BIOINFORMATIK II
Übung

MEDIZINISCHE
UNIVERSITÄT
INNSBRUCK

Dietmar Rieder

https://icbi.i-med.ac.at/bioinformatics2_ex/

Termine

Day	Date	Room	Group	Time	Type
Fri	06.03.2026	FP3_01_200	Group I	13:00-14:45	Exercise 1
Fri	06.03.2026	FP3_01_200	Group II	15:00-16:45	Exercise 1
Mon	09.03.2026	FP3_01_200	Group II	13:00-14:45	Exercise 1
Mon	09.03.2026	FP3_01_200	Group I	15:00-16:45	Exercise 1
Tue	10.03.2026	FP3_01_200	Group I	13:00-14:45	Exercise 2
Tue	10.03.2026	FP3_01_200	Group II	15:00-16:45	Exercise 2
Wed	11.03.2026	FP3_01_200	Group II	13:00-14:45	Exercise 2
Wed	11.03.2026	FP3_01_200	Group I	15:00-16:45	Exercise 2
Mon	16.03.2026	FP3_01_200	Group I	13:00-14:45	Exercise 3
Mon	16.03.2026	FP3_01_200	Group II	15:00-16:45	Exercise 3
Tue	17.03.2026	FP3_01_200	Group II	13:00-14:45	Exercise 3
Tue	17.03.2026	FP3_01_200	Group I	15:00-16:45	Exercise 3
Wed	18.03.2026	FP3_01_200	Group I	13:00-14:45	repetition
Wed	18.03.2026	FP3_01_200	Group II	15:00-16:45	repetition

Übungsziele

- **Selbstständiges** Lösen von biolgischen Fragestellungen mit Hilfe von Bioinformatik Werkzeugen
- Arbeiten mit Genexpressions Daten
- Kennenlernen von Clustering Algorithmen
- Finden von TFBS mit Hilfe von PWMs und online tools
- Netzwerkanalyse von Omics Daten
- Klassifikation mittels Gene Ontology

Organisation

- 4 Übungsblöcke zu je 4 Stunden

- Kurze Einführung am Beginn eines jeden Blocks

- Erarbeiten der Übungsziele an Hand von Beispielen am Rechner

- Protokoll:
  - 1 Protokoll / 2 Studierende
  - Elektronisch als PDF
  - Abgabe bis spätestens 20. April 2026
  - Siehe auch Guidelines auf der Übungswebseite

Übung I

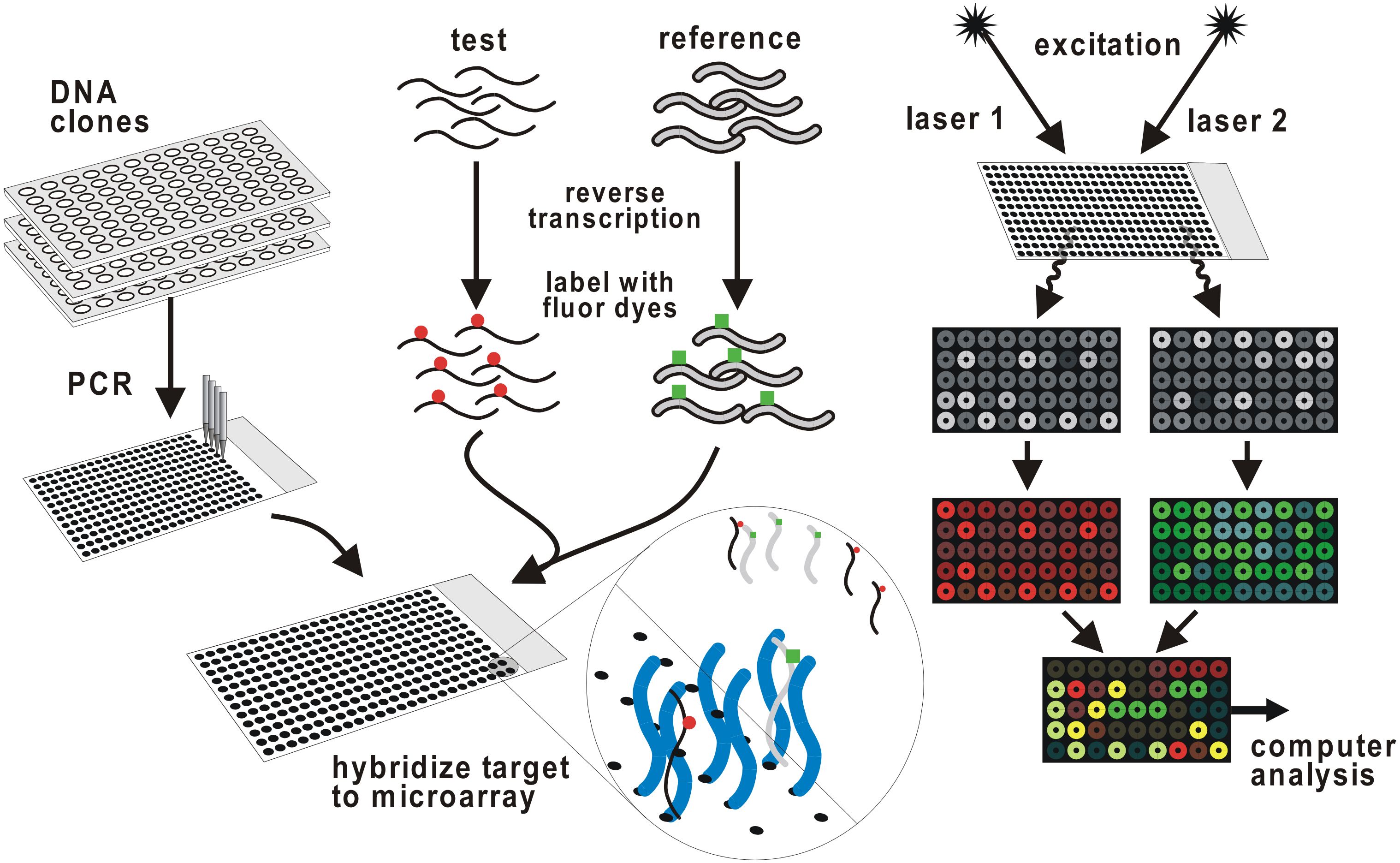
Einführung

Microarray Genexpressionsanalyse Clustering

Genome-wide gene expression analysis

- DNA microarrays consist of thousands of individual gene sequences bound to closely spaced regions on the surface of a glass microscope slide
- DNA microarrays allow the simultaneous analysis of the expression of thousands of genes
- Analysis of the complete transcriptional program of an organism during specific physiological response or developmental processes

Microarray Schema



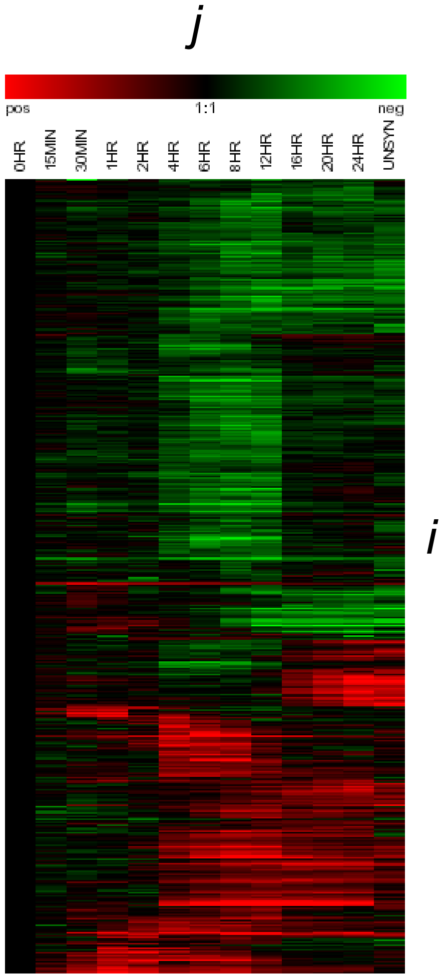
Data format

[*i* x *j*] Matrix of

i … Genes and

j … Experiments

$$x_{ij} = \log_2 \frac{C5_{ij}}{C3_{ij}}$$



$C5_{ij}$ …Cy-5 of gene *i* in microarray experiment *j*

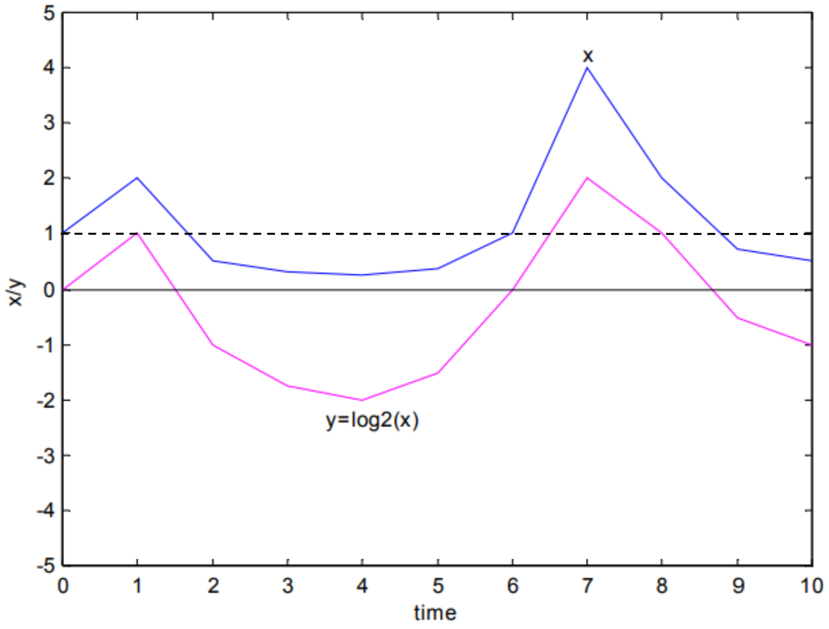
$C3_{ij}$ …Cy-3 of gene *i* in microarray experiment *j*

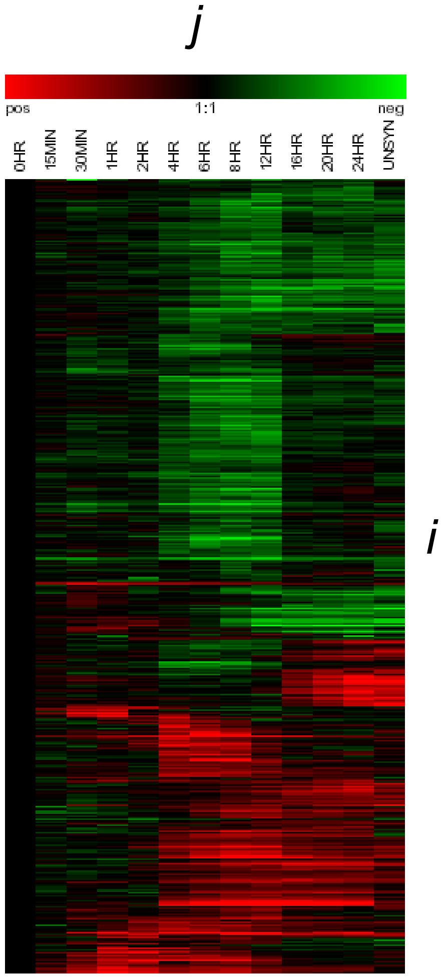
Data format

$C5_{ij}$ …Cy-5 of gene *i* in microarray experiment *j*

$C3_{ij}$ …Cy-3 of gene *i* in microarray experiment *j*

$$x_{ij} = \log_2 \frac{C5_{ij}}{C3_{ij}}$$

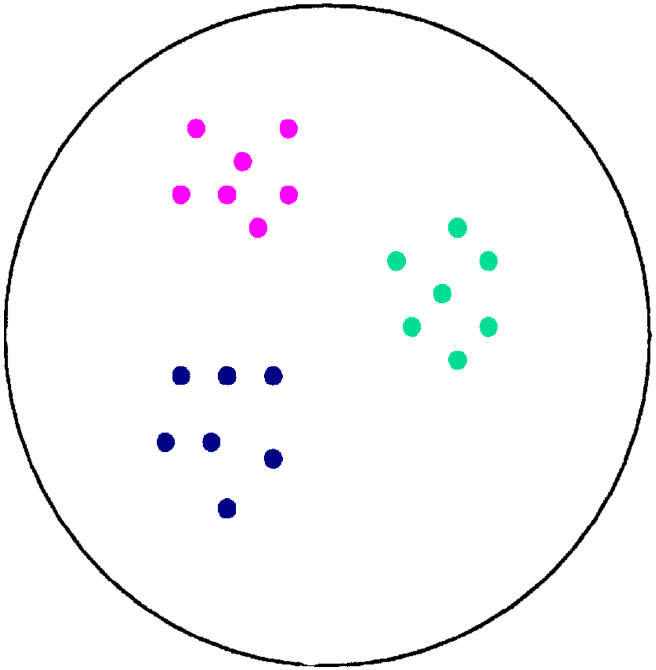




Clustering

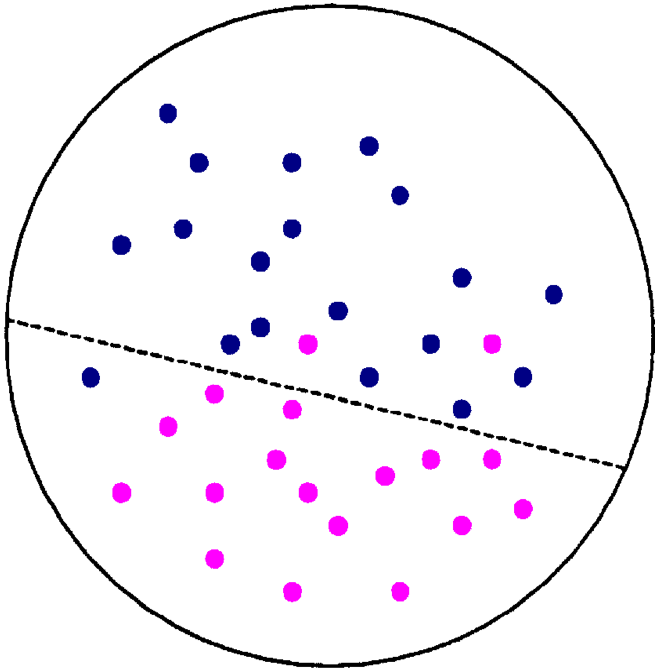
- Functionally related genes often are co-expressed
  - Grouping genes with similar expression levels can reveal the functional context of those which were previously uncharacterized.
- In general, relationship between co-expression and co-regulation is very common.
  - Co-expressed genes can reveal regulatory mechanisms.
  - Pathway activity

Clustering



Unsupervised

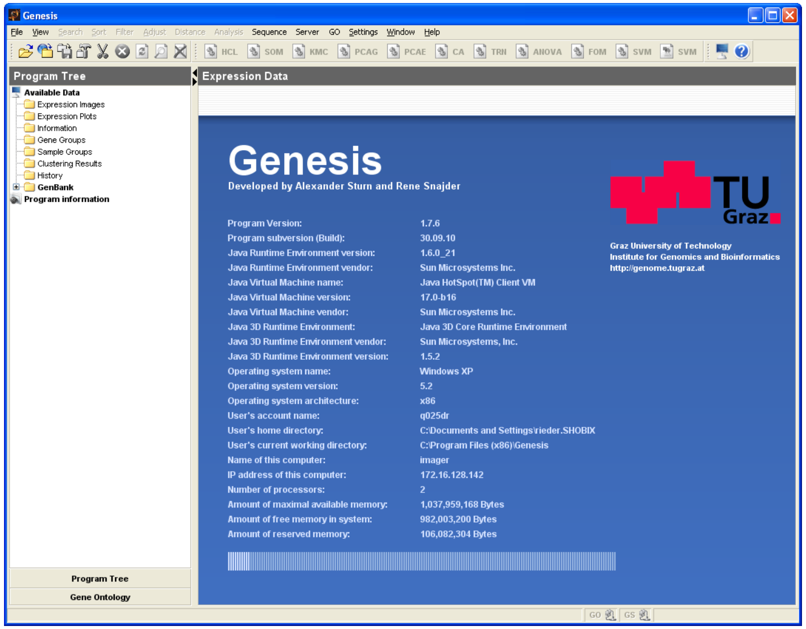
(k-means, HCL, …)



Supervised

(SVM…)

Genesis



Genesis

Developed by Alexander Sturn and Rene Snajder

Program Version:	1.7.6
Program subversion (Build):	30.09.10
Java Runtime Environment version:	1.6.0_21
Java Runtime Environment vendor:	Sun Microsystems Inc.
Java Virtual Machine name:	Java HotSpot(TM) Client VM
Java Virtual Machine version:	17.0-b16
Java Virtual Machine vendor:	Sun Microsystems Inc.
Java 3D Runtime Environment:	Java 3D Core Runtime Environment
Java 3D Runtime Environment vendor:	Sun Microsystems, Inc.
Java 3D Runtime Environment version:	1.5.2
Operating system name:	Windows XP
Operating system version:	5.2
Operating system architecture:	x86
User's account name:	q025dr
User's home directory:	C:\Documents and Settings\rieder.SHOBIX
User's current working directory:	C:\Program Files (x86)\Genesis
Name of this computer:	imager
IP address of this computer:	172.16.128.142
Number of processors:	2
Amount of maximal available memory:	1,037,959,168 Bytes
Amount of free memory in system:	982,003,200 Bytes
Amount of reserved memory:	106,082,304 Bytes

Graz University of Technology
Institute for Genomics and Bioinformatics
http://genome.tugraz.at

- Gene expression analysis tool
  - data filtering
  - data adjustment
  - clustering algorithms:
    - HCL (agglomerative)
    - SOM (paritioning)
    - K-means (partitioning)
    - PCA
    - ….

A. Sturn *et al.*, *Bioinformatics,* 18:207-208, 2002

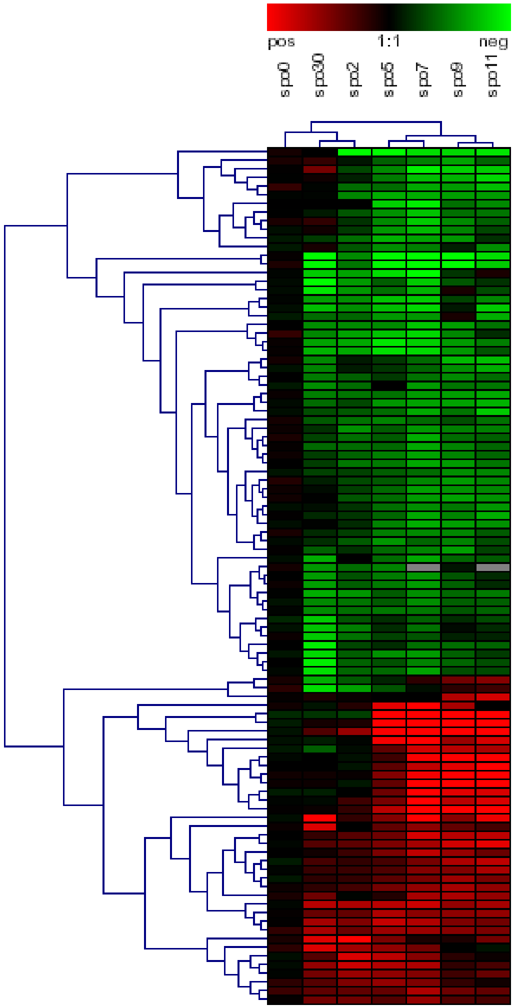
Hierarchical Clustering

- Reorders the vectors regarding similarity (i.e. distances)

- Distances are encoded in dendrogram (tree)
- Objects linked together according to a linkage rule

- Unsupervised

- computational intensive, memory intensive



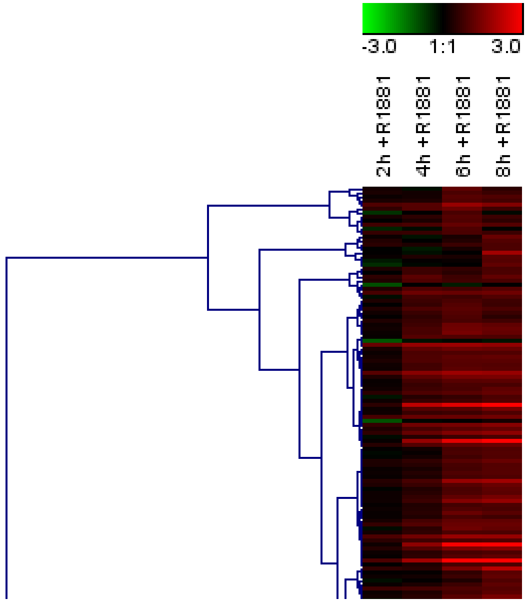
Genesis: default distance measurements

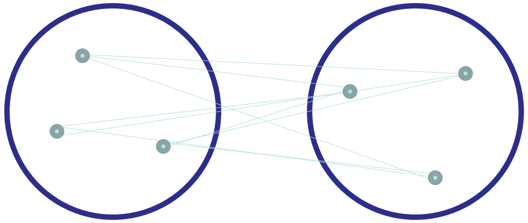
Clustering algorithm	Default similarity distance measurement
Hierarchical Clustering Genes	Euclidian Distance
Hierarchical Clustering Experiments	Euclidian Distance
k-means Clustering	Euclidian Distance
Self Organizing Maps (SOM)	Euclidian Distance
PCA Genes	Covariance
PCA Experiments	Covariance
Support Vector Machines training	Average dot product
Support Vector Machines classification	Average dot product

Implemented:
Pearson Correlation, Pearson Uncentered, Pearson Squared, Cosine Correlation, Covariance, Euclidean Distance, Average Dot Product, Manhattan Distance Chebychev Distance, Mutual Information, Spearman Rank, Kendall’s Tau

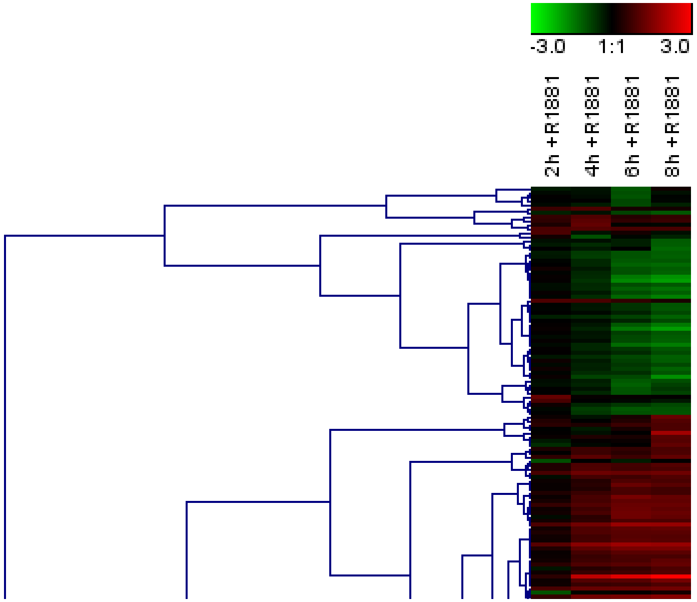
Different linkgage rules

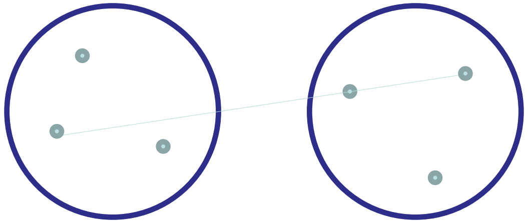
Average



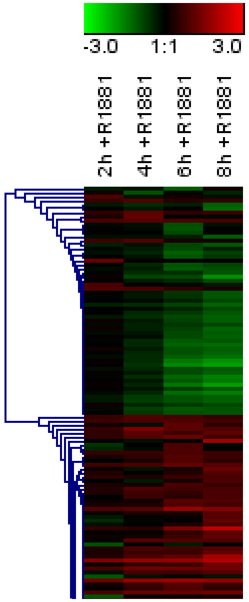


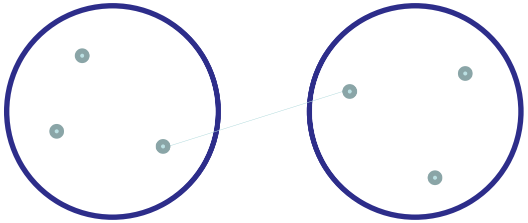
Complete



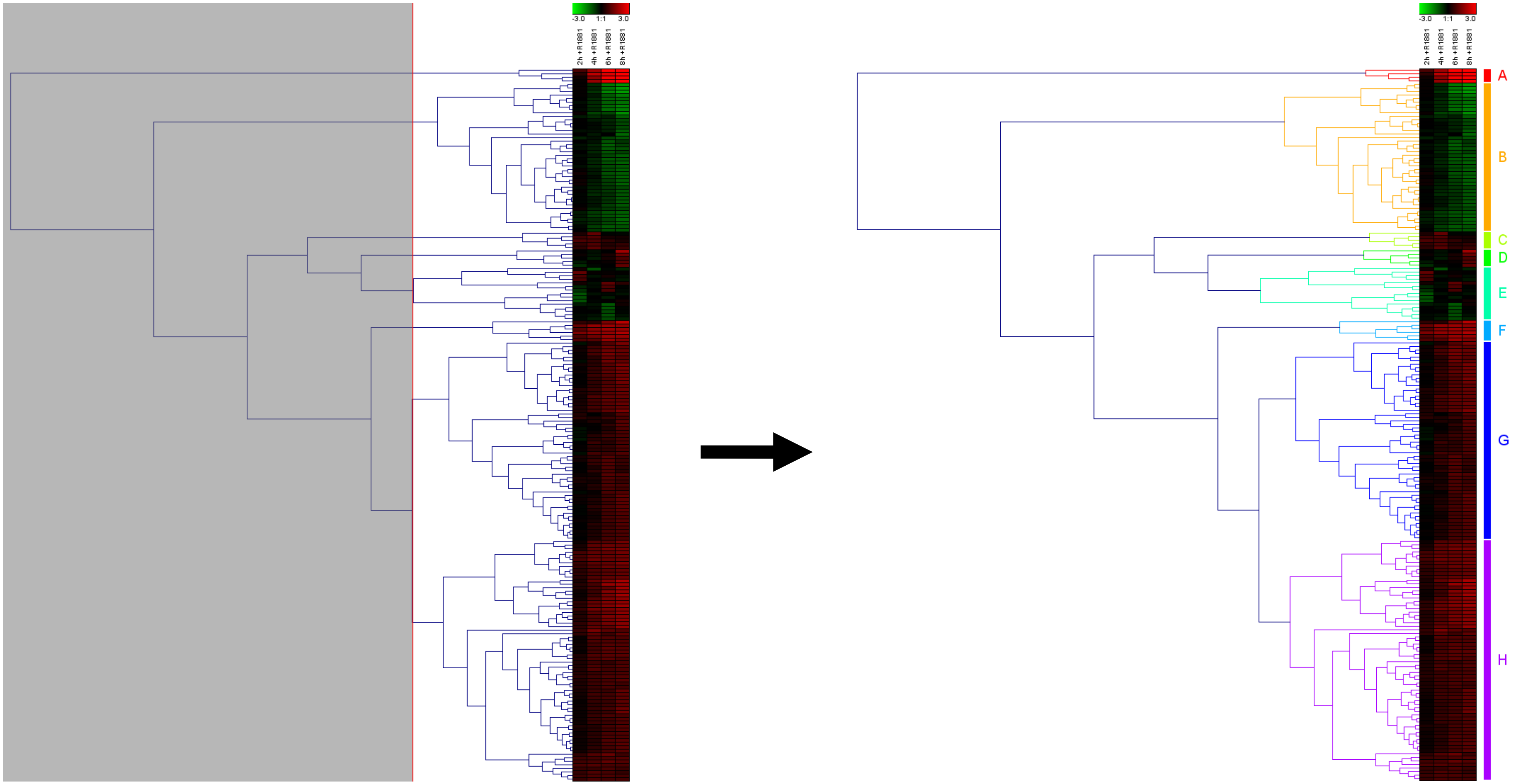


Single





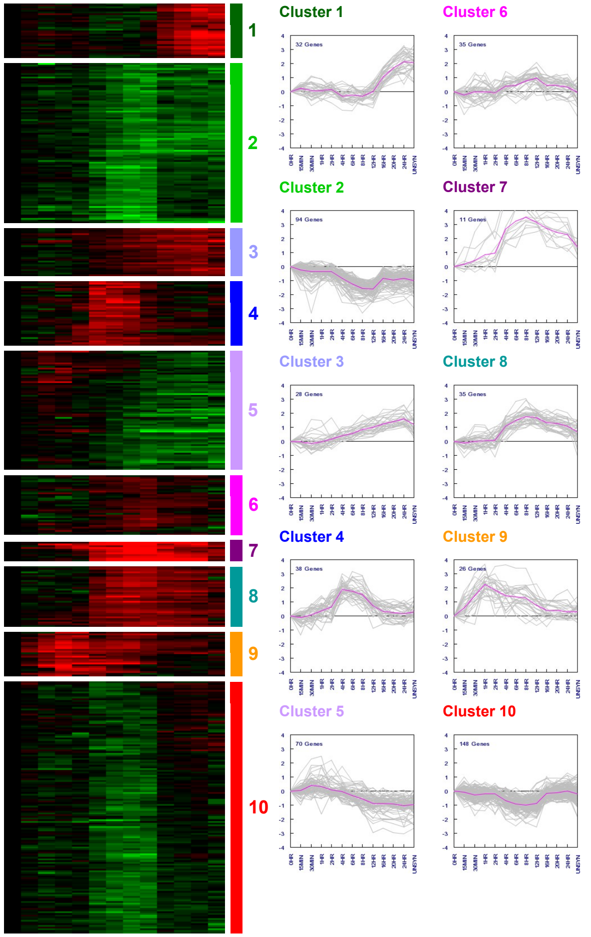
Define groups of genes by setting a threshold to select subtrees



K-Means Clustering

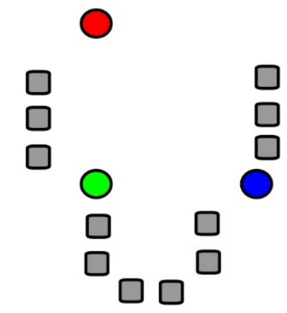
- Number of clusters has to be specified
- FOM allows for estimating the number of clusters

- Split data into ‘k’ partitions, each with an associated vector.
- Assign genes to partitions, and recalculate the vector associated with each partition as the centroid of its associated genes.
- Repeat until solution converges, or for a fixed number of iterations.



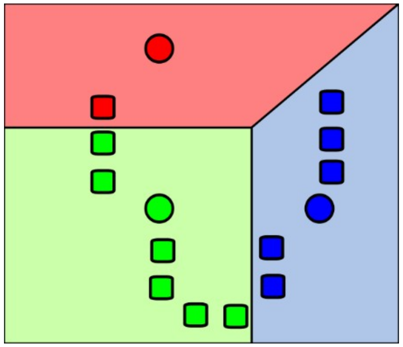
K-Means Clustering

1.



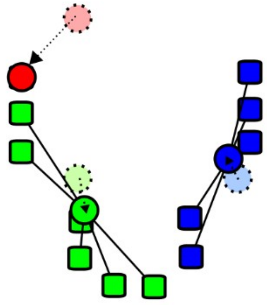
k initial „means“ (3)

2.



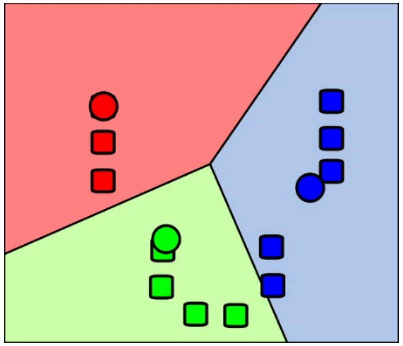
form *k* clusters by association with nearest mean

3.

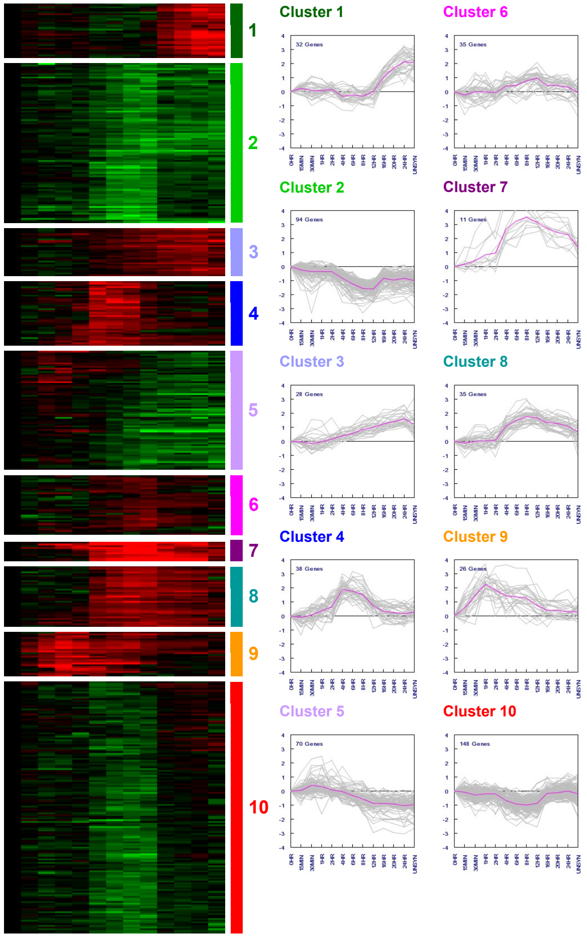


calculate centroid as new mean

4.



repeat 2 + 3 until convergence



By I, Weston.pace, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=2463053

PCA: principal component analysis

- PCA is a data reduction technique that allows to simplify multidimensional data sets into smaller number of dimensions (r<n)
- allows the identification of key variables (or combinations of variables) in a multidimensional data set that best explain the differences between observations

- **Principal Component Analysis can be used to retrieve the basic patterns of gene expression contained in a given study. It eliminates the noise part of the dataset and concentrates on the most variant aspects of the investigated observation**

PCA: how does it work

- **Standardize the data**: PCA requires that the variables are standardized so that they have a mean of 0 and a standard deviation of 1. This is important because PCA is sensitive to the scale of the variables.

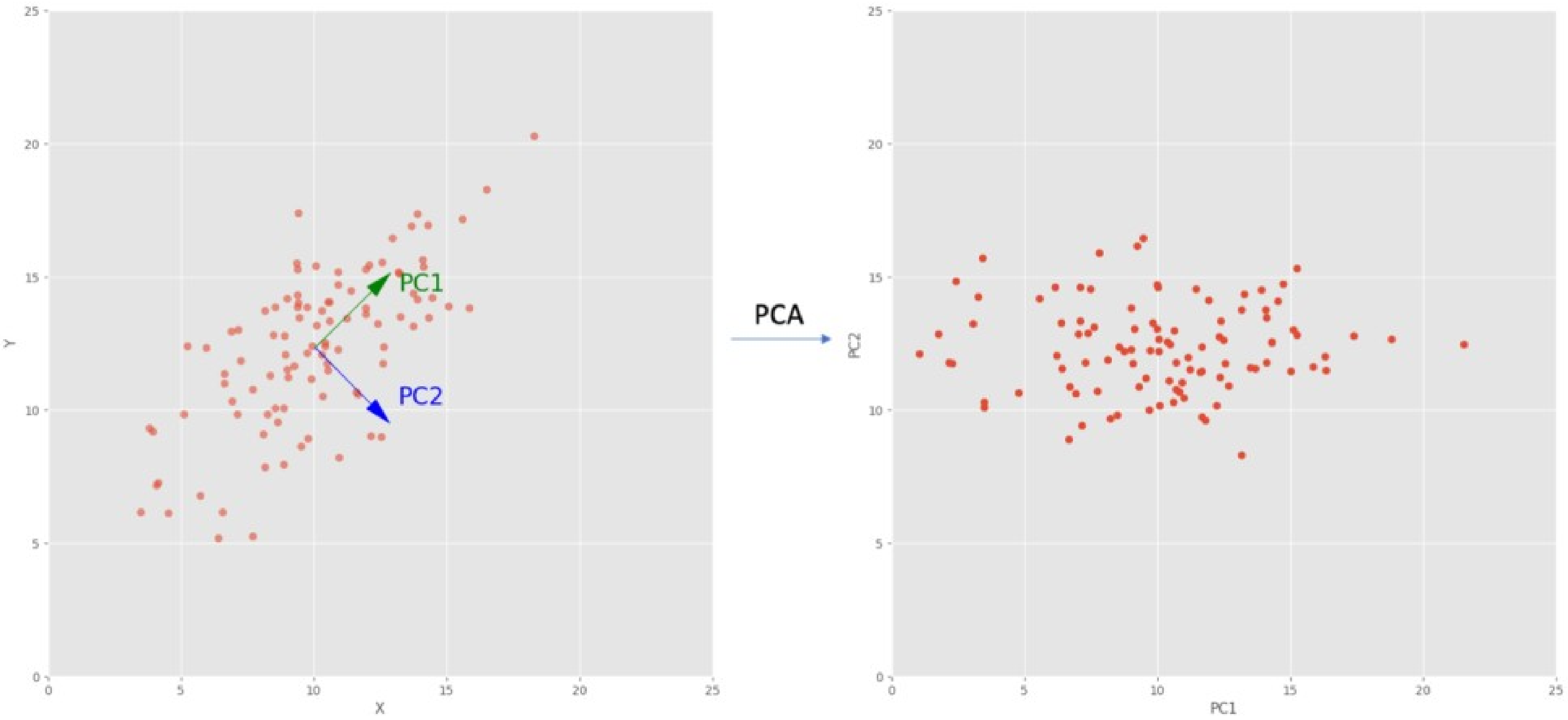
- **Compute the covariance matrix**: PCA is based on the covariance matrix of the variables. The covariance matrix shows how each variable is related to every other variable in the dataset (+ ~ same direction, - ~ opposite, 0 ~ unrelated).

- **Compute the eigenvectors and eigenvalues of the covariance matrix**: The eigenvectors represent the directions of maximum variance in the data, and the eigenvalues represent the magnitude of the variance in those directions.

- **Choose the principal components**: The principal components are chosen based on the highest eigenvalues. The first principal component is the direction of maximum variance in the data, the second principal component is the direction of the remaining maximum variance orthogonal to the first principal component, and so on.

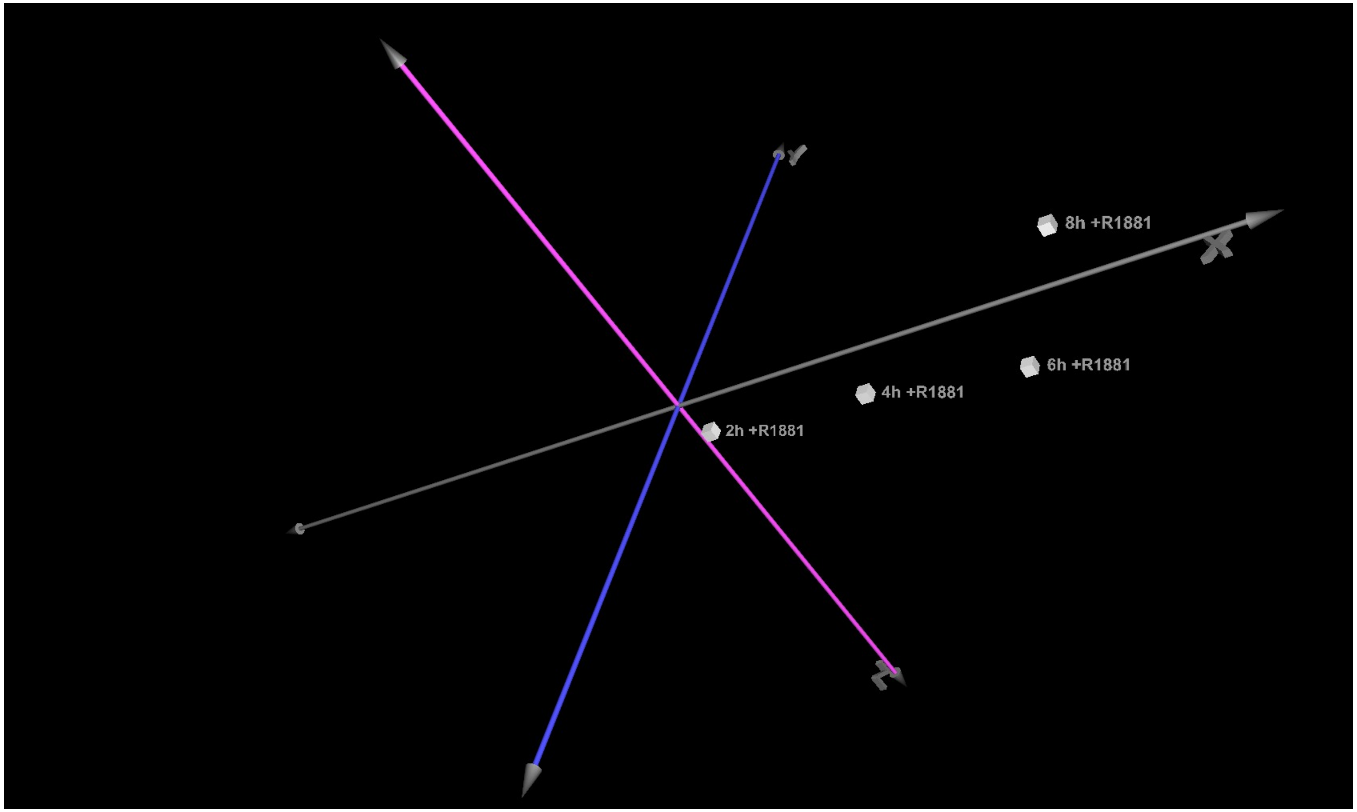
- **Compute the new data**: The original data can now be projected onto the principal components to create a new dataset with fewer dimensions. This new dataset will have uncorrelated variables and most of the variability in the original data will be retained.

PCA: how does it work



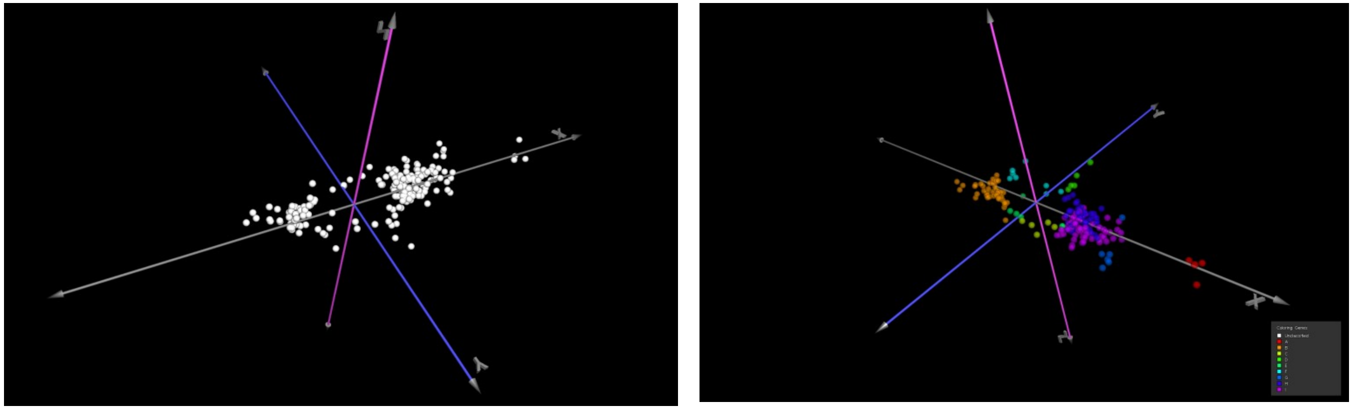
PCA Experiments

- investigate relations between different experiments
- determine the most distant or most related experiments in a study

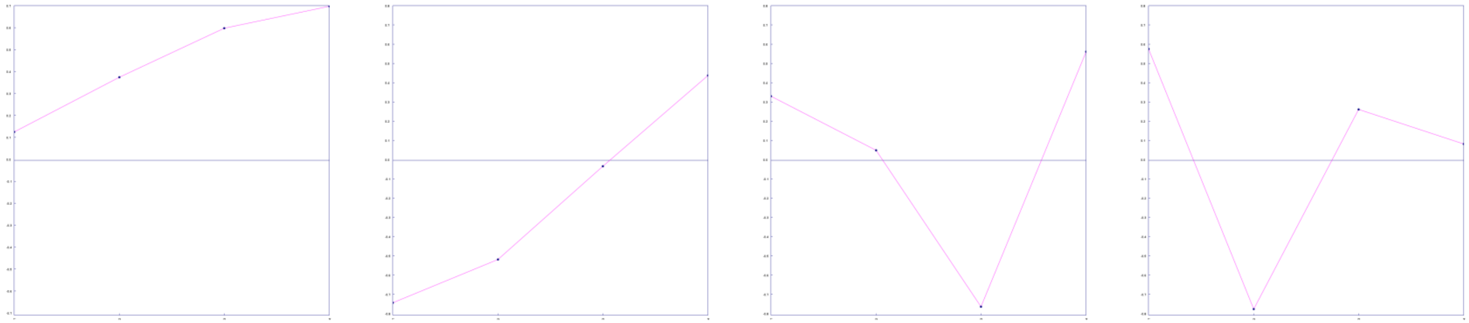


PCA Genes

3-dimensional view of the data in Eigenvector space



Component plots describing the basic patterns found in the data set



Exercise I

- Gene expression analysis using different clustering algorithms implemented in Genesis

  - Time course experiment:
    - Prostate cancer cell line LNCaP response to synthetic androgen R1881
    - 4 time points 2 replicates each
    - Human cDNA microarray
    - GEO accn. GDS2034 http://www.ncbi.nlm.nih.gov/geo/

Hendriksen PJ et al. Cancer Res 2006 May