ArrayNorm

Short Manual, 2nd version

by Roland Pieler

TUG
ArrayNORM
developed by Roland Pieler
Release 1.7
© 2004 Institute for Genomics and Bioinformatics Graz University of Technology

TUG

Technische Universität Graz
Erzherzog-Johann-Universität

Institute for Biomedical Engineering, Graz University of Technology, Graz, Austria

Contents

List of Figures

Glossary

Biological replicates biological samples from independent sources, representing the same condition, e.g. liver tissue from individual mice of the same sex and strain.

Bonferroni correction Multiple-testing adjustment in which the significance-level is divided by the total number of tests.

Class In experimental design, a *class* denotes a subset of the whole experiment. For example one single time-point out of a time-course experiment represents one class, containing all microarrays belonging to this time-point. An experiment can consist of any number of classes.

DE Short form for *differentially expressed*.

Dye-swap pair Two slides comparing the same samples of RNA, one with normal and one with reversed dye-assignment.

Fold change The ratio of RNA quanitites between two samples in a microarray experiment. Often in units of $log2$.

MARS Microarray analysis and retrieval system. A J2EE application for persisting and organizing microarray data, based on Enterprise Java Beans (EJB) and Struts framework. Developed by the *TU-Graz Bioinformatics Group*.

Microarray experiment An experiment studies a system under controlled conditions while some conditions are changed. In gene expression, one varies some parameter such as time, drug, developmental stage, or dosage on a sample. The sample is processed and labeled with a detectable tag (Cy3, Cy5) so that it can be used in hybridization with microarrays.

Normalization The process of removing the effect of all sources of non-biological variation from microarray data, making them comparable.

p-Value A measure of evidence against the null hypothesis in a statistical test.

Ratio Also referred to as ”fold change”. A ratio refers to a normalized signal intensity generated from one feature in a given channel divided by a normalized signal intensity generated by the same feature in another channel.

Replication A replicate set refers to repeated experiments where the same type of array is used, and the same probe isolation method is used to get more statistically meaningful interpretation of results. The ability to reproduce an experiment is the key to validity.

Significance level The p-value that is regarded as providing sufficient evidence against a null hypothesis. If the p-value falls below the significance-level, the null hypothesis is rejected.

Sourcefile Sourcefiles for ArrayNorm are usually textfiles provided by image aquisition software (like GenePix, ImaGene, etc.) containing intensity- and quality-data for every single spot per slide. In general, one slide describes one single slide, except for Imagene and ArrayVision: These formats need two files, one for each wavelenght-channel.

Statistical significance A result is statistically significant when it doesn’t happen by chance.

Subgrid A subarea of a single microarray. Within one subgrid all spots are printed by the same print tip (needle printing the probes to the slide).

Technical replicates Multiple hybridisations with RNA samples obtained from the same biological source.

Workflow The correct or recommended order of data manipulation steps for microarray normalization.

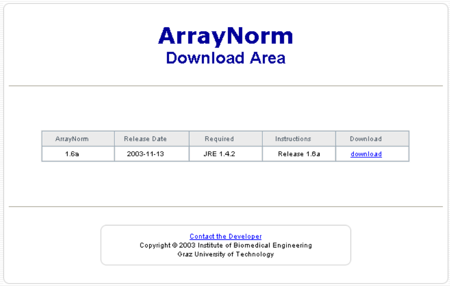
Chapter 1

Installation

1.1 System Requirements

- PC or Mac.
- Windows, Linux, Unix or Mac OS X operating system.
- 512MB RAM recommended.
- Hard disk with at least 150MB free (ArrayNorm comes with example files).

1.2 Download ArrayNorm



ArrayNorm
Download Area

ArrayNorm	Release Date	Required	Instructions	Download
1.6a	2003-11-13	JRE 1.4.2	Release 1.6a	download

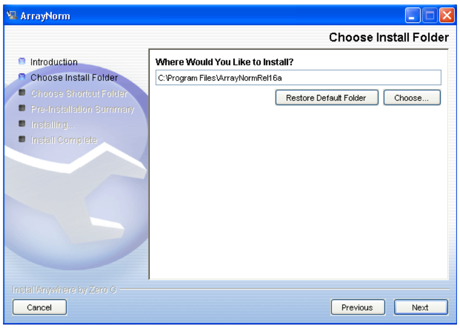
Contact the Developer
Copyright © 2003 Institute of Biomedical Engineering
Graz University of Technology

Figure 1.1: *Download ArrayNorm*

Please, visit our homepage http://genome.tugraz.at. Go to `Software` to find the installation kits for all possible operating systems. Additionally, you can get further documentation (pdf-files masterthesis and short manual).

1.3 Install ArrayNorm

Starting the setup-program, an install wizard will lead you through the installation steps. The installation contains some example microarray-files for testing the software.



Where Would You Like to Install?

C:\Program Files\ArrayNormRel16a

Figure 1.2: *Install ArrayNorm: setup-wizard*

Chapter 2

User’s manual

2.1 Introduction

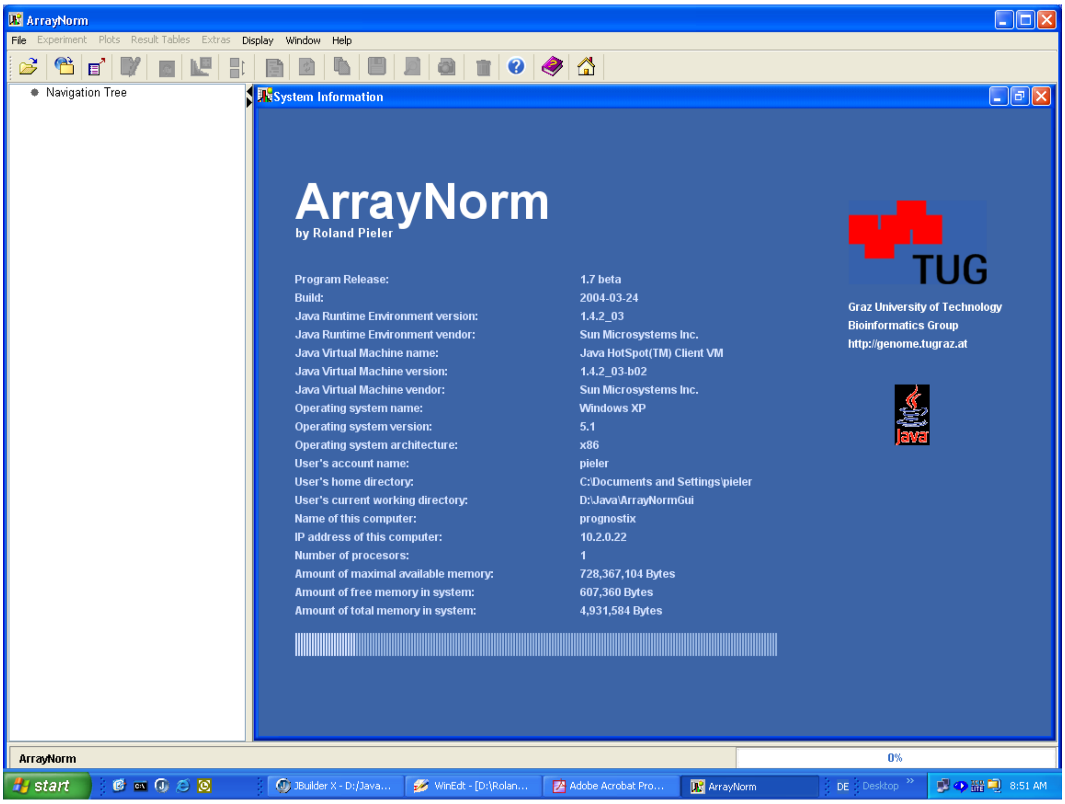
ArrayNorm is a platform independent application which provides tools for visualization, normalization and analyis of microarray data. It deals with a wide range of possible experiment designs, including replication, dyeswap-pairs or control spots.

2.1.1 Features

- Upload of any number of source-files.
- Variety of possible file-formats (GenePix, ImaGene, Agilent, etc.)
- Organization of loaded ’slides’ in experiment-classes.
- Possibility to define the experiment’s design.
- Data tree to illustrate the experiment’s organization.
- Tools for visualization (Arrayview, Scatterplot, Histogram, etc.).
- Variety of normalization methods (within and between slides)
- Possibilities of Background Subtraction and Data Reset.

- Possibility of averaging replicated data and merging of Slides.
- Simple tools for analysis (foldchange detection, etc.).
- Statistical tests for finding differentially expressed genes (T-test, Mann-Whitney test).
- Oneway-ANOVA for finding differentially expressed genes.
- Exporting of results to textfiles (compatible to Genesis software).
- Connection to the MARS-database for loading predefined microarray-experiments and uploading results.

2.2 Short Manual



ArrayNorm
by Roland Pieler

Program Release:	1.7 beta
Build:	2004-03-24
Java Runtime Environment version:	1.4.2_03
Java Runtime Environment vendor:	Sun Microsystems Inc.
Java Virtual Machine name:	Java HotSpot(TM) Client VM
Java Virtual Machine version:	1.4.2_03-b02
Java Virtual Machine vendor:	Sun Microsystems Inc.
Operating system name:	Windows XP
Operating system version:	5.1
Operating system architecture:	x86
User's account name:	pieler
User's home directory:	C:\Documents and Settings\pieler
User's current working directory:	D:\Java\ArrayNormGui
Name of this computer:	prognostix
IP address of this computer:	10.2.0.22
Number of procesors:	1
Amount of maximal available memory:	728,367,104 Bytes
Amount of free memory in system:	607,360 Bytes
Amount of total memory in system:	4,931,584 Bytes

Graz University of Technology
Bioinformatics Group
http://genome.tugraz.at

Figure 2.1: *The main GUI, splitted into data-navigation panel and visualisation panel*

After starting the program, a GUI appears, showing some system information and an empty data-navigation tree. The first step will be to define a new experiment and load its **sourcefiles**.

2.2.1 Toolbar

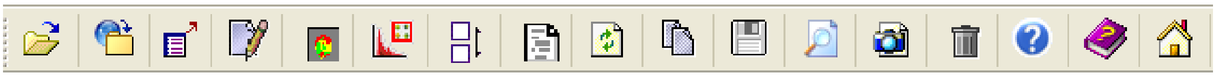


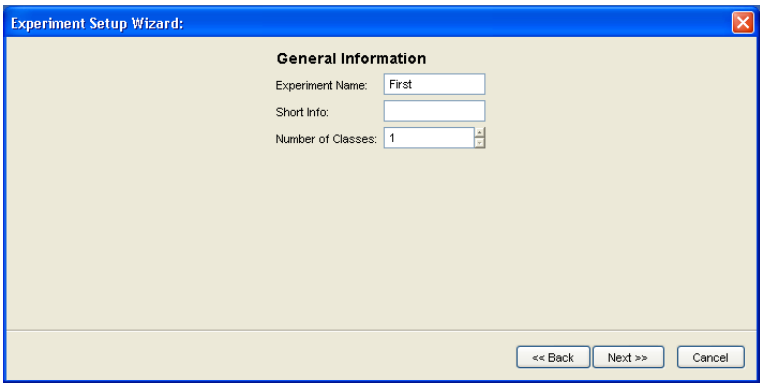
Figure 2.2: *Toolbar-buttons from left to right: open new experiment, load experiment from MARS, upload resultfiles to MARS, show reports, background subtraction, normalization, scale adjustment, print slide data files, reset data, replicate handling, export results to file, analyze, capture plot, delete open plots, aboutbox, shortmanual, homepage.*

2.2.2 Loading data, defining the experimental design

Starting a new experiment requires information about general information, sources of data and experimental design. A wizard leads through these steps.

2.2.2.1 Experimental setup

Every new experiment can be attached with general informations, like name, number of **experimental classes**, etc. Defining the number of classes is not definite, it may be changed afterwards.



Experiment Setup Wizard:

General Information

Experiment Name: First

Short Info:

Number of Classes: 1

Figure 2.3: *First step in setting up a new experiment. Defining general parameters, like name and number of experimental classes.*

2.2.2.2 Selecting source files

Since the number of possible microarrays is not limited, the wizard provides a file list, to which multiple files can be added. **Important** is to select the correct **file-filter** according to the sourcefile-vendor: In every filechooser you can choose the correct filter out of a list of possible file formats (see appendix C for possible file formats). After selection, all files are numerated in the list, without information about class-affiliation or dye-assignment. Note that all files within one experiment must have the same file-format and the same number of spots.

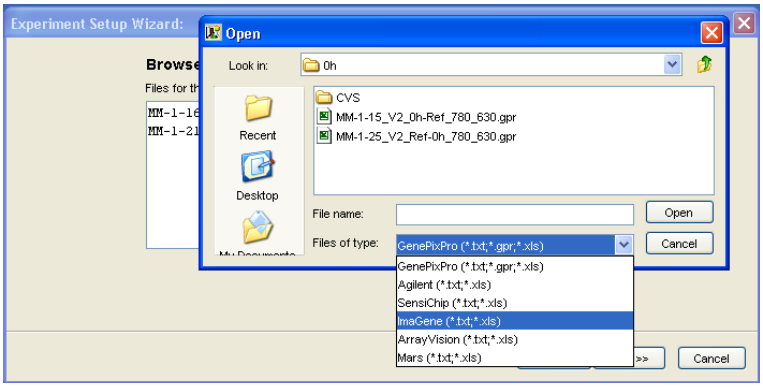


Figure 2.4: *Browsing and adding the experiment's sourcefiles. Selecting the appropriate file-format.*

2.2.2.3 ImaGene and ArrayVision file formats

These file formats come up with two files combined: one for each intensity channel. This needs some preparatory work by the user: Both files have to have the same name, containing `cy5` and `cy3`, respectively. The names just differ in `5` and `3`. For example `testslide110cy5xyz.txt` and `testslide110cy3xyz.txt`. The `cy5` and `cy3` tags can be placed anywhere in the filename. When selecting the sourcefiles in the setup wizard, just select the `cy5` files! ArrayNorm will automatically find the according `cy3` files and merge the pairs to one dataset.

2.2.2.4 Experimental design

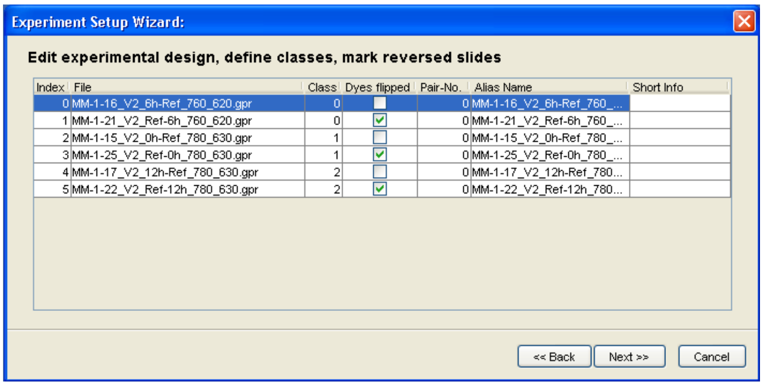
For normalization and analysis tasks, it is necessary to define relationships between microarrays. For example, which slides belong to the same class, which are **reverse-labeled** and if there are **dye-swap pairs** available. All these informations will be used for normalization, scaling between slides and replicate handling. For every hybridization the user can edit:

- **The assignment to a class.** A class is a subelement of the whole experiment. For instance, a single class represents one timepoint in a time-series experiment. All hybridisations gained from this timepoint can be assigned and grouped together into one class. For further steps (like analysis), one class will be treated as biological replicate and its contained slides as technical replicates.

- **Whether the particular slide is reverse-labeled or not.** Selecting a reverse-labeled checkbox tells the program to deal the particular slide as reverse-labeled. This is important for the organization of the experiment and for normalization issues.

- **The assignment of a dye-swap partner, if available.** If a class comes up with (at least) one normal and one reverse labeled slide, it is possible (and recommended) to build dyeswap-pairs. This is done by editing the number in the dysw-pair column. Two slides showing the same number belong to one pair. **Important** is to start numbering pairs with 1, not 0.

- **An alias-name to appear in the data-tree.** That is just for a nice look of the navigation tree.



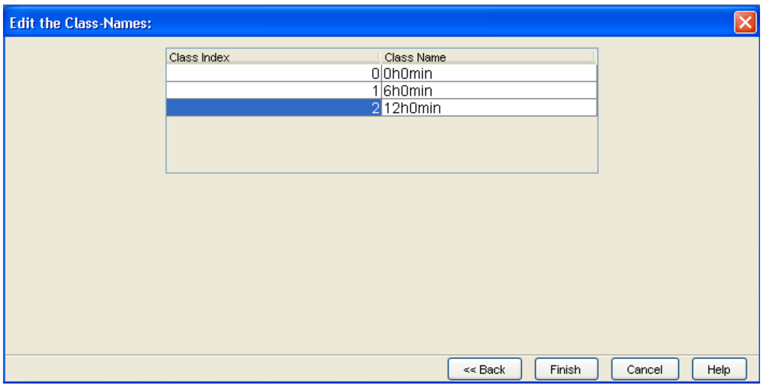
Experiment Setup Wizard:

Edit experimental design, define classes, mark reversed slides

Index	File	Class	Dyes flipped	Pair-No.	Alias Name	Short Info
0	MM-1-16_V2_6h-Ref_760_620.gpr	0	□	0	MM-1-16_V2_6h-Ref_760_...	
1	MM-1-21_V2_Ref-6h_760_620.gpr	0	☑	0	MM-1-21_V2_Ref-6h_760_...	
2	MM-1-15_V2_0h-Ref_780_630.gpr	1	□	0	MM-1-15_V2_0h-Ref_780_...	
3	MM-1-25_V2_Ref-0h_780_630.gpr	1	☑	0	MM-1-25_V2_Ref-0h_780_...	
4	MM-1-17_V2_12h-Ref_780_630.gpr	2	□	0	MM-1-17_V2_12h-Ref_780...	
5	MM-1-22_V2_Ref-12h_780_630.gpr	2	☑	0	MM-1-22_V2_Ref-12h_780...	

Figure 2.5: *Defining the assignment to classes, marking reverse-labeled slides and dyeswap-pairs, editing the alias name of every slide.*

Additionally, it is possible to edit the class-names. This will be helpful in all further steps and it affects any result-files created.



Class Index	Class Name
0	0h0min
1	6h0min
2	12h0min

Figure 2.6: *Editing the class-names. The navigation-tree folders will display these names.*

2.2.2.5 Some hints for experimental design

To be able to use the tools provided in the program, some hints should be considered:

- **Technical replicates:** If you have got technical replicated slides, put the replicates into one single class! When handling replicates (see chapter 2.2.5 about replicate handling), technical replicates are merged to one result representing the whole class.

- **Biological replicates:** Statistical testing only is possible with biological replicates. ArrayNorm can treat whole classes as biological replicates (see section 2.2.6.2 about statistical testing!!). So if you have biological replicated slides, put each into a separate class!

- **Biological and technical replicates:** This is the perfect case: Multiple classes, each standing for one biological replicate and containing technical replicated slides.

- **No biological replicates:** Sometimes or often there are no biological replicated slides available, just technical replicates. Theoretically, you will not be able to apply any statistical test. But there is a way out: Put every single slide into a single class! ArrayNorm will treat the classes as biological replicates and allows you to do statistical testing, even though it is not recommended.

2.2.2.6 Data organization tree

After setting up the experiment, all data are organized by a graphical tree, reflecting the structure of the experimental design.

- The experiment is splitted into **classes**, each class represents a biological condition (e.g. a timepoint of a timecourse experiment). The classes are named as specified in the wizard.
- Every class holds its appendant slides, marked with colors and prefixes. A red-green point indicates normal dye-assignment, a green-red point indicates a reverse-labelled microarray. The **prefix p**X states that the particular slide belongs to the Xth dye-swap pair. A dye-swap pair always contains one normal and one reverse labelled slide. Naturally, a dye-swap pair can only include slides from the same class. Multiple dye-swap pairs within a single class are possible.
- The `All Plots`-folder holds plots created by the user, allowing switching between all opened plots.

2.2.2.7 Accessing methods

In principle, all actions or methods can be accessed by

- **Rightclicking a tree's component or folder.** Depending which kind of folder (experiment, class, slide, results,...), a menu pops up with possible actions for the particular component.
- **Buttons.** Especially for Mac users. For some functions (e.g. normalization), the wanted tree component must be preselected by a mouseclick. Warning dialogs will inform the user about wrong or impossible selections.
- **Menus.** All functions which are covered with buttons can also be activated by menus.

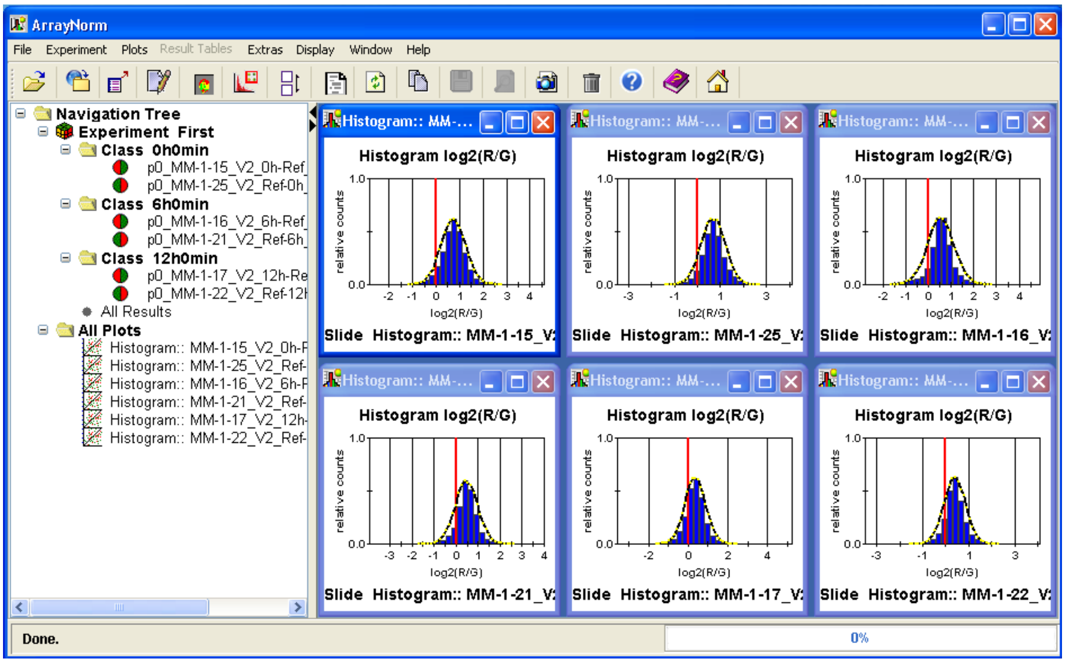

ArrayNorm
File Experiment Plots Result Tables Extras Display Window Help
Navigation Tree
Experiment First
Class 0h0min
p0_MM-1-15_V2_0h-Ref
p0_MM-1-25_V2_Ref-0h
Class 6h0min
p0_MM-1-16_V2_6h-Ref
p0_MM-1-21_V2_Ref-6h
Class 12h0min
p0_MM-1-17_V2_12h-Re
p0_MM-1-22_V2_Ref-12h
All Results
All Plots
Histogram:: MM-1-15_V2_0h-R
Histogram:: MM-1-25_V2_Ref-
Histogram:: MM-1-16_V2_6h-R
Histogram:: MM-1-21_V2_Ref-
Histogram:: MM-1-17_V2_12h-
Histogram:: MM-1-22_V2_Ref-
Histogram:: MM-...
Histogram log2(R/G)
relative counts
1.0
0.0
log2(R/G)
Slide Histogram:: MM-1-15_V2
Slide Histogram:: MM-1-25_V2
Slide Histogram:: MM-1-16_V2
Slide Histogram:: MM-1-21_V2
Slide Histogram:: MM-1-17_V2
Slide Histogram:: MM-1-22_V2
Done.
0%

Figure 2.7: *GUI with loaded experiment and opened histograms for all slides. The structure of the tree indicates 3 classes, each with one normal and one reverse labeled slide (colored icons!). All opened plots are listed below in the plots-folder.*

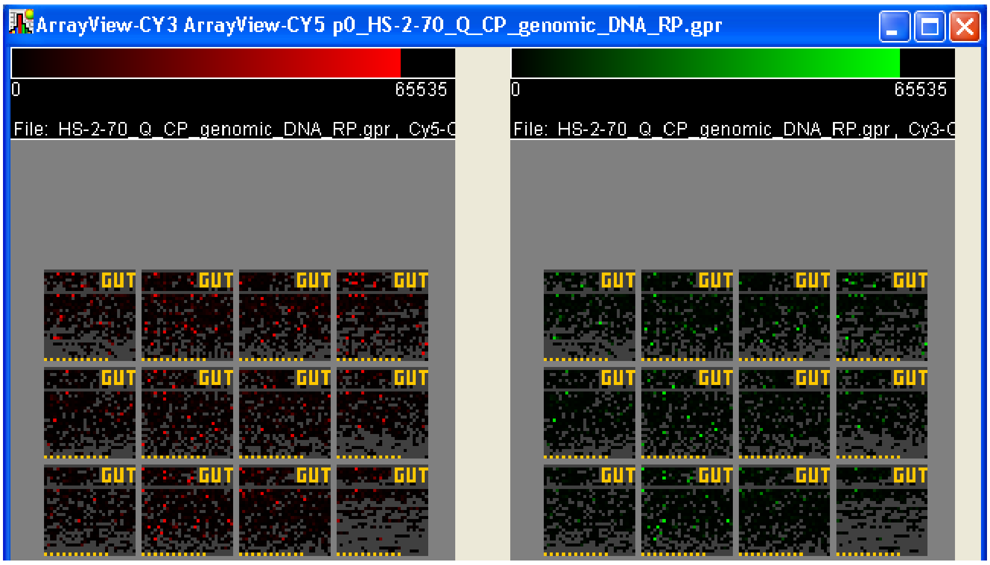
2.2.3 Visualization

To get an idea about the condition of the data sets or the effects of different normalization methods, means of graphical display can help assessing the success of the experiment and choosing the analysis tools.

2.2.3.1 Array view

The Array Viewer features false-colored images for the red and green channel per slide. It is definitely not the scanned microarray output image (e.g. provided by GenePix Pro), but a diagnostic plot showing:

- the arrangement of print-tip groups.
- rough information on spatial artifacts (e.g. scratches)
- highlightet control-spots: orange



ArrayView-CY3 ArrayView-CY5 p0_HS-2-70_Q_CP_genomic_DNA_RP.gpr
0
65535
File: HS-2-70_Q_CP_genomic_DNA_RP.gpr , Cy5-C
0
65535
File: HS-2-70_Q_CP_genomic_DNA_RP.gpr , Cy3-C
GUT

Figure 2.8: *ArrayView. This cutout from a 43.000-spots slide shows the arrangement of subgrids, how the controls are situated (orange dots) and the distribution of bad spots (grey colored spots).*

- bad spots filtered out by quality-criteria: grey

The coloring is automatically adapted to the maximum intensity occuring in the particular channel.

2.2.3.2 Scatterplot and MA-plot

Plotting the $log_2 R$ intensities versus the $log_2 G$ intensities is a common way to display single slide expression data. An alternative is to transform the axes to introduce intensity information.

2.2.3.3 Ratio histogram

Frequency histograms counts the number of ratios for every intensity value and provides information about the distribution of the ratios.

2.2.3.4 QQ plot

A QQ plot compares a given distribution with a normal distribution and therefore gives a short hint how normal the given dataset is distributed.

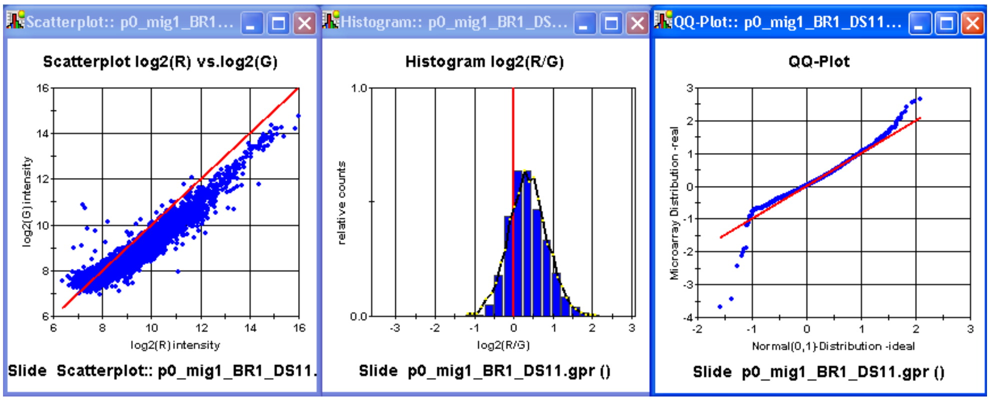


Figure 2.9: *Plots: a: scatterplot, b: ratio-histogram, c: QQ-plot*

2.2.3.5 Boxplot

Boxplots are useful for comparing ratio-values between different groups of data. That can be

- different print-tip groups on a single slide.
- all slides contained in the experiment.

They give a good display of normalization effects.

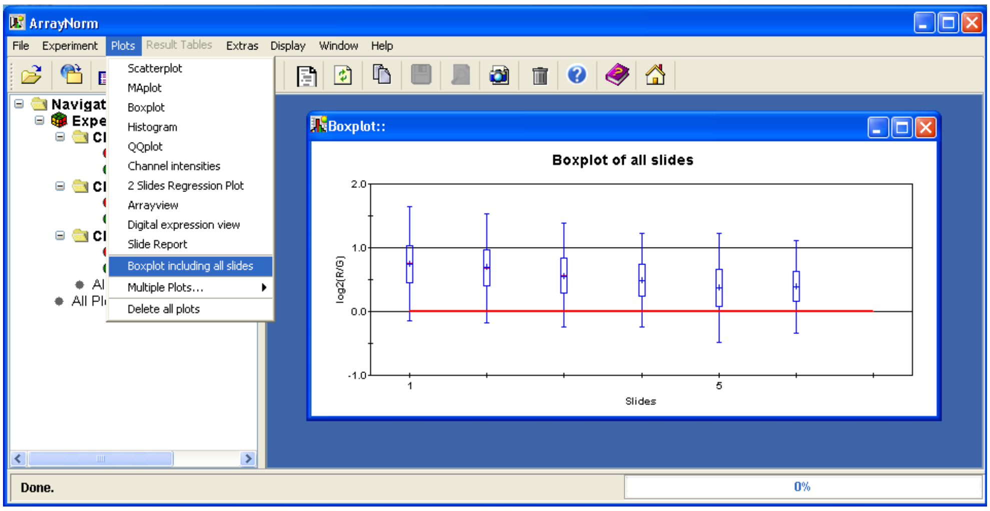


Figure 2.10: *All-slides boxplot: Select Boxplot-including all slides in the Plots-menu. Every slide is displayed by one single box.*

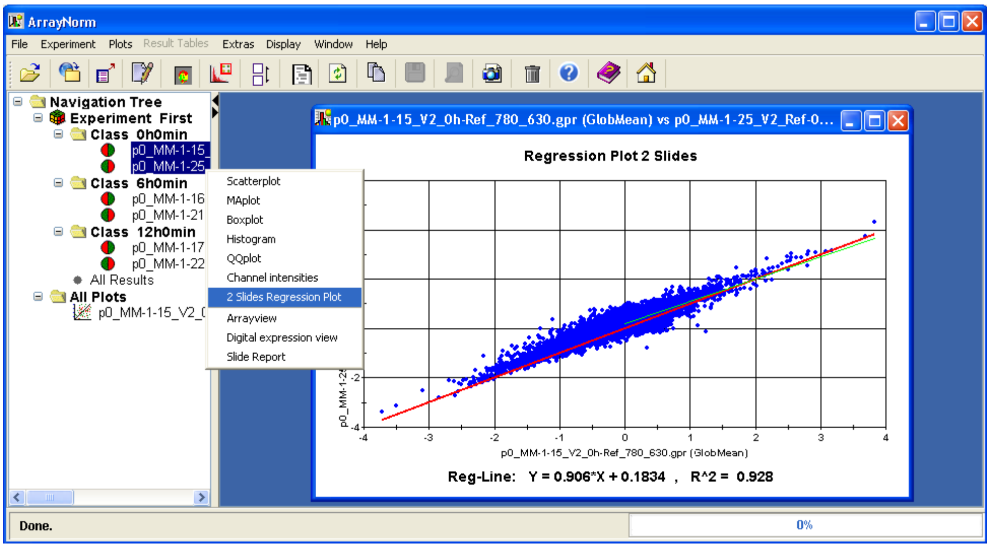


Figure 2.11: *Create a 2SlidesRegressionPlot: Multi-select 2 slides, press right mouse-button and select 2SlidesRegressionPlot!*

2.2.3.6 2-slides regression plot

can be used for comparing 2 single slides. Additionally, a regression line is computed and drawn into the diagram. The 2-slides regression plot is generated by multiselection of the two particular slides and selection of the `2SlidesRegressionPlot`-item in the popup-menu. Note that if a selected slide is reverse labeled, the intensity-channels will be flipped for plotting!

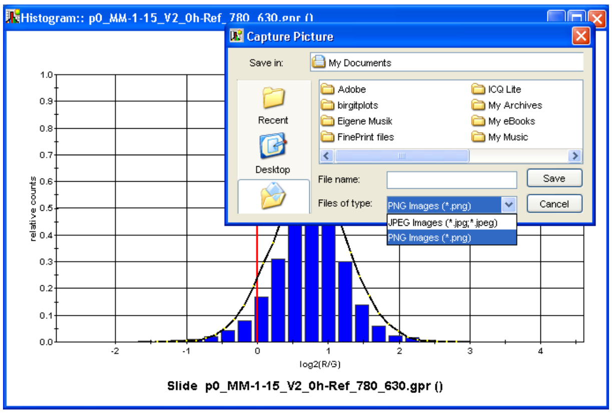
2.2.3.7 Channel intensities plot

It simply draws the raw intensity value of every single spot for both channels. This can be used for detecting patterns due to subgrid effects.

2.2.3.8 Special functions

- **Capturing a plot:** Every open plot can be exported to a file. Possible encoding formats are PNG and JPEG. To capture a plot, select its frame and press the `Capture` button. A filechooser will open for editing filename and encoding-format.

- **Zooming in:** All plots (except ArrayView) come up with a simple zoom-in function. Simply drag a box over the area of interest by holding the left mouse-button. Zoom out



Save in: My Documents

File name:

Files of type: PNG Images (*.png)

Figure 2.12: *Save a plot: Select the plot, press the Capture-button, select a fileformat and edit the filename.*

by pressing R on the keyboard.

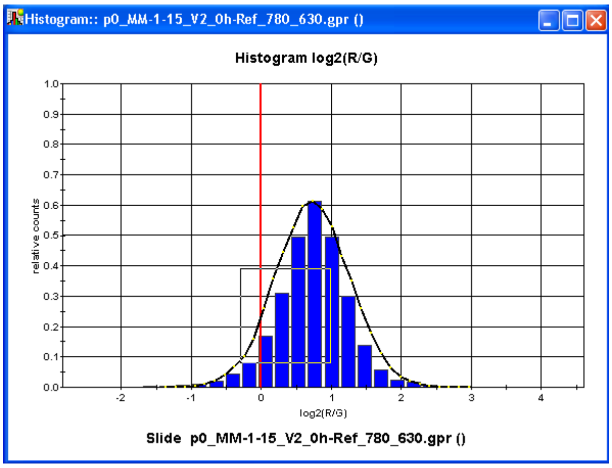

Histogram:: p0_MM-1-15_V2_0h-Ref_780_630.gpr ()
Histogram log2(R/G)
relative counts
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
-2
-1
0
1
2
3
4
log2(R/G)
Slide p0_MM-1-15_V2_0h-Ref_780_630.gpr ()

Figure 2.13: *Zoom in: Drag a box in the plot for zooming in by holding the left mouse-button. Type R for zoom out.*

- **Show all plots:** It is possible to create plots from all loaded slides at once! No matter the number of plots, they will be resized and arranged on the GUI automatically.

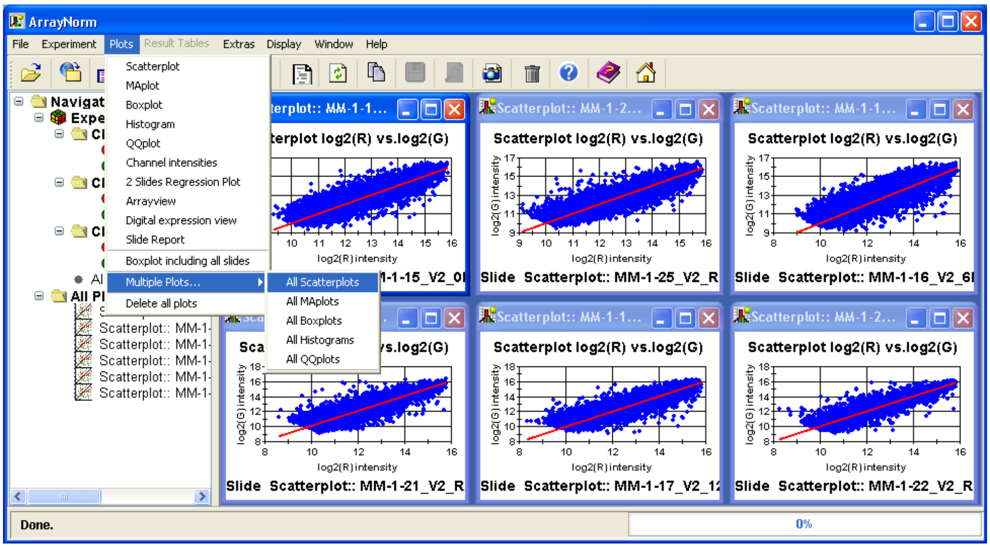

ArrayNorm
File Experiment Plots Result Tables Extras Display Window Help
Scatterplot
MAplot
Boxplot
Histogram
QQplot
Channel intensities
2 Slides Regression Plot
Arrayview
Digital expression view
Slide Report
Boxplot including all slides
Multiple Plots...
Delete all plots
All Scatterplots
All MAplots
All Boxplots
All Histograms
All QQplots
Scatterplot log2(R) vs.log2(G)
log2(R) intensity
log2(G) intensity
Slide Scatterplot:: MM-1-25_V2_R
Slide Scatterplot:: MM-1-16_V2_6I
Slide Scatterplot:: MM-1-21_V2_R
Slide Scatterplot:: MM-1-17_V2_12
Slide Scatterplot:: MM-1-22_V2_R
Done.
0%

Figure 2.14: *All plots at once: Go to the plots-menu and select a plot in the All Plots - submenu.*

2.2.4 Background correction

Background subtraction can be done separately for each class or for the whole experiment at once. If a negative intensity value (`background` > `foreground intensity`) occurs, the particular spot will be marked bad and therefore ignored. Generally, this case will not arise, because pre-filtering (e.g. flagging in GenePix) has already filtered out bad quality spots (see appendix B).

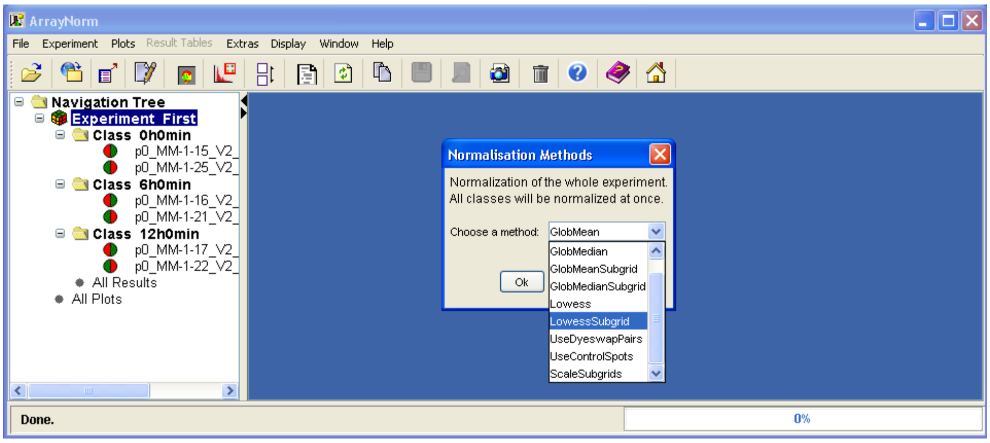
2.2.5 Normalization methods

The user can choose among several normalization methods, depending on the experimental design.

2.2.5.1 Available methods

- *Global-median* or *mean* normalization
- *Global-median* or *mean print-tip group dependent* normalization
- *Intensity-dependent* normalization (Lowess fit)

- *Intensity-dependent* normalization (Lowess fit), *print-tip group dependent*
- Normalization *using control-spots* (positive controls)
- *Paired-slides normalization* (for dye-swap experiments)
- *Scale subgrids* (adjust all subgrids within a single slide)



Normalization of the whole experiment.
All classes will be normalized at once.

Choose a method: GlobMean

Figure 2.15: *Choose a normalization method.*

2.2.5.2 Applying normalization

All methods can be applied for:

- **single classes:** The chosen method will be performed on every slide belonging to the particular class. This allows to use different normalization methods for different classes (e.g. class c1 has got dyeswap-pairs, class c2 just normal labeled slides: normalize c1 with the paired-slides method, c2 with a global or intensity-dependent method).
- **the whole experiment:** All classes will obtain the same method. This is the recommended way to keep classes comparable for analysis.

Using different normalization methods within a class is not possible. It would introduce additional errors and inhibit sensible analysis. In some cases it is necessary to use different normalization methods for different classes (e.g. one class includes dye-swap pairs, other classes just replicated slides). But this should always be an exception. Different methods treat data in different ways and that makes comparisons fewer meaningful.

2.2.5.3 Normalization examples

Two examples illustrate the normalization step.

- **Paired-slides normalization:** Since the already loaded experiment contains dye-swap pairs, the use of self-normalization would be recommended. Each pair will be corrected individually. If no pairs were predefined, the reverse-labelled slides will be averaged for building a template. This template is used as *dye-swap partner* for every normal-labelled slide. Thus, the number of self-normalizations within a class equals the number of normal labelled slides. An informative way to illustrate the effect of self-normalization are boxplots. After normalization, the medians of logratios should be shifted to zero. Additionally, the plot of the normal-labelled slide should be exactly the reversed to its reverse-labelled partner.

- **Normalization with control-spots:** Providing control-spots, slides can be normalized by this subset of genes. This assumes, that the gene-names of the controls are marked with a specific prefix (see Appendix A for ’Marking control genes’). Using the Levenberg-Marquardt algorithm, a polynomial function is fitted to the control spots. This function is used to correct all other genes on the slide. The idea is similar to intensity-dependent normalization, just using another set of genes to fit the function. A common way is to apply Lowess to the set of controls. But this can be critical with less reliable control spots. The example-slide (43000 spots, 1900 controls) just has 150 good-quality controls, after serious quality-filtering in GenePix Pro. That would be too little for Lowess.

- **Scale subgrids:** There are often substantial scale differences between the subgrids of a single slide. A simple scaling of the intensity values is useful to assure that each subgrid has the same median absolute deviation (the same principle is used in the next chapter for scaling slides).

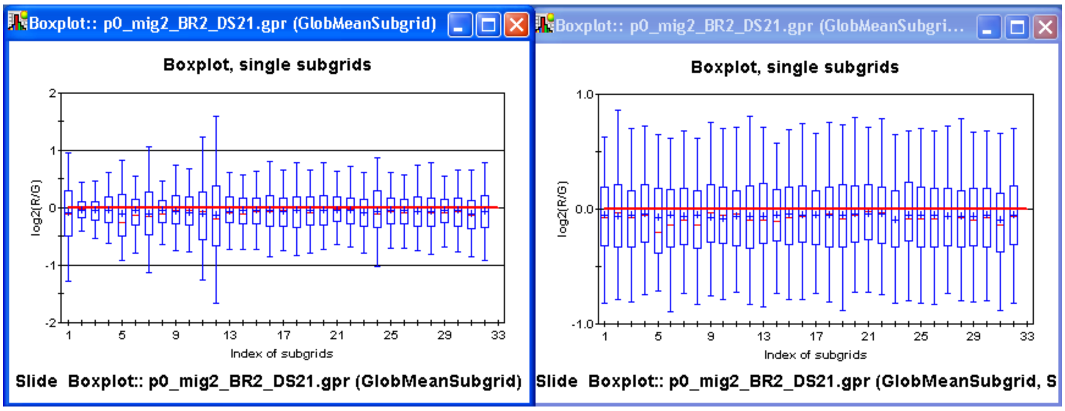

Boxplot:: p0_mig2_BR2_DS21.gpr (GlobMeanSubgrid)
Boxplot, single subgrids
log2(R/G)
2
1
0
-1
-2
1 5 9 13 17 21 25 29 33
Index of subgrids
Slide Boxplot:: p0_mig2_BR2_DS21.gpr (GlobMeanSubgrid)
Boxplot:: p0_mig2_BR2_DS21.gpr (GlobMeanSubgri...
Boxplot, single subgrids
log2(R/G)
1.0
0.0
-1.0
1 5 9 13 17 21 25 29 33
Index of subgrids
Slide Boxplot:: p0_mig2_BR2_DS21.gpr (GlobMeanSubgrid, S

Figure 2.16: *Scale subgrids: a: slide after GlobalMeanSugrid-normalization, b: slide after normalization AND ScaleSubgrids-adjustment.*

2.2.6 Between slides normalization

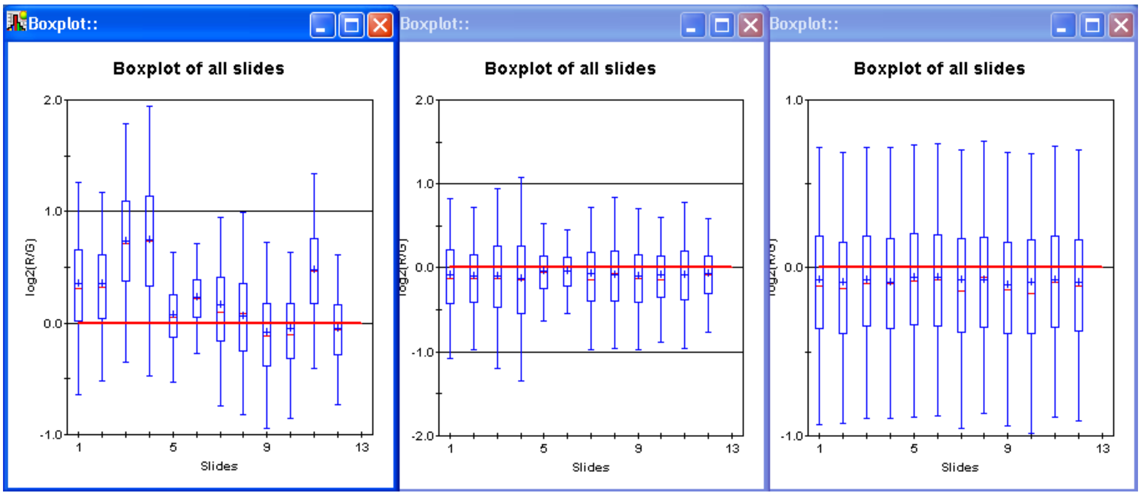
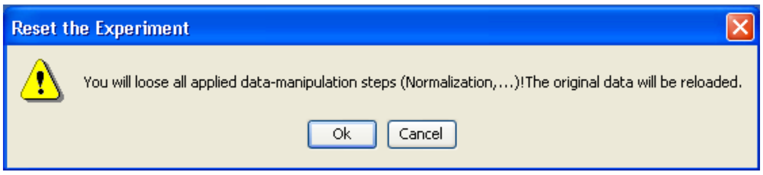


Figure 2.17: *Scale across slides: The effects of normalization and scale adjustment can be displayed using boxplots. a: Raw data before any normalization step, b: slides after global-mean normalization, c: slides after global-mean normalization AND scale-adjustment across slides.*

Individual slides in a multi-slides comparison may need to be adjusted for scale when the single slides have seriously different spreads in their log-ratios. This is done by a simple scaling of the M-values ($log2$-ratios) from all experiment's slides so that each array has the same median absolute deviation (that means consistent widths of the boxplot). Failing to perform such adjustment could lead to one or more slides having undue weight when averaging log-ratios across slides (see chapter replicate handling). Generally, there is a trade-off between the gains achieved by scale-adjustment and the possible increase in variability introduced by this step. In cases with fairly small scale differences it may be preferable to skip the scaling. In practice, the need for scaling will be determined empirically.

2.2.7 Resetting the data



Reset the Experiment

You will loose all applied data-manipulation steps (Normalization,...)!The original data will be reloaded.

Figure 2.18: *Resetting data: reloading the original datasets.*

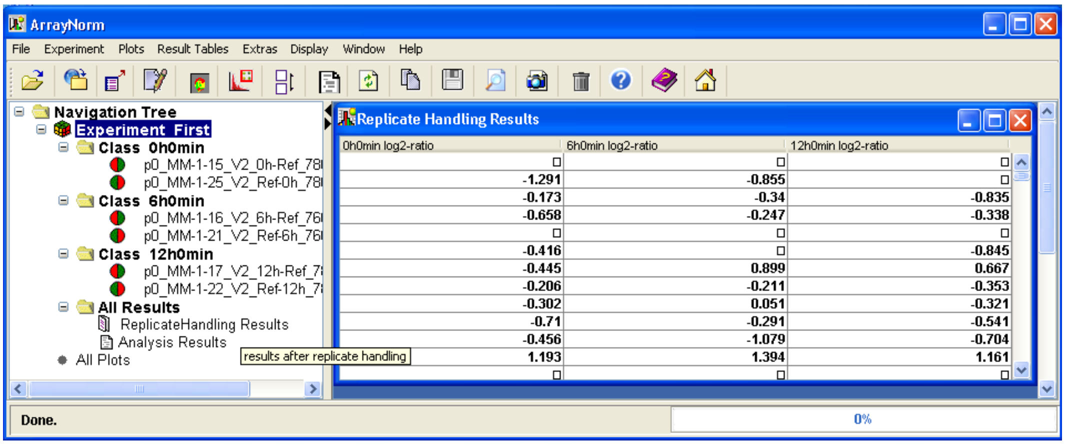
Having applied any changes to the data (like background subraction, normalization, scaling), one might undo all steps. By clicking `Reset Experiment` on the experiment's popup-menu or reset-button, the origin will be reloaded, all changes will be cancelled. Note that already opened plots will keep unaffected.

2.2.8 Finalize, replicate handling and generating results

Before analysis can be carried out, some steps have to be done:

- **Replicate handling.** If there are replicated spots on a single slide, find and average them.
- **Merging slides.** If replicated slides within a class are available, average them. The results are ratio-values for each gene on the slide.
- **Data transformation.** The ratios can be log_2-transformed or not.
- **Export to file.** The resulting values (i.e. ratios or log_2-ratios) can be exported to a file, which is suitable for further software (e.g. Genesis).

These steps are applied to each class. For each gene, the results contain values for ratios, standard-deviations and sample-size. All these values will be needed for statistical testings.



0h0min log2-ratio	6h0min log2-ratio	12h0min log2-ratio
□	□	□
-1.291	-0.855	□
-0.173	-0.34	-0.835
-0.658	-0.247	-0.338
□	□	□
-0.416	□	-0.845
-0.445	0.899	0.667
-0.206	-0.211	-0.353
-0.302	0.051	-0.321
-0.71	-0.291	-0.541
-0.456	-1.079	-0.704
1.193	1.394	1.161
□	□	□

Figure 2.19: *GUI after replicate handling: New icons on the navigation tree indicate successful replicate handling. The table displays all computed ratios for each experimental class.*

Guided by a wizard, the user can define the replicate treatment, the transformation of ratios and if control spots should be printed to the file or not. 2 `Results`-icons added to the data-tree indicates correct results: The first for the general results after replicate handling, the second for all analysis results. Each icon provides a popup-menu for displaying srollable result-tables, which can be exported to a file, too.

2.2.8.1 View replicate handling results via MA-plot

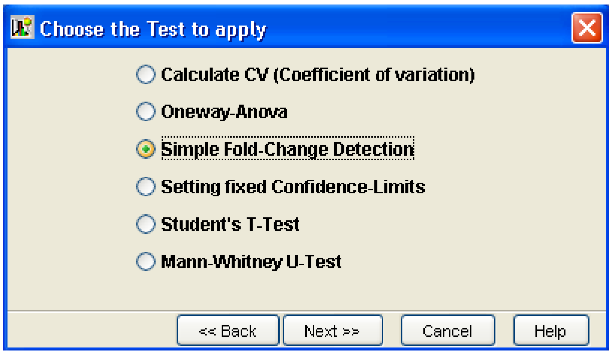
By selecting `Plots->MAPlots of replicate handling results`, MA-plots displaying the M and A values after replicate handling will be opened. One plot reports the M and A values of a particular class. Note that this feature is only accessible after replicate handling.

2.2.9 Analysis

After replicate handling, ratios for every spot and every class are available. This is the assumtion to apply analysis methods for finding differentially expressed genes. ArrayNorm provides some nice possibilities to find subsets of interesting genes.

2.2.9.1 Simple methods

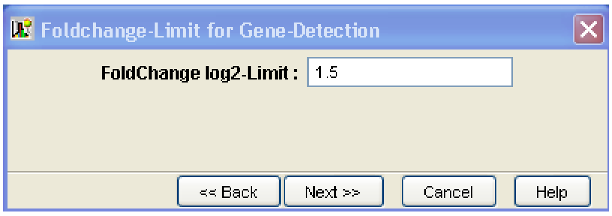
The simple methods address experiments with no or insufficient use of replication (i.e. replicated slides).



○ Calculate CV (Coefficient of variation)
○ Oneway-Anova
● Simple Fold-Change Detection
○ Setting fixed Confidence-Limits
○ Student's T-Test
○ Mann-Whitney U-Test

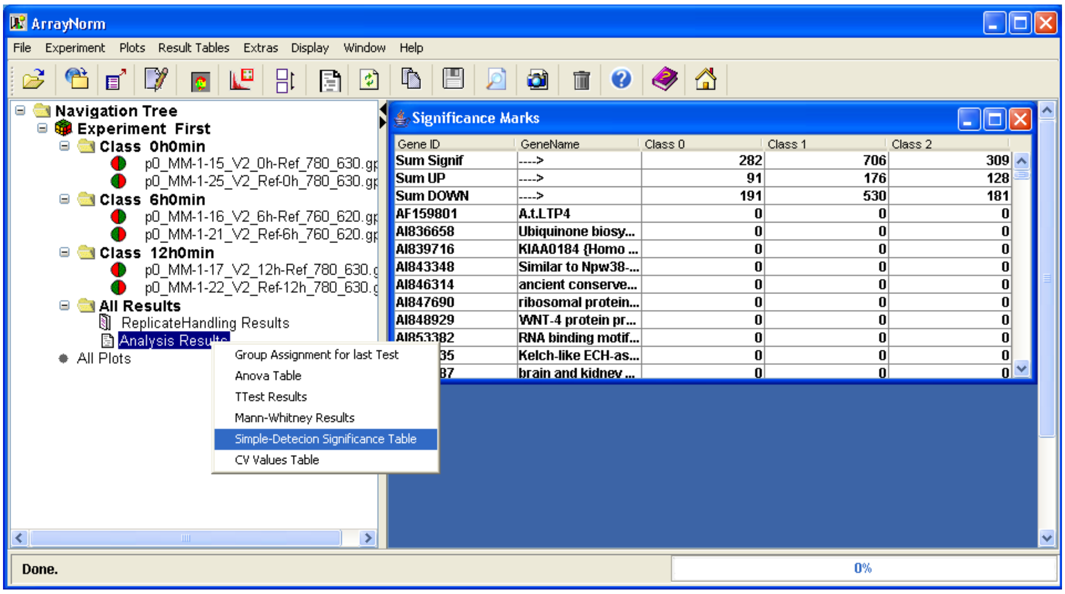
Figure 2.20: *Select Simple foldchange detection in the analysis-wizard.*

- **Defining a fixed fold-change cut-off:** Every gene in every class will be marked as DE, if its absolute intensity-ratio exceeds this threshold. In experiments with more than one class, a gene will be printed to the output file if it is DE in any of the classes. The user can edit the log_2-scaled cut-off. For example, a value of 1.0 denotes a two-fold change in expression.
- **Setting a confidence limit:** For every gene in every class, z-values are computed and compared to a user-defined cut-off score. Those genes with higher z-scores will be marked and treated as above.



FoldChange log2-Limit : 1.5

Figure 2.21: *Edit the foldchange-limit in units of* $log2$ *for deciding whether a gene is DE or not.*


ArrayNorm
File
Experiment
Plots
Result Tables
Extras
Display
Window
Help
Navigation Tree
Experiment First
Class 0h0min
p0_MM-1-15_V2_0h-Ref_780_630.gp
p0_MM-1-25_V2_Ref-0h_780_630.gp
Class 6h0min
p0_MM-1-16_V2_6h-Ref_760_620.gp
p0_MM-1-21_V2_Ref-6h_760_620.gp
Class 12h0min
p0_MM-1-17_V2_12h-Ref_780_630.g
p0_MM-1-22_V2_Ref-12h_780_630.g
All Results
ReplicateHandling Results
Analysis Results
All Plots
Group Assignment for last Test
Anova Table
TTest Results
Mann-Whitney Results
Simple-Detecion Significance Table
CV Values Table
Significance Marks
Done.
0%

Gene ID	GeneName	Class 0	Class 1	Class 2
Sum Signif	---->	282	706	309
Sum UP	---->	91	176	128
Sum DOWN	---->	191	530	181
AF159801	A.t.LTP4	0	0	0
AI836658	Ubiquinone biosy...	0	0	0
AI839716	KIAA0184 {Homo ...	0	0	0
AI843348	Similar to Npw38-...	0	0	0
AI846314	ancient conserve...	0	0	0
AI847690	ribosomal protein...	0	0	0
AI848929	WNT-4 protein pr...	0	0	0
AI853382	RNA binding motif...	0	0	0
35	Kelch-like ECH-as...	0	0	0
87	brain and kidney ...	0	0	0

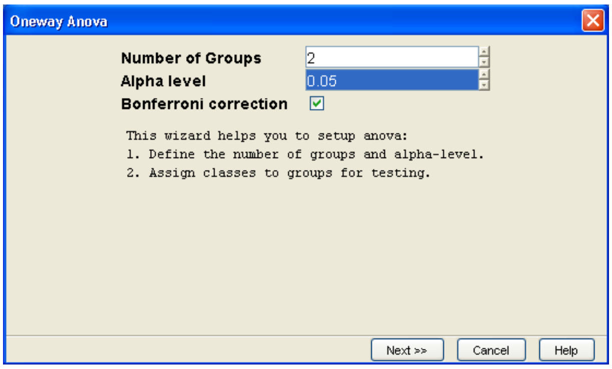
Figure 2.22: *Select the menu-item of the analysis-results icon for displaying the results-table.*

2.2.9.2 Statistical tests

Providing a sufficient number of biological replicates for each gene, statistical tests can be applied. In terms of replication, ArrayNorm treats different classes as biological replicates. All slides within one class are treated as technical replicates. The following statistical methods require biological replication. For instance: 3 classes belonging to one single timepoint of an experiment can be used as 3 biological replicates.

Before starting any statistical test, you have to define groups of classes. A wizard leads you through this step. You can edit the number of groups (except t-test and mann-whitney: always 2 groups!) and the critical probability-level. A table with checkboxes gives the opportunity to assign classes to particular groups. Logically, one class can only belong to one group. In terms of replication, the classes within one group are treated like biological replicates (as described above).

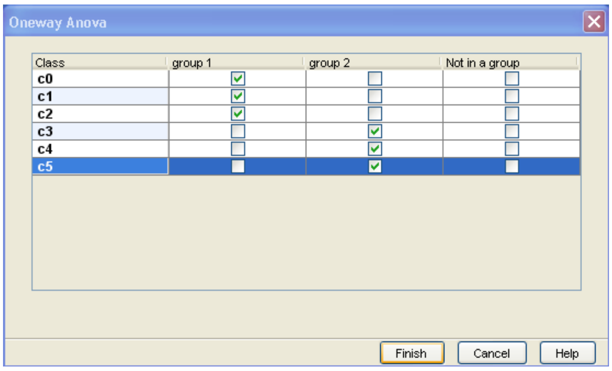
- **Oneway-ANOVA:** Analysis of variance allows to detect significant differences between multiple groups. For every gene, ANOVA calculates F-statistic and p-value. If the p-value falls below the critical alpha level, the particular gene is marked as significant or



Number of Groups: 2
Alpha level: 0.05
Bonferroni correction: ☑

```
This wizard helps you to setup anova:
1. Define the number of groups and alpha-level.
2. Assign classes to groups for testing.
```

Figure 2.23: *ANOVA: Select the number of groups to compare, the significance-limit and if Bonferroni-stepdown should be applied.*



Class	group 1	group 2	Not in a group
c0	☑	□	□
c1	☑	□	□
c2	☑	□	□
c3	□	☑	□
c4	□	☑	□
c5	□	☑	□

Figure 2.24: *ANOVA group assignment: Define which classes belong to which groups.*

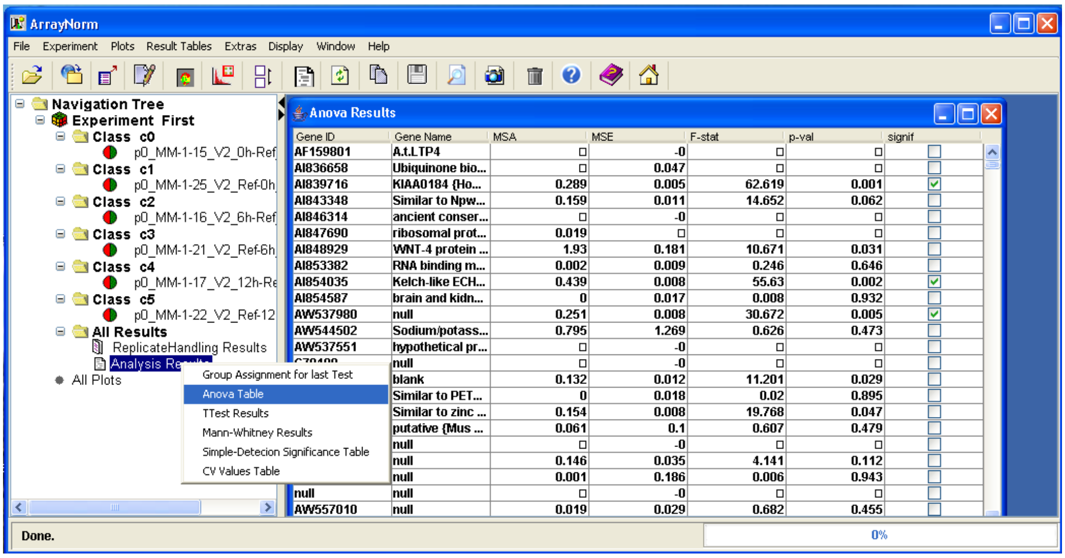
differentially expressed. Note that ANOVA does not tell you between which groups this difference has appeared!

- **T-Test:** A Student's t-test detects differences between 2 groups. The main assumption for the t-test is normality of your data. Unfortunately, this is rarely the case. With aid of t-statistic and sample size, a p-value for every gene is computed and compared to a certain alpha level. Like ANOVA, significant genes are marked as DE.

- **Mann-Whitney Test:** This is another statistical test for finding differences between 2 groups. Here, the assumption of normality need not to be fulfilled.

- **CV Values:** In principle, the coefficient of variation CV is the standard deviation divided by the mean. The quality of data is inversely related to its CV. ArrayNorm computes a CV value for each spot in each group (you have to define groups of classes analogous to ANOVA or T-test). You can interpret a low CV as high consistency of the biological replicates. For every gene in every class, a CV value is computed and compared to an editable upper CV-limit.

After every test, a dialog tells you how many genes were found as DE. To get an overview which genes were marked, right-click the Analysis-Results folder in the navigation tree and select the table you want to view. Each **table** can be **saved** to a textfile. Simply rightclicking the table will open a filechooser. To account for **multiple testing**, it is possible to adjust the critical alpha limit by **Bonferroni step-down** (set by default). This means dividing the edited alpha-limit by the number of tests (here: the number of genes).



Gene ID	Gene Name	MSA	MSE	F-stat	p-val	signif
AF159801	A.t.LTP4	□	-0	□	□	☐
AI836658	Ubiquinone bio...	□	0.047	□	□	☐
AI839716	KIAA0184 {Ho...	0.289	0.005	62.619	0.001	☑
AI843348	Similar to Npw...	0.159	0.011	14.652	0.062	☐
AI846314	ancient conser...	□	-0	□	□	☐
AI847690	ribosomal prot...	0.019	□	□	□	☐
AI848929	WNT-4 protein ...	1.93	0.181	10.671	0.031	☐
AI853382	RNA binding m...	0.002	0.009	0.246	0.646	☐
AI854035	Kelch-like ECH...	0.439	0.008	55.63	0.002	☑
AI854587	brain and kidn...	0	0.017	0.008	0.932	☐
AW537980	null	0.251	0.008	30.672	0.005	☑
AW544502	Sodium/potass...	0.795	1.269	0.626	0.473	☐
AW537551	hypothetical pr...	□	-0	□	□	☐
[illegible]	null	□	-0	□	□	☐
	blank	0.132	0.012	11.201	0.029	☐
	Similar to PET...	0	0.018	0.02	0.895	☐
	Similar to zinc ...	0.154	0.008	19.768	0.047	☐
	putative {Mus ...	0.061	0.1	0.607	0.479	☐
	null	□	-0	□	□	☐
	null	0.146	0.035	4.141	0.112	☐
	null	0.001	0.186	0.006	0.943	☐
null	null	□	-0	□	□	☐
AW557010	null	0.019	0.029	0.682	0.455	☐

Figure 2.25: *ANOVA results table: Showing all important statistics and genes marked as DE.*

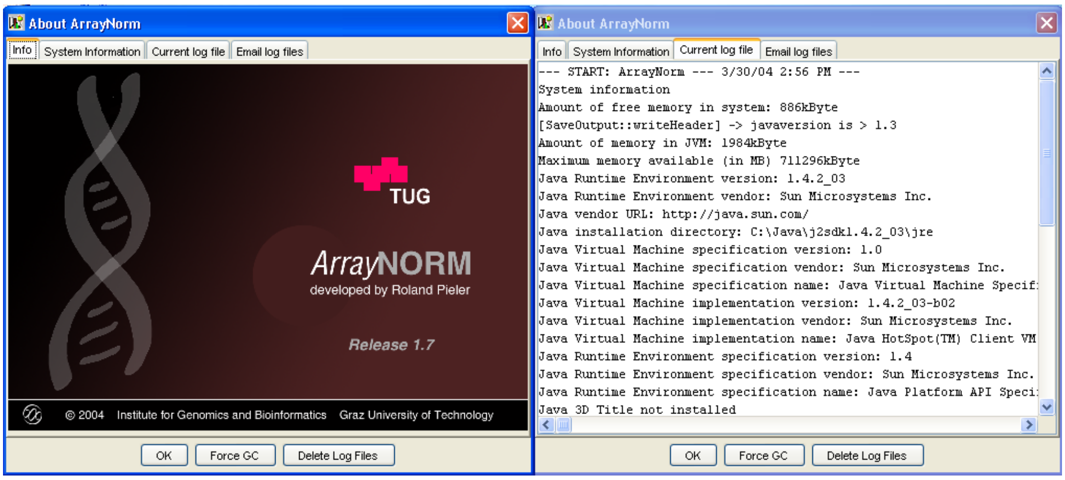
2.2.9.3 Output files

Having figured out DE genes, the gene-names, gene-IDs and log_2-ratio values (or optional the p-values) for every class can be saved to a text-file. The file's format is compatible with the Genesis clustering software. The textfile will contain one ratio-column per experimental class. The columns will be named with the correct class-names (edited by the user when setting up the experiment).

2.2.10 Extras

2.2.10.1 About box

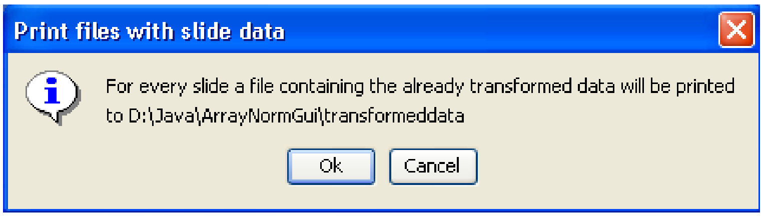
The About box allows to get system informations (e.g. release-number, Java VM) and to view the current logfile. The user can send a mail to the developer with the logfile automatically attached.



```
--- START: ArrayNorm --- 3/30/04 2:56 PM ---
System information
Amount of free memory in system: 886kByte
[SaveOutput::writeHeader] -> javaversion is > 1.3
Amount of memory in JVM: 1984kByte
Maximum memory available (in MB) 711296kByte
Java Runtime Environment version: 1.4.2_03
Java Runtime Environment vendor: Sun Microsystems Inc.
Java vendor URL: http://java.sun.com/
Java installation directory: C:\Java\j2sdk1.4.2_03\jre
Java Virtual Machine specification version: 1.0
Java Virtual Machine specification vendor: Sun Microsystems Inc.
Java Virtual Machine specification name: Java Virtual Machine Specif:
Java Virtual Machine implementation version: 1.4.2_03-b02
Java Virtual Machine implementation vendor: Sun Microsystems Inc.
Java Virtual Machine implementation name: Java HotSpot(TM) Client VM
Java Runtime Environment specification version: 1.4
Java Runtime Environment specification vendor: Sun Microsystems Inc.
Java Runtime Environment specification name: Java Platform API Speci:
Java 3D Title not installed
```

Figure 2.26: *Aboutbox: The user can get information about system properties and the current logfile. It is possible to mail the logfile to the developer and delete all logfiles.*

2.2.10.2 Microarray data files



For every slide a file containing the already transformed data will be printed to D:\Java\ArrayNormGui\transformeddata

Figure 2.27: *Generate files with slide data.*

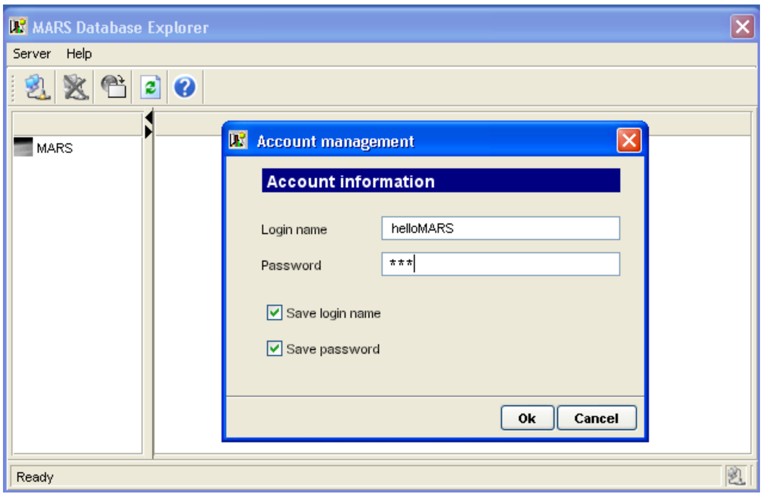
It might be useful to have output files for every single slide of the experiment. By clicking the *Generate slide-data files*-button or -menu-item, textfiles for every slide will be printed. They come with columns for GeneID, GeneName, F635- and F532-intensity, M-value and A-value. It is possible to generate these files in every step of the normalization process.

2.2.10.3 Export MA values to textfile

Additionally to the outputfiles (including the ratios or $log2$-ratios) the user can save M- and A-values for every class to a textfile. For each class, the file will contain a column with M and a column with A, respectively. This colums present the already averaged values within the particular class. Note that this file is not compatible with the Genesis-software. The major output of ArrayNorm remains the $log2$-ratio output file.

2.2.11 Connecting to MARS (in development)

ArrayNorm provides connectivity to the MARS database for downloading experiments (source-files and design parameters) and uploading analysis results. Currently, only the download was implemented.



Login name helloMARS

Password ***

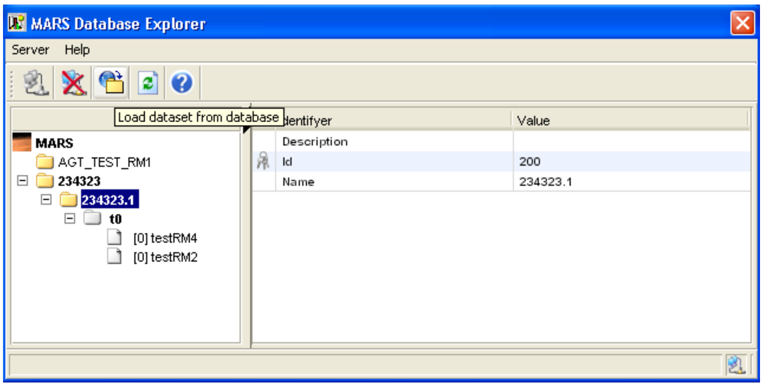
☑ Save login name

☑ Save password

Figure 2.28: *MARS explorer: Edit user-account settings and server properties. Then click Connect to Database for displaying all microarray-experiments stored in MARS.*

2.2.11.1 Loading experiments from MARS

- **MARS Explorer:** Press the `Open DB` button to access the MARS explorer.
- **Edit server settings:** In the `Server` menu, edit server- and user-properties. For proper settings, please ask the MARS developer at the Insitute for Genomics and Bioinformatics,



Identifyer	Value
Description	
Id	200
Name	234323.1

Figure 2.29: *Loading a subexperiment: Select the subexperiment and click the Download-button.*

TU Graz.

- **Connect to MARS:** Press the `Connect to Mars` button to open a connection to MARS. It will take some time to display a naviation-tree of all stored experiments.
- **Select and load subexperiment:** In MARS, the experiment's structure is organized in subexperiments. Expanding the appropriate subexperiment-folder should show all classes and slides belonging to this experiment. Select your subexperiment and press the `Download Subexperiment` button. It may take plenty of time to load all files and the experiment's structure.

In detail, ArrayNorm gets the sourcefiles as attachments from MARS, stores them locally and then parses the files. Since the experiment's structure is already defined in MARS, ArrayNorm builds up the navigation tree automatically without the need to edit anything in the setup-wizard.

2.2.11.2 Upload results to MARS

After successful replicate handling, ArrayNorm automatically generates files for uploading to MARS. For every single microarray, one file will be printed, including all necessary data-columns for MARS. By clicking the *uploading slide result files* - button, a connection to MARS will be opened to save the files to the database. Once the files are sent, they will be deleted from

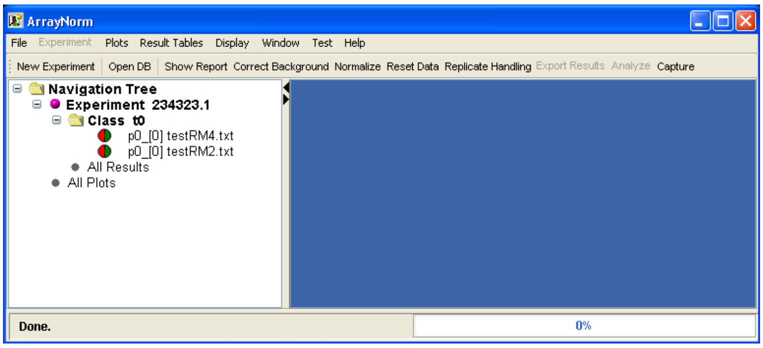

ArrayNorm
File Experiment Plots Result Tables Display Window Test Help
New Experiment Open DB Show Report Correct Background Normalize Reset Data Replicate Handling Export Results Analyze Capture
Navigation Tree
Experiment 234323.1
Class t0
p0_[0] testRM4.txt
p0_[0] testRM2.txt
All Results
All Plots
Done.
0%

Figure 2.30: *Successful download: The loaded experiment consists of class t0, containing 2 normal labeled slides.*

the local directory. Of course, the upload is only possible if the experiment was downloaded from MARS.

Chapter 3

Normalization workflow

3.1 Order of steps

Normalization of microarray experiments require a kind of *workflow*. Different steps of data manipulation have to happen in a certain order (e.g. analysis is simply impossible without having already applied replicate handling, or background subtraction should always happen before normalization) to produce reasonable output. Here is a **checklist of the correct order of functions** provided by the software and which are required, recommended, optional or done automatically.

1. **Experiment setup**
   - (a) select source files *(required)*
   - (b) define experiment design *(required)*
   - (c) load files *(automatic)*
2. **Visualization**
   - (a) generate different plots to view data *(recommended)*
   - (b) save important plots to image-files *(optional)*
3. **Data manipulation**

   (a) correct background *(optional)*

   (b) choose normalization method for every class *(recommended)*

   (c) apply normalization to classes *(recommended)*

   (d) run scale-adjustment across slides *(recommended)*

   (e) check normalization effects via plots *(recommended)*

   (f) save textfile with data for every microarray *(optional)*

4. **Replicate handling**

   (a) average replicated spots within every slide *(optional)*

   (b) average and merge all slides within a class *(automatic)*

   (c) calculate ratios *(automatic)*

   (d) $log2$-transform the ratios *(recommended)*

   (e) save results to a tab-delimited textfile *(recommended)*

5. **Analysis**

   (a) find DE genes with simple or statistics methods *(recommended)*

   (b) save list of found DE genes to textfile *(recommended)*

   (c) do further analysis with other software (e.g. cluster analysis)

3.2 Bad examples

ArrayNorm does not always interdict users to violate the above order of steps. Here are some examples how data should not be modified.

- **Normalization before background correction:** If the background should be subtracted, do it always before any other normalization or scaling step.

- **ScaleSubgrids-adjustment before normalization:** Scaling subgrids just looks for differences in the spreads of the subgrids and assumes that the dye effects are already corrected.
- **Scale-adjustment across slides before within-slide normalization:** The same reason as the last point.
- **Replicate handling before normalization:** Any method of normalization does not affect already finished results. Always repeat replicate handling to update the results.

Chapter 4

For technical assistance:

This software was developed by Roland Pieler.

This manual was written by Roland Pieler.

mailto: roland.pieler@tugraz.at

web: genome.tugraz.at

Address: Krenngasse 37, A-8010 Graz, Austria

Appendix A

Marking Control Genes

A simple strategy to mark control-elements is to add a well defined prefix to the gene-name. ArrayNorm can use these information to identify control-spots and make use of it. To distinguish between different types,a list of possible control-elements and appropriate prefixes was considered. This list should be binding if any control-elements are spotted to microarray-slides.

Prefix	**Describtion**
C_	Control
CN_	negative control
CPG_	pos. control - Genomic DNA
CPH_	pos. control - housekeeping genes
CPS_	pos. control - spike control
CPM_	pos. control - microarray sample pool (MSP)

Table A.1: Table of possible prefixes. A prefix is added to a gene-name.

Appendix B

GenePix Flagging Criteria

Bad-quality spots on a microarray heavily affects all data analysis steps and the experiment's results. Most important is to mark bad spots to remove them from further analysis.

The GenePix Pro software features a 'Flag feature' dialog box, where multiple criterias can be linked to a boolean query. Here is one example for such a query.

```
[Flags]            =  [Bad]       Or
[Flags]            = [Absent]    Or [Flags]            = [Not
Found] Or
[F635 % Sat.]      >  10           Or
[F532 % Sat.]      >  10           Or
[Sum of Medians]   <  1000  Or [Sum of Means] < 1000 Or
([% > B635+1SD] < 55  Or
([% > B532+1SD] < 55  Or
(([F532 Mean]-[F532 Median])/[F532 Mean])> 0.2 Or (([F635
Mean]-[F635 Median])/[F635 Mean])> 0.2
```

Every single spot on a microarray has to pass these criteria, otherwise it would be marked as ’bad’.

This reference-query was developed by Hubert Hackl and is used for the majority of microarray-experiments in the *TU-Graz Bioinformatics Group*.

Appendix C

Supported file formats

These file formats can be loaded into ArrayNorm:

Name	Extensions
GenePixPro	gpr, txt, xls
GenePixSinglChannel	gpr, txt, xls
Agilent	txt, xls
SensiChip	txt, xls
ImaGene	txt, xls
ArrayVision	txt, xls
Mars	txt, xls

Table C.1: Table of possible file formats.