

# Microarray data analysis with Chipster

13.-15.1.2016

Eija Korpelainen, Massimiliano Gentile  
chipster@csc.fi

# What will I learn?

- **How to operate the Chipster software**
- **How to analyze microarray data**
  - Central concepts
  - Analysis workflow
  - What happens in the different analysis steps
- **How to design experiments**

# Microarray data analysis workflow

- **Importing data to Chipster**
- **Normalization**
- **Describing samples with a phenodata file**
- **Quality control**
  - Array level
  - Experiment level
- **Filtering (optional)**
- **Statistical testing**
  - Parametric and non-parametric tests
  - Linear modeling
  - Multiple testing correction
- **Annotation**
- **Pathway analysis**
- **Clustering**
- **Saving the workflow**

# Introduction to Chipster

# Chipster

- **Provides an easy access to over 350 analysis tools**
  - No programming or command line experience required
- **Free, open source software**
- **What can I do with Chipster?**
  - analyze and integrate high-throughput data
  - visualize data efficiently
  - share analysis sessions
  - save and share automatic workflows



# Chipster

Open source platform for data analysis



- Home
- Getting access
- Analysis tool content
- Screenshots
- Manual
- Tutorial videos
- Cite
- FAQ
- Contact
  
- For developers:
  - Open source project
  - Tool editor

## Welcome to Chipster

Chipster is a user-friendly analysis software for high-throughput data. It contains over 350 analysis tools for next generation sequencing (NGS), microarray, proteomics and sequence data. Users can save and share automatic analysis workflows, and visualize data interactively using a [built-in genome browser](#) and many other visualizations.

Chipster's client software uses Java Web Start to install itself automatically, and it connects to computing servers for the actual analysis. Chipster is open source and the server environment is available as a [virtual machine image](#) free of charge. If you would like to use Chipster running on CSC's server, you need a [user account](#).



### Launch Chipster v3.6

...or launch with more memory: [3 GB](#) or [6 GB](#)

*If you have trouble launching Chipster, read [this](#)*

## News:

- 2.10.2015 [Version 3.6 released](#)
- 10.7.2015 [Chipster tutorial videos](#) now in YouTube
- 19.8.2014 [RNA-seq data analysis guidebook](#) with Chipster instructions

## Training:

- 13.-15.1.2016 Expression data analysis, DKFZ Heidelberg
- 1.12.2015 RNA-seq data analysis, University of Helsinki
- 16.11.2015 [RNA-seq data analysis](#), Cape Town
- 11.11.2015 [NGS data analysis](#), Bari
- 9.11.2015 [RNA-seq data analysis](#), CSC
- 29.10.2015 [RNA-seq data analysis](#), DPPS
- 26.-27.10.2015 RNA-seq data analysis, Biomedicum

**Datasets**

- two-sample.tsv
- column-value-filter.tsv
- hc.tre
- kmeans.pdf
- kmeans.tsv
- extract.tsv
- seqs.txt.wee
- seqs.html
- annotations.tsv
- annotations.html
- cpdb-pathways.html
- cpdb-pathways.tsv
- cpdb-genes.tsv

**Analysis tools**

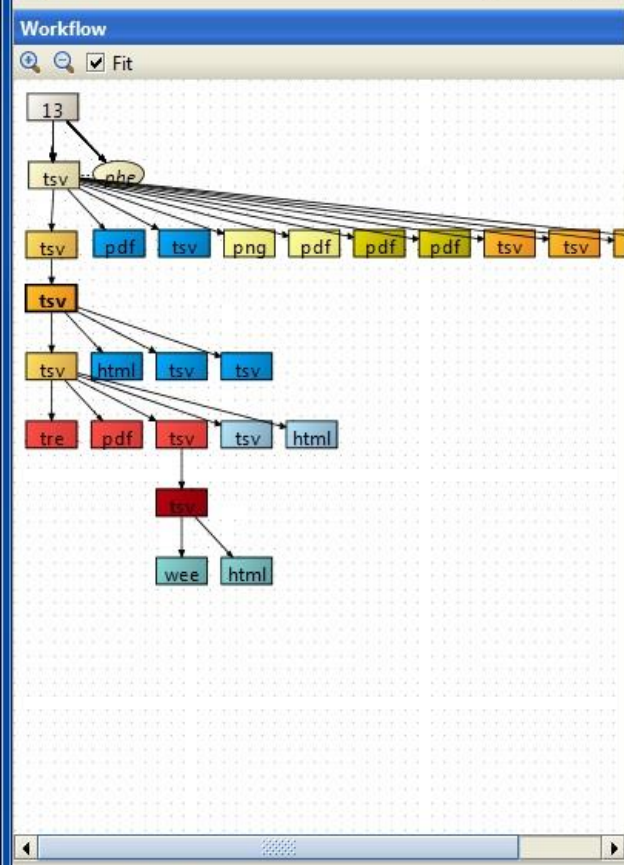
Microarrays | NGS | Misc

Normalisation  
 Quality control  
 Preprocessing  
 **Statistics**  
 Clustering  
 Annotation  
 Pathways  
 Promoter analysis  
 Copy number aberrations  
 Visualisation  
 Utilities

Show parameters  Run

One sample tests  
 Two groups tests  
 ROTS  
 SAM  
 Several groups tests  
 Linear modelling  
 Linear modelling using user-defined design matrix  
 Test proportions  
 Correlate with phenodata  
 Correlate miRNA with target expression  
 Time series  
 Association analysis

Tests for comparing the mean gene expression of two groups. LPE only works, if the whole normalized data is used, i.e., the data should not be filtered. Other than empiricalBayes might be slow, if run on unfiltered data.



**Visualisation**

Maximise Detach Close

two-sample.tsv

472 kB, Wed Sep 03 06:56:07 EEST 2014

(Click here to add your notes)

[Analysis history](#)

**Statistics / Two groups tests**

Column	group
Pairing	EMPTY
Test	empirical Bayes
p-value adjustment method	BH
p-value threshold	0.01
Show NA	no

- Spreadsheet
- Heatmap
- Expression profile
- Volcano plot
- Scatterplot
- 3D Scatterplot
- Histogram
- Open in external web browser

# Mode of operation

Select: data → tool category → tool → run → visualize

The screenshot displays the Chipster 3.4.0 (build 1441) interface, which is used for managing genomic data and running analysis tools. The interface is divided into several main sections:

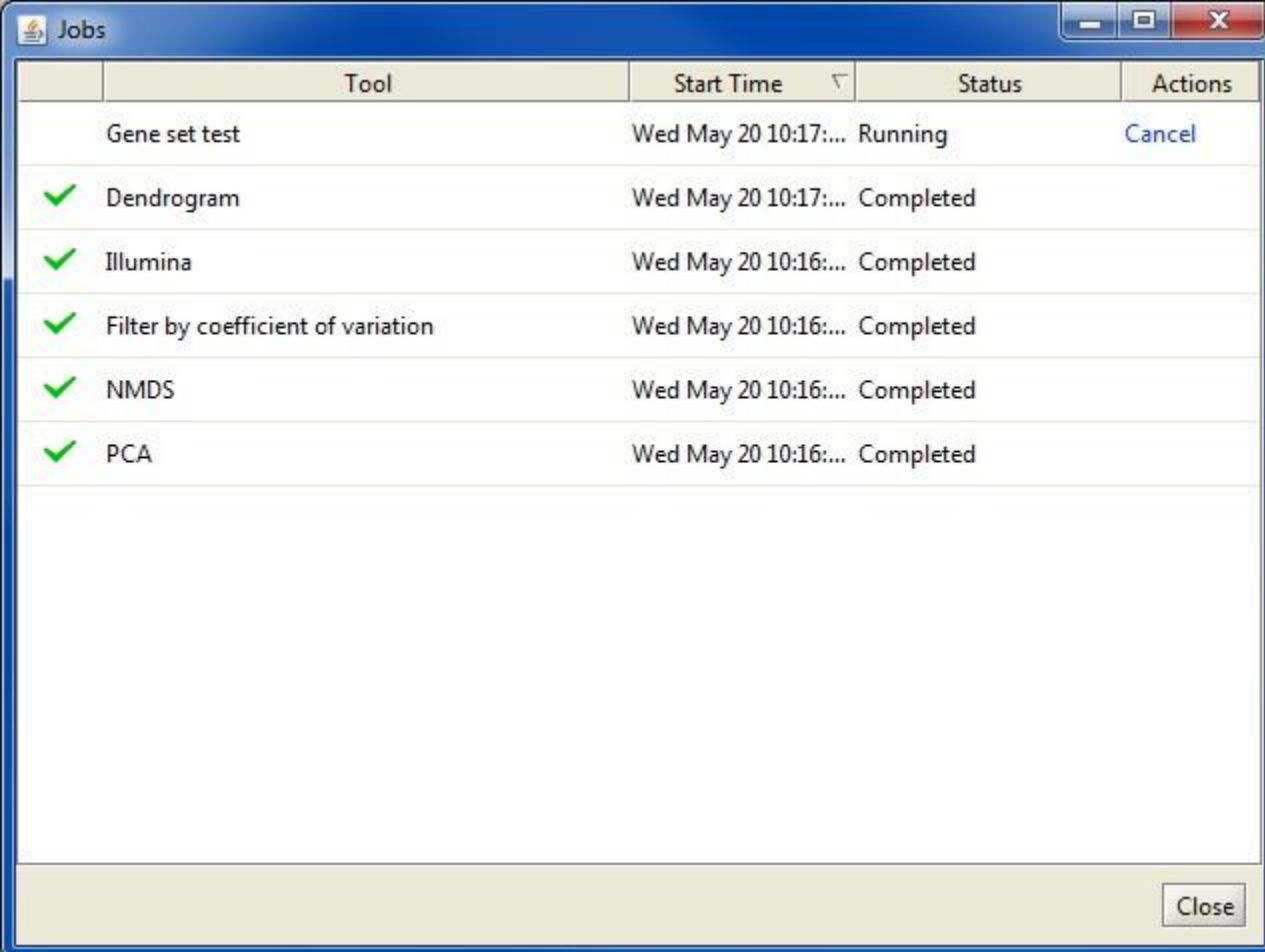
- Datasets:** A list of input files including control and treatment BAM files, MACS2 results, and various TSV and PDF files.
- Workflow:** A visual representation of the analysis pipeline. It shows a sequence of steps: BAM files are converted to bed files, which are then processed through various tools (TSV, PDF, TXT) to produce final HTML and TSV outputs. A red circle highlights a 'bed' file in the workflow.
- Analysis tools:** A menu of tools categorized by type (Microarrays, NGS, Misc). The 'Find peaks using MACS2' tool is selected, and its description is shown in a tooltip: "Detects statistically significantly enriched genomic regions in ChIP-seq data, using a control sample if available. If you have several samples, you need to merge them first to one ChIP file and one control file. BAM files can be merged with the Utilities tool 'Merge BAM'".
- Visualisation:** A genome browser view showing the genomic region around the RNF115-001 gene. It displays tracks for annotations (POLR3C-001, POLR3C-002), control ChIP-seq data (control\_chr\_1\_sorted.bam), treatment ChIP-seq data (treatment\_chr\_1\_sorted.bam), and MACS2 peaks (macs2-peaks.bed). A red arrow points from the 'bed' file in the workflow to the MACS2 peaks track.
- Settings and Options:** A panel on the right allows users to configure the visualization, including selecting the genome (Homo sapiens), chromosome (1), location (144322773), and view size (4 kb). It also includes options for highlighting SNPs and displaying density graphs.
- View jobs:** A button at the bottom right, circled in red, allows users to view the status of their analysis jobs.

At the bottom of the interface, it shows the connection to chipster.csc.fi, the status of 0 jobs running, and the current memory usage of 199M / 870M.



# Job manager

- You can run many analysis jobs at the same time
- Use Job manager to
  - view status
  - cancel jobs
  - view time
  - view parameters



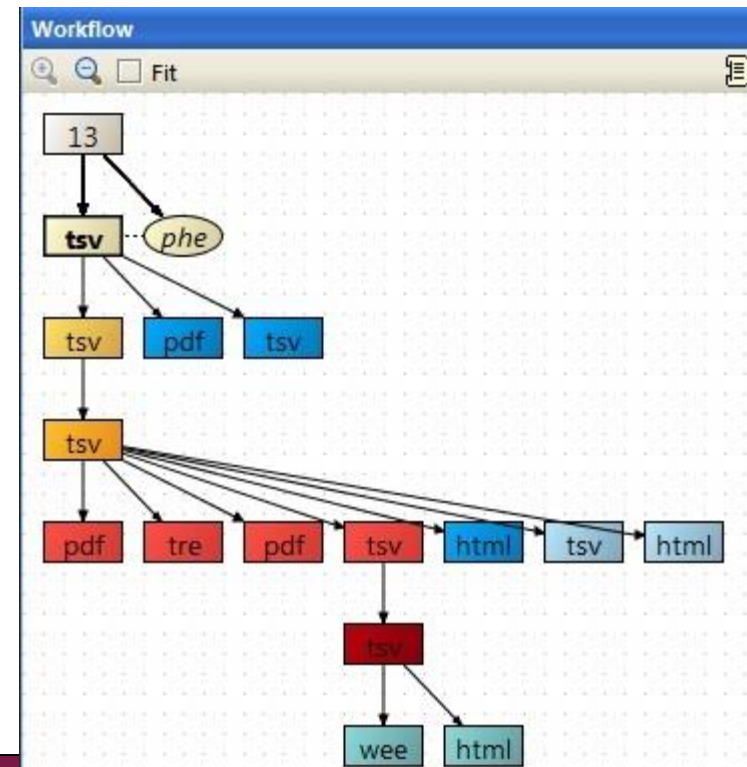
The screenshot shows a window titled "Jobs" with a table of analysis jobs. The table has columns for Tool, Start Time, Status, and Actions. The jobs listed are:

	Tool	Start Time	Status	Actions
	Gene set test	Wed May 20 10:17:...	Running	Cancel
✓	Dendrogram	Wed May 20 10:17:...	Completed	
✓	Illumina	Wed May 20 10:16:...	Completed	
✓	Filter by coefficient of variation	Wed May 20 10:16:...	Completed	
✓	NMDS	Wed May 20 10:16:...	Completed	
✓	PCA	Wed May 20 10:16:...	Completed	

A "Close" button is visible in the bottom right corner of the window.

# Workflow panel

- Shows the relationships of the files
- You can move the boxes around, and zoom in and out.
- Several files can be selected by keeping the Ctrl key down
- Right clicking on the data file allows you to
  - Save an individual result file ("Export")
  - Delete
  - Link to another data file
  - Save workflow

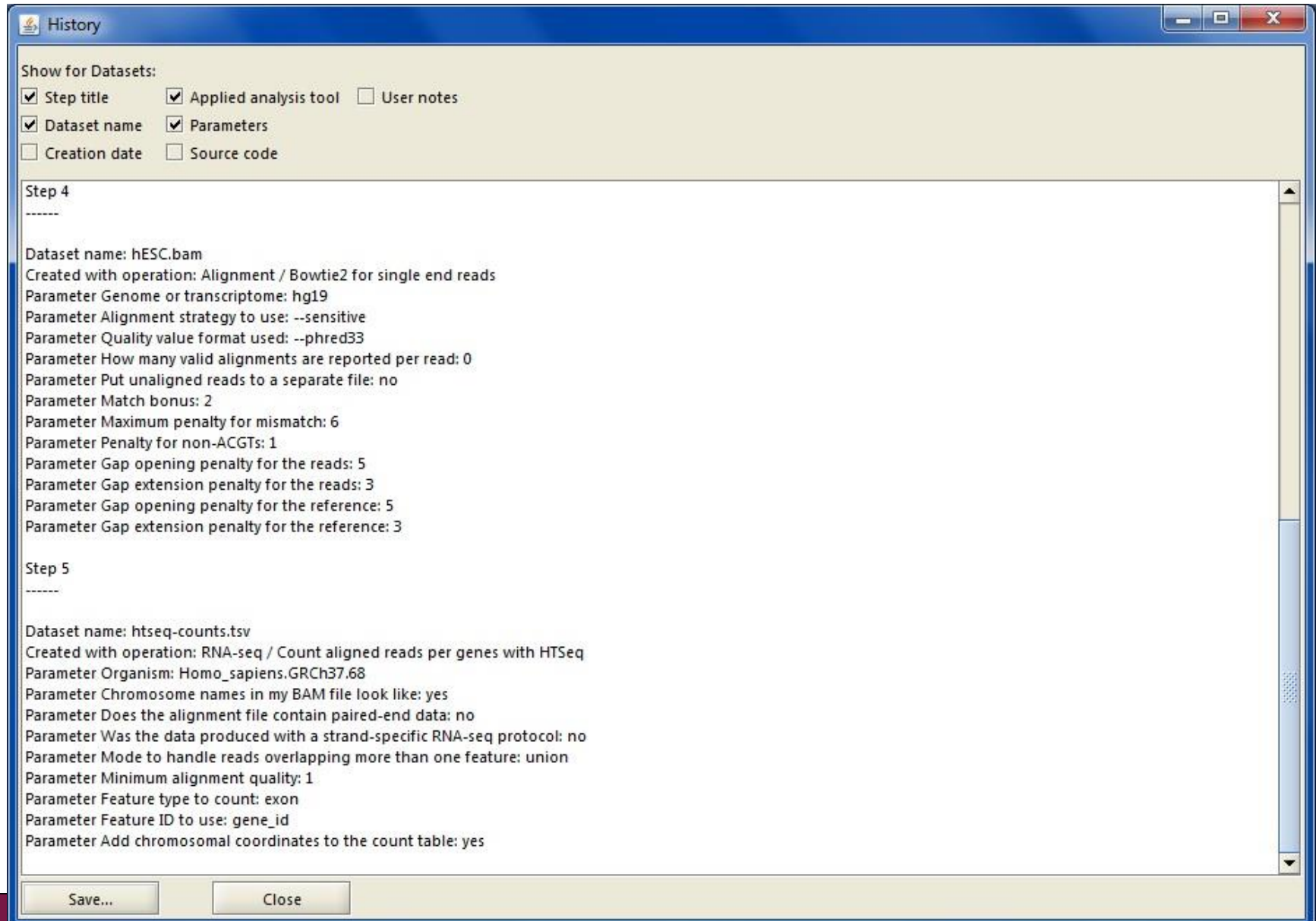


# Workflow – reusing and sharing your analysis pipeline

- **You can save your analysis steps as a reusable automatic "macro", which you can apply to another dataset**
- **When you save a workflow, all the analysis steps and their parameters are saved as a script file, which you can share with other users**

# Analysis history is saved automatically

-you can add tool source code to reports if needed



The screenshot shows a 'History' window with a title bar containing a folder icon and the text 'History'. The window has standard Windows window controls (minimize, maximize, close) in the top right corner. The main content area is divided into two sections, 'Step 4' and 'Step 5', each separated by a dashed line. Each section lists the dataset name, the operation used, and various parameters. At the bottom of the window, there are two buttons: 'Save...' and 'Close'.

Show for Datasets:

- Step title
- Applied analysis tool
- User notes
- Dataset name
- Parameters
- Creation date
- Source code

Step 4  
-----

Dataset name: hESC.bam  
Created with operation: Alignment / Bowtie2 for single end reads  
Parameter Genome or transcriptome: hg19  
Parameter Alignment strategy to use: --sensitive  
Parameter Quality value format used: --phred33  
Parameter How many valid alignments are reported per read: 0  
Parameter Put unaligned reads to a separate file: no  
Parameter Match bonus: 2  
Parameter Maximum penalty for mismatch: 6  
Parameter Penalty for non-ACGTs: 1  
Parameter Gap opening penalty for the reads: 5  
Parameter Gap extension penalty for the reads: 3  
Parameter Gap opening penalty for the reference: 5  
Parameter Gap extension penalty for the reference: 3

Step 5  
-----

Dataset name: htseq-counts.tsv  
Created with operation: RNA-seq / Count aligned reads per genes with HTSeq  
Parameter Organism: Homo\_sapiens.GRCh37.68  
Parameter Chromosome names in my BAM file look like: yes  
Parameter Does the alignment file contain paired-end data: no  
Parameter Was the data produced with a strand-specific RNA-seq protocol: no  
Parameter Mode to handle reads overlapping more than one feature: union  
Parameter Minimum alignment quality: 1  
Parameter Feature type to count: exon  
Parameter Feature ID to use: gene\_id  
Parameter Add chromosomal coordinates to the count table: yes

Save... Close

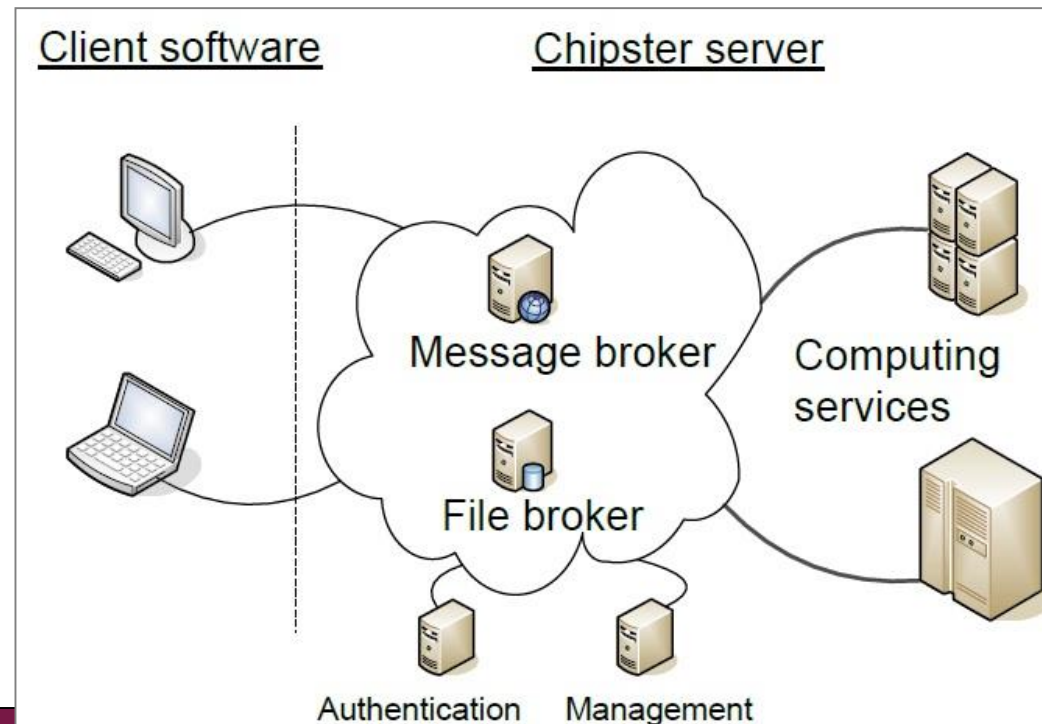
# Technical aspects

## ➤ Client-server system

- Enough CPU and memory for large analysis jobs
- Centralized maintenance

## ➤ Easy to install

- Client uses Java Web Start
- Server available as a virtual machine



# Analysis sessions

- **In order to continue your work later, you have to save the analysis session.**
  - Session includes all the files, their relationships and metadata (what tool and parameters were used to produce each file).
- **Session is saved into a single .zip file on your computer.**
  - In Chipster v3.7 you can also save it on the server
- **Session files allow you to continue the analysis on another computer, or share it with a colleague.**
- **You can have multiple analysis sessions saved separately, and combine them later if needed.**

# Problems? Send a support request

-request includes the error message and link to analysis session (optional)

```
Hi,  
I'm trying to normalise my Illumina microarray data (obtained with the Illumina HT-12 v4.0)  
For that purpose I have selected the Normalisation option "Illumina - lumi pipeline"  
However, the normalisation did not complete successfully.
```

Any advice to solve this problem ?

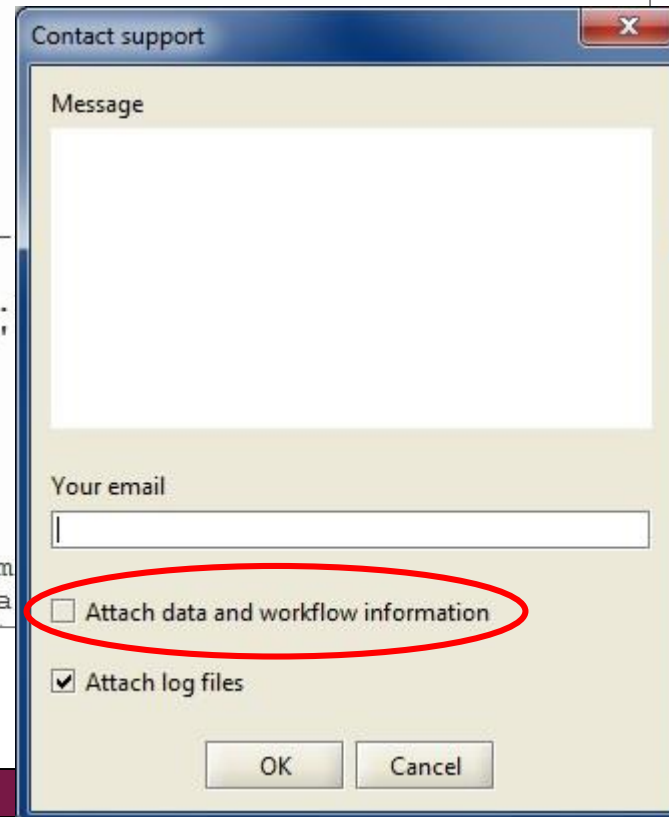
Thank you in advance for your precious help.

Best regards

Error message:

```
in library(chiptype, character.only = T) :  
  there is no package called 'Illumina.db'
```

```
-----  
> chipster.common.path = '/opt/chipster/comp/modules/common/R-2.  
> chipster.module.path = '/opt/chipster/comp/modules/microarray'  
> setwd("271661a6-946c-450f-bb21-5d5b5a2837aa")  
> probe.identifier <- "Probe_ID"  
> transformation <- "log2"  
> background.correction <- "none"  
> normalize.chips <- "quantile"  
> chiptype <- "empty"  
> # TOOL norm-illumina-lumi.R: "Illumina - lumi pipeline" (Illum  
BeadSummaryData files, and using lumi methodology. If you have a
```



Contact support

Message

Your email

Attach data and workflow information

Attach log files

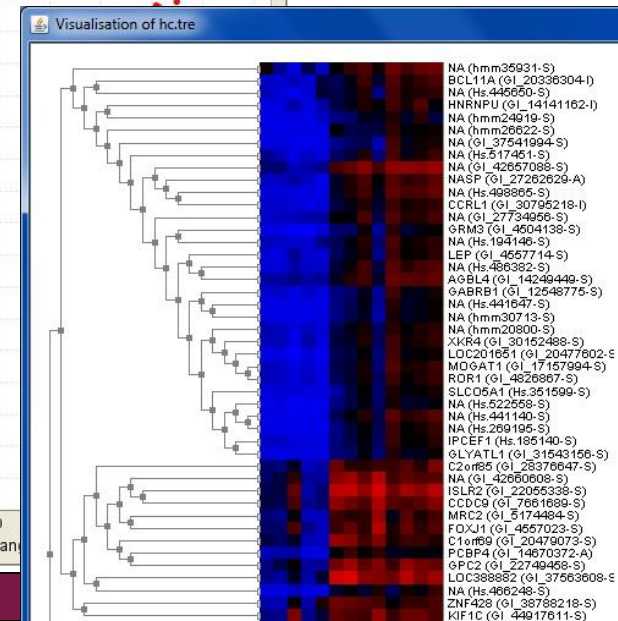
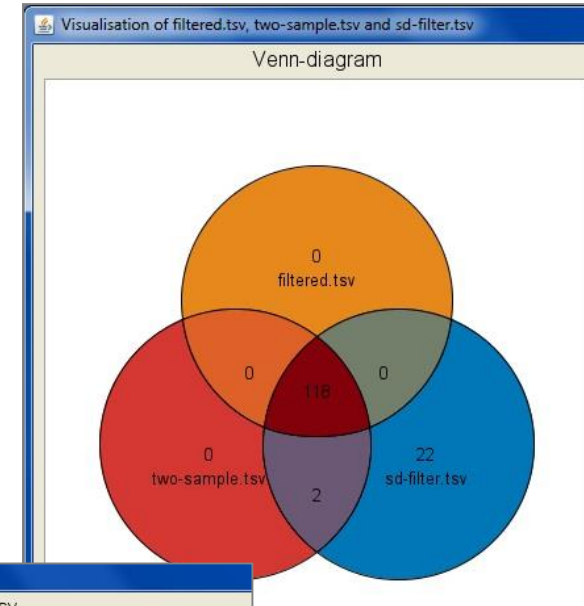
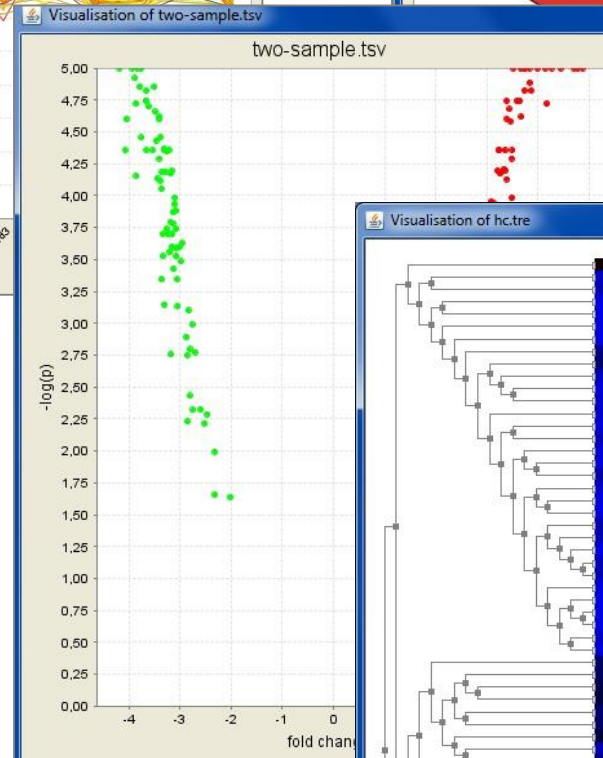
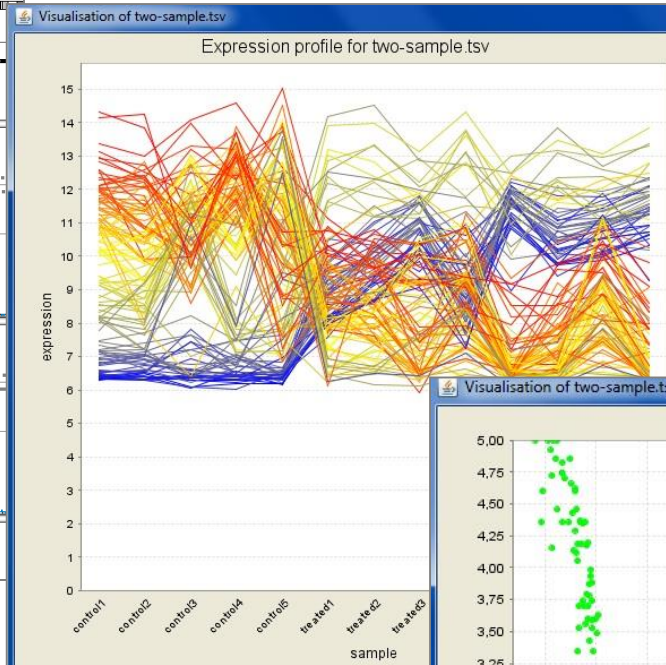
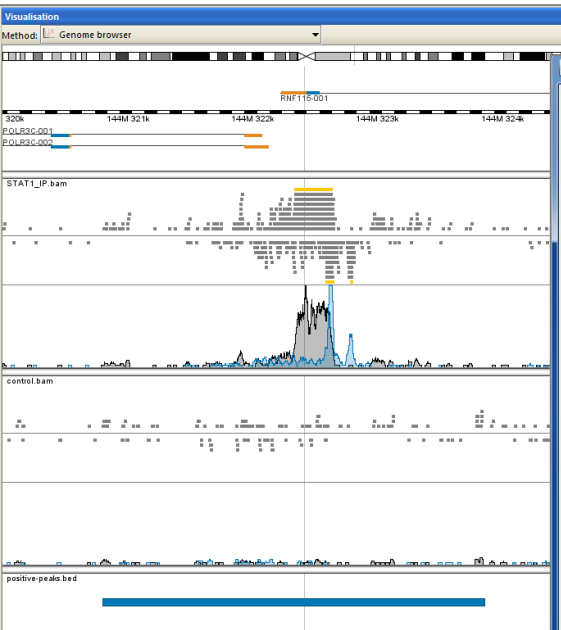
OK Cancel

# Two types of data visualizations

- 1. Interactive visualizations produced by the client program**
  - Select the visualization method from visualization panel icons
  - Save by right clicking on the image
- 2. Static images produced by the analysis tools on the server**
  - Select from Analysis tools / Visualisation
  - Save by right clicking on the file name and choosing "Export"



# Interactive visualizations



## Available actions:

- Select genes and create a gene list
- Change titles, colors etc
- Zoom in/out
- Venn diagram: select genes with a list

### Datasets

- teratospermi...GSM160624\_...
- teratospermiGSM160626\_(6474973047278781905)
- teratospermiGSM160627\_(7690701737716377477)
- teratospermiGSM160628\_(6016938503863357191)
- normalized.tsv
- phenodata.tsv
- sd-filter.tsv
- multitest.pdf
- globaltest-result-table.tsv
- two-sample.tsv**
- resample.pdf
- hc.tr
- kmeans.pdf

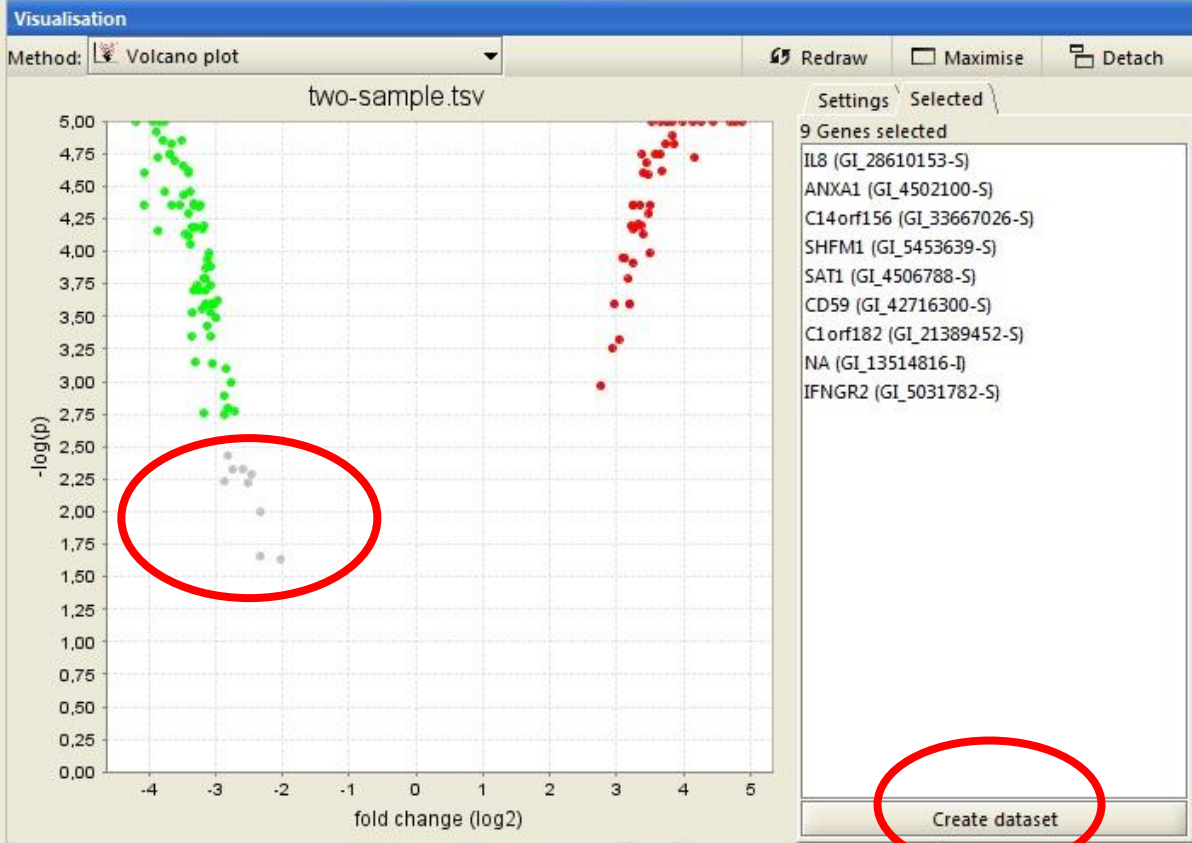
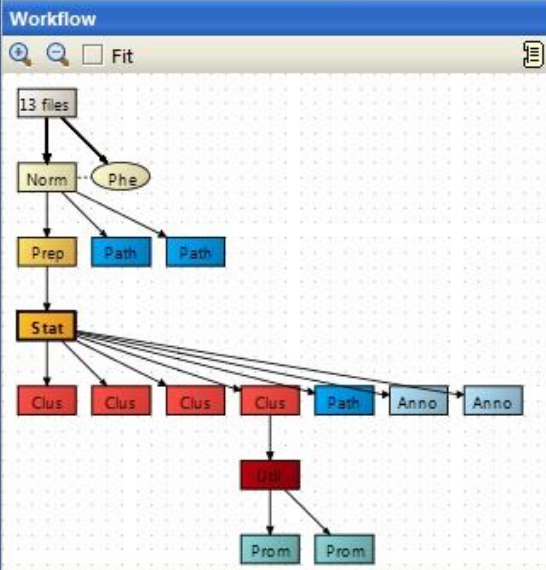
### Analysis tools

Microarrays NGS

- Normalisation
- Quality control
- Preprocessing
- Statistics**
  - One sample tests
  - Two groups tests**
    - ROTS
    - SAM
  - Several groups tests
  - Linear modelling
  - Test proportions
  - Correlate with phenodata
  - Correlate miRNA with target expression
  - Time series
  - Association analysis
- Clustering
- Annotation
- Pathways
- Promoter analysis
- aCGH
- Visualisation
- Utilities

Tests for comparing the mean gene expression of two groups. LPE only works, if the whole data is used, i.e., the data should not be pre-filtered, if LPE is used. Other than empiricalBayes might be slow, if run on unfiltered data.

More help Show tool sourcecode



### Notes for dataset

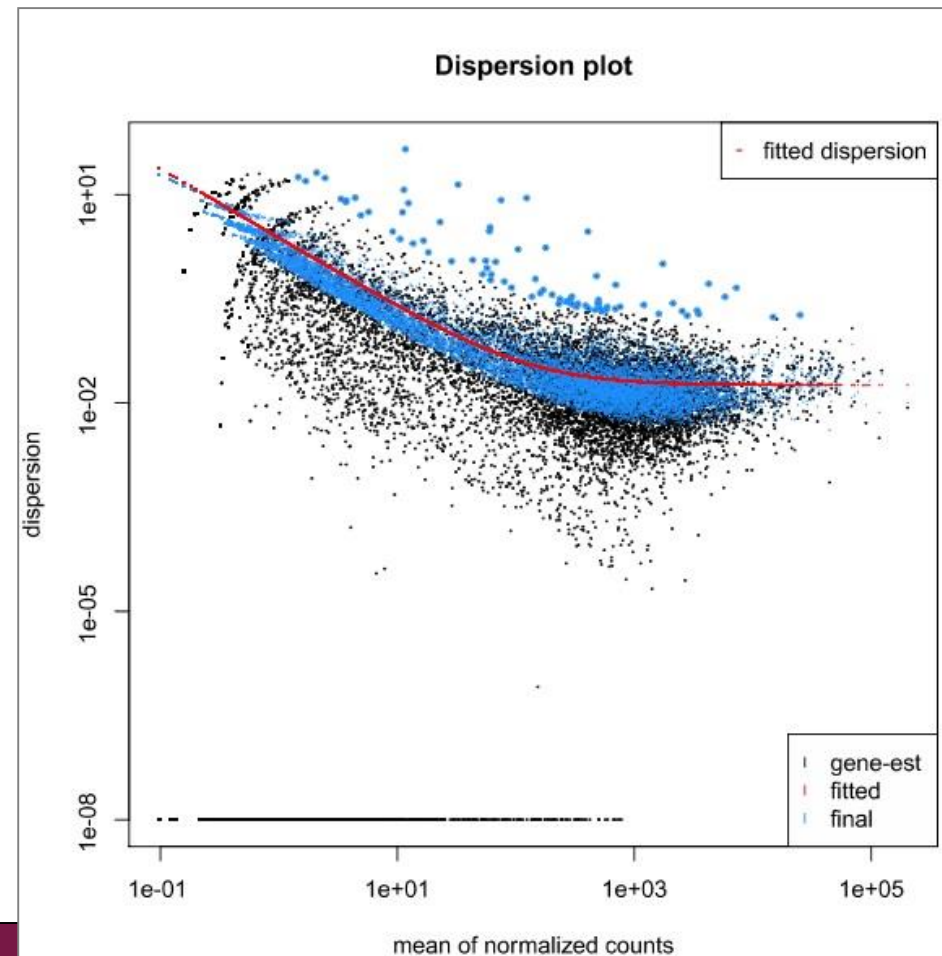
Statistics / Two groups tests Hide

Wed Oct 17 12:11:05 EEST 2012  
 column=group, test=empiricalBayes,  
 p.value.adjustment.method=BH, p.value.threshold=0.05

Add your notes here...

# Static images produced by analysis tools

- MA plot
- MDS plot
- Box plot
- Histogram
- Heatmap
- Idiogram
- Chromosomal position
- Correlogram
- Dendrogram
- K-means clustering
- SOM-clustering
- Dispersion plot
- etc



# Analysis tool overview

## ➤ 150 NGS tools for

- RNA-seq
- miRNA-seq
- exome/genome-seq
- ChIP-seq
- FAIRE/DNase-seq
- MeDIP-seq
- CNA-seq
- Metagenomics (16S rRNA)

## ➤ 140 microarray tools for

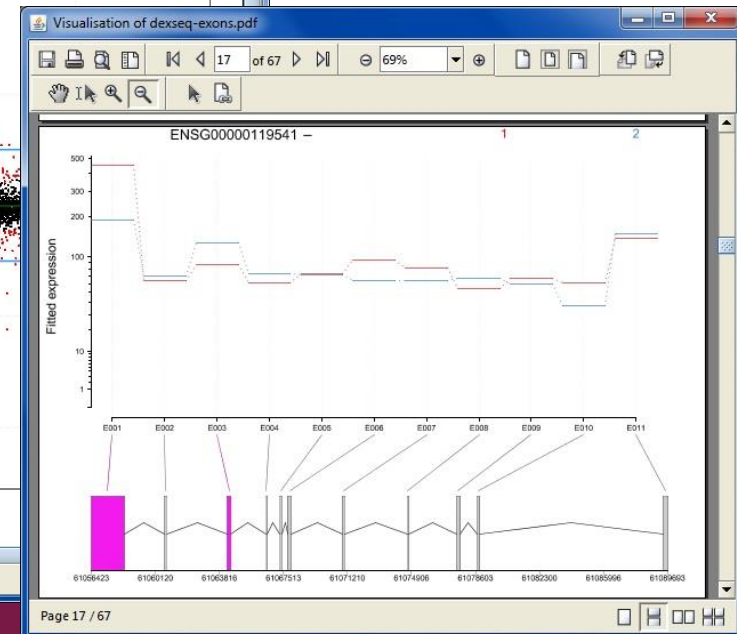
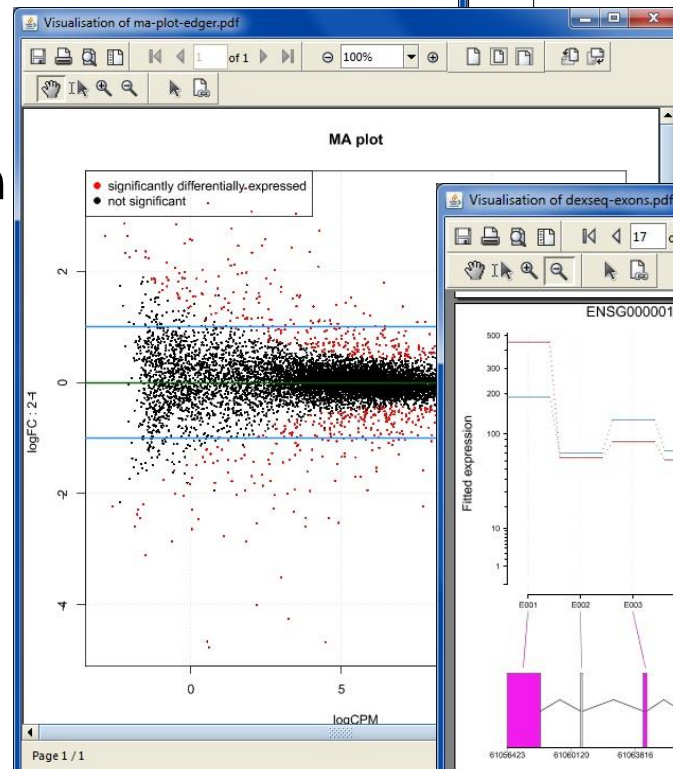
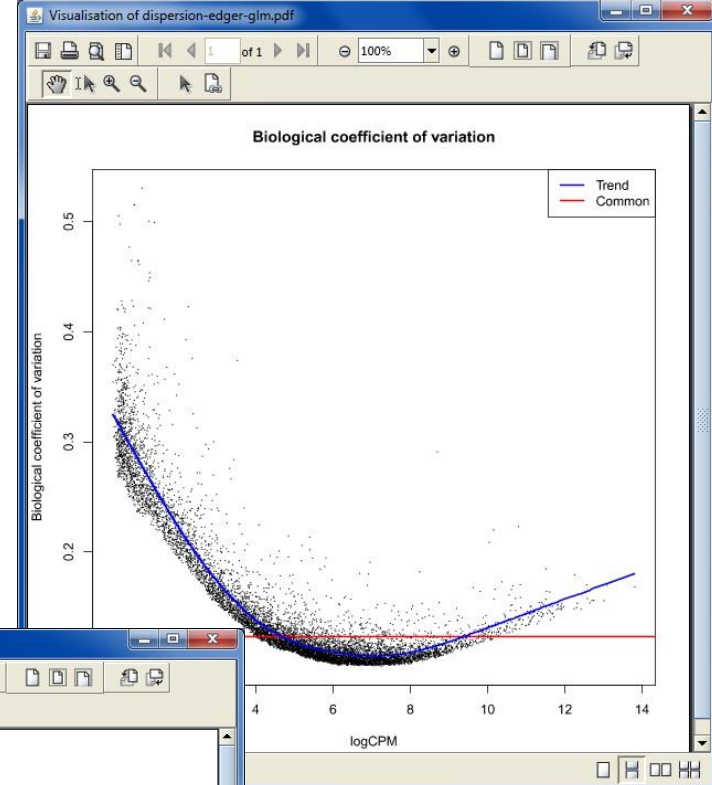
- gene expression
- miRNA expression
- protein expression
- aCGH
- SNP
- integration of different data

## ➤ 60 tools for sequence analysis

- BLAST, EMBOSS, MAFFT
- Phylip

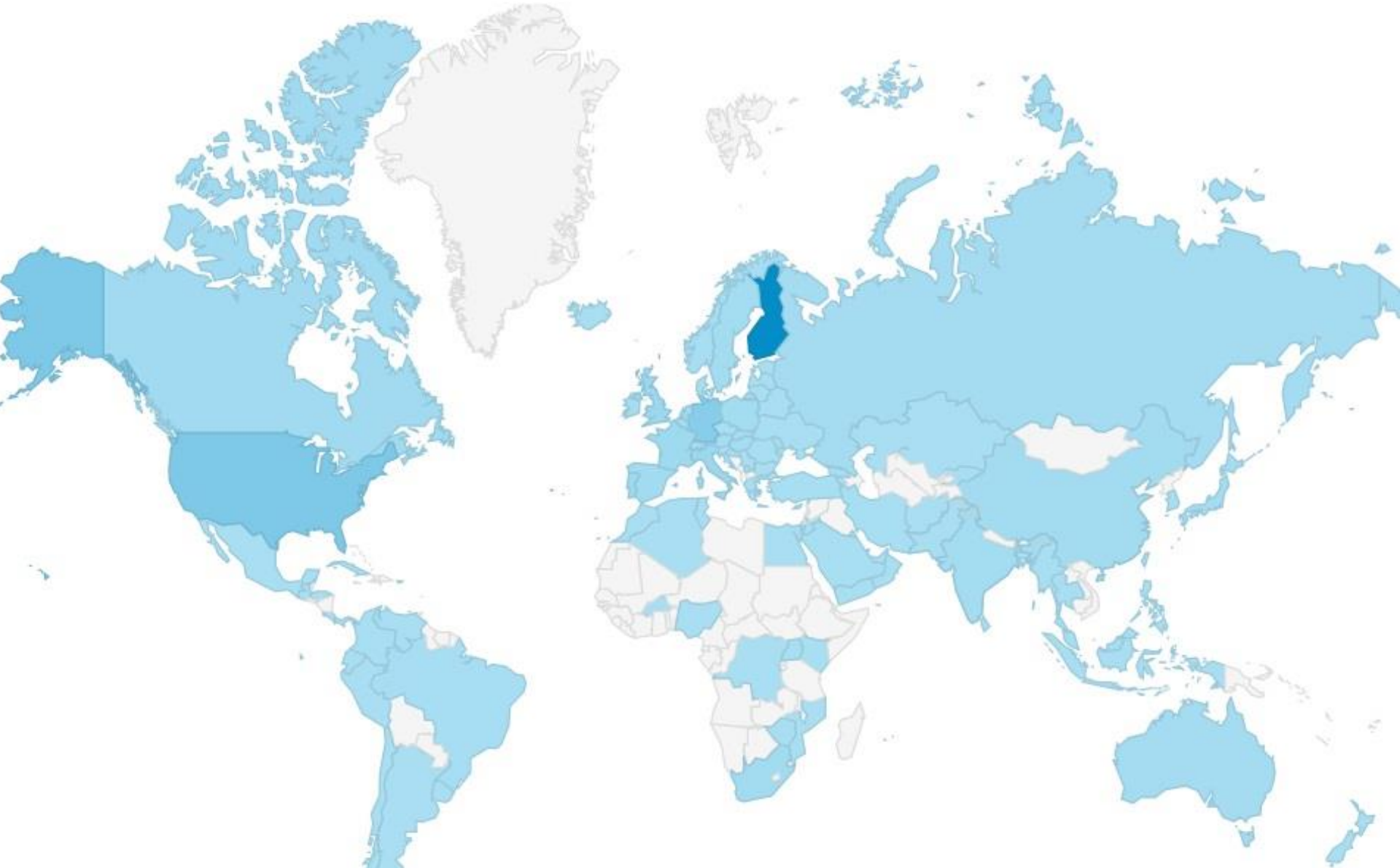
# Example: RNA-seq tools

- **Quality control**
  - RseQC
- **Counting**
  - HTSeq
  - eXpress
- **Transcript discovery**
  - Cufflinks
- **Differential expression**
  - edgeR
  - DESeq
  - Cuffdiff
  - DEXSeq
- **Pathway analysis**
  - ConsensusPathDB



# Acknowledgements to Chipster users and contributors

---



# More info

- [chipster@csc.fi](mailto:chipster@csc.fi)
- <http://chipster.csc.fi>
- Chipster tutorials in YouTube

GitHub

This repository Search

Explore Features

chipster / chipster

Chipster is a user-friendly analysis software for high-throughput data.

7,565 commits

18 branches

123 releases

14 contributors

BMC  
Genomics

IMPACT  
FACTOR  
4.21

[home](#) | [journals A-Z](#) | [subject areas](#) | [advanced search](#) | [authors](#) | [reviewers](#) | [libraries](#) | [about](#) | [my BioMed Central](#)

Software

Highly accessed Open Access

## Chipster: user-friendly analysis software for microarray and other high-throughput data

M Aleksis Kallio ✉, Jarno T Tuimala ✉, Taavi Hupponen ✉, Petri Klemela ✉, Massimiliano Gentile ✉, Ilari Scheinin ✉, Mikko Koski ✉, Janne Kaki ✉ and Eija I Korpelainen ✉

BMC Genomics 2011, 12:507 doi:10.1186/1471-2164-12-507

RNA-seq Data Analysis

Korpelainen, Tuimala,  
Somervuo, Huss, and Wong

Chapman & Hall/CRC  
Mathematical and Computational Biology Series

## RNA-seq Data Analysis A Practical Approach



Eija Korpelainen, Jarno Tuimala,  
Panu Somervuo, Mikael Huss, and Garry Wong



CRC Press  
Taylor & Francis Group  
A CHAPMAN & HALL BOOK

# Microarray data analysis



# Microarray data analysis workflow

- **Importing data to Chipster**
- **Normalization**
- **Describing samples with a phenodata file**
- **Quality control**
  - Array level
  - Experiment level
- **Filtering (optional)**
- **Statistical testing**
  - Parametric and non-parametric tests
  - Linear modeling
  - Multiple testing correction
- **Annotation**
- **Pathway analysis**
- **Clustering**
- **Saving the workflow**

# Importing data

## ➤ Affymetrix

- CEL-files are recognized by Chipster automatically

## ➤ Illumina: two importing options

1. Import the GenomeStudio file as it is

- All the samples need to be in one file.
- Need columns AVG, BEAD\_STDERR, Avg\_NBEADS and DetectionPval
- When imported this way, the data has to be normalized in Chipster using the lumi method

2. Use Import tool to define the sample columns in the file(s)

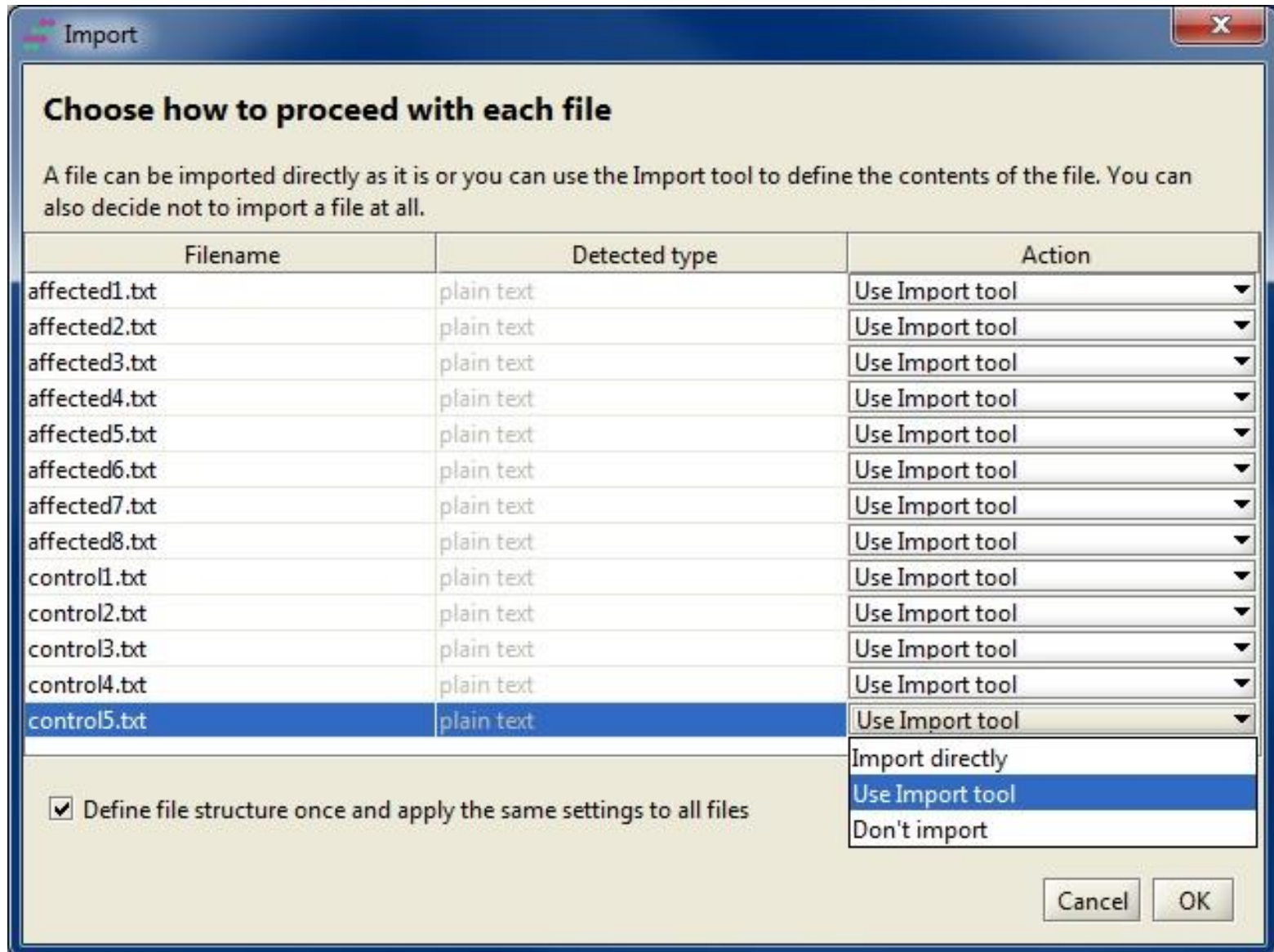
- Use the tool "Normalization / Illumina" to normalize the data

➔ **The import option influences your normalization options later**

## ➤ Agilent (and any other tab delimited files)

- Use Import tool to define the sample columns

# 1. Import tool: Select what to do



## 2. Import tool: Define rows (header, title, etc)

The screenshot shows the 'Import tool' window with the following settings and data:

- Tools Panel:**
  - Column Delimiter:** Tab (selected), Space, Comma, Semicolon, Other:  Use
  - Decimal Separator:** Dot (selected), Comma
- Select rows (affected1.txt):**
  - Mark header: 0
  - Mark footer: 47295
  - Mark title row:
  - Reset:
  - Showing columns 5 of 9
- Data Table:**

	1	2	3	4	5
1	TargetID	MIN_Signal...	AVG_Signal...	MAX_Signal...	...
2	GI_1004708...	73.7	73.7	73.7	...
3	GI_1004709...	312.7	312.7	312.7	...
4	GI_1004709...	170.6	170.6	170.6	...
5	GI_1004709...	98.0	98.0	98.0	...
6	GI_1004710...	354.3	354.3	354.3	...
7	GI_1004710...	213.0	213.0	213.0	...
8	GI_1004712...	90.9	90.9	90.9	...
9	GI_1004712...	92.4	92.4	92.4	...
10	GI_1004713...	83.8	83.8	83.8	...
11	GI_10047133-I	92.3	92.3	92.3	...
12	GI_1009257...	599.3	599.3	599.3	...
13	GI_1009258...	99.0	99.0	99.0	...
14	GI_1009259...	122.1	122.1	122.1	...
15	GI_1009260...	3789.0	3789.0	3789.0	...
16	GI_1009260...	85.4	85.4	85.4	...
17	GI_1009260...	96.0	96.0	96.0	...
18	GI_1009261...	93.8	93.8	93.8	...
19	GI_1009261...	455.9	455.9	455.9	...
20	GI_1009261...	135.8	135.8	135.8	...
21	GI_1009263...	100.0	100.0	100.0	...
22	GI_1009265...	71.9	71.9	71.9	...
23	GI_1009266...	05.8	05.8	05.8	...

Buttons at the bottom: Help, Back, Next, Finish, Cancel.

# 3. Import tool: Define columns (identifier, sample)

Import tool

Tools

Chip counts

Complete with pattern

Complete the rest Undo

Data Modification

Column: 1 - TargetID

Look For:

Replace With:

Use Regular Expressions

Replace Undo

Select columns (affected1.txt)

	Identifier	Sample	Sample BG	Control	Control BG	Flag	Annotation	Unused	Reset
Showing rows 100 of 47294	1 - TargetID	2 - MIN_Signal-1412091085_A	3 - AVG_Signal-1412091085_A	4 - MAX_Signal-1412091085_A	5 - N/				
	Identifier	Unused	Sample	1	Unused			Unus	
2	GI_10047089-S	73.7	73.7	73.7	1.0				
3	GI_10047091-S	312.7	312.7	312.7	1.0				
4	GI_10047093-S	170.6	170.6	170.6	1.0				
5	GI_10047099-S	98.0	98.0	98.0	1.0				
6	GI_10047103-S	354.3	354.3	354.3	1.0				
7	GI_10047105-S	213.0	213.0	213.0	1.0				
8	GI_10047121-S	90.9	90.9	90.9	1.0				
9	GI_10047123-S	92.4	92.4	92.4	1.0				
10	GI_10047133-A	83.8	83.8	83.8	1.0				
11	GI_10047133-I	92.3	92.3	92.3	1.0				
12	GI_10092578-S	599.3	599.3	599.3	1.0				
13	GI_10092585-S	99.0	99.0	99.0	1.0				
14	GI_10092596-S	122.1	122.1	122.1	1.0				
15	GI_10092600-S	3789.0	3789.0	3789.0	1.0				
16	GI_10092602-S	85.4	85.4	85.4	1.0				
17	GI_10092603-S	96.0	96.0	96.0	1.0				
18	GI_10092611-A	93.8	93.8	93.8	1.0				
19	GI_10092616-S	455.9	455.9	455.9	1.0				
20	GI_10092618-S	135.8	135.8	135.8	1.0				
21	GI_10092638-S	100.0	100.0	100.0	1.0				
22	GI_10092658-S	71.9	71.9	71.9	1.0				

Select sample

Help Back Next Finish Cancel

# Import tool - which columns should I mark?

- <http://chipster.csc.fi/manual/import-help.html>
  - **Illumina BeadStudio version 3 file and GenomeStudio files**
    - Identifier (ProbeID)
    - Sample (text “AVG”)
  - **Illumina BeadStudio version 1-2 file**
    - Identifier (TargetID)
    - Sample (text “AVG”)
  - **Agilent**
    - Identifier (ProbeName)
    - Sample (rMeanSignal or rMedianSignal)
    - Sample background (rBGMedianSignal)
    - Control (gMeanSignal or gMedianSignal)
    - Control background (gBGMedianSignal)
    - Flag (Control type)
- 
- 1-color
- 2-color

# Exercise 1. Import Illumina data directly

## ➤ **Import Illumina data directly**

- Select **File / Import files**.
- Select the file **IlluminaHuman6v1\_BS1.txt**
- In the Import files -window choose the action **“Import directly”**
- Select the file and view it as text.

## ➤ **Normalize the data with the lumi tool**

- Select the file and the tool **Normalization/ Illumina – lumi pipeline**. Set the **chiptype** parameter to **Human** and click **Run**.
- Inspect the result file **normalized.tsv**. How does the first column containing identifiers look like?
- Inspect the file **phenodata.tsv**. How many samples are there?

# Exercise 2: Import Illumina data with Import tool

## ➤ Import the same file using the Import tool

- Select **File / Import files** and select the file **IlluminaHuman6v1\_BS1.txt**
- In the Import files -window choose the action **Use Import tool**
- Click the **Mark header** button and paint the header rows.
- Click the **Mark title row** button and click on the title row. Click **Next**.
- Click the **Identifier** button and click in the **TargetID** column.
- Click the **Sample** button and click in a couple of **AVG** columns. Click the **Complete the rest** button and check that all the AVG columns were selected.
- Click **Finish**.
- How many files do you get now? Inspect one of them.

## ➤ Normalize the data

- Select the 8 files and run **Normalization/ Illumina** so that:
  - **Illumina software version = BeadStudio1**
  - **identifier type = TargetID**
  - **chiptype = Human-6v1**.
- Inspect the **normalized.tsv** and **phenodata.tsv**. Are there differences if compared to the files from exercise 1?



# Exercise 3: Import several files with Import tool

- **Save the session and start a new one.**
  - Select **File / Save local session.**
  - Select **File / New session.**
  
- **Import a new dataset containing several files (one for each sample) using the Import tool**
  - Select **File / Import folder** and
  - Select the folder **IlluminaTeratospermiaHuman6v1\_BS1**
  - Choose the action **Use Import tool** for each file
  - Click the **Mark title row** button and click on the title row. Click **Next.**
  - Click the **Identifier** button and click in the **TargetID** column.
  - Click the **Sample** button and click in the **AVG** column. Click **Finish.**
  - How many files do you get now?

# Importing normalized data

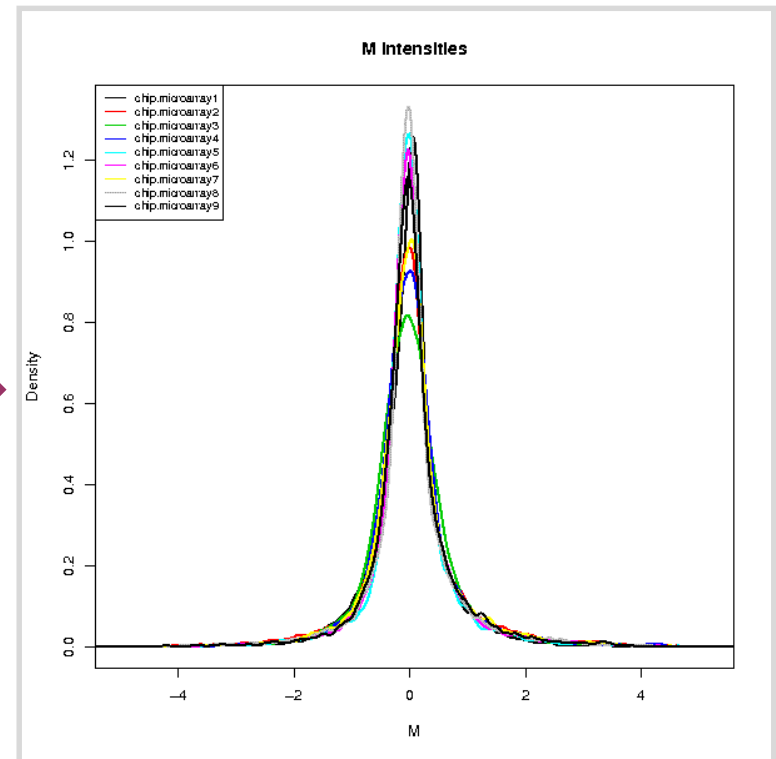
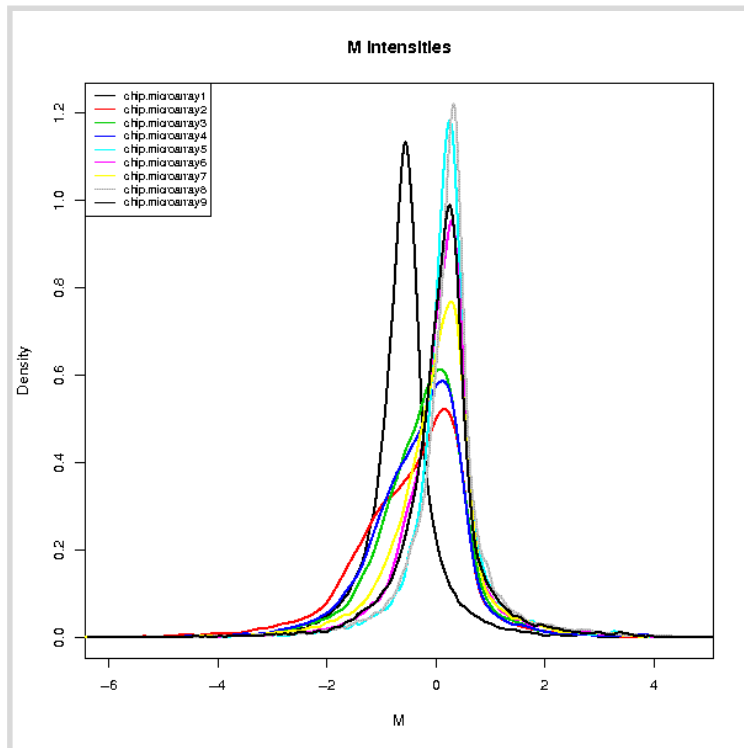
- **The data should be tab delimited and preferably log-transformed**
  - If your data is not log-transformed, you can transform it with the tool “Change interpretation”
- **Import the data file to Chipster using the Import tool. Mark the identifier column and all the sample columns.**
- **Run the tool Normalize / Process prenormalized. This**
  - Converts data to Chipster format by adding “chip.” to expression column names
  - Creates the phenodata file. You need to indicate the chiptype using names given at <http://chipster.csc.fi/manual/supported-chips.html>

# Microarray data analysis workflow

- Importing data to Chipster
- **Normalization**
- Describing samples with a phenodata file
- **Quality control**
  - Array level
  - Experiment level
- **Filtering (optional)**
- **Statistical testing**
  - Parametric and non-parametric tests
  - Linear modeling
  - Multiple testing correction
- **Annotation**
- **Pathway analysis**
- **Clustering**
- **Saving the workflow**

# Normalization

- **The goal is to make the arrays comparable to each other**
  - Makes the expression value distributions similar
  - Assumes that most genes don't change expression
- **After normalization the expression values are in log<sub>2</sub>-scale**
  - Hence a fold change of 2 means 4-fold up



# Illumina normalization: two analysis tools

## 1. Illumina

- Normalization method  
Quantile, vsn (variance stabilizing normalization), scale, none
- Illumina software version  
GenomeStudio or BeadStudio3, BeadStudio2, BeadStudio1
- Chiptype
- Identifier type  
Probe ID (for BeadStudio version 3 data and newer), Target ID

## 2. Lumi pipeline (data needs to be in one file, imported directly!)

- Normalization method  
Quantile, vsn, rsn (robust spline normalization), loess, none
- Transformation  
Log2, vst (variance stabilizing transformation), none
- Chiptype  
human, mouse, rat
- Background correction (usually done already in GenomeStudio)  
none, bgAdjust.Affy

# Quantile normalization procedure

	Sample A	Sample B	Sample C
Gene 1	20	10	350
Gene 2	100	500	200
Gene 3	300	400	30

1. Raw data

	Sample A	Sample B	Sample C	Median
Quantile 1	20	10	30	20
Quantile 2	100	400	200	200
Quantile 3	300	500	350	350

2. Rank data within sample and calculate median intensity for each row

	Sample A	Sample B	Sample C	Median
Quantile 1	20	20	20	20
Quantile 2	200	200	200	200
Quantile 3	350	350	350	350

3. Replace the raw data of each row with its median (or mean) intensity

	Sample A	Sample B	Sample C
Gene 1	20	20	350
Gene 2	200	350	200
Gene 3	350	200	20

4. Restore the original gene order

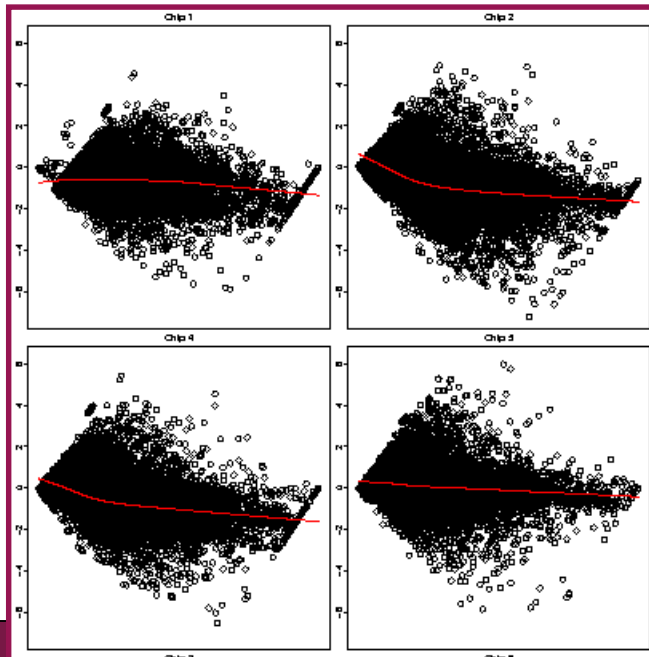
# Normalization of Affymetrix data

- **Normalization = background correction + expression estimation + summarization**
- **Methods**
  - **RMA** (Robust Multichip Averaging) uses only PM probes, fits a model to them, and gives out expression values after quantile normalization and median polishing. Works nicely if you have more than a few chips
  - **GCRMA** is similar to RMA, but takes also GC% content into account
  - **MAS5** is the older Affymetrix method, **Plier** is a newer one
  - **Li-Wong** is the method implemented in dChip
- **Custom chip type parameter to use remapped probe information**
  - Because some of the Affymetrix probe-to-transcript mappings are not correct, probes have been remapped in the Bioconductor project.
  - To use these remappings (alt CDF environments), select the matching chip type from the Custom chip type menu.
- **Variance stabilization option makes the variance similar over all the chips**
  - Works only with MAS5 and Plier (the other methods log<sub>2</sub>-transform the data, which corrects for the same phenomenon)

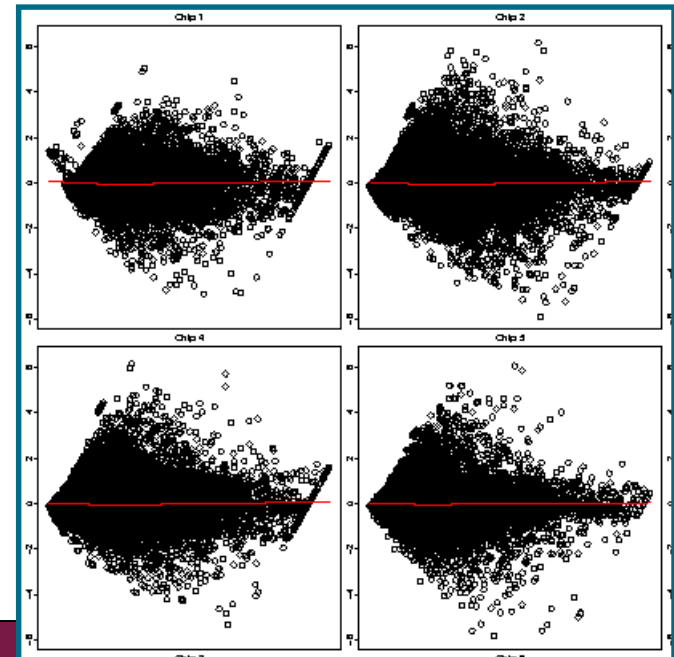
# Normalization of Agilent data

- **Background correction + averaging duplicate spots + normalization**
- **Background subtraction often generates negative values, which are coded as missing values after log<sub>2</sub>-transformation.**
  - Using normexp + offset 50 will not generate negative values, and it gives good estimates
- **Loess removes curvature from the data (recommended)**

Before



After





# Agilent normalization parameters in Chipster

## ➤ **Background treatment**

- Normexp, Subtract, Edwards, None

## ➤ **Background offset**

- 50 or 0

## ➤ **Normalize chips**

- Loess, median, none

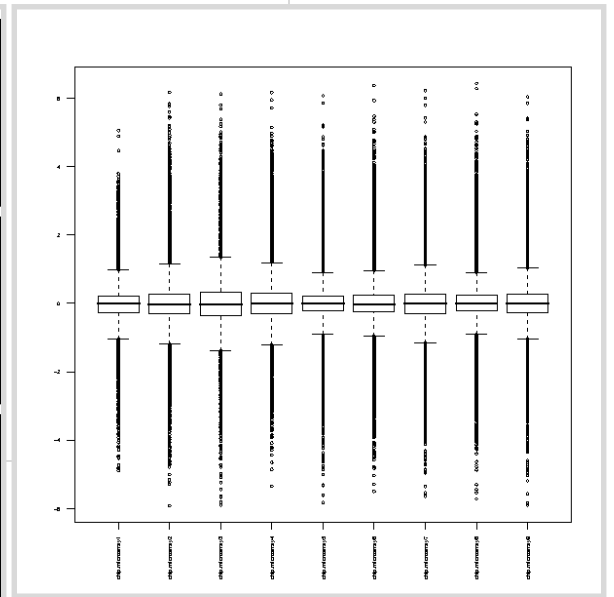
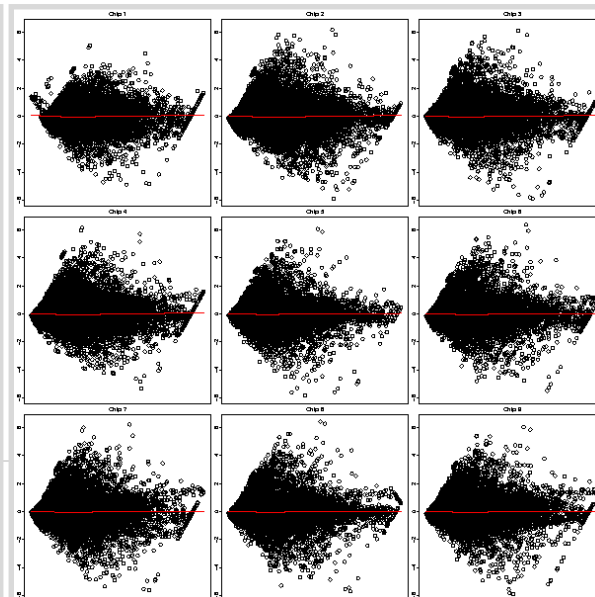
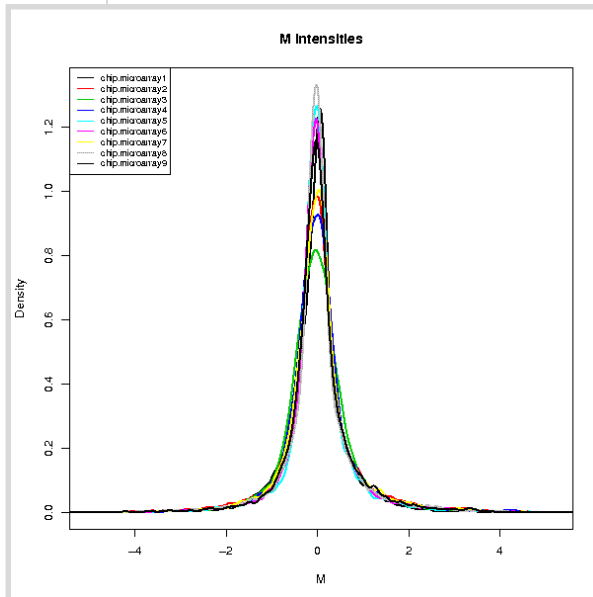
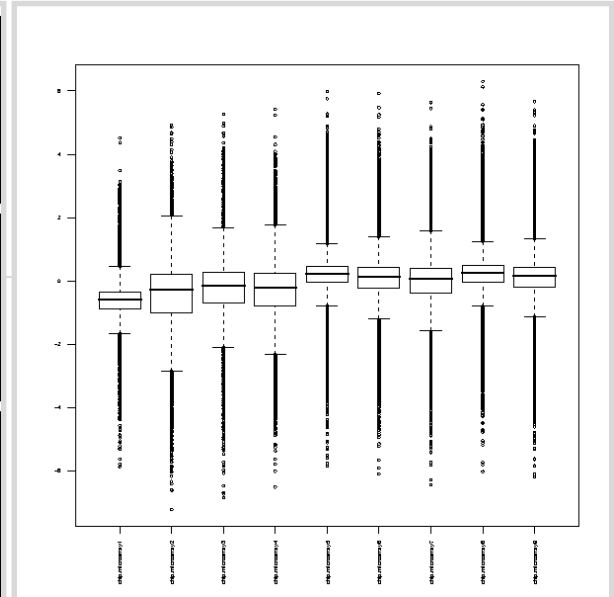
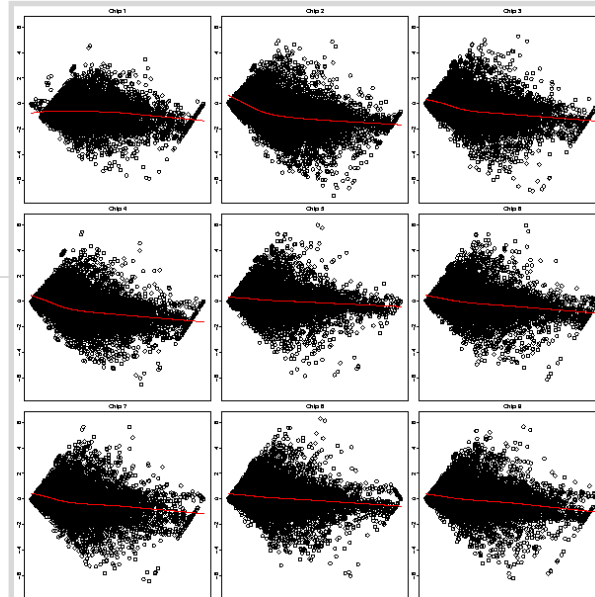
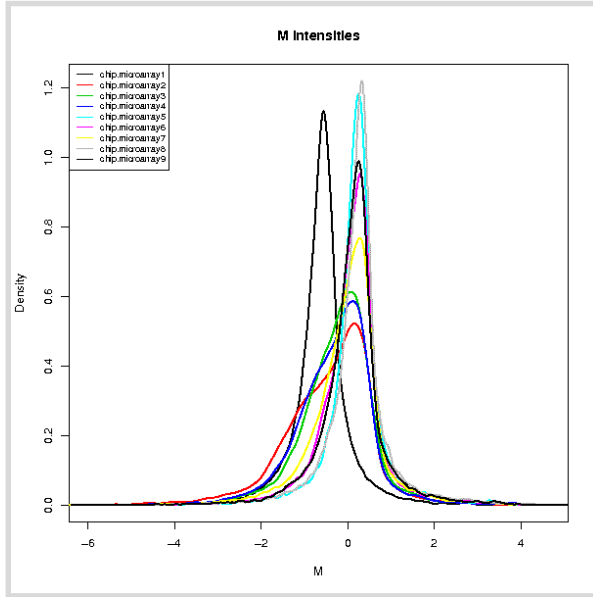
## ➤ **Chiptype**

- You must give this information in order to use annotation-based tools later

## ➤ **Normalize genes**

- None, scale (to median), quantile
- not needed for statistical analysis

# Checking normalization



# Exercise 4: Normalize Illumina data

- **Select all the files by clicking on the box "13" in the Workflow view**
- **Select the tool Normalization / Illumina. Set parameters so that**
  - Illumina software version = BeadStudio1
  - identifier type = TargetID
  - chiptype = Human-6v1
- **Make an unnormalized, log-transformed file to be used as a comparison in exercise 6 when checking the normalization effect**
  - Repeat the run as before, but change Normalization method = none
  - Rename the result file to unnormalized.tsv

# Microarray data analysis workflow

- Importing data to Chipster
- Normalization
- **Describing samples with a phenodata file**
- Quality control
  - Array level
  - Experiment level
- Filtering (optional)
- Statistical testing
  - Parametric and non-parametric tests
  - Linear modeling
  - Multiple testing correction
- Annotation
- Pathway analysis
- Clustering
- Saving the workflow

# Phenodata file

- Experimental setup is described with a phenodata file, which is created during normalization
- Fill in the group column with numbers describing your experimental groups
  - e.g. 1 = control sample, 2 = cancer sample
  - necessary for the statistical tests to work
  - note that you can sort a column by clicking on its title

The screenshot shows a software interface with two main panels. The left panel, titled 'Workflow view', displays a flowchart with '17 files' at the top, branching into 'Norm' and 'Phe' (circled in red). The right panel, titled 'Visualisation', shows a table with the following columns: sample, original\_name, chiptype, group (circled in red), training, and description. The table contains 17 rows of data, with the first 8 rows having a 'group' value of 1 and the remaining 9 rows having a 'group' value of 2. The 'Method' dropdown is set to 'Phenodata editor'.

sample	original_name	chiptype	group	training	description
microarray010.cel	GSM11805.cel	hgu133a	1		GSM11805.cel
microarray011.cel	GSM11823.cel	hgu133a	1		GSM11823.cel
microarray012.cel	GSM12075.cel	hgu133a	1		GSM12075.cel
microarray013.cel	GSM12098.cel	hgu133a	1		GSM12098.cel
microarray014.cel	GSM12268.cel	hgu133a	1		GSM12268.cel
microarray015.cel	GSM12283.cel	hgu133a	1		GSM12283.cel
microarray016.cel	GSM12300.cel	hgu133a	1		GSM12300.cel
microarray017.cel	GSM12444.cel	hgu133a	1		GSM12444.cel
microarray001.cel	cancerGSM1181...	hgu133a	2		cancerGSM11814.cel
microarray002.cel	cancerGSM1183...	hgu133a	2		cancerGSM11830.cel
microarray003.cel	cancerGSM1206...	hgu133a	2		cancerGSM12067.cel
microarray004.cel	cancerGSM1207...	hgu133a	2		cancerGSM12079.cel
microarray005.cel	cancerGSM1210...	hgu133a	2		cancerGSM12100.cel
microarray006.cel	cancerGSM1210...	hgu133a	2		cancerGSM12105.cel
microarray007.cel	cancerGSM1227...	hgu133a	2		cancerGSM12270.cel
microarray008.cel	cancerGSM1229...	hgu133a	2		cancerGSM12298.cel

# How to describe pairing, replicates, time, etc?

- **You can add new columns to the phenodata file**
- **How to describe different variables**
  - **Time:** Use either real time values or recode with group codes
  - **Replicates:** All the replicates are coded with the same number
  - **Pairing:** Pairs are coded using the same number for each pair
  - **Gender:** Use numbers
  - **Anything else:** Use numbers

# Creating phenodata for normalized data

- **When you import data which has been already normalized, you need to create a phenodata file for it**
  - Use Import tool to bring the data in
  - Use the tool Normalize / Process prenormalized to create phenodata
    - Remember to give the chiptype
  - Fill in the group column
- **Note: If you already have a phenodata file, you can import it too**
  - Choose "Import directly" in the Import tool
  - Right click on normalized data, choose "Link to phenodata"

# Exercise 5: Describe the experiment

- **Double click the phenodata file of the real normalization**
- **In the phenodata editor, fill in the group column so that you enter**
  - 1 for control samples
  - 2 for teratospermia affected samples
- **For the interest of visualizations later on, give shorter names for the samples in the Description column**
  - Name the teratospermia samples t1, t2,.....t8
  - Name the control samples c1, c2 ,..c5



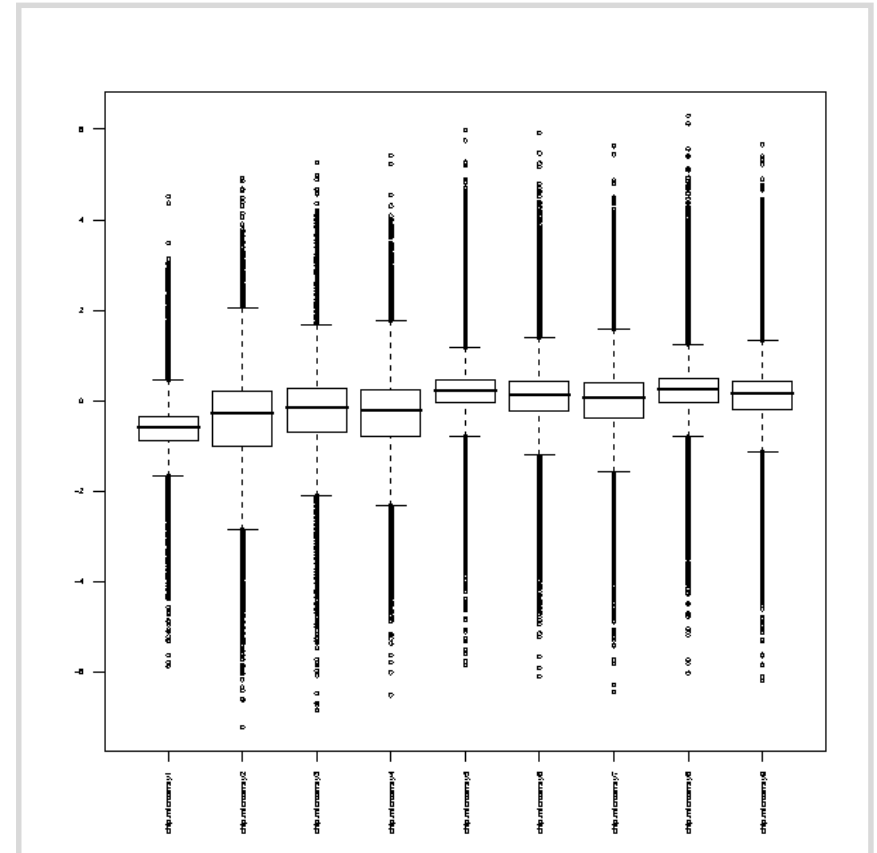
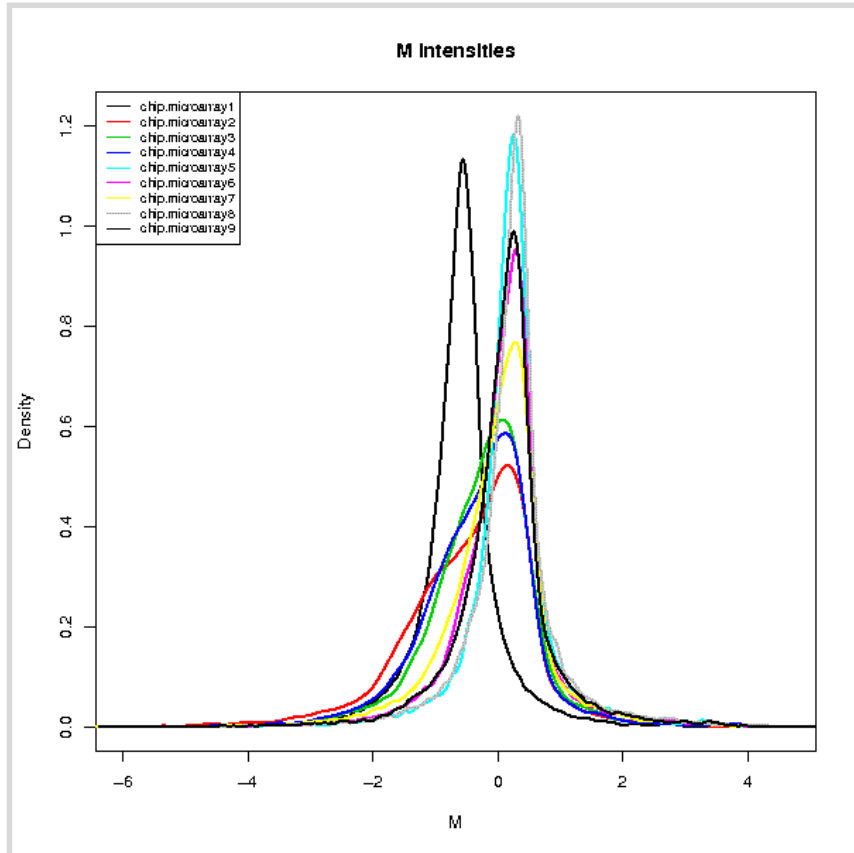
# Microarray data analysis workflow

- Importing data to Chipster
- Normalization
- Describing samples with a phenodata file
- **Quality control**
  - Array level
  - Experiment level
- **Filtering (optional)**
- **Statistical testing**
  - Parametric and non-parametric tests
  - Linear modeling
  - Multiple testing correction
- **Annotation**
- **Pathway analysis**
- **Clustering**
- **Saving the workflow**

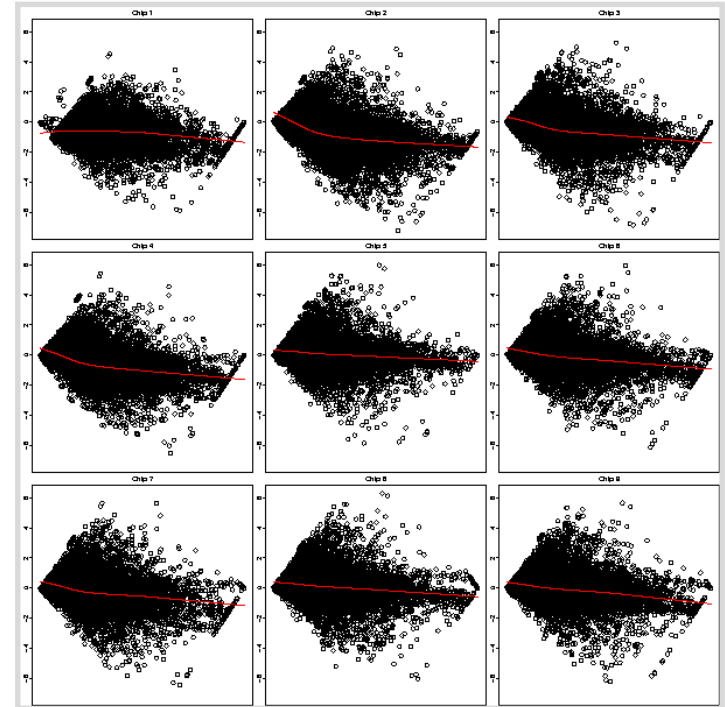
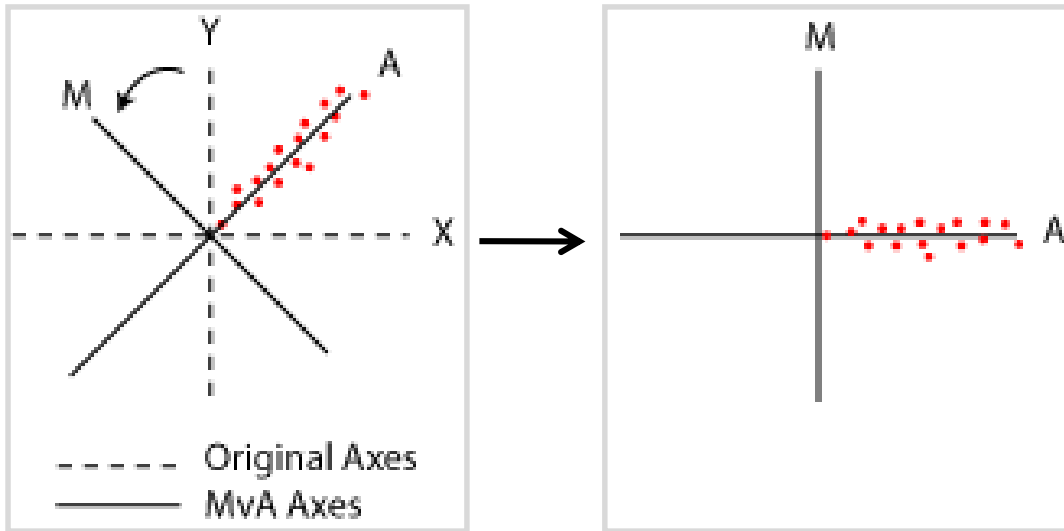
# Array level quality control

- **Allows you to check if arrays are comparable to each other**
- **Tools in Chipster**
  - Illumina: density plot and boxplot
  - Agilent 1-color: density plot and boxplot
  - Agilent 2-color: MA-plot, density plot and boxplot
  - Affymetrix basic: RNA degradation and Affy QC
  - Affymetrix RLE and NUSE: fit a model to expression values

# Density plot and box plot



# Agilent QC: MA-plot



- Scatter plot of log intensity ratios  $M = \log_2(R/G)$  versus average log intensities  $A = \log_2 \sqrt{R \cdot G}$ , where R and G are the intensities for the sample and control, respectively
- M is a mnemonic for minus, as  $M = \log R - \log G$
- A is mnemonic for add, as  $A = (\log R + \log G) / 2$

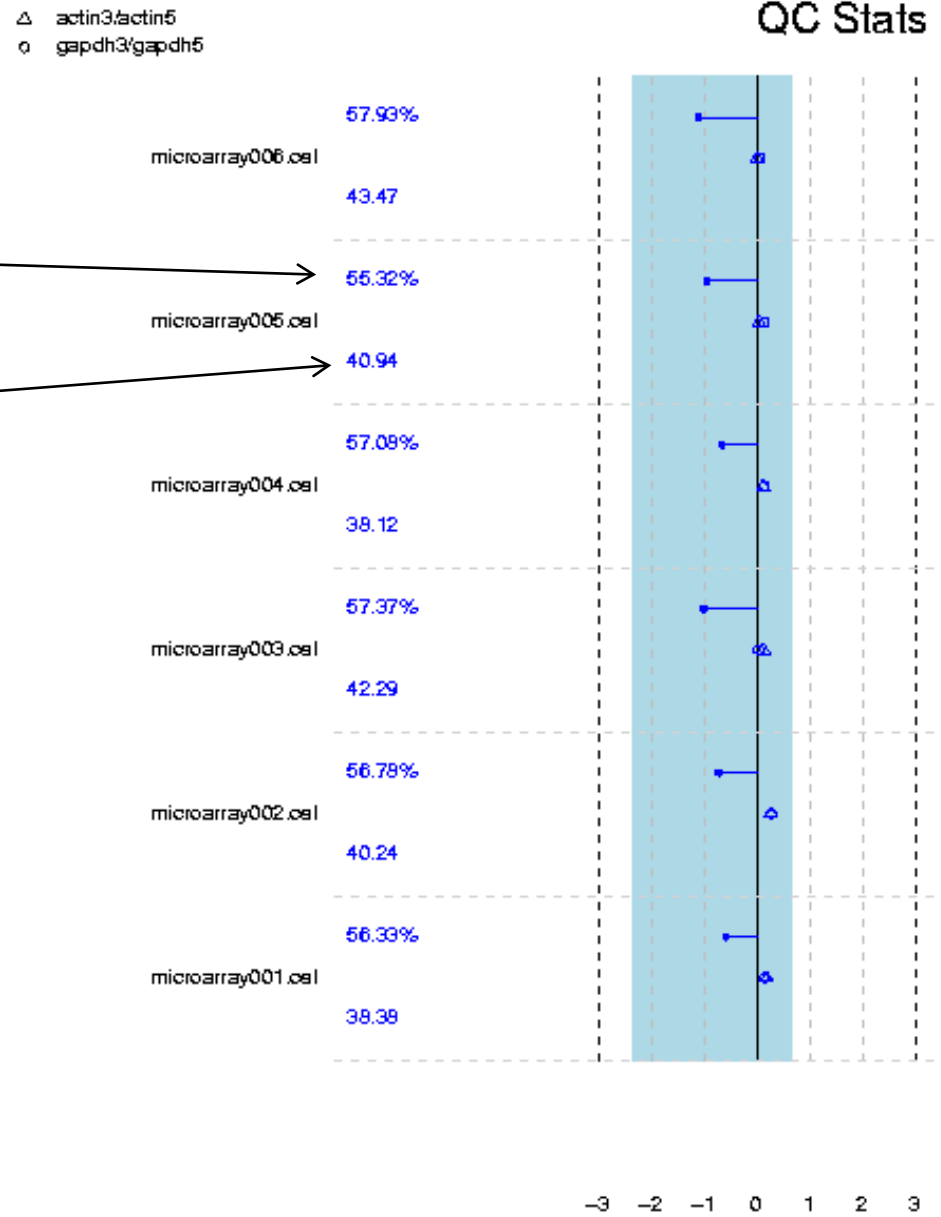
# Affymetrix QC

## ➤ **Several options**

- Affymetrix QC metrix
- RNA degradation
- Spike-in controls linearity
- RLE (relative log expression)
- NUSE (normalized unscaled standard error plot)

➤ **Note that Affymetrix array level QC tools are run on raw data (CEL files), not on normalized data**

# Affymetrix QC metrics



probesets with present flag

average background on the chip

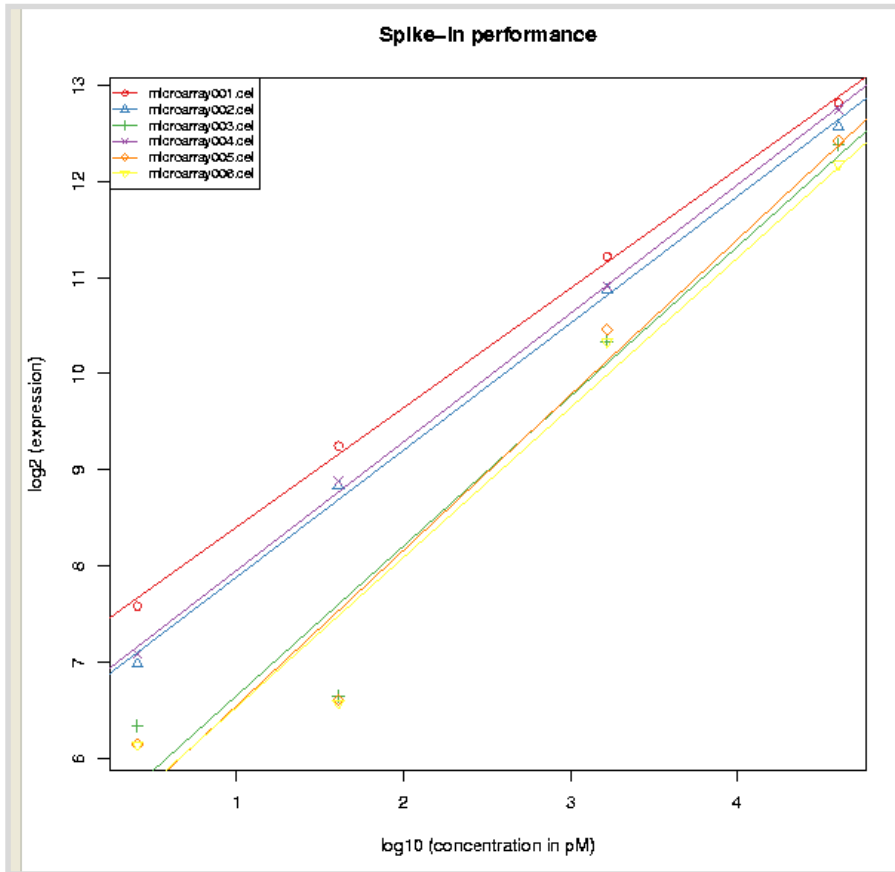
- scaling factors for the chips
- ▲ beta-actin 3':5' ratio
- GADPH 3':5' ratio

Blue area shows where scaling factors are less than 3-fold of the mean.

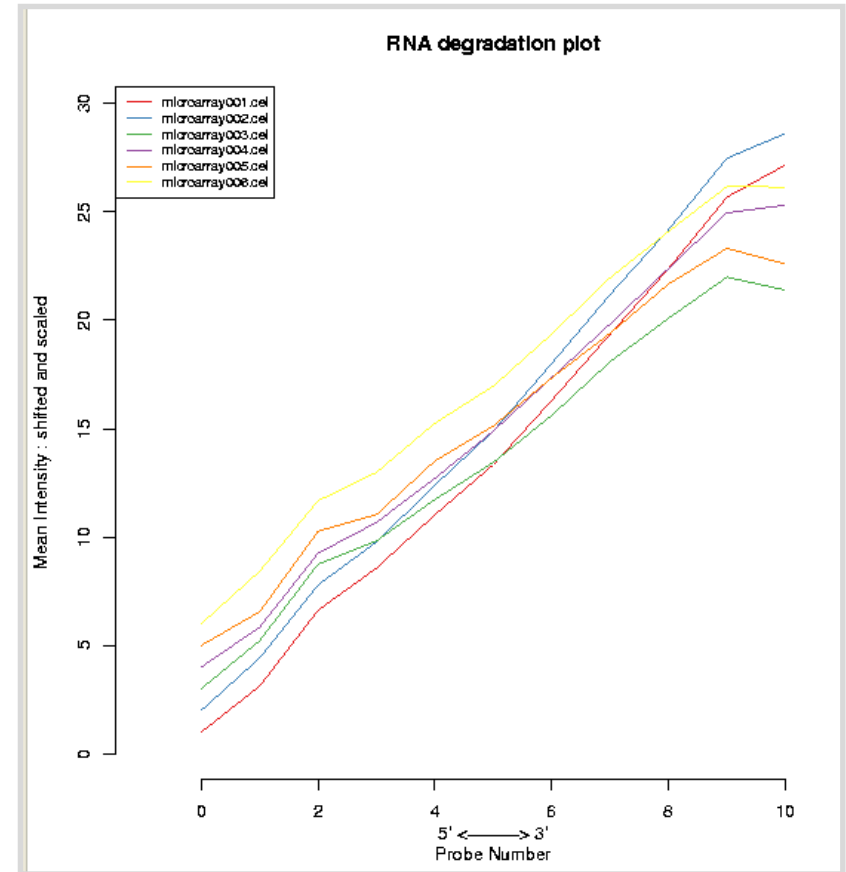
- If the scaling factors or ratios fall within this region (1.25-fold for GADPH), they are colored **blue**, otherwise **red**

# Affymetrix spike-ins and RNA degradation

## Spike-in linearity

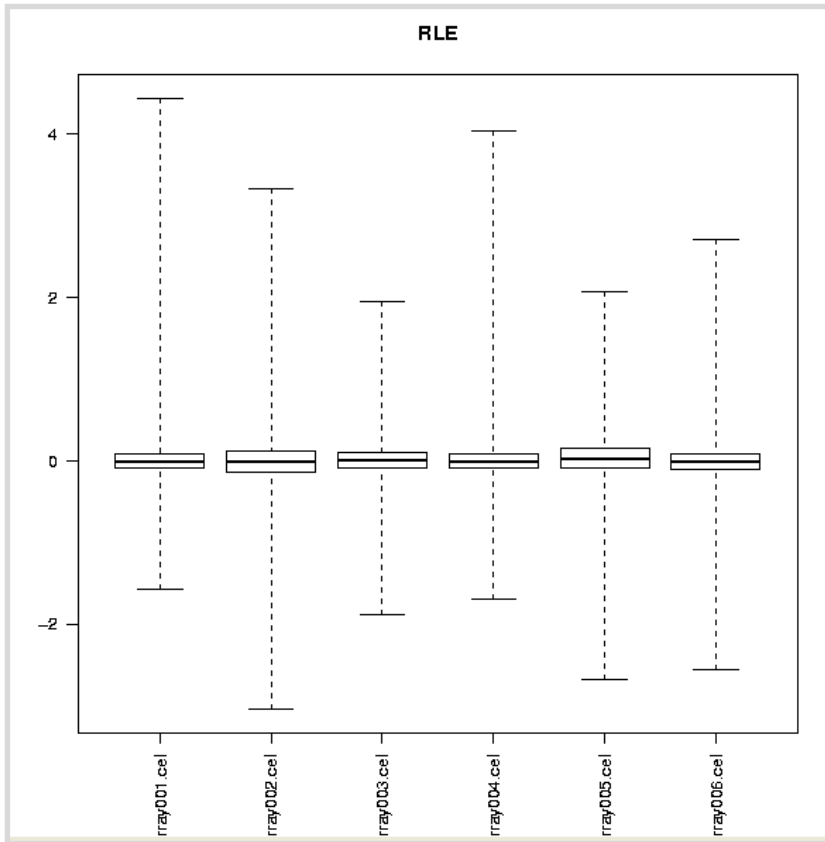


## RNA degradation plot

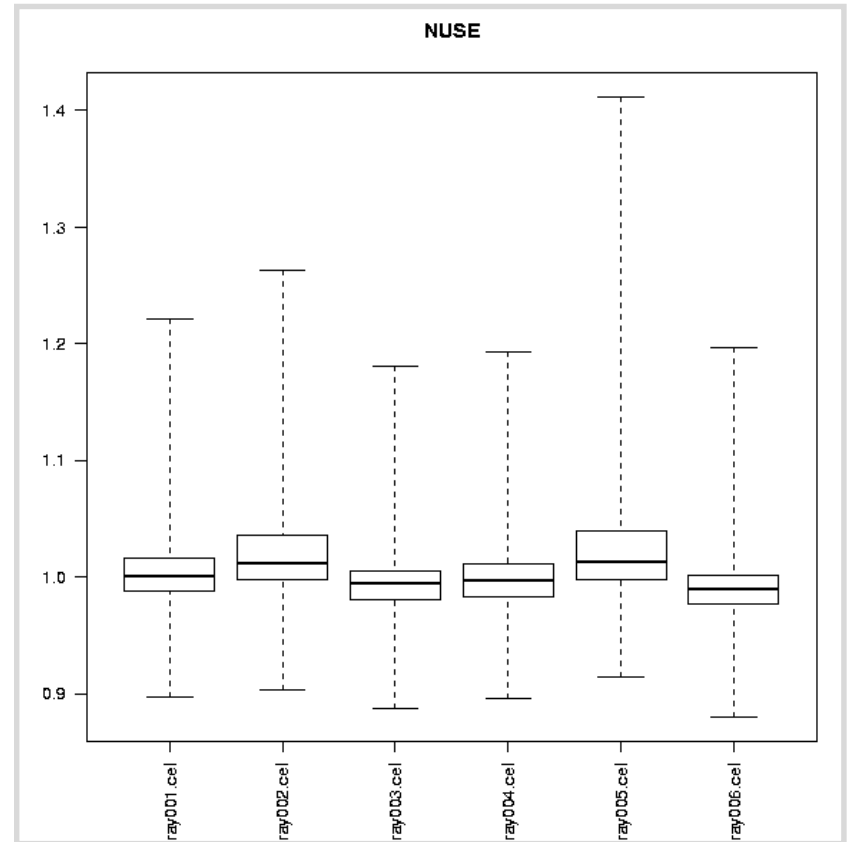


# Affymetrix: RLE and NUSE

RLE (relative log expression)



NUSE (normalized unscaled standard error)





# Exercise 6: Illumina array level quality control

- Run Quality control / Illumina for the normalized data
- Repeat this for the file `unnormalized.tsv` and compare the results (use the **Detach** button to view the images side by side). Can you see the effect of normalization?

# Microarray data analysis workflow

- Importing data to Chipster
- Normalization
- Describing samples with a phenodata file
- **Quality control**
  - Array level
  - Experiment level
- Filtering (optional)
- **Statistical testing**
  - Parametric and non-parametric tests
  - Linear modeling
  - Multiple testing correction
- Annotation
- Pathway analysis
- Clustering
- Saving the workflow

# Experiment level quality control

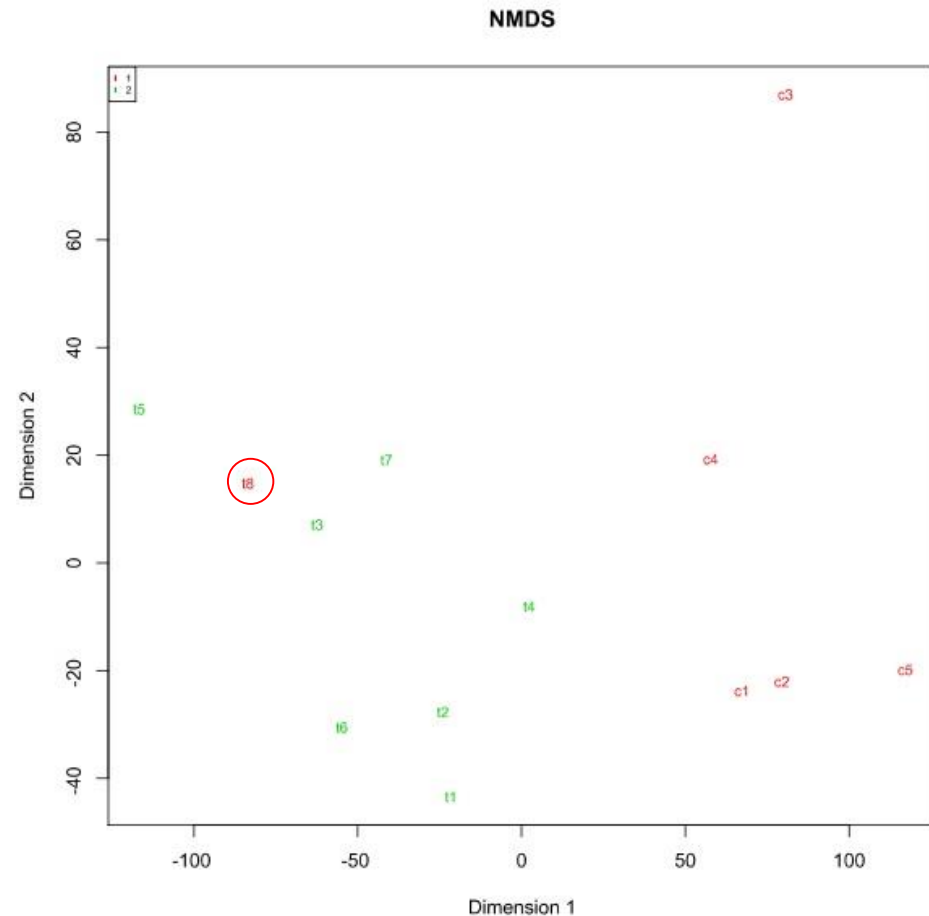
- **Getting an overview of similarities and dissimilarities between samples allows you to check**
  - Do the experimental groups separate from each other?
  - Is there a confounding factor (e.g. batch effect) that should be taken into account in the statistical analysis?
  - Are there sample outliers that should be removed?
- **Several methods available**
  - NMDS (non-metric multidimensional scaling)
  - PCA (principal component analysis)
  - Clustering
  - Dendrogram
  - Correlogram

# Non-metric multidimensional scaling (NMDS)

- **Goal is to reduce dimensions from several thousands to two**
  - High dimensional space is projected into a 2-dimensional space
- **Check that the experimental groups separate on dimension 1**
  - Do the samples separate according to something else on dimension 2?

## ➤ Method

- Computes a distance matrix for all genes
- Constructs the dimensions so that the similarity of distances between the original and the 2-dimensional space is maximized

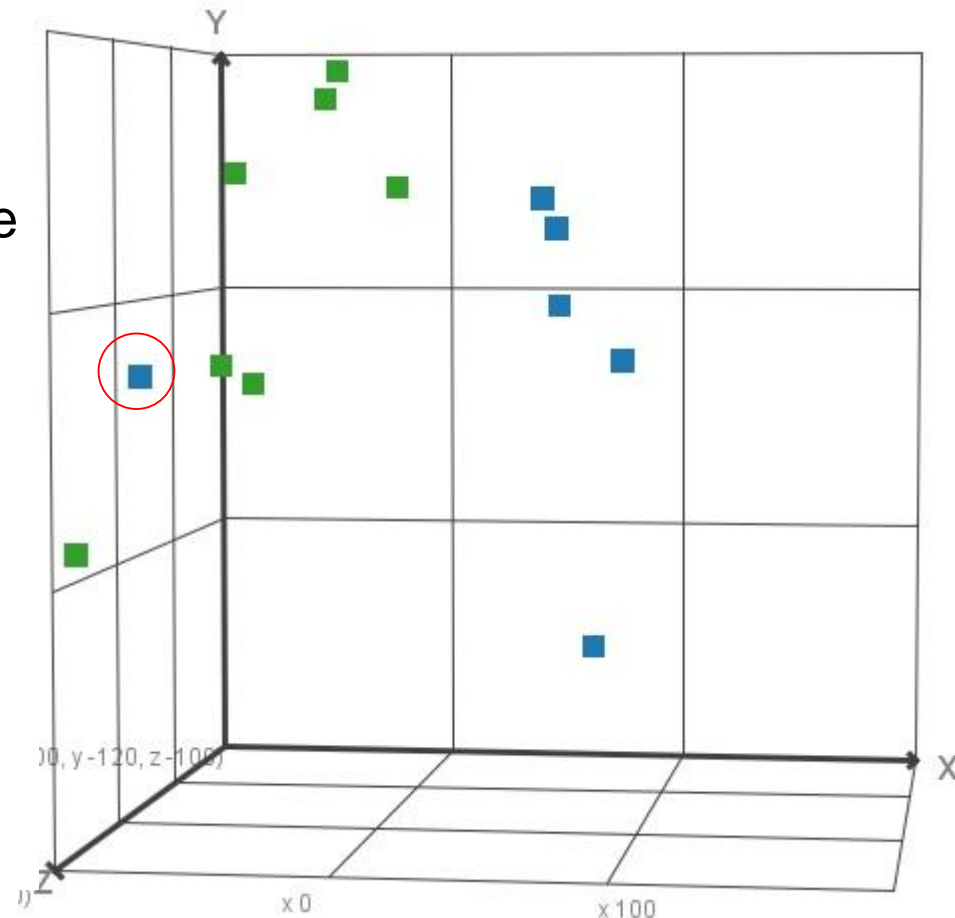


# Principal component analysis (PCA)

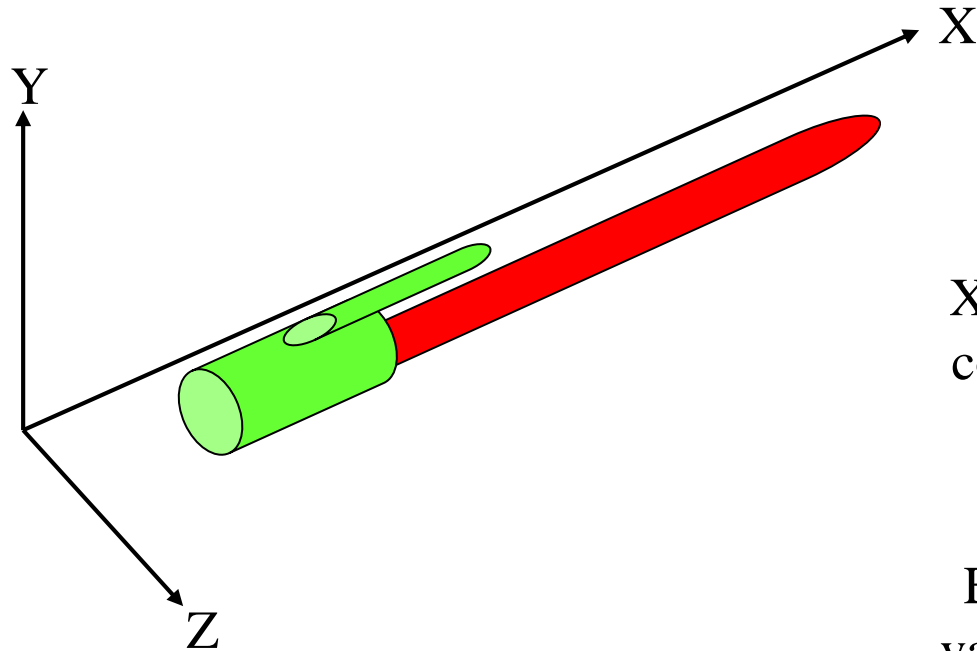
- **Goal is to reduce dimensions**
  - High dimensional space is projected into a lower dimensional space
- **Check the percentage of variance explained by each component**
  - If PC2 explains only a small percentage of variance, it can be ignored.

## ➤ Method

- Computes a variance-covariance matrix for all genes
- PC1, the first principal component, is the linear combination of variables that maximizes the variance
- PC2 is a linear combination orthogonal to the previous one which maximizes variance.
- etc



# PCA illustration



X is the first principal component of the pen



Explains most of the variability in the shape of the pen



Z-Y



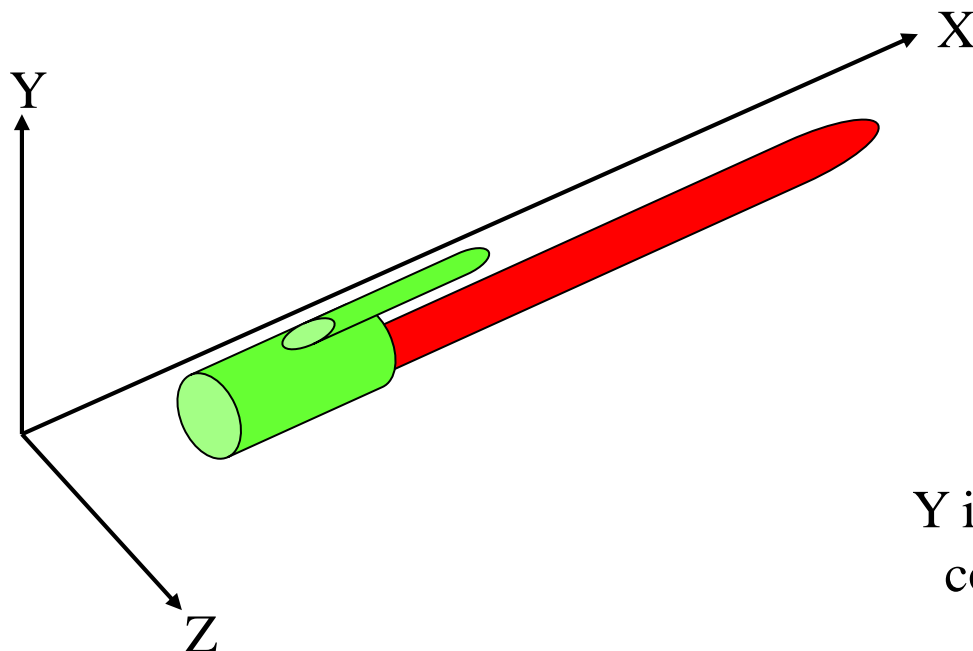
Z-X



X-Y



# PCA illustration, continued



Y is the second principal component of the pen



Explains most of the remaining variability in the shape of the pen



Z-Y



Z-X

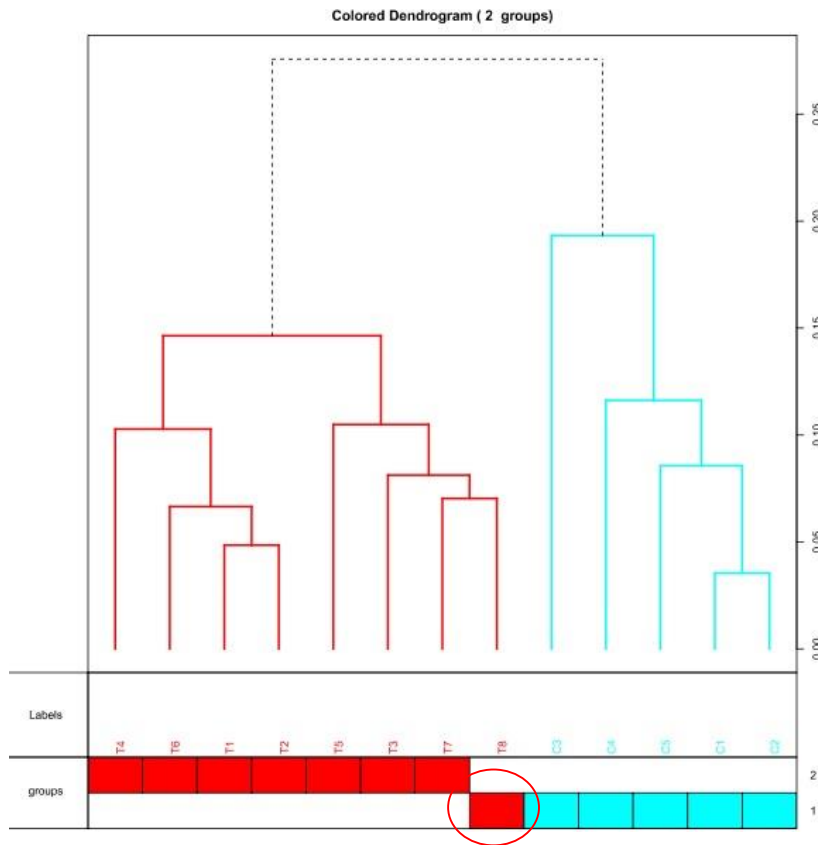


Y-X

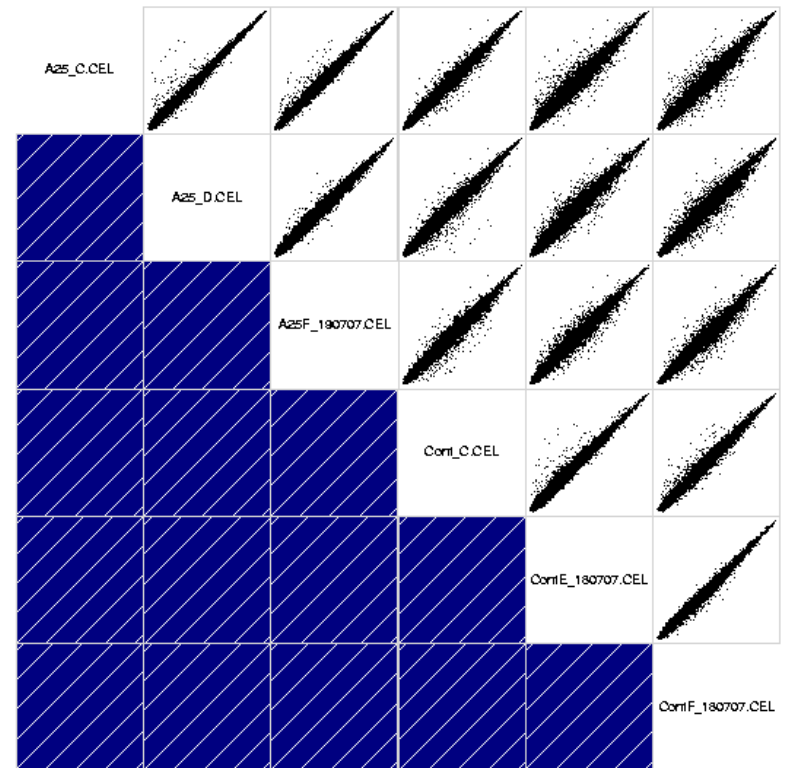


# Dendrogram and correlogram

## Dendrogram



## Correlogram





# Exercise 7: Experiment level quality control

- **Run Statistics / NMDS for the normalized data**
  - Do the groups separate along the first dimension?
- **Run Statistics / PCA on the normalized data.**
  - View **pca.tsv** as **3D scatter plot for PCA**. Can you see 2 groups?
  - Check in **variance.tsv** how much variance the first principal component explains? And the second one?
- **Run Visualization / Dendrogram for the normalized data**
  - Do the groups separate well?
- **Save the analysis session with name `sessionTeratospermia.zip`**

# Microarray data analysis workflow

- Importing data to Chipster
- Normalization
- Describing samples with a phenodata file
- Quality control
  - Array level
  - Experiment level
- **Filtering (optional)**
- Statistical testing
  - Parametric and non-parametric tests
  - Linear modeling
  - Multiple testing correction
- Annotation
- Pathway analysis
- Clustering
- Saving the workflow

# Filtering

## ➤ Why?

- Reducing the number of genes tested for differential expression reduces the severity of multiple testing correction of p-values. As the p-values remain better, we detect more differentially expressed genes.

## ➤ Why not?

- Some statistical testing methods (inc. the empirical Bayes option in Chipster) need many genes, because they estimate variance by borrowing information from other genes which are expressed at similar level. Hence the more genes the better.

## ➤ Filtering should

- remove genes which don't have any chance of being differentially expressed: genes that are not expressed or don't change
- be independent: should not use the sample group information

# Filtering tools in Chipster

- **Filter by standard deviation (SD)**
  - Select the percentage of genes to be filtered out
- **Filter by coefficient of variation (CV = SD / mean)**
  - Select the percentage of genes to be filtered out
- **Filter by flag**
  - Flag value and number of arrays
- **Filter by expression**
  - Select the upper and lower cut-offs
  - Select the number of chips required to fulfil this rule
- **Filter by interquartile range (IQR)**
  - Select the IQR

# Exercise 8: Filtering

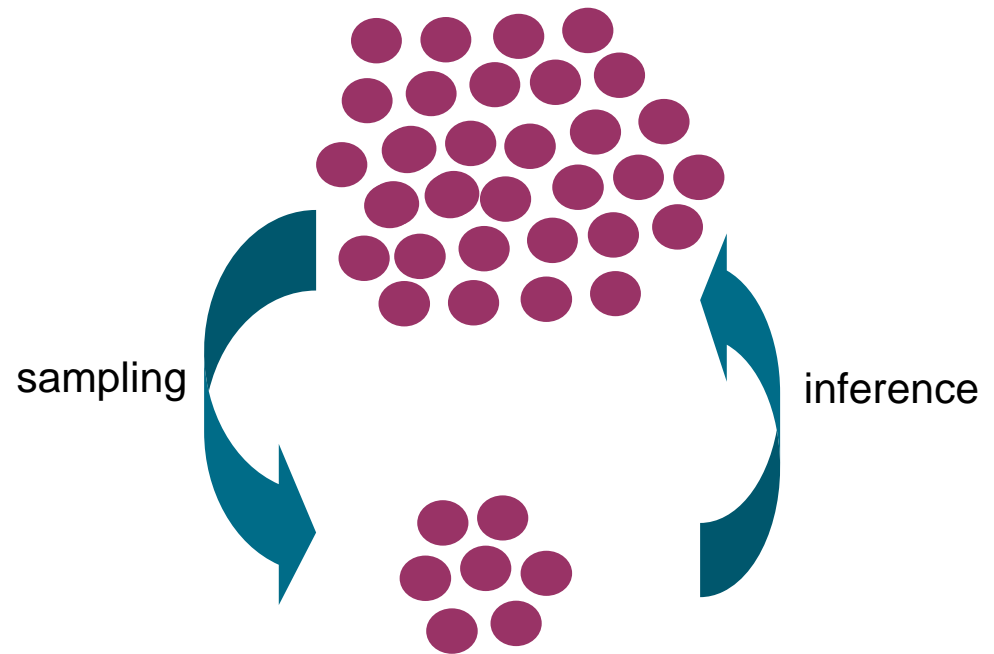
- **Select the normalized data and play with the SD filter and CV filter.**
  - Set the cutoffs so that you filter out 90% of genes (Percentage to filter out = 0.9).
  - Preprocessing / Filter by SD
  - Preprocessing / Filter by CV
  
- **Select the result files and compare them using the interactive Venn diagram visualization**
  - Save the genes specific to SD filter to a new file. Rename it sd.tsv.
  - Save the genes specific to CV filter to a new file. Rename it cv.tsv.
  - View both as expression profiles. Is there a difference in expression levels of the two sets?

# Microarray data analysis workflow

- Importing data to Chipster
- Normalization
- Describing samples with a phenodata file
- Quality control
  - Array level
  - Experiment level
- Filtering (optional)
- **Statistical testing**
  - Parametric and non-parametric tests
  - Linear modeling
  - Multiple testing correction
- Annotation
- Pathway analysis
- Clustering
- Saving the workflow

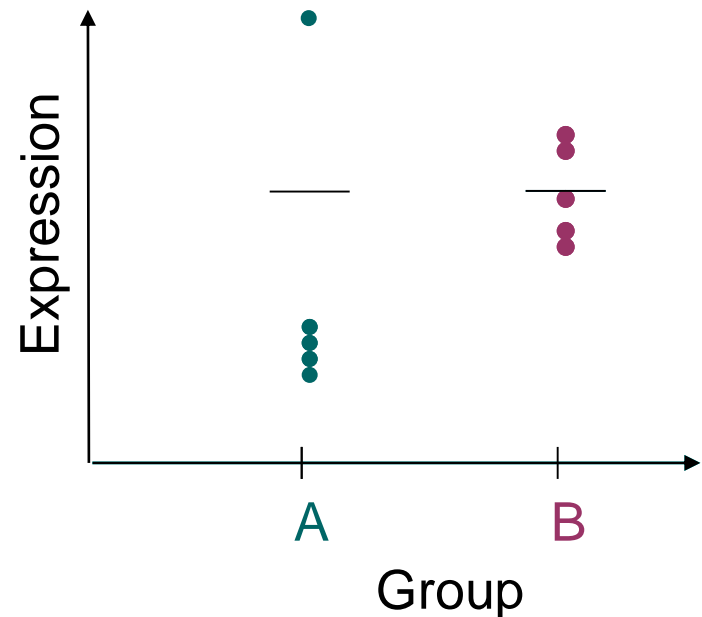
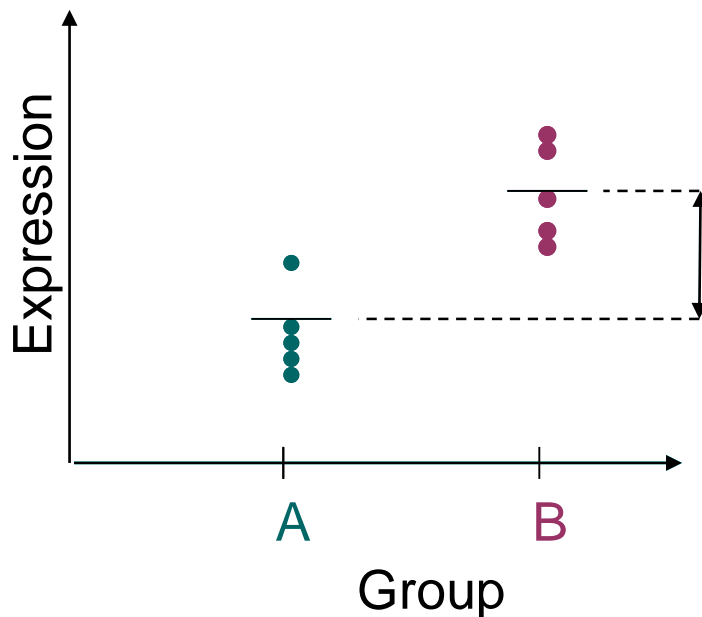
# Statistical analysis: Why?

- **Distinguish the treatment effect from biological variability and measurement noise**
  - replicates
  - estimation of uncertainty (variability)
  
- **Generalisation of results**
  - representative sample
  - statistical inference



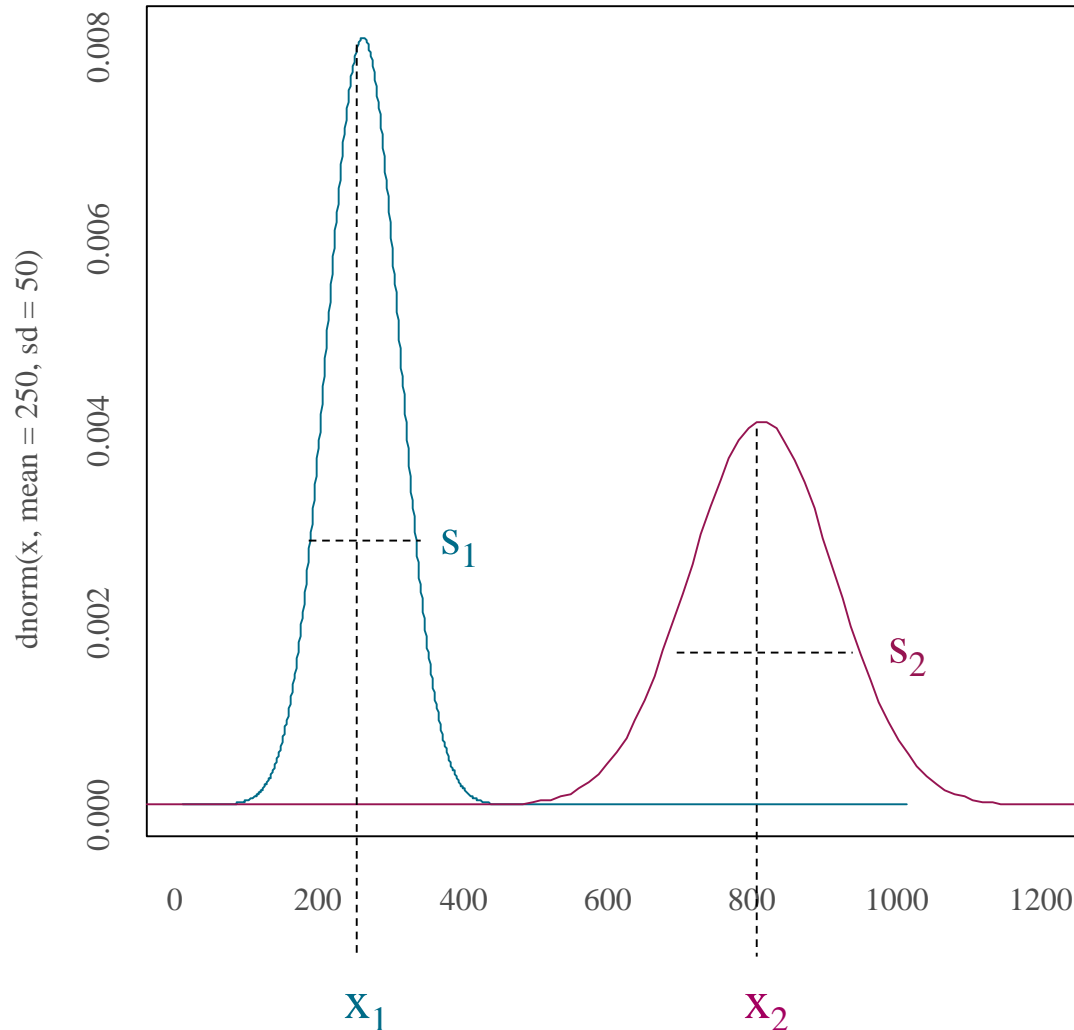
# Parametric statistical methods

- **Comparing means of 1-2 groups**
  - student's t-test
- **Comparing means of more than 2 groups**
  - 1-way ANOVA
- **Comparing means in a multifactor experiment**
  - 2-way ANOVA





# Parametric statistics



$$t = \frac{x_1 - x_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}}$$

$$H_0 : \mu_A = \mu_B, \mu_A - \mu_B = 0$$

$$H_1 : \mu_A \neq \mu_B$$

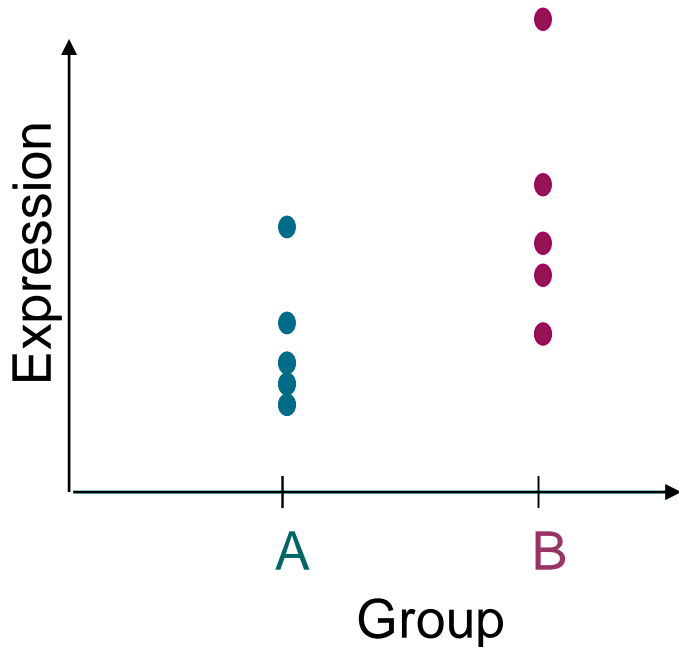
Type 1 error,  $\alpha$

Type 2 error,  $\beta$

Power =  $1 - \beta$

# Non-parametric statistical methods

- **Comparing ranks of 2 groups**
  - Mann-Whitney
- **Comparing ranks of more than 2 groups**
  - Kruskal-Wallis



Ranks	
group A	group B
1	4
2	6
3	7
5	9
8	10

$$U_1 = n_1 * n_2 + \frac{n_1 * (n_1 + 1)}{2} - R_1$$

$$U_2 = n_1 * n_2 + \frac{n_2 * (n_2 + 1)}{2} - R_2$$

# Non-parametric compared to parametric tests

## Benefits

- Do not make any assumptions on data distribution
  - ⇒ robust to outliers
  - ⇒ allow for cross-experiment comparisons

## Drawbacks

- Lower power than parametric counterpart
- Granular distribution of calculated statistic
  - ⇒ many genes get the same rank
  - ⇒ requires at least 6 samples / group

# How to improve statistical power?

- **Need more accurate estimates of variability and effect size**
- **Improved analysis methods**
  - Variance shrinking: Empirical Bayes method
  - Partitioning variability: ANOVA, linear modeling
- **Improved experimental design**
  - Increase number of biological replicates
  - Use paired samples if possible
  - Randomization
  - Blocking

# Pairing = matched samples from the same individual

## Unpaired analysis

	Before	After
	2	3
	2	4
	3	2
	1	3
Mean	2	3
Stdev	0.8	0.8

## Paired analysis

Before	After	Difference
2	3	1
2	3	1
3	4	1
1	2	1

One sample T-test

# Improving power with variance shrinking

## ➤ **Concept**

- Borrow information from other genes which are expressed at similar level, and form a pooled error estimate

## ➤ **How?**

- models the error - intensity dependence by comparing replicates
- uses a smoothing function to estimate the error for any given intensity
- calculates a weighted average between the observed gene specific variance and the model-derived variance (pooling)
- incorporates the pooled variance estimate in the statistical test (usually t- or F-test)

## ➤ **Available in Chipster**

- Two group test: Select empirical Bayes as the test
- Linear modeling tool

# Microarray data analysis workflow

- Importing data to Chipster
- Normalization
- Describing samples with a phenodata file
- Quality control
  - Array level
  - Experiment level
- Filtering (optional)
- **Statistical testing**
  - Parametric and non-parametric tests
  - Linear modeling
  - Multiple testing correction
- Annotation
- Pathway analysis
- Clustering
- Saving the workflow

# Linear modeling

- **Models the expression of a gene as a linear combination of explanatory factors (e.g. group, gender, time, patient,...)**

$$y = a + (b \cdot \text{group}) + (c \cdot \text{gender}) + (d \cdot \text{group} \cdot \text{gender})$$

y = gene's expression

a, b, c and d = parameters estimated from the data

a = intercept (expression when factors are at "reference" level)

b and c = main effects

d = interaction effect



# Taking multiple factors into account

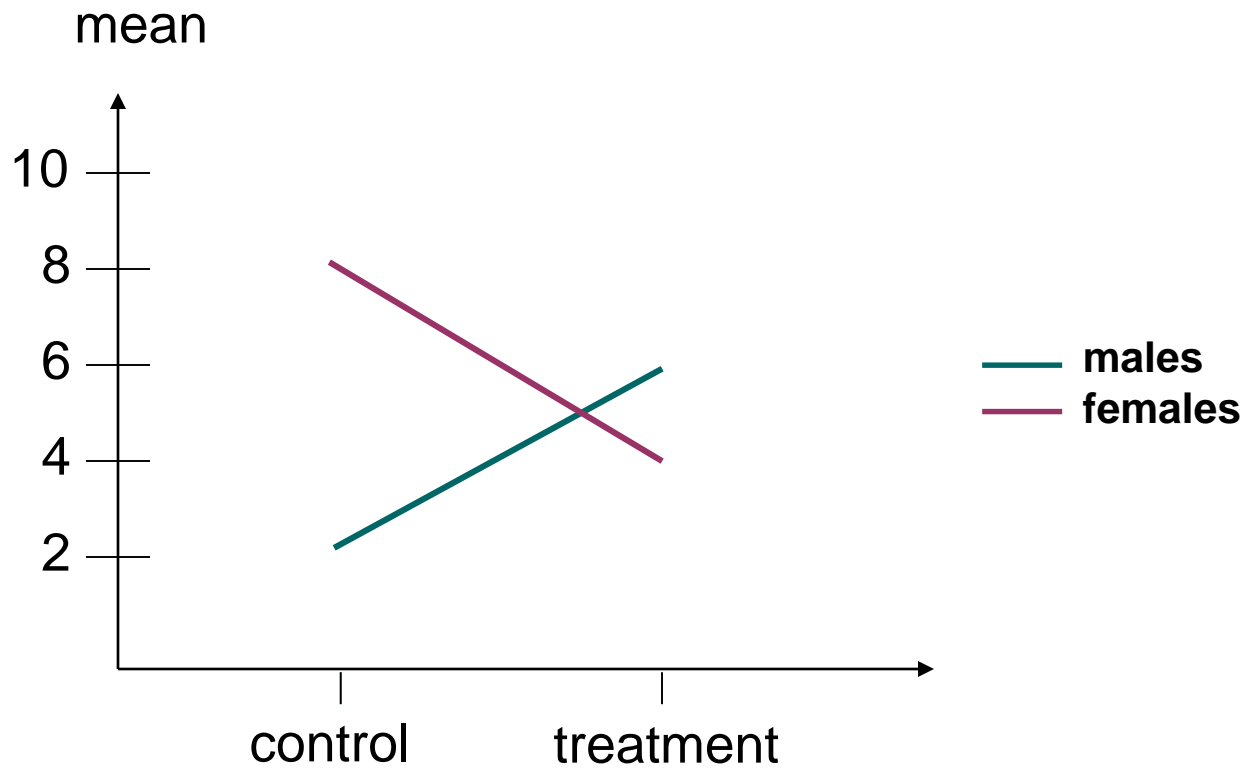
## 1 factor: treatment

	Control	Treatment
	2	5
	9	7
	1	3
	7	5
	8	4
	3	6
Mean	5	5

## 2 factors: treatment and gender

	Control	Treatment
Males	2	6
	3	7
	1	5
	Mean	2
Females	8	4
	9	5
	7	3
	Mean	8

# Linear modeling: Interaction effect



# Linear modeling tool in Chipster

## ➤ **Linear modeling tool in Chipster can take into account**

- 3 main effects
- Their interactions
- Pairing
- Technical replication (one sample is hybridized to several arrays)

## ➤ **Main effects can be tested as**

- Linear = is there a trend towards higher numbers?
- Factor = are there differences between the groups?

If the main effect has only two levels (e.g. gender), selecting linear or factor gives the same result

## ➤ **Note that the result table contains all the genes, so to get the differentially expressed genes you have to filter it**

- Use the tool **Filter using a column value**
- Select the p.adjusted column that corresponds to the comparison of your interest

# Microarray data analysis workflow

- Importing data to Chipster
- Normalization
- Describing samples with a phenodata file
- Quality control
  - Array level
  - Experiment level
- Filtering (optional)
- **Statistical testing**
  - Parametric and non-parametric tests
  - Linear modeling
  - Multiple testing correction
- Annotation
- Pathway analysis
- Clustering
- Saving the workflow

# Multiple testing correction

- **Problem: When thousands of genes are tested for differential expression, a gene can get a good p-value just by chance.**

1 gene,  $\alpha = 0.05$

⇒ false positive incidence = 1 / 20

30 000 genes,  $\alpha = 0.05$

⇒ false positive incidence = 1500

- **Solution: Correct the p-values for multiple testing. Methods:**

- Bonferroni
- Holm (step down)
- Westfall & Young
- Benjamini & Hochberg



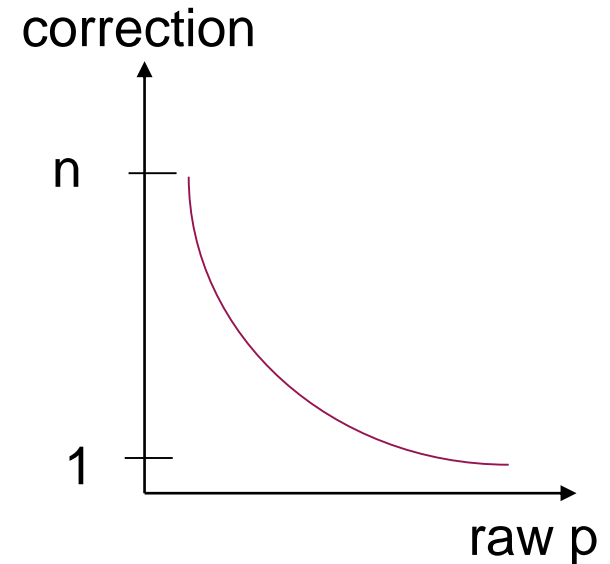
more false negatives

more false positives

# Benjamini & Hochberg method (BH)

## ➤ How does it work?

- rank p-values from largest to smallest
- largest p-value remains unaltered
- second largest p-value =  $p * n / (n-1)$
- third largest p-value =  $p * n / (n-2)$
- ...
- **smallest p-value** =  $p * n / (n-n+1) = p * n$



## ➤ We can reduce the severity of multiple testing correction by reducing the number of genes tested (n)

- use independent filtering

## ➤ The adjusted p-value is FDR (false discovery rate)

- Tells what proportion of results can be false positives

# Exercise 9: Statistical testing

## ➤ Run different two group tests

- Select the file **cv-filter.tsv** and **Statistics / Two group test**. What is the default value of parameter test? How many differentially expressed genes do you get?
- Repeat the run but change **test = t-test**. Rename the result file to **t.tsv**. How many differentially expressed genes do you get now?
- Repeat the run but change **test = Mann-Whitney**. Rename the result file to **MW.tsv**. How many differentially expressed genes do you get now?

## ➤ Compare the results with a Venn diagram

- Which method seems most powerful?
- Select the genes common to all three datasets and create a new dataset.

# Exercise 10: Visualize and filter results

- **View the Empirical Bayes result as an interactive volcano plot and filter genes based on visual selection**
  - Select the **two-sample.tsv** and visualization method **Volcano plot**
  - Draw a box around the genes whose  $\log_2 \text{FC} > 3$  and create a new dataset from this selection.
  - Visualize the new file as **Expression profile**
  
- **Filter genes based on fold change using an analysis tool**
  - Select **two-sample.tsv** and the tool **Preprocessing / Filter using a column value**. Keep genes whose  $\log_2 \text{FC} > 3$ :
    - Column = FC
    - Cut-off = 3
    - Smaller or larger = larger-than.



# Microarray data analysis workflow

- **Importing data to Chipster**
- **Normalization**
- **Describing samples with a phenodata file**
- **Quality control**
  - Array level
  - Experiment level
- **Filtering (optional)**
- **Statistical testing**
  - Parametric and non-parametric tests
  - Linear modeling
  - Multiple testing correction
- **Annotation**
- **Pathway analysis**
- **Clustering**
- **Saving the workflow**

# Annotation

- **Gene annotation = information about biological function, pathway involvement, chromosomal location etc**
- **Annotation information is collected from different biological databases to a single database by the Bioconductor project**
  - Bioconductor provides annotation packages for many microarrays
- **Annotation package is required by many analysis tools**
  - Annotation, GO/KEGG enrichment, promoter analysis, chromosomal plots
  - These tools don't work for those chiptypes which don't have Bioconductor annotation packages

## Annotations for the selected gene list

Probe	Symbol	Description	Chromosome	Chromosome Location	GenBank	Gene	Cytoband	UniGene	PubMed	Gene Ontology	Pathway
<a href="#">205626_s_at</a>	CALB1	calbindin 1, 28kDa	8	-91140013	<a href="#">NM_004929</a>	<a href="#">793</a>	<a href="#">8q21.3-q22.1</a>	<a href="#">Hs.65425</a>	<a href="#">22</a>	<a href="#">locomotory behavior</a> <a href="#">cytoplasm</a> <a href="#">vitamin D binding</a> <a href="#">calcium ion binding</a> <a href="#">protein binding</a>	
<a href="#">220281_at</a>	SLC12A1	solute carrier family 12 (sodium/potassium/chloride transporters), member 1	15	46285789	<a href="#">AI632015</a>	<a href="#">6557</a>	<a href="#">15q15-q21.1</a>	<a href="#">Hs.123116</a>	<a href="#">13</a>	<a href="#">ion transport</a> <a href="#">potassium ion transport</a> <a href="#">sodium ion transport</a> <a href="#">chloride transport</a> <a href="#">membrane fraction</a> <a href="#">plasma membrane</a> <a href="#">membrane</a> <a href="#">integral to membrane</a> <a href="#">transporter activity</a> <a href="#">sodium:potassium:chloride symporter activity</a> <a href="#">symporter activity</a> <a href="#">potassium ion binding</a> <a href="#">sodium ion binding</a>	
<a href="#">206054_at</a>	KNG1	kininogen 1	3	187917813	<a href="#">NM_000893</a>	<a href="#">3827</a>	<a href="#">3q27</a>	<a href="#">Hs.77741</a>	<a href="#">86</a>	<a href="#">smooth muscle contraction</a> <a href="#">inflammatory response</a> <a href="#">negative regulation of cell adhesion</a> <a href="#">elevation of cytosolic calcium ion concentration</a> <a href="#">blood coagulation</a> <a href="#">diuresis</a> <a href="#">natriuresis</a> <a href="#">negative regulation of blood coagulation</a> <a href="#">vasodilation</a> <a href="#">positive regulation of apoptosis</a> <a href="#">extracellular region</a> <a href="#">cysteine protease inhibitor activity</a> <a href="#">receptor binding</a> <a href="#">heparin binding</a> <a href="#">zinc ion binding</a>	<a href="#">Complement and coagulation cascades</a>
										<a href="#">behavior</a> <a href="#">gamma-aminobutyric acid catabolic process</a> <a href="#">neurotransmitter catabolic</a>	<a href="#">Glutamate</a>

# Alternative CDF environments for Affymetrix

- **CDF is a file that links individual probes to gene transcripts**
- **Affymetrix default annotation uses old CDF files that map many probes to wrong genes**
- **Alternative CDFs fix this problem**
- **In Chipster selecting "custom chiptype" in Affymetrix normalization takes altCDFs to use**
- **For more information see**
  - Dai et al, (2005) *Nuc Acids Res*, 33(20):e175: *Evolving gene/transcript definitions significantly alter the interpretation of GeneChip data*
  - [http://brainarray.mbni.med.umich.edu/Brainarray/Database/CustomCDF/genomic\\_curated\\_CDF.asp](http://brainarray.mbni.med.umich.edu/Brainarray/Database/CustomCDF/genomic_curated_CDF.asp)

# Also a problem with Illumina

- **Probes are remapped in the R/Bioconductor project**
- **Chipster uses remapped probes**
- **For more information see**
  - Barbosa-Morais NL, Dunning MJ, Samarajiwa SA, Darot JFJ, Ritchie ME, Lynch AG, Tavaré S. "**A re-annotation pipeline for Illumina BeadArrays: improving the interpretation of gene expression data**". *Nucleic Acids Research*, 2009 Nov 18, doi:10.1093/nar/gkp942
  - <https://prod.bioinformatics.northwestern.edu/nuID/>

# Exercise 11: Annotation

## ➤ Annotate genes

- Select the file **column-value-filter.tsv**
- Run **Annotation / Illumina gene list** so that you include the FC and p-value information to the result file
- Open the result file **annotations.html** in external browser and explore the NASP gene by clicking on the link in the Gene column. Find the LEP gene and read about the JAK-STAT signaling pathway by clicking on the link in the pathway column.

# Microarray data analysis workflow

- **Importing data to Chipster**
- **Normalization**
- **Describing samples with a phenodata file**
- **Quality control**
  - Array level
  - Experiment level
- **Filtering (optional)**
- **Statistical testing**
  - Parametric and non-parametric tests
  - Linear modeling
  - Multiple testing correction
- **Annotation**
- **Pathway analysis**
- **Clustering**
- **Saving the workflow**

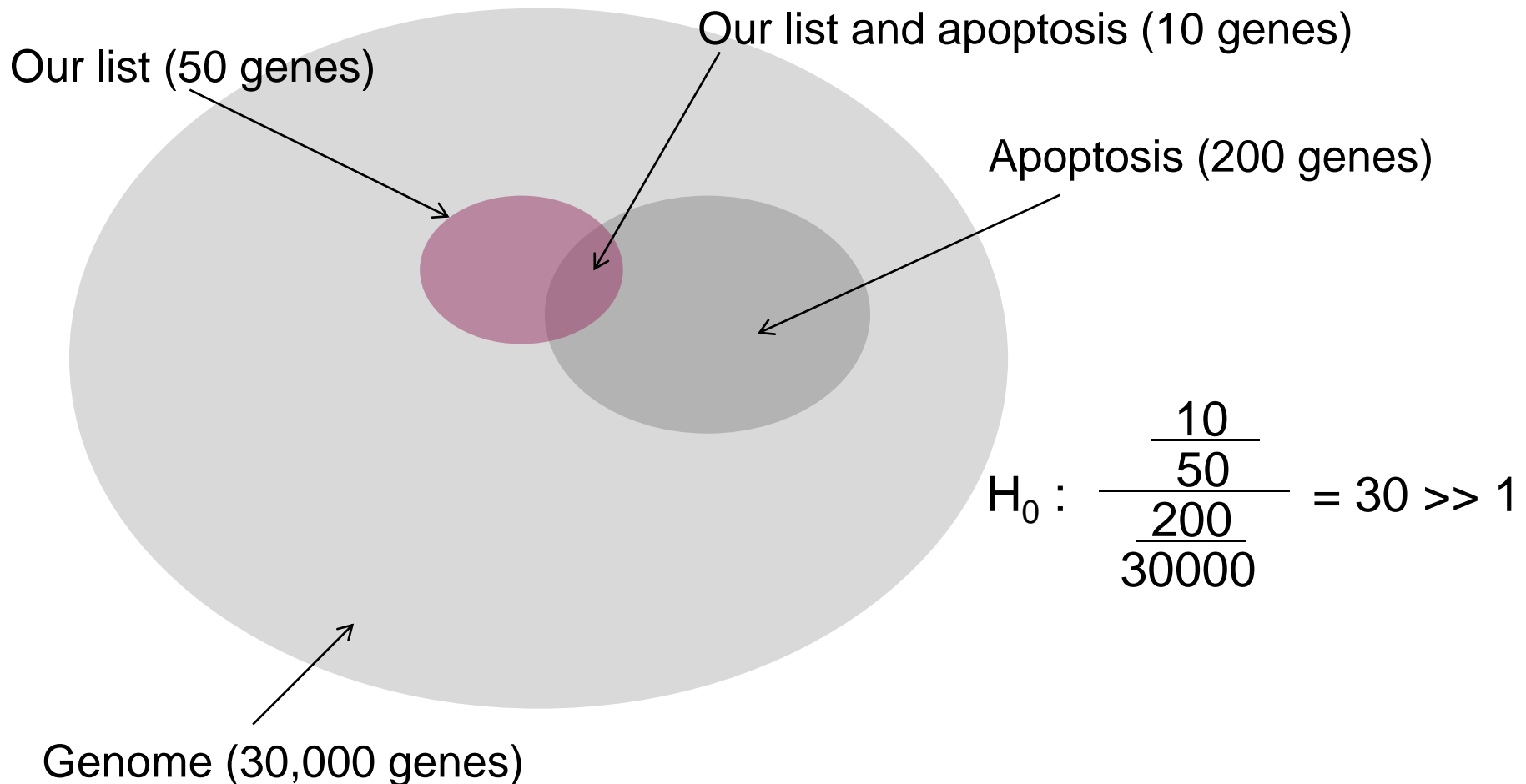
# Pathway analysis – why?

- **Statistical tests can yield thousands of differentially expressed genes**
- **It is difficult to make "biological" sense out of the result list**
- **Looking at the bigger picture can be helpful, e.g. which pathways are differentially expressed between the experimental groups**
- **Databases such as KEGG, GO, Reactome and ConsensusPathDB provide grouping of genes to pathways, biological processes, molecular functions, etc**
- **Two approaches to pathway analysis**
  - Gene set enrichment analysis
  - Gene set test



# Approach I: Gene set enrichment analysis

1. Perform a statistical test to find differentially expressed genes
2. Check if the list of differentially expressed genes is "enriched" for some pathways

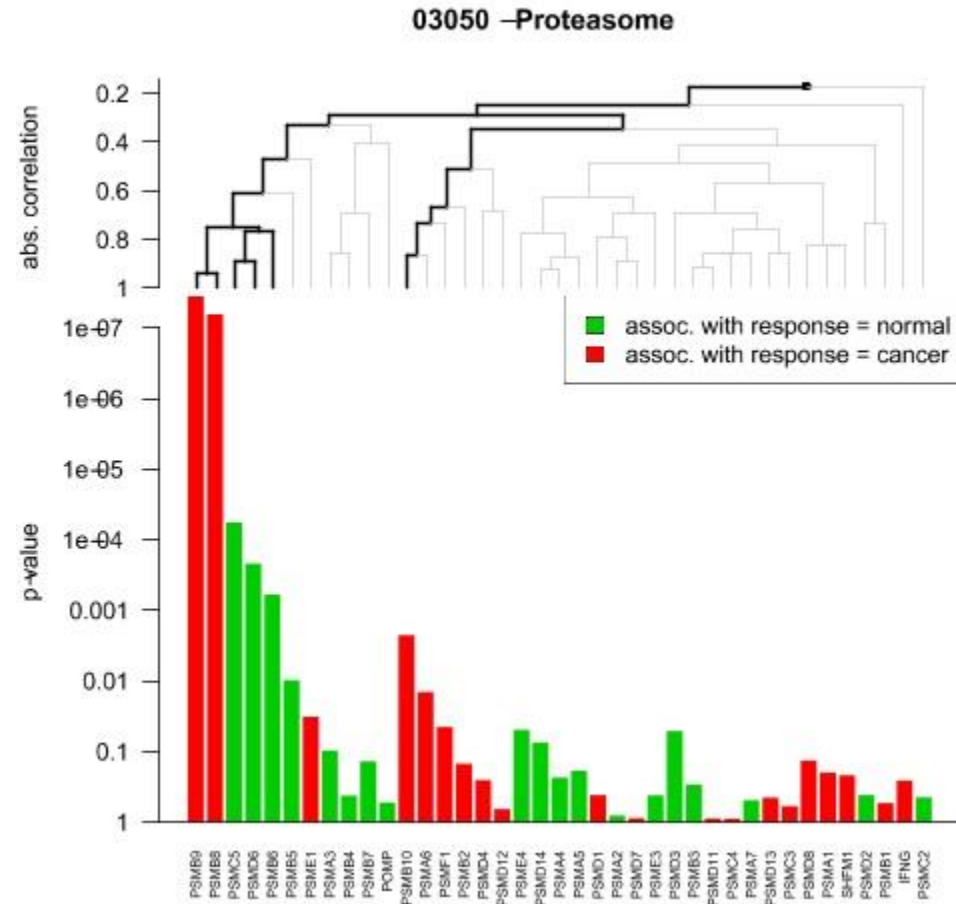


# Approach II: Gene set test

1. Do NOT perform differential gene expression analysis
2. Group genes to pathways and perform differential expression analysis for the whole pathway

## ➤ Advantages

- More sensitive than single gene tests
- Reduced number of tests  
→ less multiple testing correction  
→ increased power



# ConsensusPathDB

- **One-stop shop: Integrates pathway information from 32 databases covering**
  - biochemical pathways
  - protein-protein, genetic, metabolic, signaling, gene regulatory and drug-target interactions
- **Developed by Ralf Herwig's group at the Max-Planck Institute in Berlin**
- **ConsensusPathDB over-representation analysis tool is integrated in Chipster**
  - runs on the MPI server in Berlin

# GO (Gene Ontology)

➤ **Controlled vocabulary of terms for describing gene product characteristics**

➤ **3 ontologies**

- Biological process
- Molecular function
- Cellular component

➤ **Hierarchical structure**

▣ all : all [841457 gene products]

⊕ ⓘ GO:0008150 : biological\_process [660879 gene products]

⊕ ⓘ GO:0065007 : biological regulation [145630 gene products]

⊕ ⓘ GO:0050789 : regulation of biological process [134091 gene products]

⊕ ⓘ GO:0048518 : positive regulation of biological process [42078 gene products]

⊕ ⓘ GO:0048522 : positive regulation of cellular process [34658 gene products]

⊕ ⓘ GO:0031325 : positive regulation of cellular metabolic process [21272 gene products]

⊕ ⓘ GO:0032270 : positive regulation of cellular protein metabolic process [6797 gene products]

⊕ ⓘ GO:0031401 : positive regulation of protein modification process [5757 gene products]

⊕ ⓘ GO:0001934 : positive regulation of protein phosphorylation [4638 gene products]

⊕ ⓘ GO:0045860 : positive regulation of protein kinase activity [2860 gene products]

⊕ ⓘ GO:0032147 : activation of protein kinase activity [1745 gene products]

⊕ ⓘ GO:0000185 : activation of MAPKKK activity [82 gene products]

⊕ ⓘ GO:0071902 : positive regulation of protein serine/threonine kinase activity [1815 gene products]

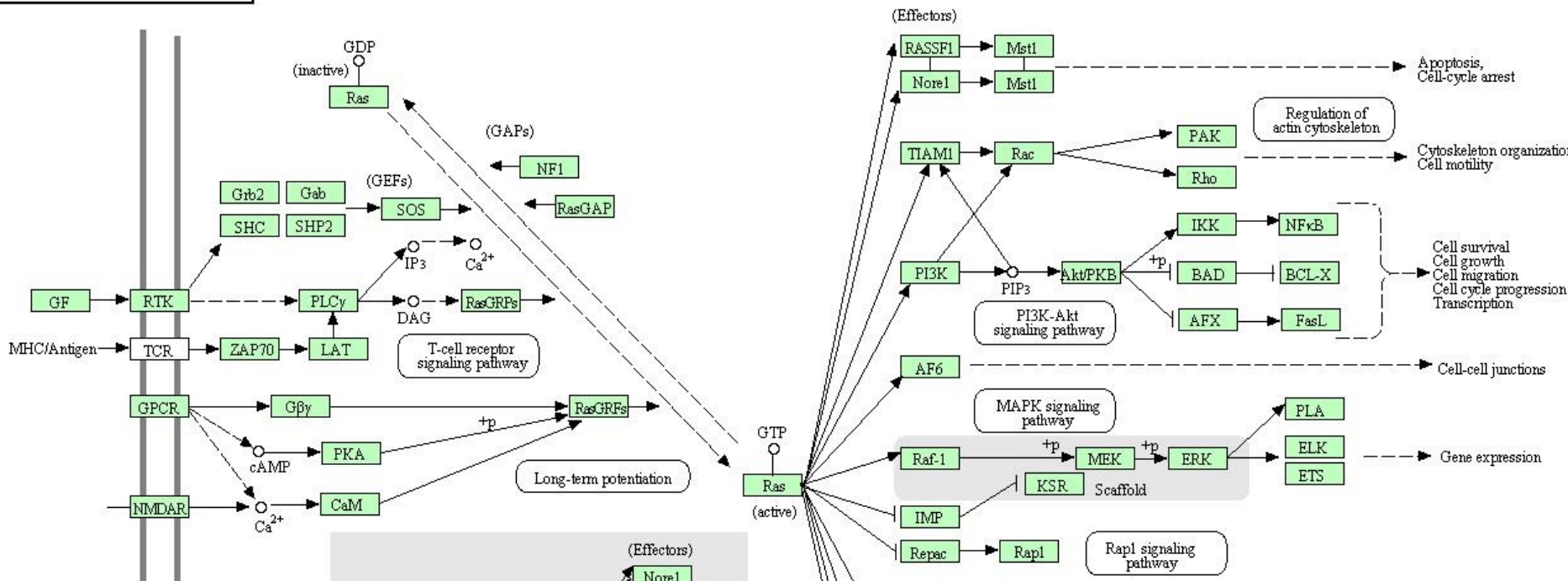
⊕ ⓘ GO:0000185 : activation of MAPKKK activity [82 gene products]

⊕ ⓘ GO:0010562 : positive regulation of phosphorus metabolic process [6341 gene products]

# KEGG

- **Kyoto Encyclopedia for Genes and Genomes**
- **Collection of pathway maps representing molecular interaction and reaction networks for**
  - metabolism
  - cellular processes
  - diseases, etc

RAS SIGNALING PATHWAY



# Exercise 12: Gene set enrichment analysis

## ➤ Identify over-represented GO terms

- Select the **two-sample.tsv** file and run **Pathways / Hypergeometric test for GO**. Open **hypergeo.html** and read about the first term. Check in **hypergeo.tsv** how many terms do you get.

## ➤ Extract genes for a specific GO term

- Copy the GO identifier for the top term (GO:0000184).
- Select **two-sample.tsv** and run tool **Utilities / Extract genes from GO**, pasting the GO identifier into the parameter field.
- Open **extracted-from-GO.tsv**. How many genes do you get? Are they up- or down-regulated (use also Volcano plot and Expression profile)?

## ➤ Identify over-represented ConsensusPathDB pathways

- Select **two-sample.tsv** and run **Pathways / Hypergeometric test for ConsensusPathDB**.
- Click on the links in the **cpdb.html** file to read about the pathways.

# Exercise 13: Gene set test

## ➤ Identify differentially expressed KEGG pathways

- Select the normalized.tsv file and **Pathways / Gene set test**. Set the **Number of pathways to visualize = 4**
- Explore **global-test-result-table.tsv**. How many differentially expressed KEGG pathways do you get?
- Explore **multtest.png**. Which gene contributes most to the first pathway?

# Microarray data analysis workflow

- **Importing data to Chipster**
- **Normalization**
- **Describing samples with a phenodata file**
- **Quality control**
  - Array level
  - Experiment level
- **Filtering (optional)**
- **Statistical testing**
  - Parametric and non-parametric tests
  - Linear modeling
  - Multiple testing correction
- **Annotation**
- **Pathway analysis**
- **Clustering**
- **Saving the workflow**



# Clustering in Chipster

## ➤ **Hierarchical**

- Includes reliability checking of the resulting tree with bootstrapping

## ➤ **K-means**

- Additional tool to estimate K

## ➤ **Quality threshold**

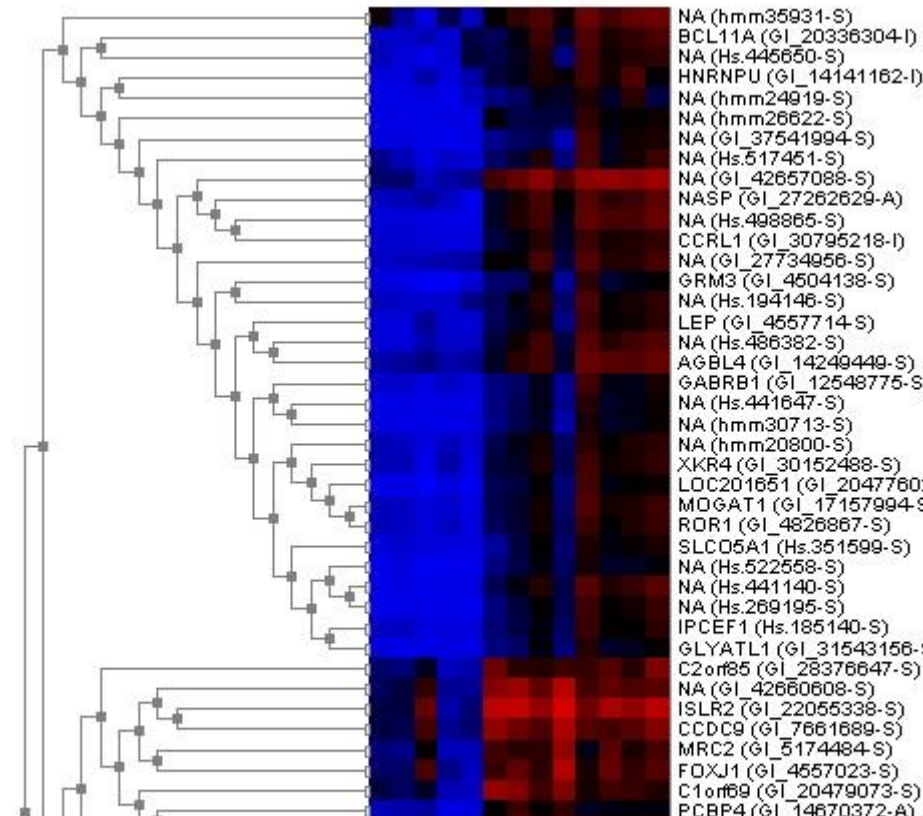
## ➤ **Self-organizing maps**

## ➤ **K-nearest neighbor (KNN)**

- Classification aka class prediction

# Hierarchical clustering

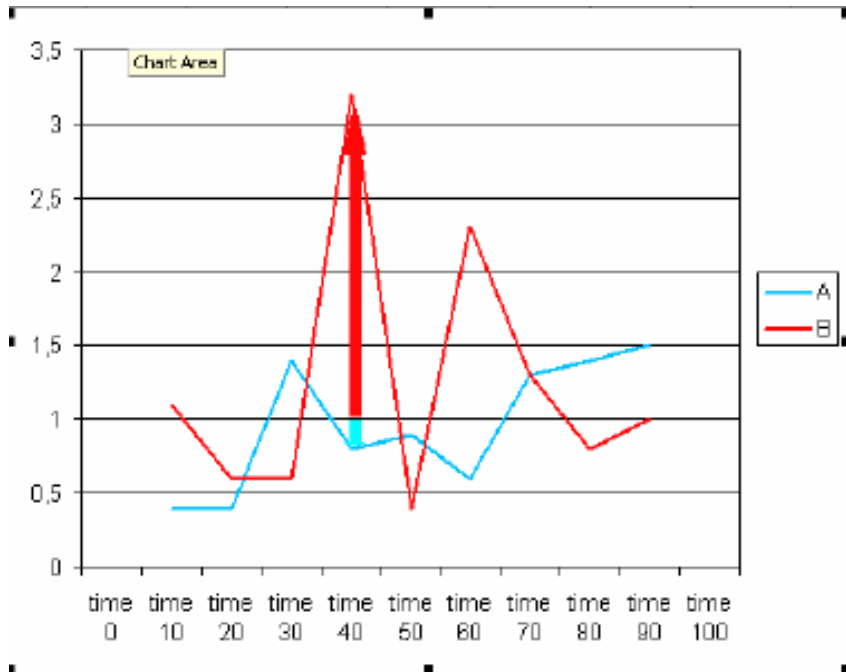
- Provides stable clusters
- Assumes pairwise correlations
- Early mistakes cannot be corrected
- Computationally intensive
- Drawing methods
  - Single / average / complete linkage
- Distance methods
  - Euclidean distance
  - Pearson / Spearman correlation



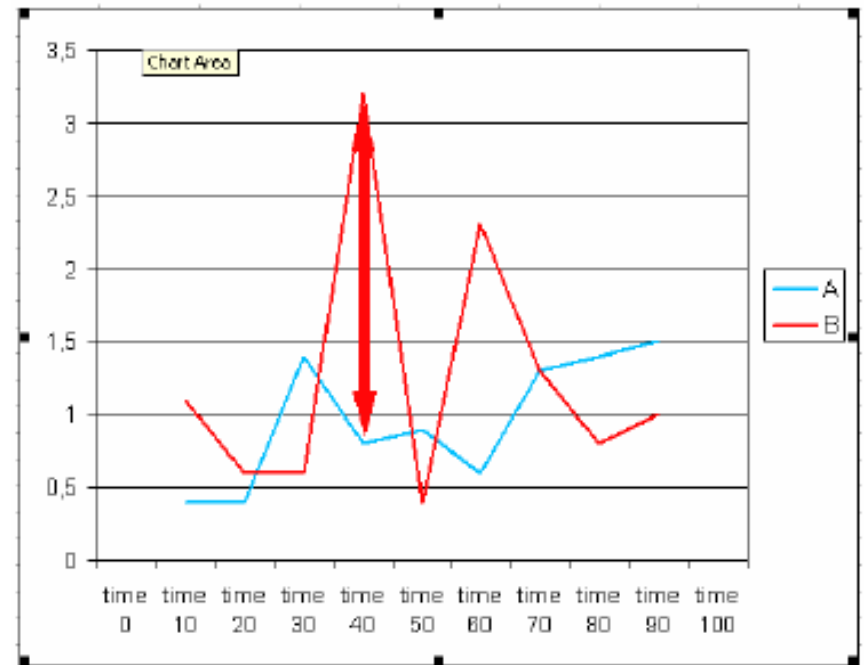
# Hierarchical clustering: distance methods

One can either calculate the distance between two pairs of data sets (e.g. samples) or the similarity between them

Pearson correlation



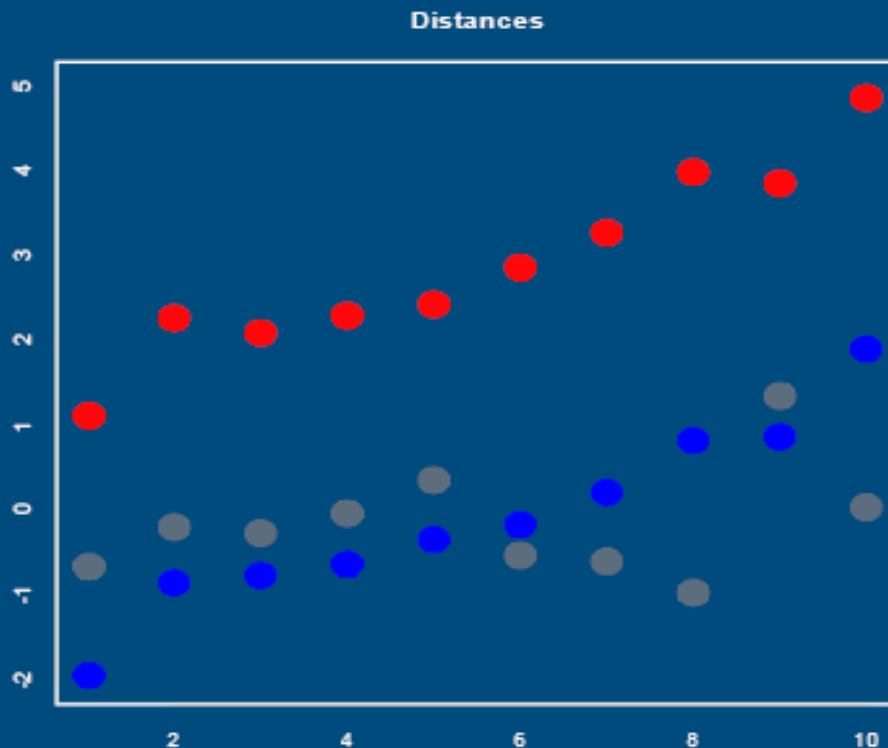
Euclidean distance



# Distance methods can yield very different results

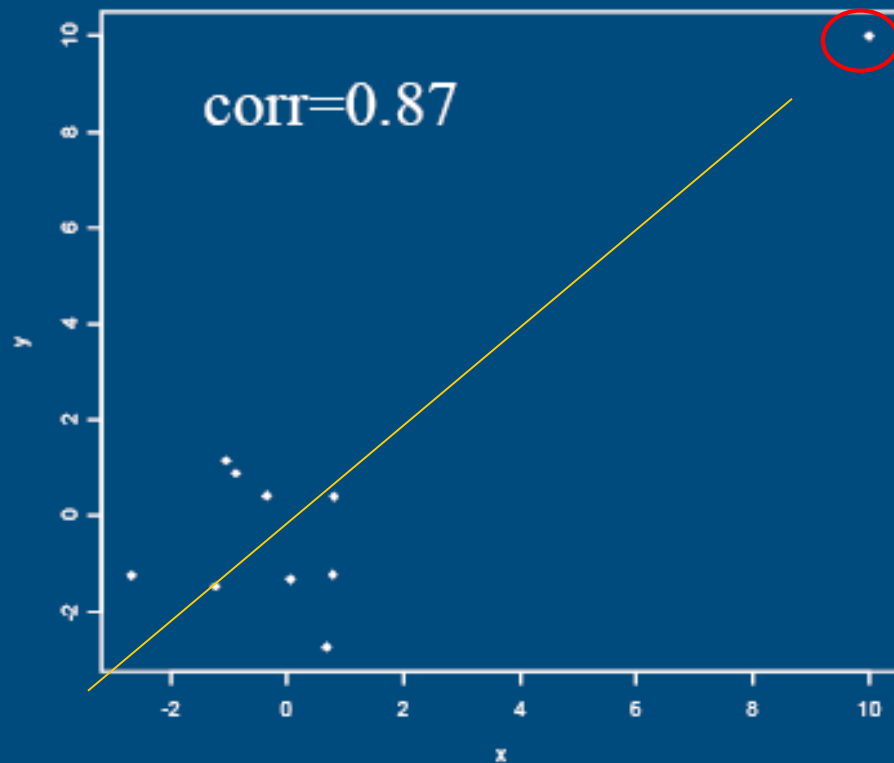
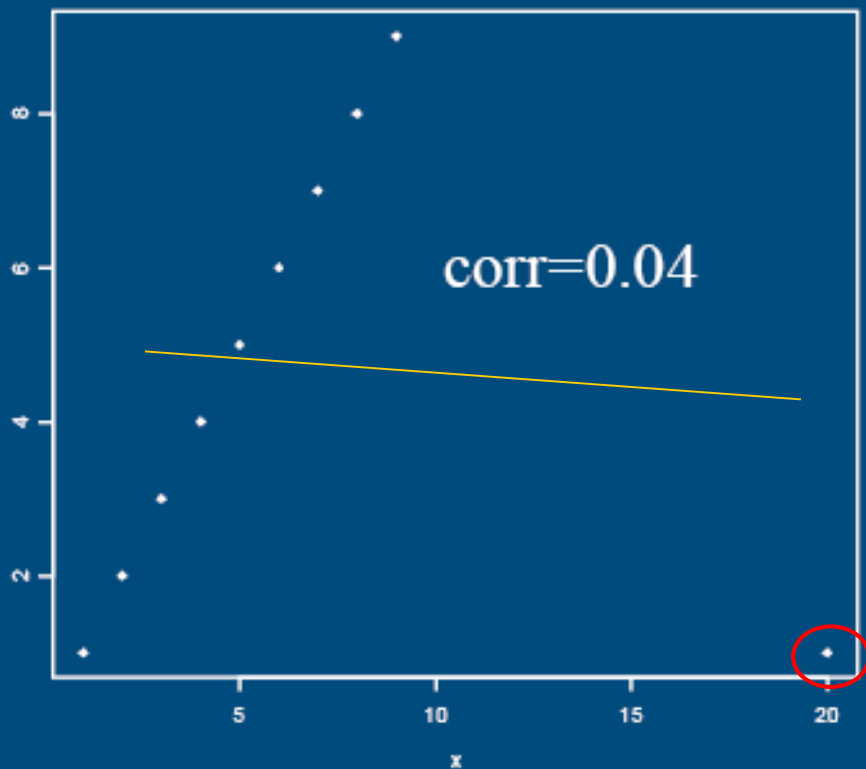
## Distances

- the Correlation distance
  - red-blue is 0.006
  - red-gray is 0.768
  - blue-gray is 0.7101
- Euclidean distance:
  - red-blue is 9.45
  - red-gray is 10.26
  - blue-gray is 3.29



Correlations are sensitive to outliers (use Spearman)!

## Correlations gone wrong

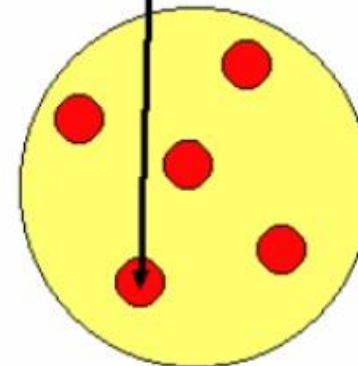
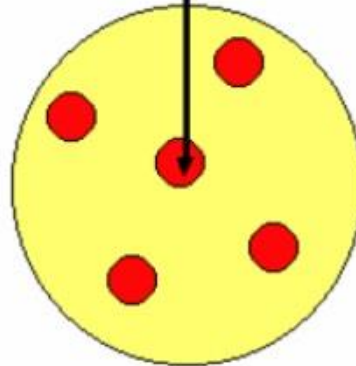
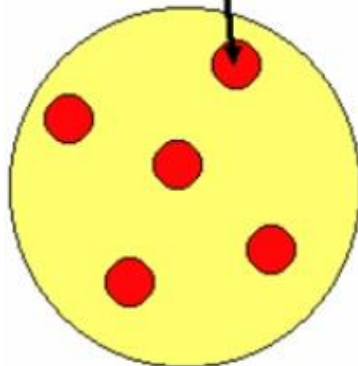
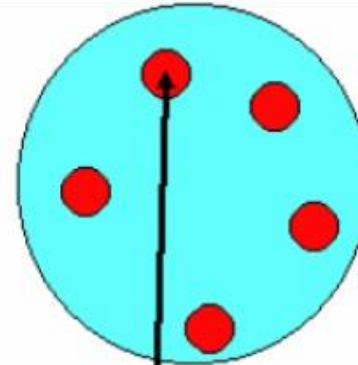
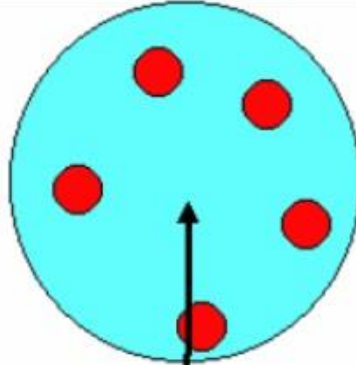
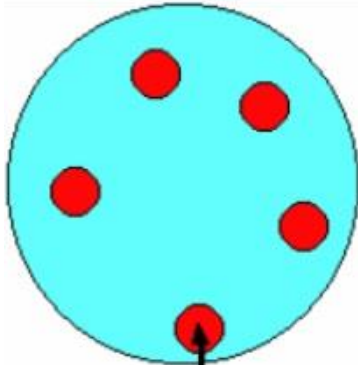


# Hierarchical clustering: drawing methods

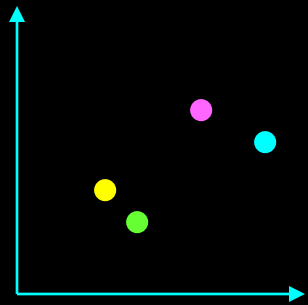
single linkage

average linkage

complete linkage

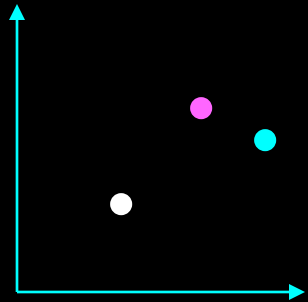


# Hierarchical clustering (euclidean distance)



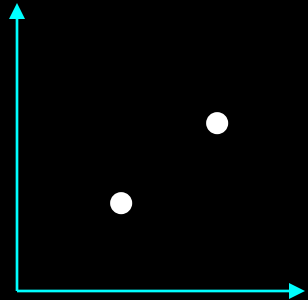
calculate distance matrix

	gene 1	gene 2	gene 3	gene 4
gene 1	0			
gene 2	2	0		
gene 3	8	7	0	
gene 4	10	12	4	0



calculate averages of most similar

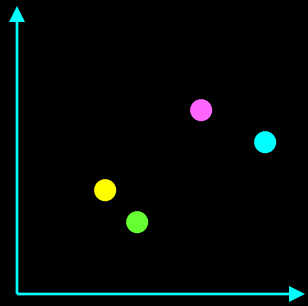
	gene 1,2	gene 3	gene 4
gene 1,2	0		
gene 3	7.5	0	
gene 4	11	4	0



calculate averages of most similar

	gene 1,2	gene 3,4
gene 1,2	0	
gene 3,4	9.25	0

# Hierarchical clustering (avg. linkage)



calculate  
distance  
matrix



	gene 1	gene 2	gene 3	gene 4
gene 1	0			
gene 2	2	0		
gene 3	8	7	0	
gene 4	10	12	4	0

calculate averages of  
most similar



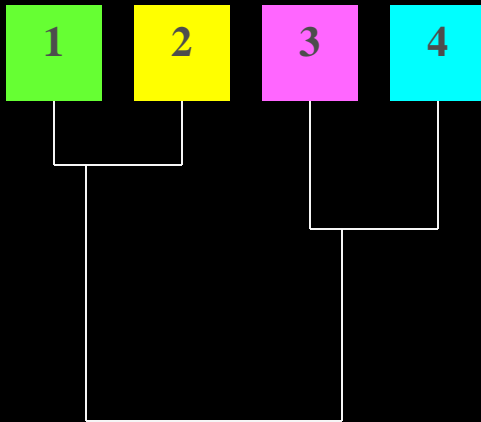
	gene 1,2	gene 3	gene 4
gene 1,2	0		
gene 3	7.5	0	
gene 4	11	4	0

calculate averages of  
most similar



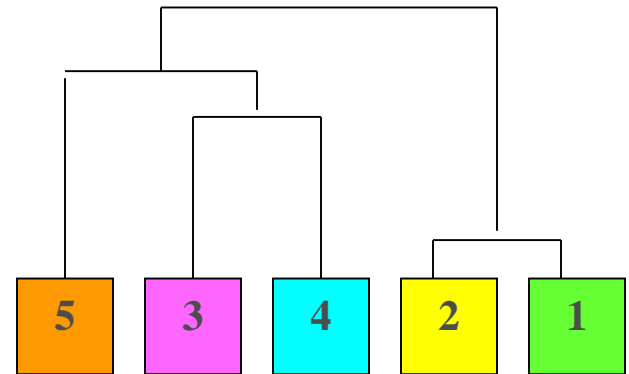
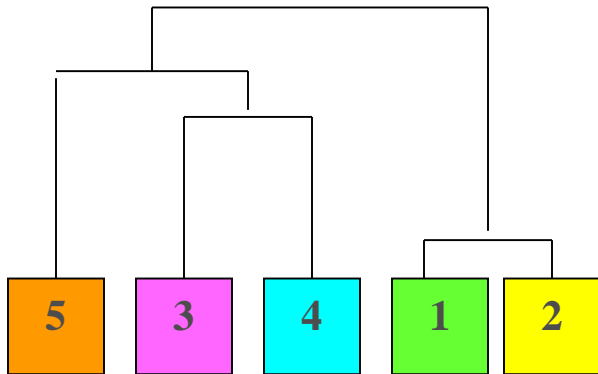
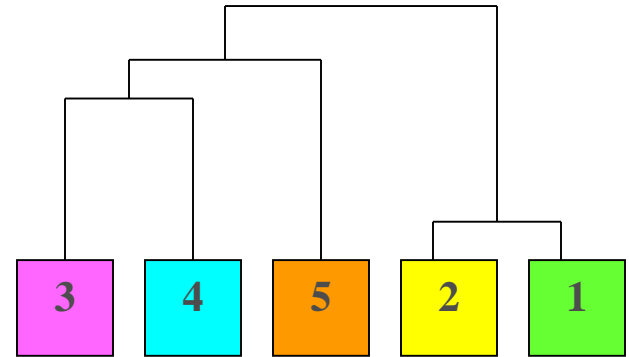
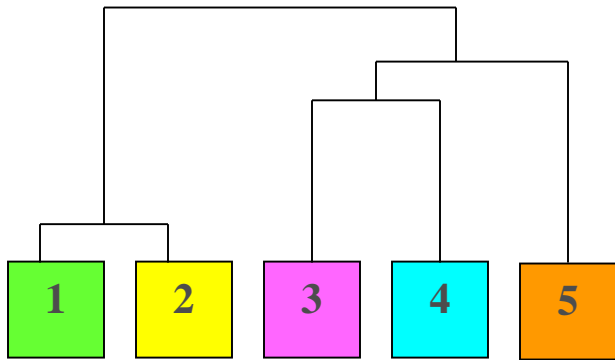
	gene 1,2	gene 3,4
gene 1,2	0	
gene 3,4	9.25	0

Dendrogram

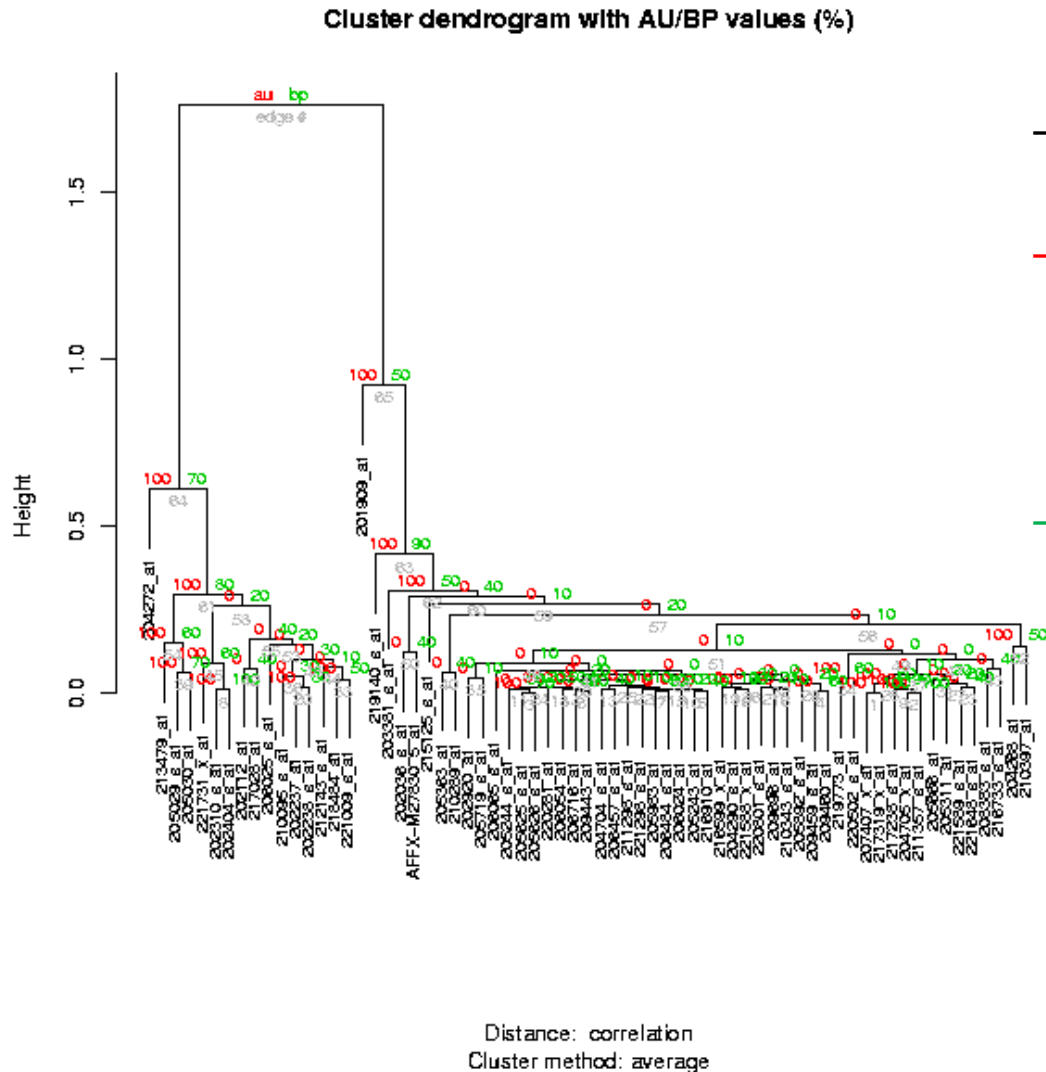




When assessing similarity, look at the branching pattern instead of sample order

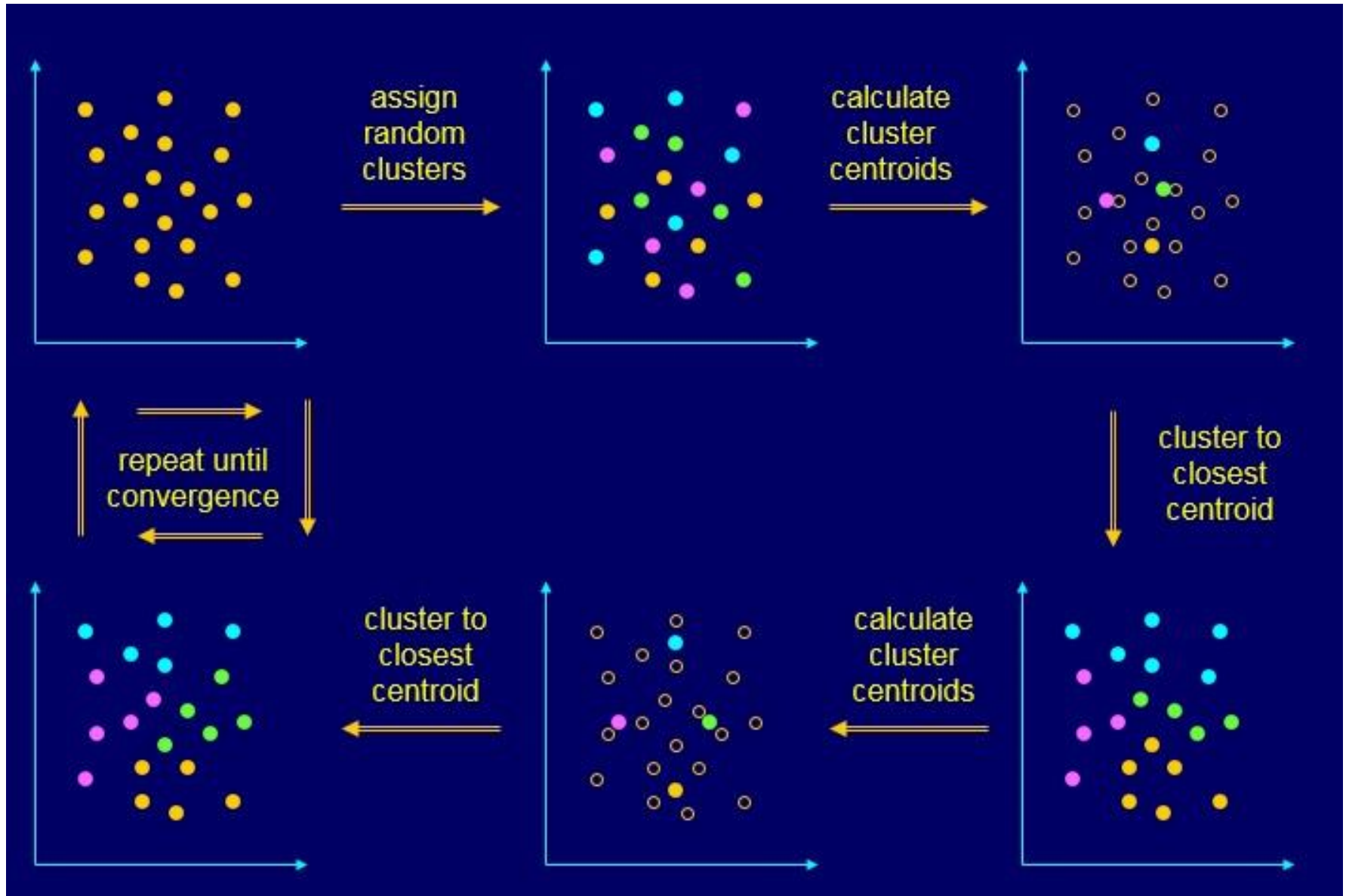


# Bootstrap resampling

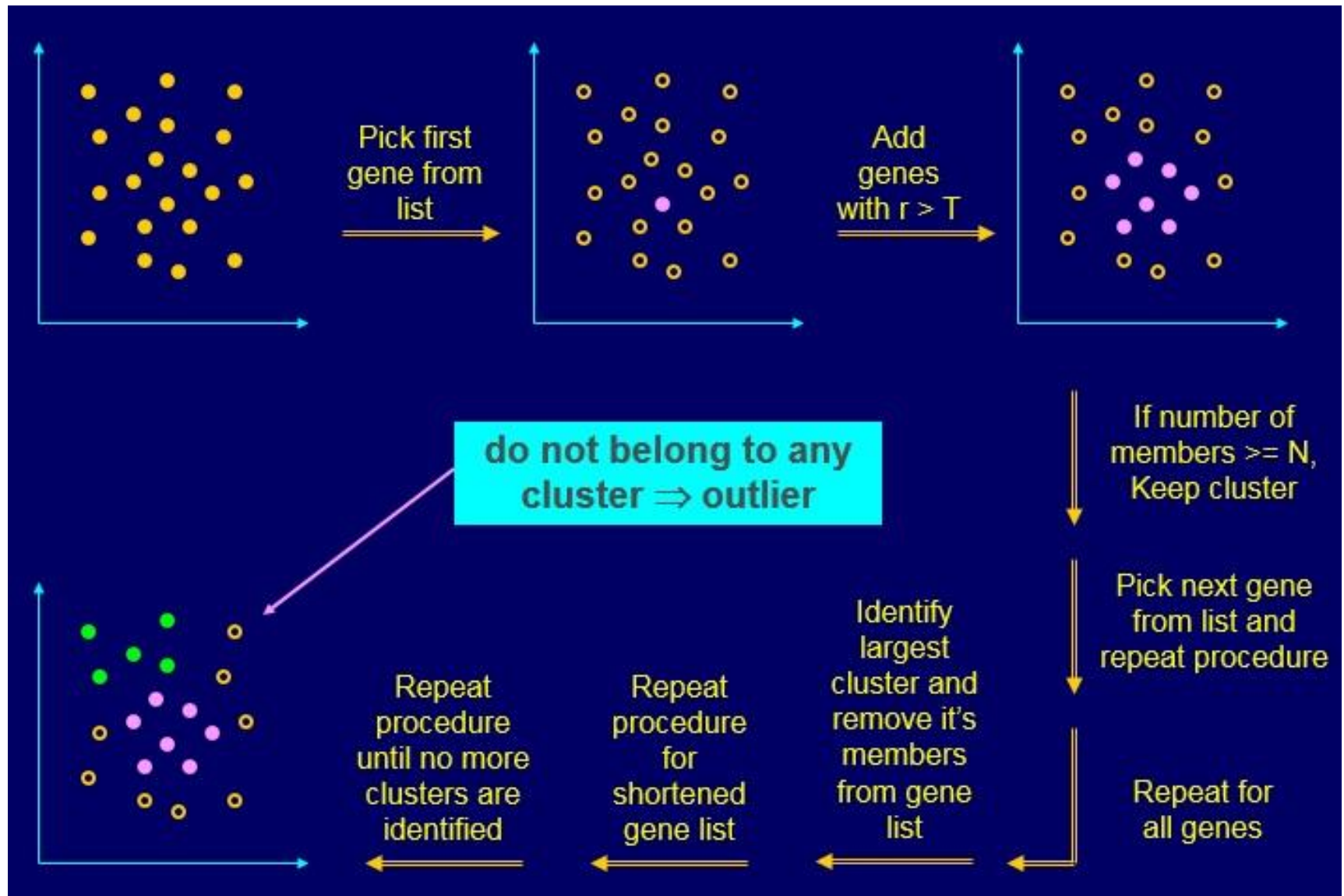


- checks uncertainty in hierarchical cluster analysis
- **AU** = approximately unbiased p-value, computed by multiscale bootstrap resampling. Clusters with AU larger than 95% are strongly supported by data.
- **BP** = bootstrap probability p-value, computed by normal bootstrap resampling

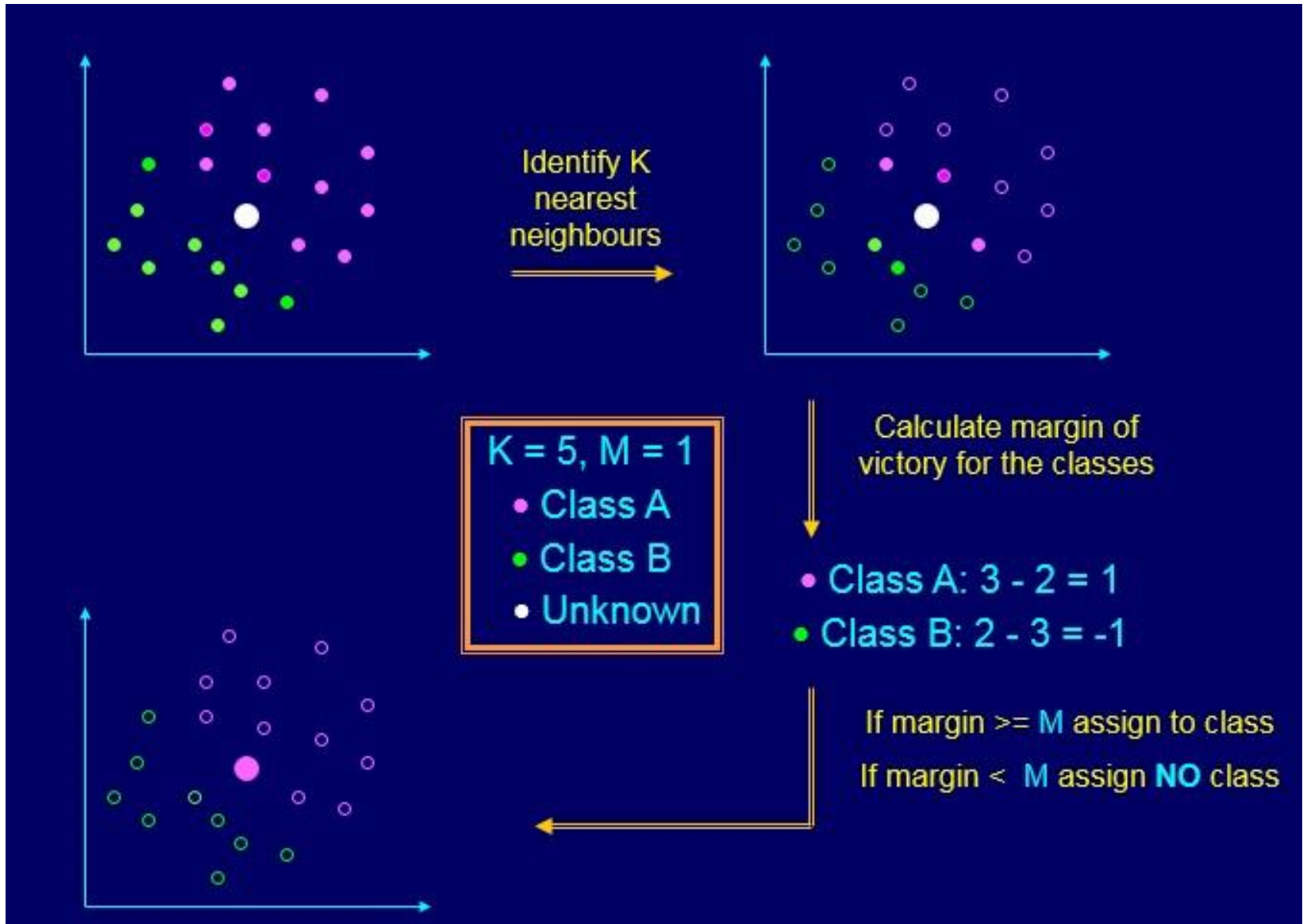
# K-means clustering



# Quality threshold clustering



# K nearest neighbour clustering



# Exercise 14: Hierarchical clustering

## ➤ Cluster genes

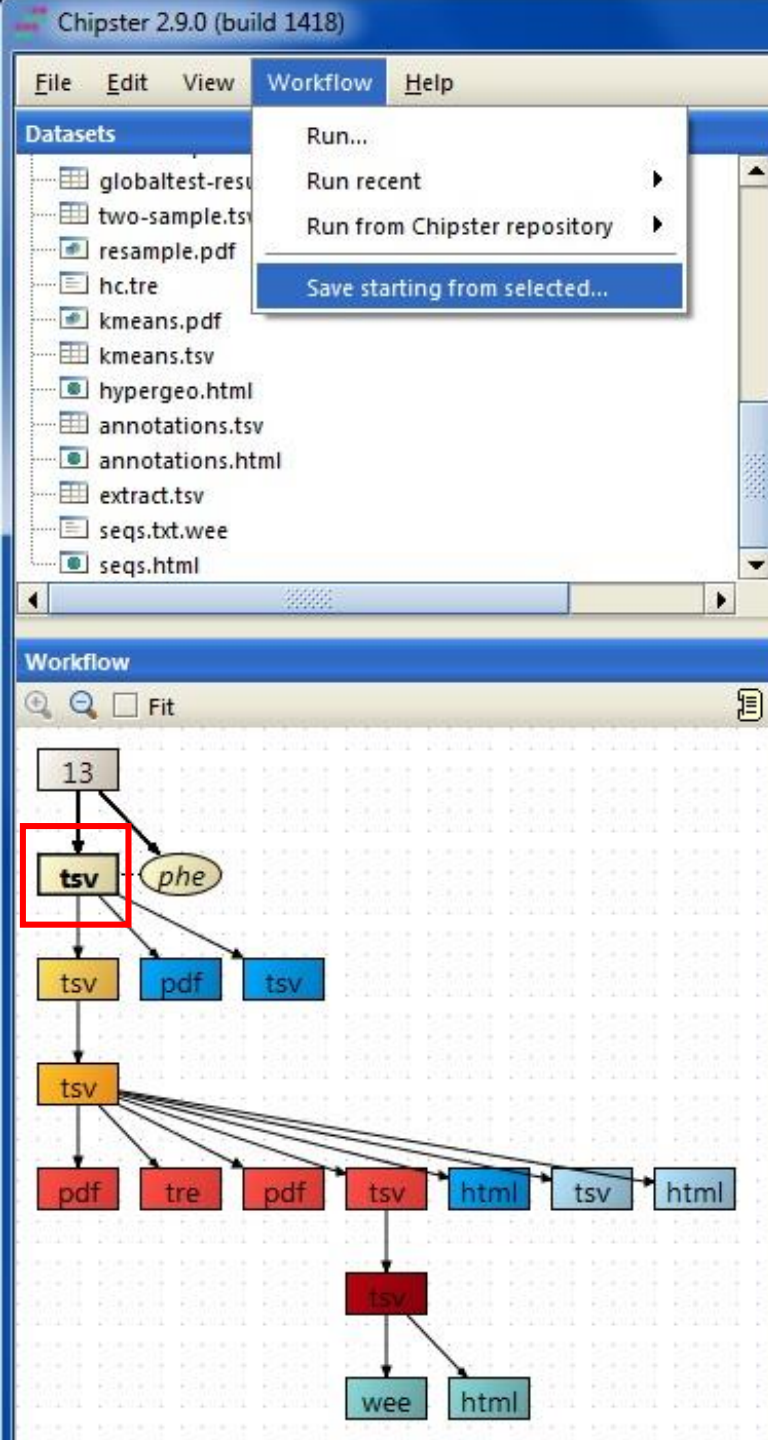
- Select the **column-value-filter.tsv** and run **Clustering / Hierarchical**.
- View the resulting file **hc.tre** as **Hierarchical clustering**. Select the genes in the last cluster by clicking on the heatmap rows. Create a new dataset out of the selection.

## ➤ Cluster genes and samples

- Select the **column-value-filter.tsv** and run the tool **Visualization / Heatmap**.
- Select the **column-value-filter.tsv** and run the tool **Visualization / Annotated heatmap**, using parameters
  - Coloring scheme = Blue - white – red
  - Cluster samples only = no

# Microarray data analysis workflow

- **Importing data to Chipster**
- **Normalization**
- **Describing samples with a phenodata file**
- **Quality control**
  - Array level
  - Experiment level
- **Filtering (optional)**
- **Statistical testing**
  - Parametric and non-parametric tests
  - Linear modeling
  - Multiple testing correction
- **Annotation**
- **Pathway analysis**
- **Clustering**
- **Saving the workflow**



# Saving and using workflows

- Select the starting point for your workflow
- Select "Workflow/ Save starting from selected"
- Save the workflow file on your computer with a meaningful name
  - Don't change the ending (.bsh)
- To run a workflow on another dataset, select
  - Workflow → Open and run
  - Workflow → Run recent (if you saved the workflow recently).



# Exercise 15: Saving a workflow

- **Prune your workflow if necessary by removing**
  - cyclic structures
  - files produced by visual selection (gray boxes)
- **Save the workflow**
  - Select **normalized.tsv** and click on **Workflow / Save starting from selected**. Give your workflow a meaningful name and save it.

# Illumina data analysis: summary

- **Normalization**
  - lumi method
- **Quality control at array level: are there outlier arrays?**
  - density graph and boxplot
- **Quality control at experiment level: do the sample groups separate? Are there batch effects or outliers?**
  - PCA, NMDS, dendrogram
- **Independent filtering of genes**
  - e.g. 50% based on coefficient of variation
  - Depends on the statistical test to be used later
- **Statistical testing**
  - Empirical Bayes method (two group test / linear modeling)
- **Annotation, pathway analysis, promoter analysis, clustering, classification...**

# Design of experiments

# When planning an experiment, pay attention to

- **The number of biological replicates**
  - Technical replicates are a different thing!
- **Sample pairing**
  - Use paired samples if you can
- **Pooling**
  - Avoid it if possible
- **Reference samples**
  - Should be as similar as possible: same individual, tissue...

# Technical vs. biological replicates

- **Biological replicates are separate individuals/samples**
  - Necessary for a properly controlled experiment
- **Technical replicates are repeated measurements using the same RNA isolate or sample**
  - Waste of resources?
  - Can cause unnecessary variance reduction → increases number of false positives
- **Avoid mixing biological and technical replicates!**

# Technical vs. biological replicates

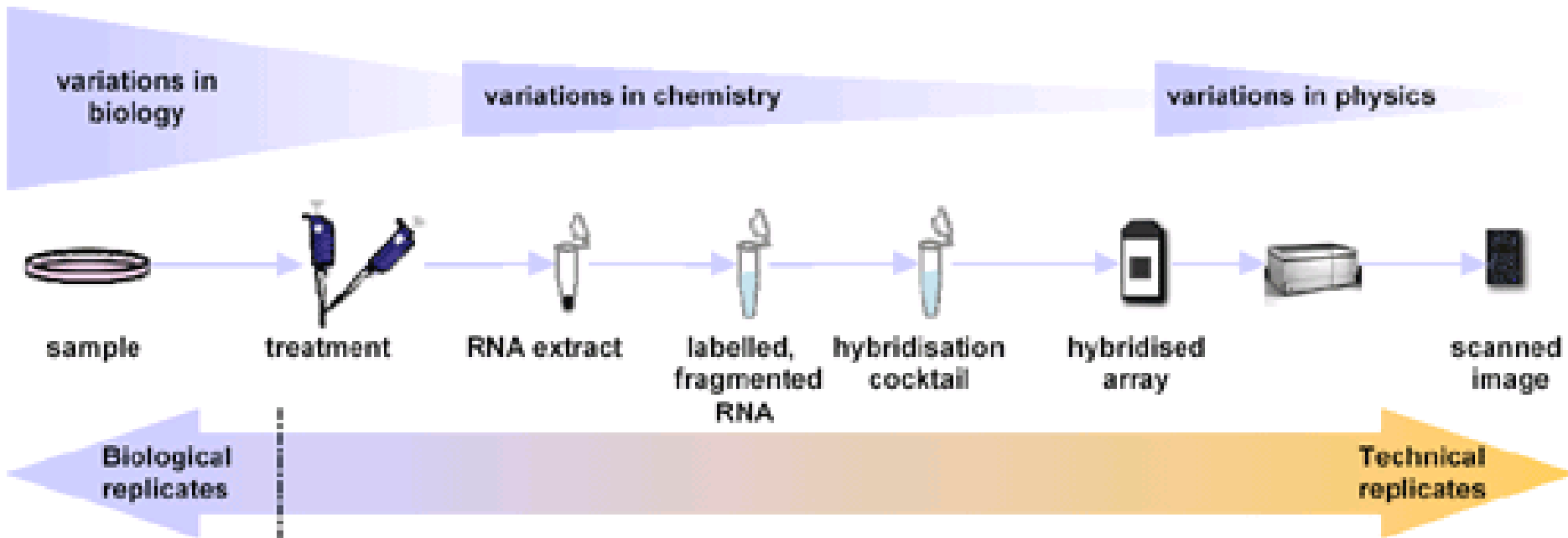
➤ **Dummy example: Let's measure the average height of a Finnish male.**

- **Biological** replicates: different individuals
- **Technical** replicates: measure the same individual with different measuring tape



# Technical vs. biological replicates

Distinction between technical and biological replicates is fuzzy.



# Replicate number

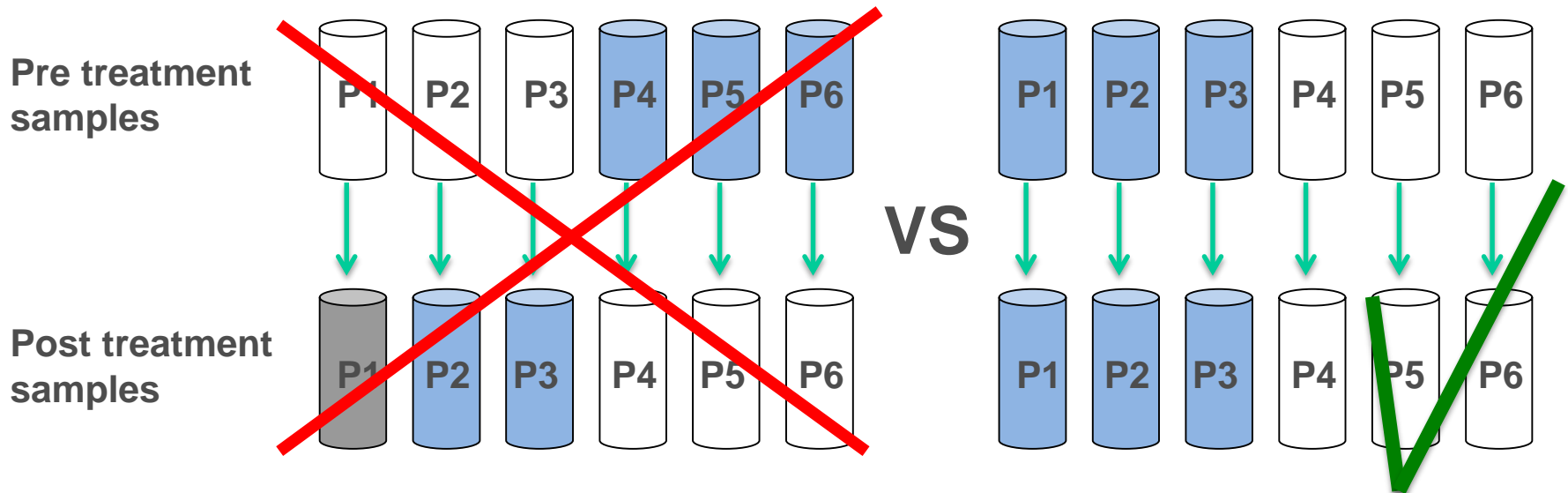
- **Publication quality data needs at least 3 biological replicates per sample group.**
  - This can be sufficient for cell-cultures and test animals
- **More reasonable numbers:**
  - Cell cultures / test animals: 3 is minimum, 4-5 OK, >7 excellent
  - Patients: 3 is minimum, 10-20 OK, >50 good
  - Power analysis can be used to estimate sample sizes





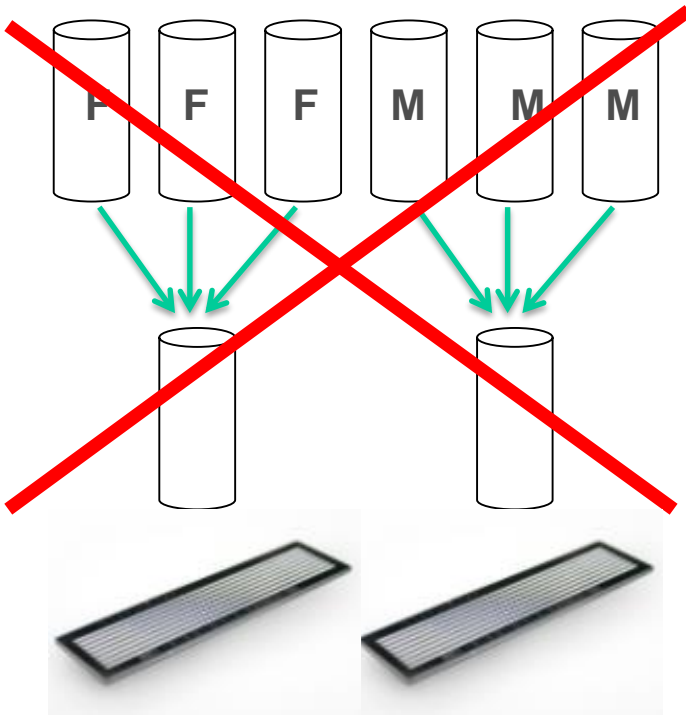
# Paired samples

- Using paired samples reduces variance, as individual variation can be tackled using a matched control
  - Pre vs. post treatment samples
  - Tumor vs. normal samples from the same patient
- Example
  - 6 patients, 2 samples from each. Enough money to analyze only 6 samples. Which option do you choose?



# Pooling

- If possible, don't pool samples.
- If you don't have enough material to analyze each sample on its own, you might have to pool.
- Careful with concentrations!
- Make pools as similar as possible



VS

