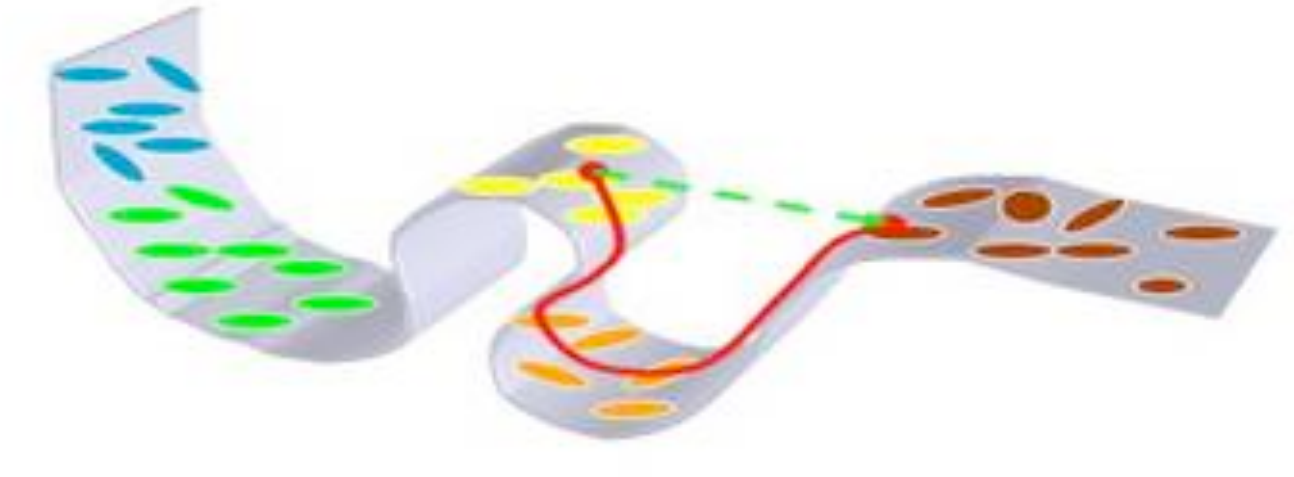
2 objects, 2 dimensions ➔ 1 dimension	2 objects, 3 dimensions ➔ 1 dimension	3 objects, 3 dimensions ➔ 2 dimension

Maximum number of dimensions: #objects -1

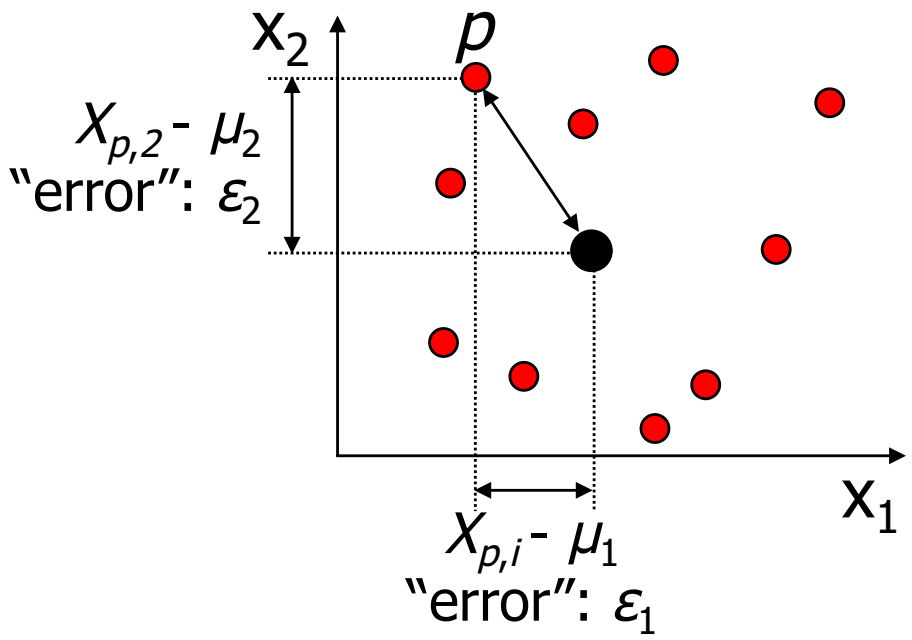
Dimensionality reduction (4)

Intrinsic dimensionality

Data may lie also be spread across a lower dimensional space (manifold)



Dimensionality reduction (5)
Accumulation of noise



In *D* dimensions distance becomes:

$$\sum_{i=1}^{D}(x_{p,i} - \mu_p)^2$$

If a feature *i* is not relevant than can be considered as error

$$\epsilon_i = (x_{p,i} - \mu_p)$$

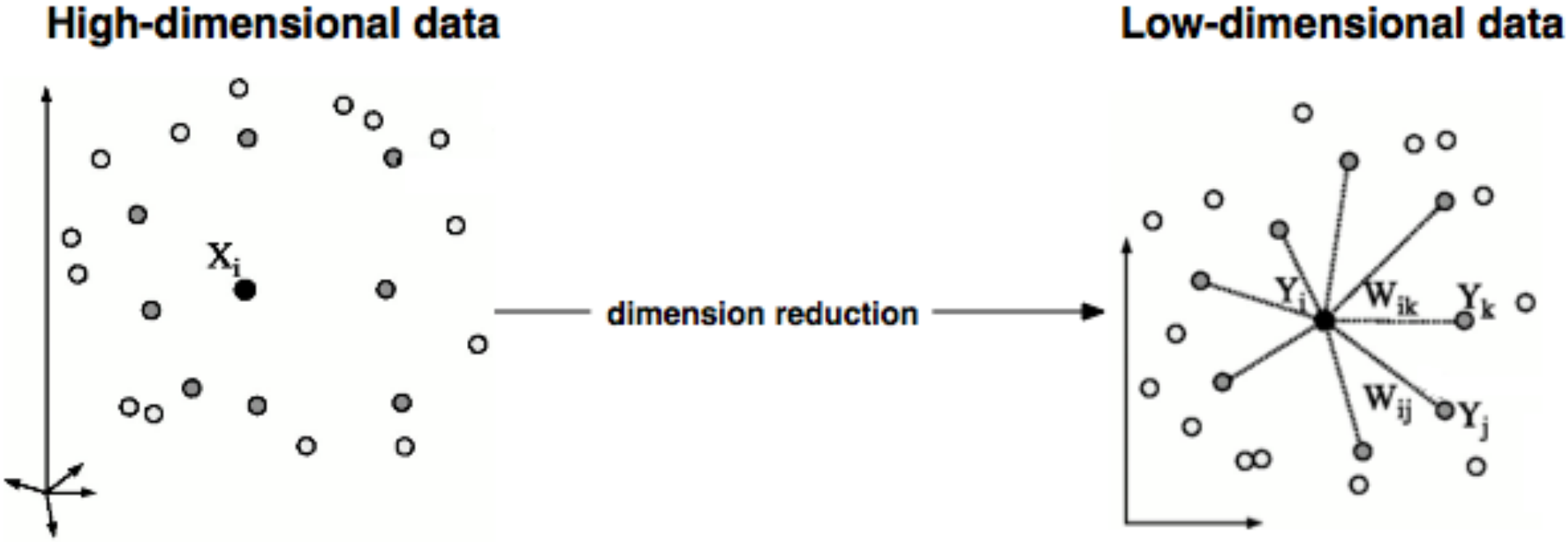
When there are *K* non-relevant features: accumulation of errors:

$$\sum_{j=1}^{K}(\epsilon_j)^2$$

If *K* >> *D-K* than error becomes dominating

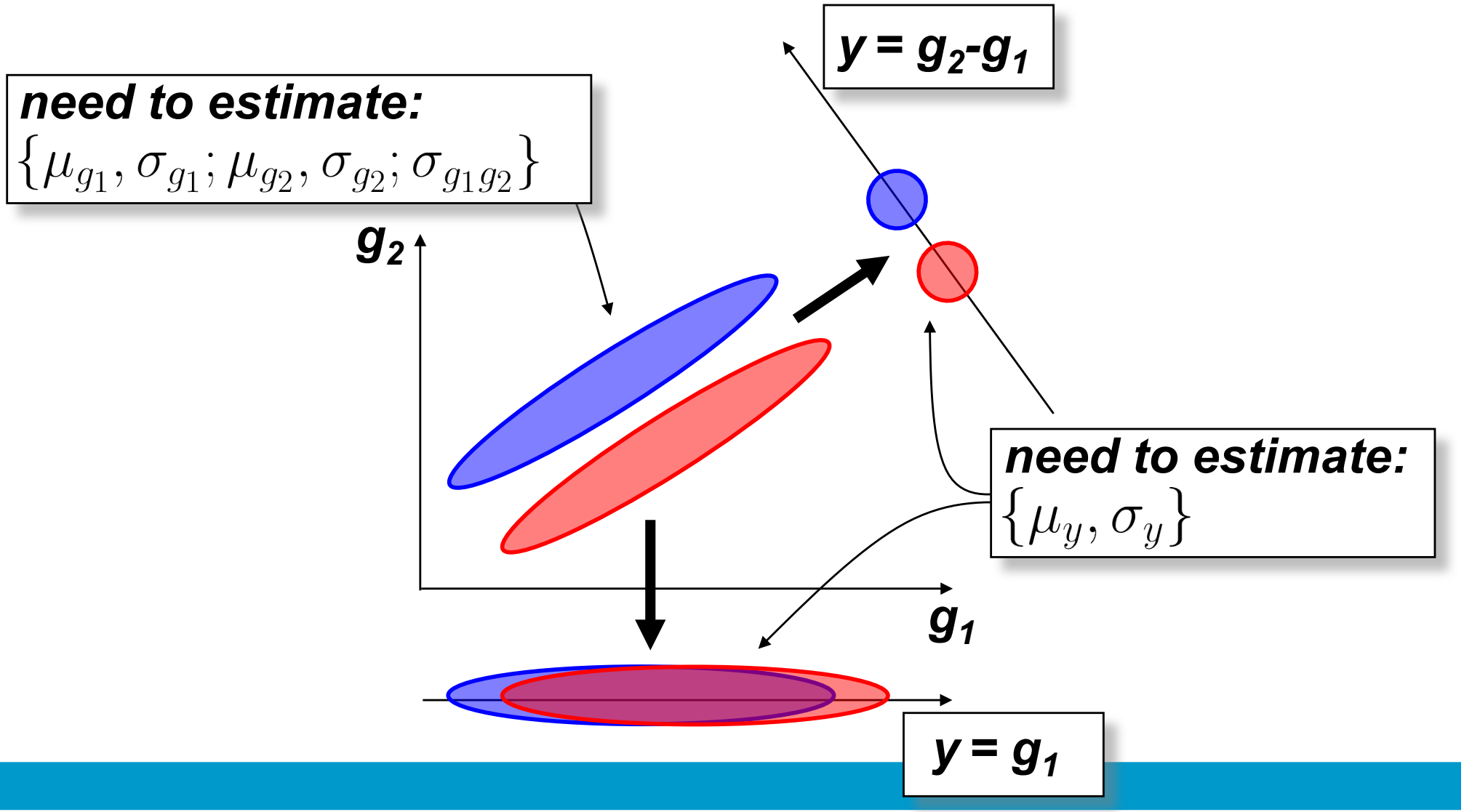
Dimensionality reduction (6)

Transform high-dimensional data to data of lower dimensionality, whilst *preserving the structure* in the original data as good as possible:

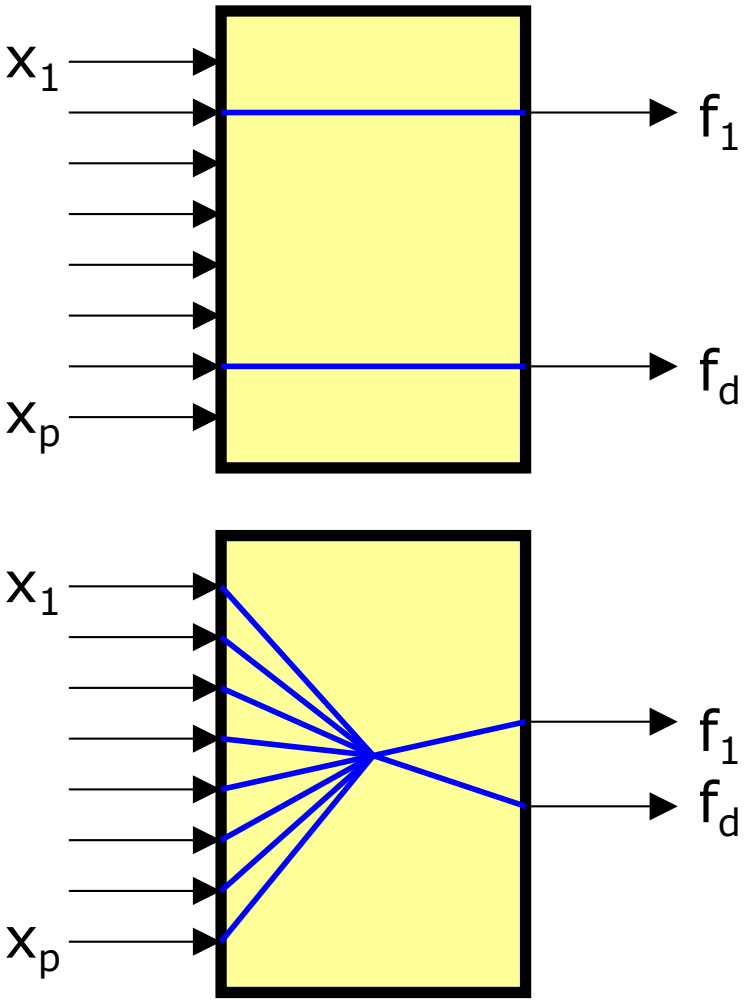


$$y_i = f(X)$$

Projecting to lower dimensions

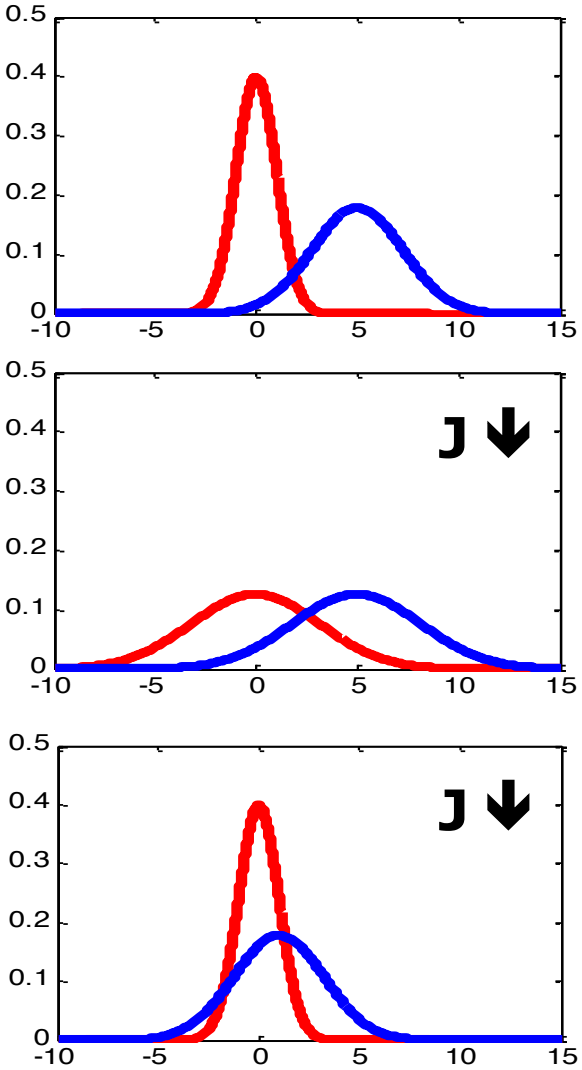


Feature reduction



- **Find most discriminating features**
- **Feature filtering (selection)**
  - Select *d* features out of *p* features
  - Interpretable, but, expensive and approximate
- **Feature mapping (extraction)**
  - Map *p* features to *d* features
  - Fast and optimal, but, still needs all original features
- **More advanced techniques**
  - Multi-variate selection criteria
  - Search algorithms
  - Supervised mappings
  - Non-linear mapping

Feature filtering

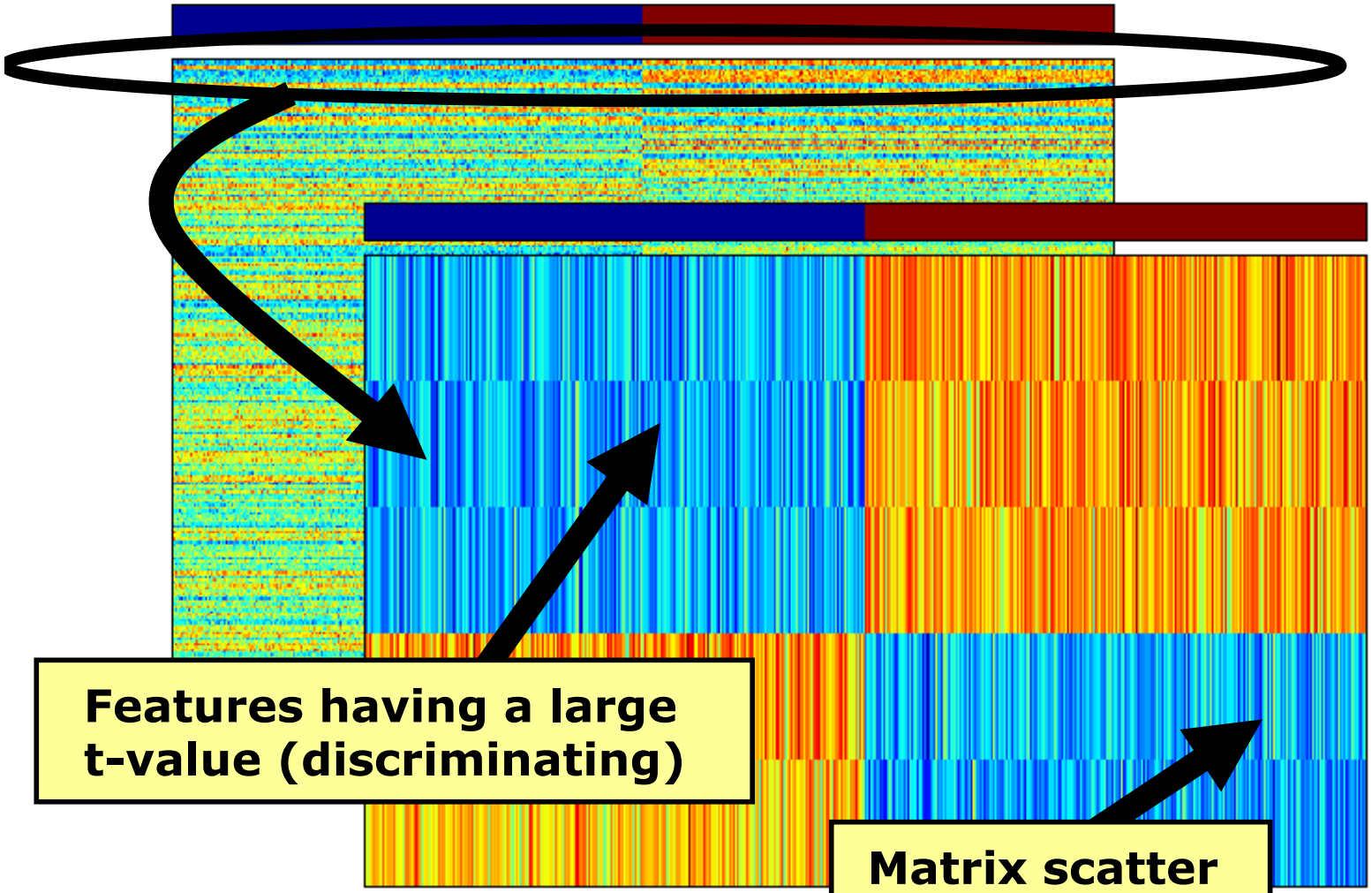


- **Define criterion *J* that expresses *separability* of classes**
- **T-test**
  - Test on difference in mean assuming Gaussian distribution

$$t = \frac{\mu_A - \mu_B}{\sqrt{\frac{\sigma_A^2}{n_A} + \frac{\sigma_B^2}{n_B}}}$$

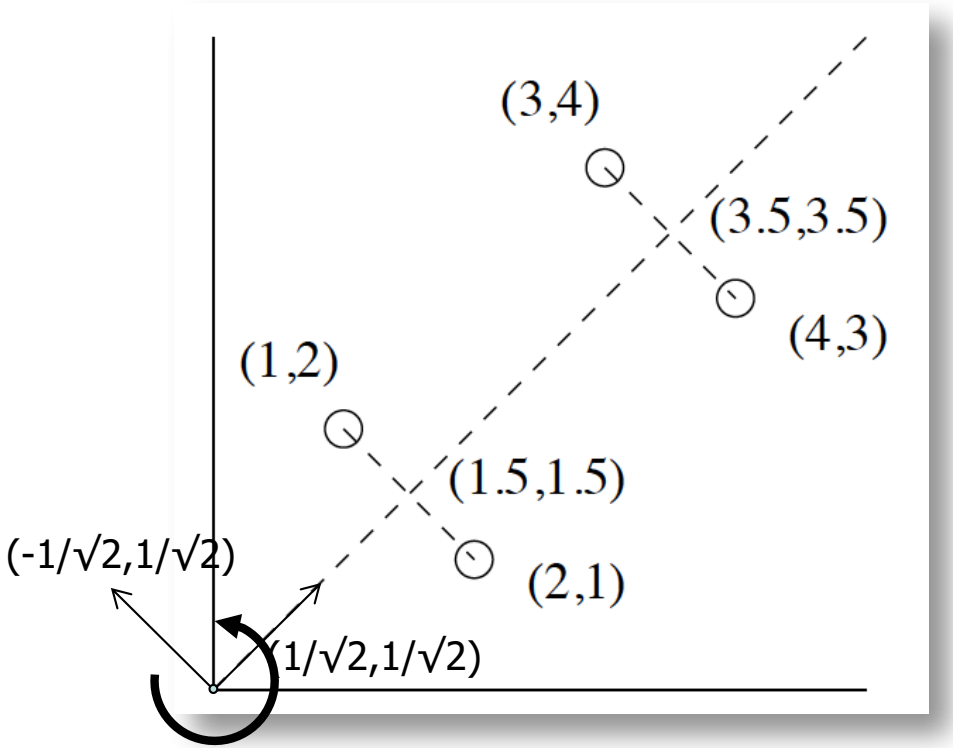
- **Feature filtering**
  - Rank individual features (on *J*)
  - Select the top-N features (scoring best on the criterion *J*)

Example: Feature filtering

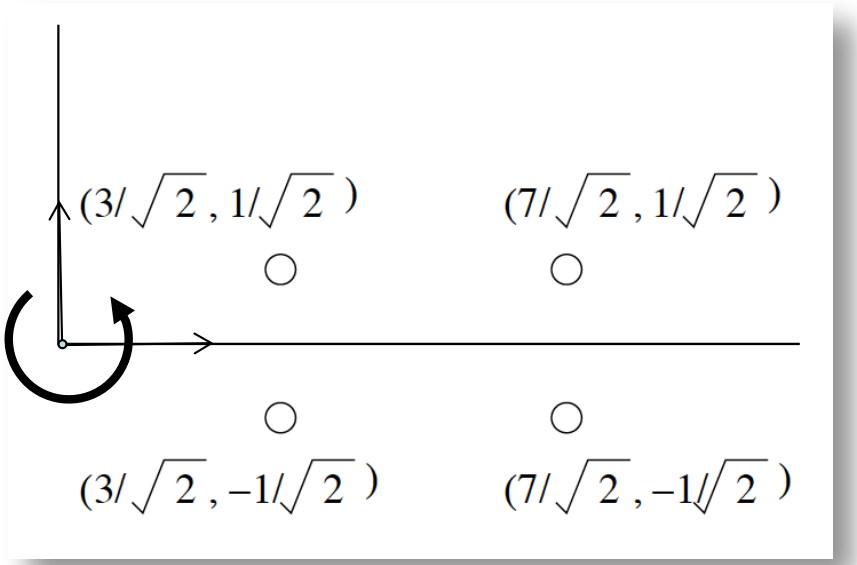


Principal component analysis

PCA is a *linear* techniques to reduce dimensions

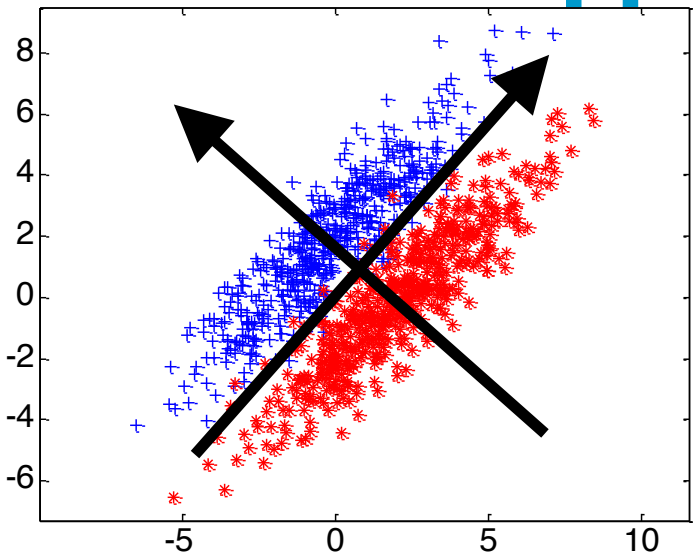


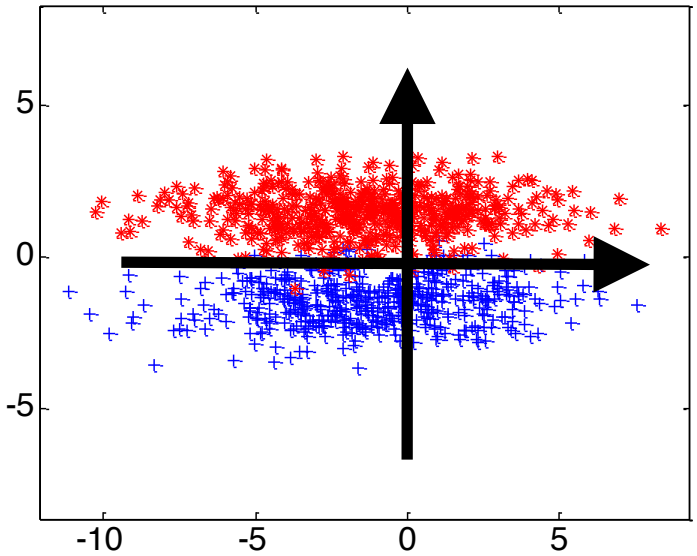
Rotate coordinate system such that variation in data is captures best



Project data on new coordinate system

Feature Mapping: PCA





- **Decorrelate data**
  Find rotation (& translation) of the space such that the data does not show correlations (linear relations)
- **Principal Component Analysis**
  Eigenvalue decomposition of the covariance matrix

$$Y = Xw \quad (\textit{rotation of the data})$$

$$Cov(Y) = I \quad (\textit{mapped data no correlation})$$

$$Cov(Y) = \left((Xw)^T(Xw)\right) = \left(w^T X^T\right)(Xw) = I$$

$$X^T X = \left(w^T\right)^{-1}(w)^{-1} = \left(ww^T\right)^{-1}$$

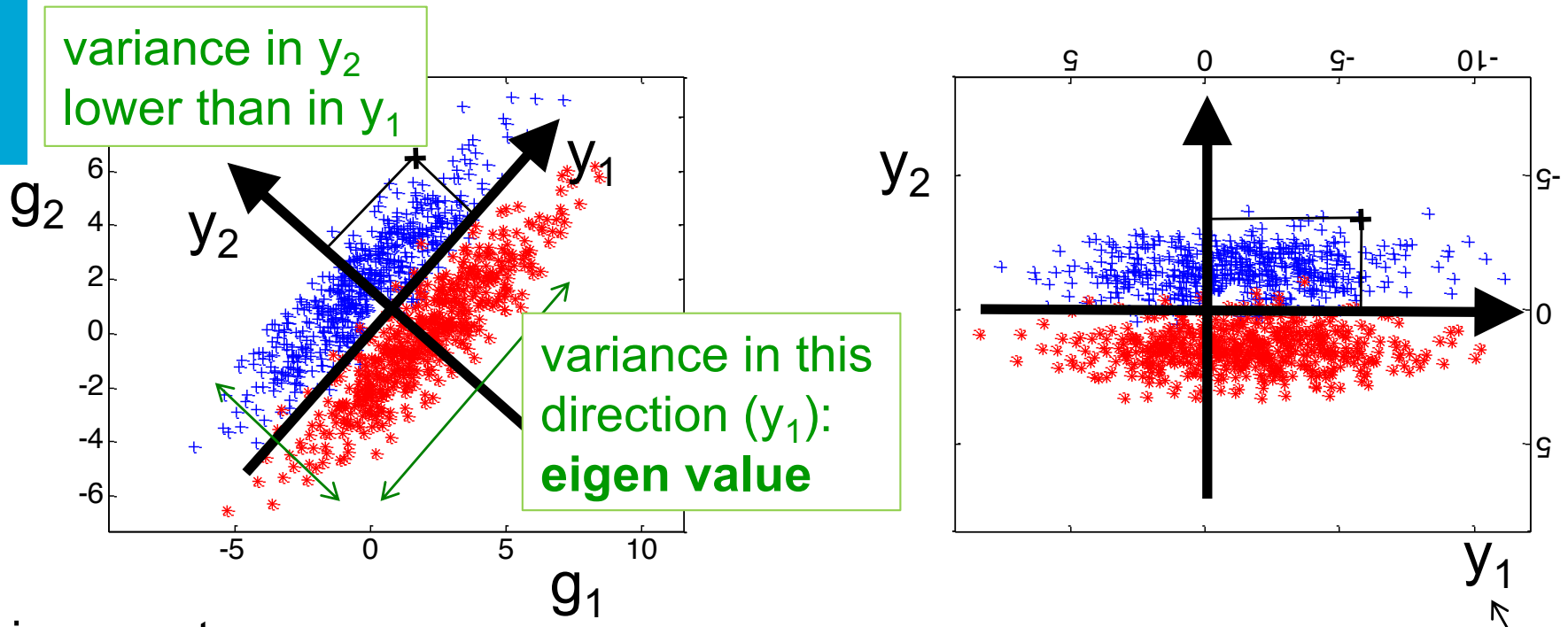
$$\left(X^T X\right)^{-1} = \left(\left(ww^T\right)^{-1}\right)^{-1} = ww^T$$

$$ww^T = \left(Cov(X)\right)^{-1} = \left(\Sigma_X\right)^{-1} = \left(E \Lambda E^T\right)^{-1} = E\Lambda^{-1}E^T$$

$$ww^T = \left(E\Lambda^{-\frac{1}{2}}\right)\left(\Lambda^{-\frac{1}{2}}E^T\right) = \left(E\Lambda^{-\frac{1}{2}}\right)\left(E\Lambda^{-\frac{1}{2}}\right)^T$$

$$Y = XE\Lambda^{-\frac{1}{2}}$$

PCA transforms the space



- eigen vector
- principal component
- eigen gene

$$y_1 = w_{11}g_1 + w_{12}g_2$$
$$y_2 = w_{21}g_1 + w_{22}g_2$$

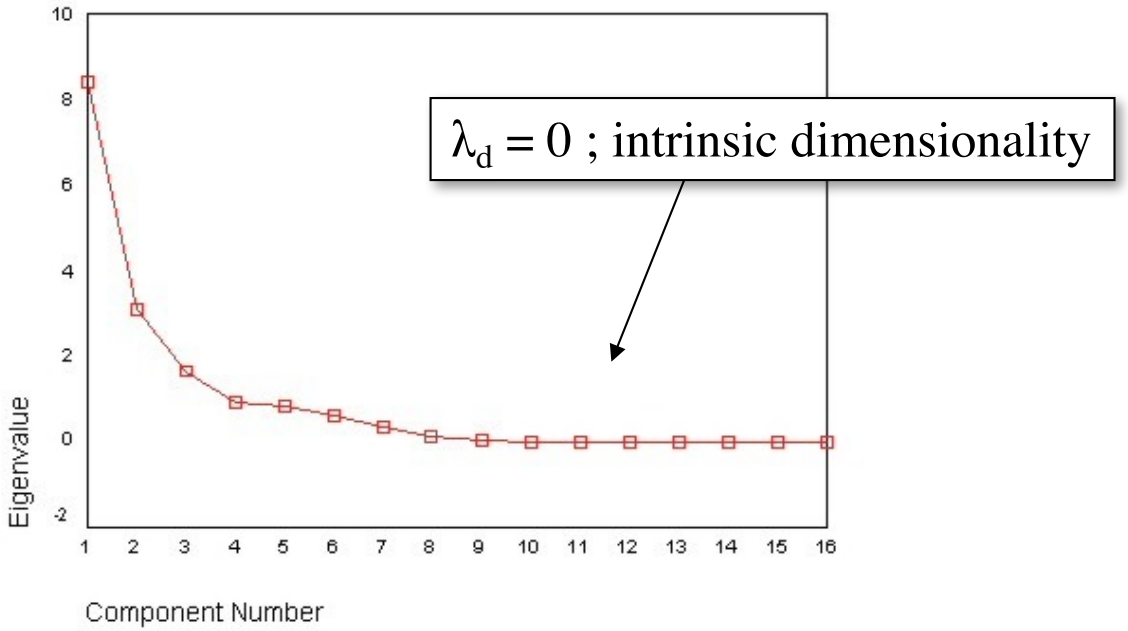
loading factor

(importance of gene in contributing to variance)

component showing the most variation

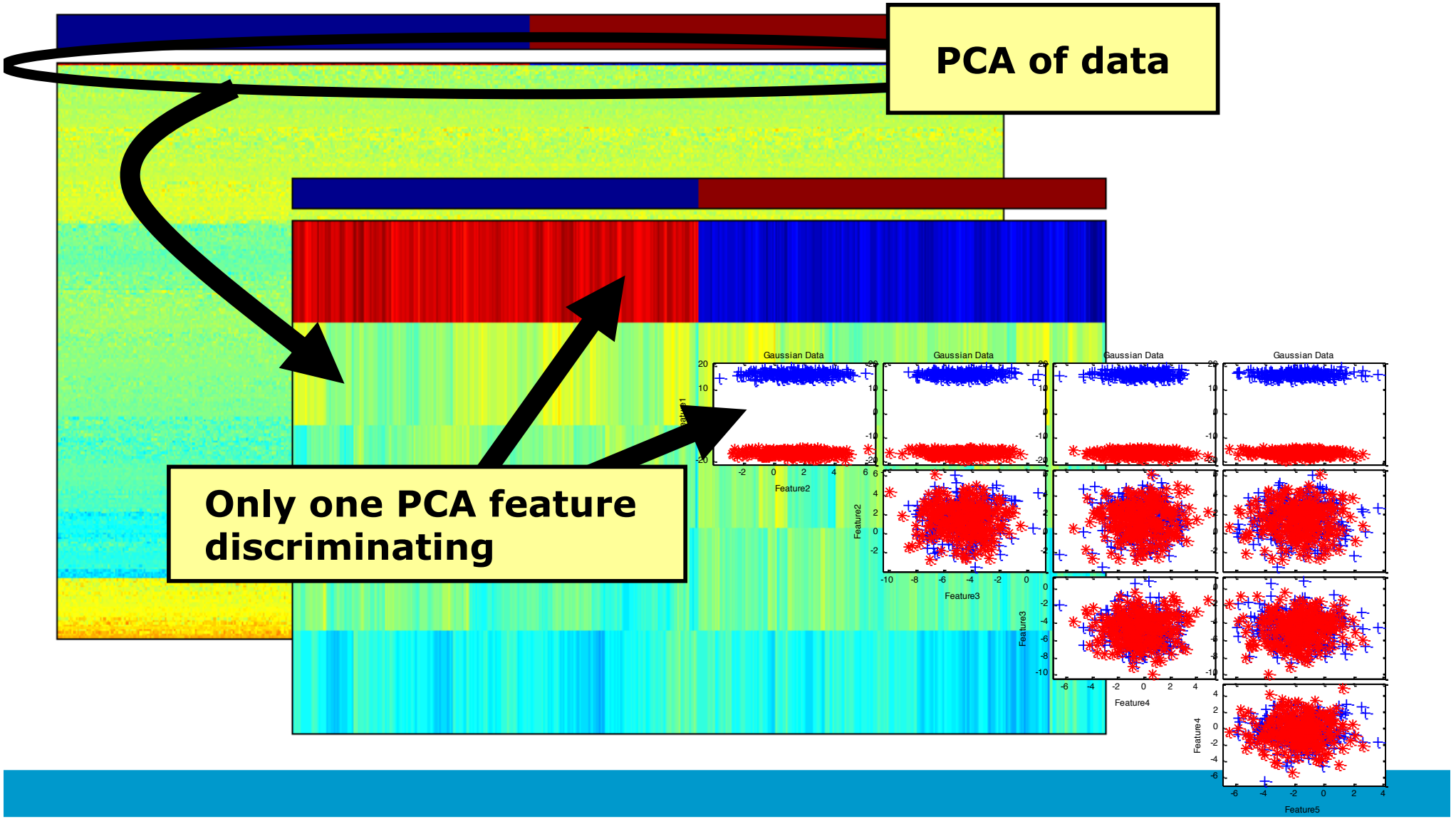
PCA scree plot

- *Scree plot* of eigenvalues shows amount of variance retained by the eigenvectors (*principal components, PCs*):



- First *K* PCs explain $\frac{\sum_{d=1}^{K} \lambda_d}{\sum_{d'=1}^{D} \lambda_{d'}} \times 100\%$ of variance

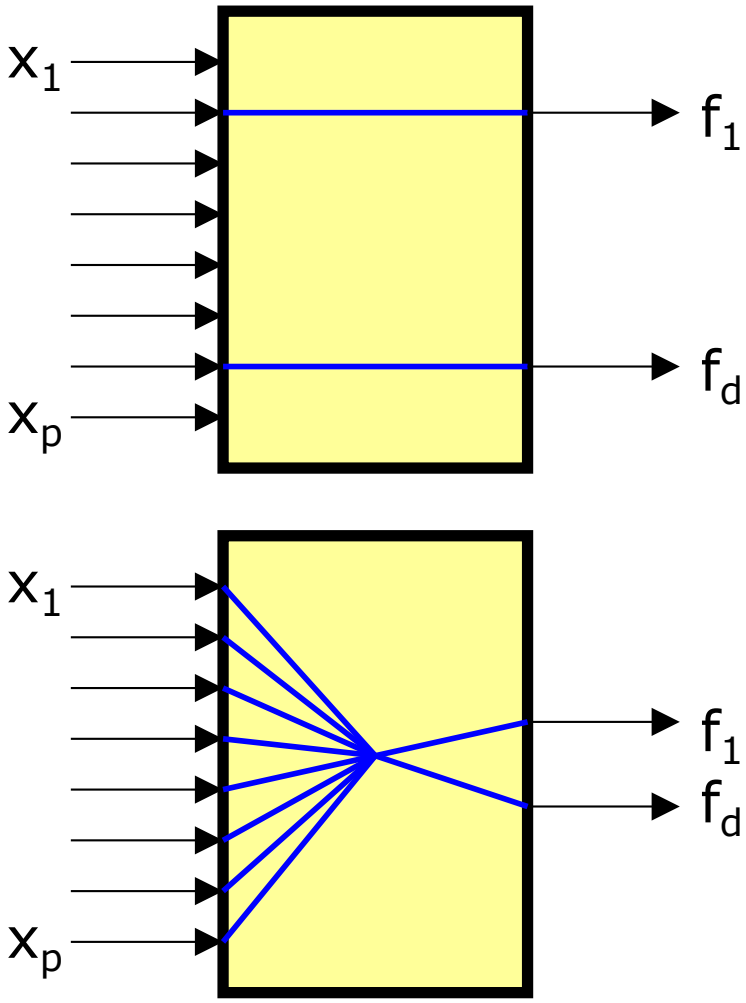
Example: PCA



The Delft Bioinformatics Lab

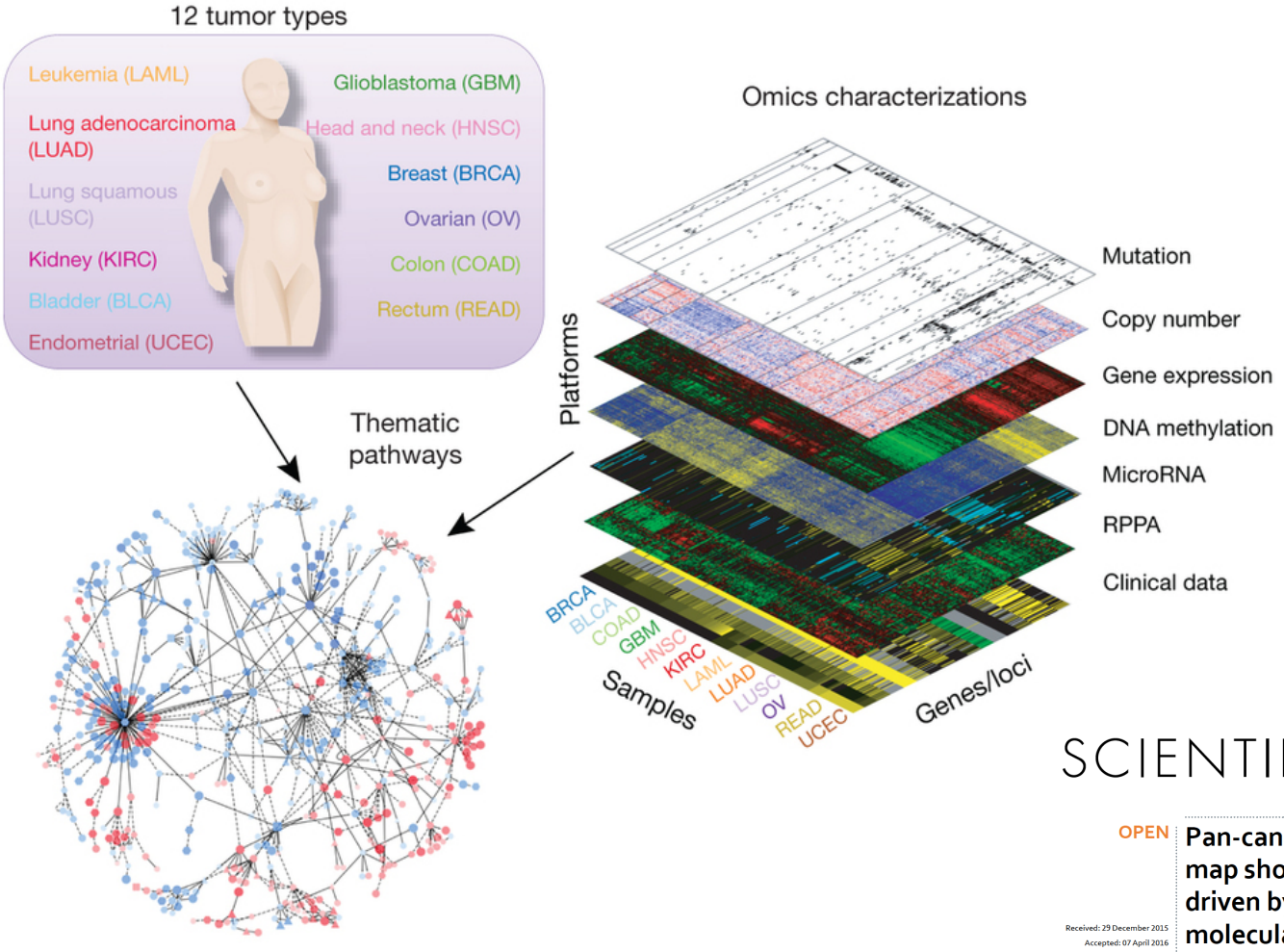
TU Delft

Feature reduction (2)



- **But: How many features to use ?**
- **Naïve approach**
  - Feature filtering: threshold the t-values (retained separability)
  - Feature mapping: threshold the eigenvalues of the covariance matrix (variance retained)
- **Or, more advanced**
  - based on some performance that can be reached on the basis of the final *d* features

Example: TCGA data (1)



SCIENTIFIC REPORTS

OPEN

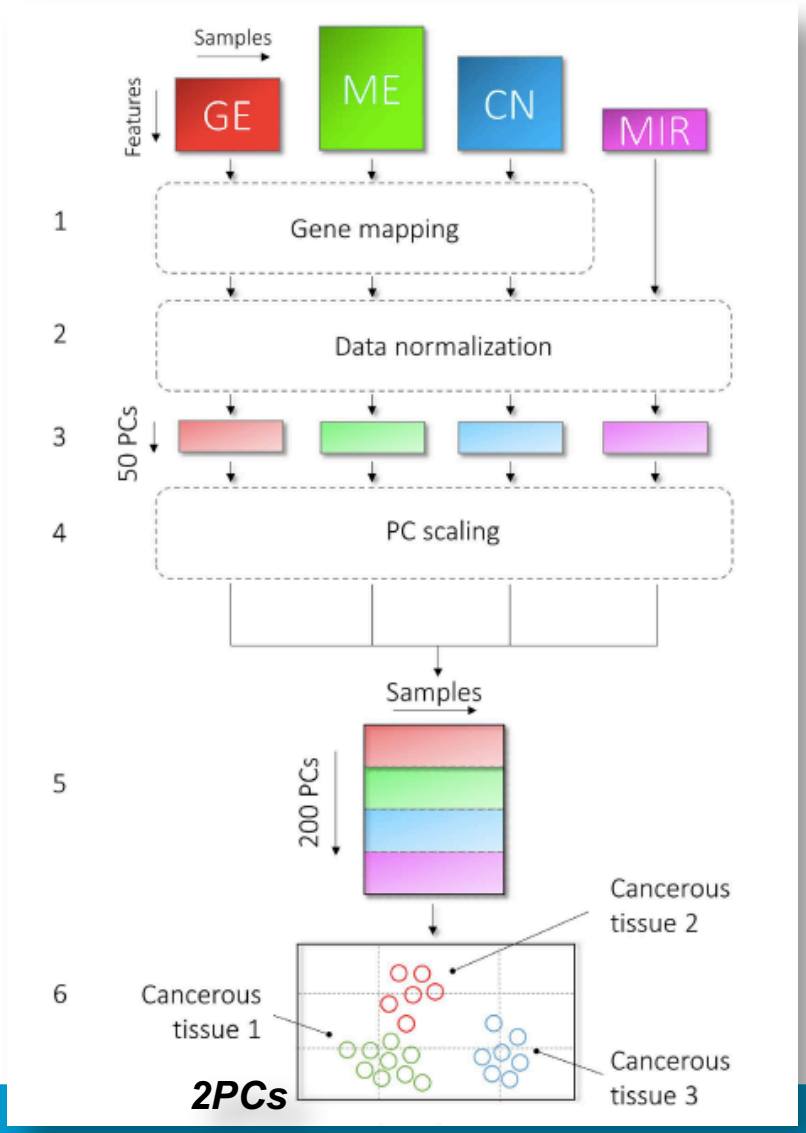
Pan-cancer subtyping in a 2D-map shows substructures that are driven by specific combinations of molecular characteristics

Received: 29 December 2015
Accepted: 07 April 2016
Published: 25 April 2016

Erdogan Taskesen[1], Sjoerd M. H. Huisman[1,2], Ahmed Mahfouz[1,2], Jesse H. Krijthe[1], Jeroen de Ridder[1], Anja van de Stolpe[3], Erik van den Akker[1], Wim Verheagh[3] & Marcel J. T. Reinders[1]

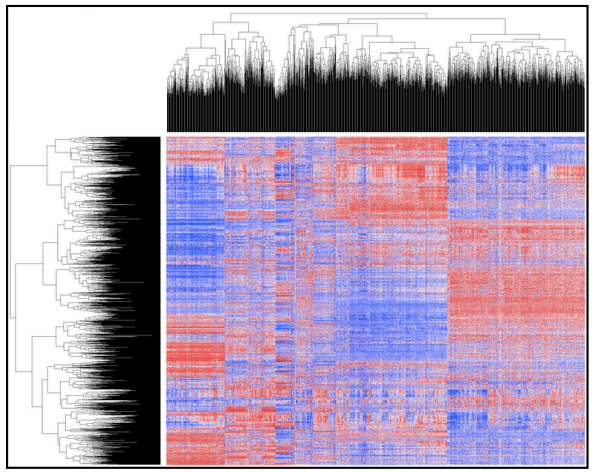
Taskesen, Reinders et al. Scientific Reports. 2016.

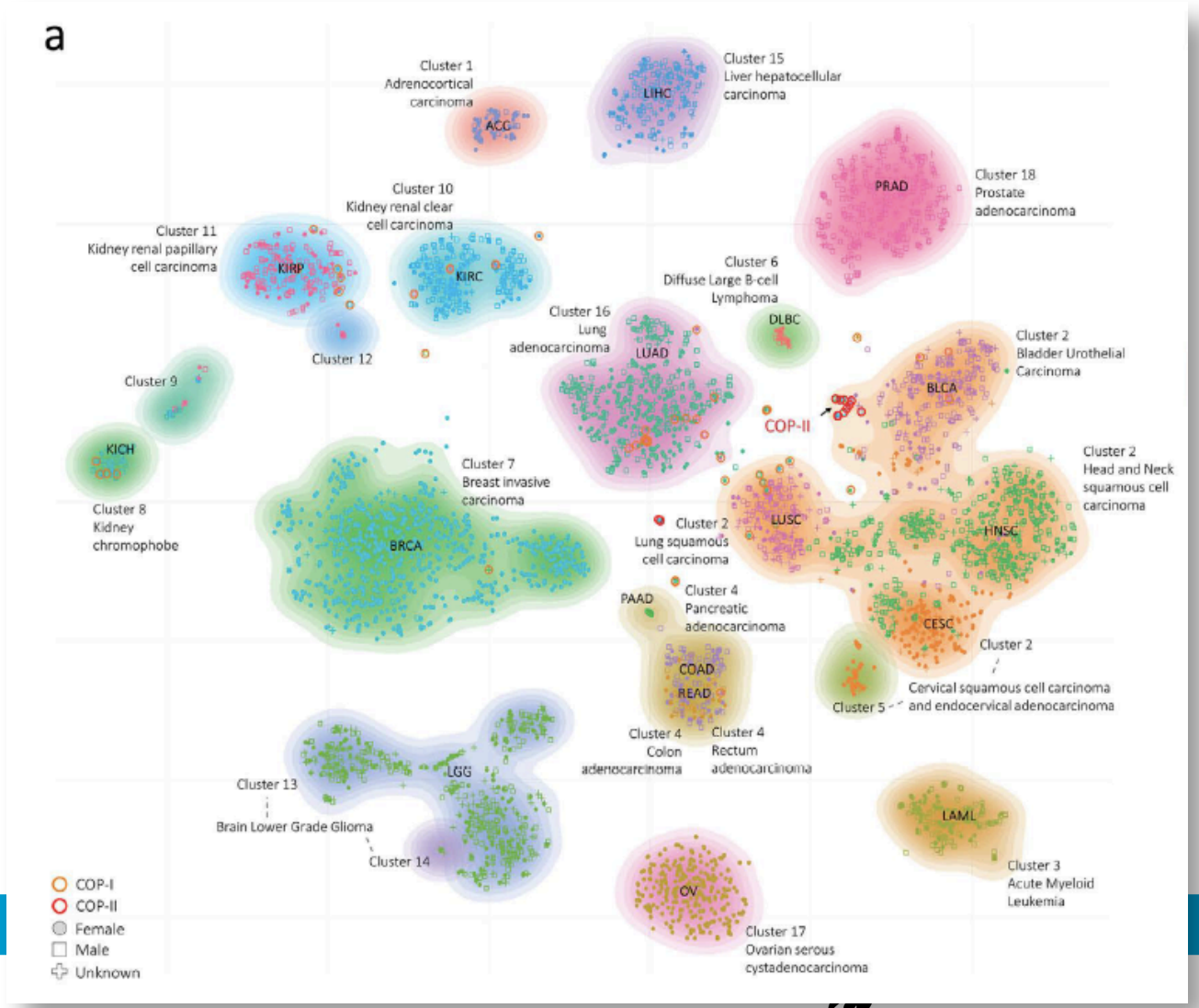
Example: TCGA data (2)



Dimension reduction (4)

Visualize in scatter plot





The Delft Bioinformatics Lab

TU Delft

Dimension reduction: Summary

- Dimensionality reduction can be helpful to visually inspect data and to remove uninformative information
- There are two ways: selection and extraction
- Principal Component Analysis is a linear approach for feature extraction which removes correlations between the features
- First eigenvectors are the most important, and loading factors indicate which (original) features contribute to these PCs

Thank you!

TU Delft

Delft University of Technology

References/Reading material

Simon, R.M., Korn, E.L., McShane, L.M., Radmacher, M.D., Wright, G.W., Zhao, Y. *Design and Analysis of DNA Microarray Investigations.* Springer. 2003.

D'haeseleer, P. *How does gene expression clustering work?* Nature Biotechnology, 23, 1499 - 1501 (2005)
https://www.nature.com/nbt/journal/v23/n12/full/nbt1205-1499.html

Ringer M. *What is principal component analysis?* Nature Biotechnology, 6, 303 - 304 (2008)
https://www.nature.com/nbt/journal/v26/n3/full/nbt0308-303.html