

# oneChannelGUI Package Vignette

Raffaele A Calogero, Francesca Cordero, Remo Sanges

May 12 2011

## 1 Introduction

This package is an add-on of affyImGUI for *mouse-click* based QC, statistical analysis and data mining for one channel microarray data. It is designed for Bioconductor beginners having limited or no experience in interacting with Bioconductor line commands. OneChannelGUI is a set of functions extending the affyImGUI capabilities, rearranging and extending affyImGUI menus.

This package performs, in a graphical environment, the analysis pipe-line shown in figure 1, green box.



Figure 1: Microarray analysis pipe-line.

This vignette gives a general overview of the graphical interfaces available in oneChannelGUI for microarray gene-level analysis.

N.B:

All the oneChannelGUI graphical outputs are visualized in the R main window, to reduce RAM consumption, which is a critical issue when Affymetrix array data or large set of data are loaded.

Furthermore, exon data generated with APT tools produce, in the working directory,

a certain amount of temporary files and directories.

A cleanup function is under development.

At the present time, user can manually remove, from the working folder, any file starting with target, elevels, glevels, e.g. target51f81aeb, elevels3e9f6b76, and folders starting with out and outMidas, e.g. out17fb164, outMidas4a31ac4, without affecting the results stored in oneChannelGUI.

## 2 Installation

For the complete functionality of oneChannelGUI some external softwares and data need to be installed. Please refer to the *install vignette* of oneChannelGUI package.

## 3 Main graphical window

oneChannelGUI inherits the core functionalities of affylmGUI and its main GUI. In oneChannelGUI some extra topics are available in the main affylmGUI info left frame, e.g. maSigPro results, Normalized Exon data, APT DABG, APT MiDAS, Splice Index, etc. Furthermore, four different menus are automatically exchanged depending on the type of array loaded:

1. .CEL IVT Affymetrix arrays.
2. .CEL exon 1.0 ST arrays uploaded in oneChannelGUI by Affymetrix APT tools or gene/exon data exported from Affymetrix Expression Console.
3. .CEL Gene 1.0 ST arrays uploaded in oneChannelGUI by Affymetrix APT tools.
4. GEO/flat tab delimited expression data file.
5. ILLUMINA output from BeadStudio software version 1, 2, 3 and GenomeStudio.

Each item in the menus is simply a graphical implementation of a function of a specific Bioconductor library , e.g. ssize: sample size and statistical power estimation. To get more information on those libraries please refer to their specific vignettes, accessible from the *Help menu*.

## 4 File

This menu allows the loading of .CEL IVT Affymetrix arrays as well as exon arrays, GEO Matrix Series files, tab delimited files containing only expression data and ILLUMINA data produced by BeadStudio and GenomeStudio. In this menu, fig. 2, are given the main functionalities to handle a microarray analysis project.

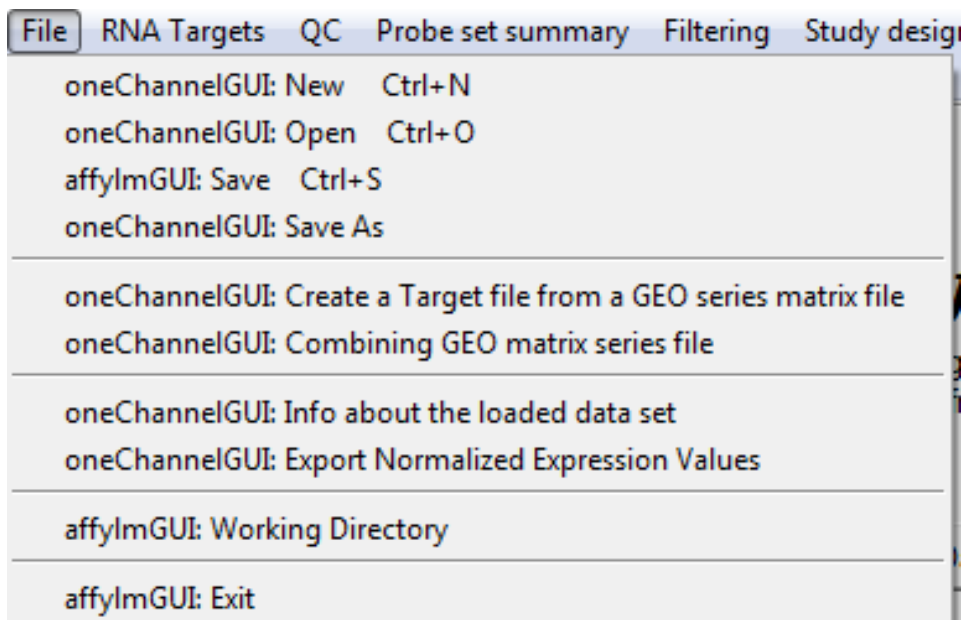


Figure 2: File menu.

### 4.1 New

The item *New*, fig. 2, allow to load various types of array data, using the sub menu shown in fig. 3,

#### 4.1.1 Target file structure

To load arrays oneChannelGUI uses the information available in a file describing the experimental structure of the data set. This file is called *target file* and it is a tab delimited file with a fixed header structure also used by affylmGUI, fig. 4.

**IMPORTANT:**

**TARGET FILE MUST NOT CONTAIN CHARACTERS LIKE ; , : \_ - | \ ! ? + \* ^ ( ) [ ] { }**



Figure 3: New: array type selection menu.

	A	B	C
1	<b>Name</b>	<b>FileName</b>	<b>Target</b>
2	mC1	M1.CEL	mcf-7ctrl
3	mC2	M4.CEL	mcf-7ctrl
4	mC3	M7.CEL	mcf-7ctrl
5	mE1	M3.CEL	mcf-7E2
6	mE2	M6.CEL	mcf-7E2
7	mE3	M9.CEL	mcf-7E2
8	ml1	M2.CEL	mcf-7IGF
9	ml2	M5.CEL	mcf-7IGF
10	ml3	M8.CEL	mcf-7IGF
11	sC1	S1.CEL	sk-er3ctrl
12	sC2	S4.CEL	sk-er3ctrl
13	sC3	S7.CEL	sk-er3ctrl
14	sE1	S3.CEL	sk-er3E2
15	sE2	S6.CEL	sk-er3E2
16	sE3	S9.CEL	sk-er3E2
17	sl1	S2.CEL	sk-er3IGF
18	sl2	S5.CEL	sk-er3IGF
19	sl3	S8.CEL	sk-er3IGF

Targets file

Selected the "targets" file.  
Then press OK to continue

Please select a filename

No filename is selected at the moment. Press the Select Targets File Button.

Select Targets File

OK Cancel

Targets file is a tab delimited **text file** containing the description of the experiment. It is made of three columns:  
**Name**: the name you want to assign to each array.  
**FileName**: the names of the corresponding .CEL file  
**Target**: the experimental condition associated to the array (e.g. mock, treated, etc). At least two conditions should be present.

Figure 4: Target file structure.

#### 4.1.2 Loading Affy .CEL files

This sub menu, fig. 3, is entirely inherited by affylmGUI and allows to load .CEL files, if a Bioconductor cdf file is available. User will be asked to select the working folder, i.e. the one in which are present the .CEL files and the target file.

#### 4.1.3 Loading EXON/GENE ARRAYS

This sub menu, fig. 3, allows to load exon/gene 1.0 ST arrays starting from .CEL, taking advantage of Affymetrix APT tools (<http://www.affymetrix.com/support/developer/powertools/index.affx>), or flat tab delimited files containing gene/exon level expression data exported from Affymetrix Expression Console (EC, [http://www.affymetrix.com/support/technical/software\\_downloads.affx](http://www.affymetrix.com/support/technical/software_downloads.affx)). If APT tool option is not used (it works only for Exon 1.0 ST data exported from EC), a sub-menu allows to select, for tab delimited data, the organism and the subset of exon data to be evaluated, fig. 5

**IMPORTANT:**

*TO USE APT TOOLS THE DOWNLOAD OF GENE/EXON LIBRARY FILES IS REQUIRED.*

*THIS CAN BE DONE WITH THE FUNCTION*

*oneChannelGUI: Set library folder and install Affy gene/Exon library files  
LOCATED IN THE GENERAL TOOLS MENU*

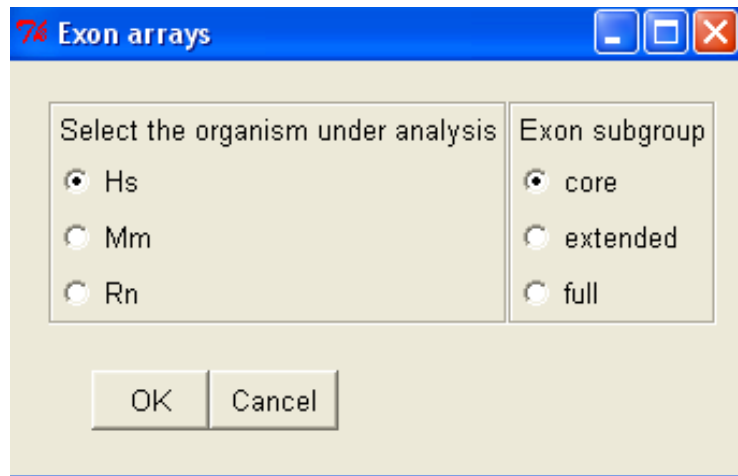


Figure 5: Sub menu to define the organism and the subset of exon data that will be loaded.

Subsequently, the user will select:

1. a working directory, a target file,

2. the flat tab delimited files containing respectively gene-level and exon-level data.

If instead, APT tool option is selected, user will select:

1. the organism and the subset of exon arrays to be evaluated, fig. 5,
2. a working directory,
3. a target file,
4. the type of probe set summary to be applied to gene/exon level data, fig. 6.

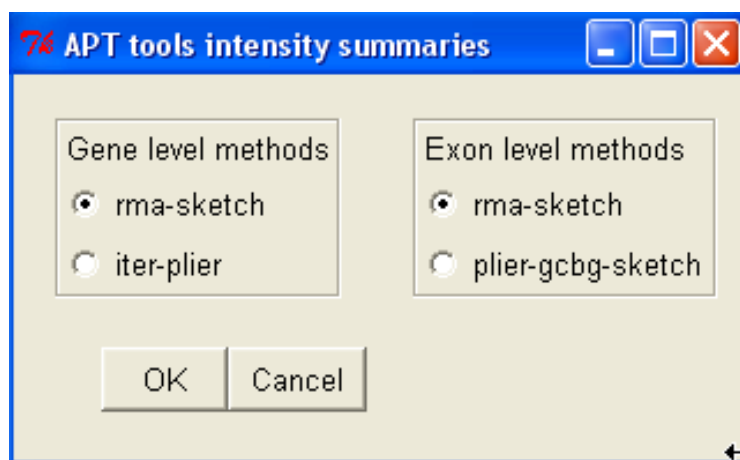


Figure 6: Sub menu to define the type of probe set summary to be applied.

Concerning probe set summary options, fig. 6, PLIER/RMA model-based algorithms are available. Probe set summary calculation and uploading will take few minutes depending on the number of .CEL to be loaded and the PC in use. Once probe set summary has been calculated, using APT tool, it is also possible to calculate DABG p-values, fig. 7.

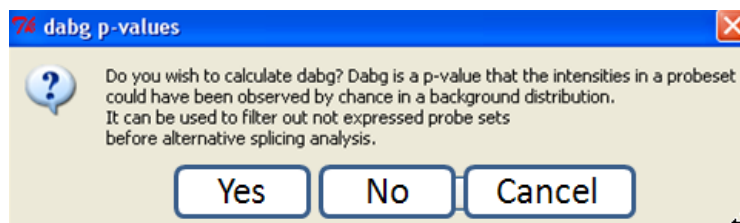


Figure 7: Selecting DABG p-value calculation.

DABG p-values represent *data above background*, it is a p-value similar to that used to derive presence/absence calls in MAS 5.0. DABG p-values could be useful to remove

low intensity signals which could produce mis-leading results when alternative splicing events are evaluated using the Splice Index, where signal intensity information is not considered.

The progress of the probe set summary calculation is shown in the main R window.

```
Gene level probe sets summary started
Read 6 cel files from: target3d92750
Opening bgp file: HuEx-1_0-st-v2.r2.antigenomic.bgp
Opening clf file: HuEx-1_0-st-v2.r2.clf
Opening pgf file: HuEx-1_0-st-v2.r2.pgf
Expecting 1 iteration.
Doing iteration: 1
Opening clf file: HuEx-1_0-st-v2.r2.clf
Opening pgf file: HuEx-1_0-st-v2.r2.pgf
Loading 22011 probesets and 908532 probes.
Reading 6 cel files.....Done.
Processing Probesets.....Done.
Cleaning up.
Done.
Run took approximately: 9.56 minutes.
```

Gene level probe sets summary ended

Gene level probe sets summary ended

Exon level probe sets summary started

```
Exon level probe sets summary started
Read 6 cel files from: target3d92750
Opening bgp file: HuEx-1_0-st-v2.r2.antigenomic.bgp
Opening clf file: HuEx-1_0-st-v2.r2.clf
Opening pgf file: HuEx-1_0-st-v2.r2.pgf
Expecting 1 iteration.
Doing iteration: 1
Opening clf file: HuEx-1_0-st-v2.r2.clf
Opening pgf file: HuEx-1_0-st-v2.r2.pgf
Loading 287329 probesets and 1111849 probes.
Reading 6 cel files.....Done.
Processing Probesets.....Done.
Cleaning up.
Done.
Run took approximately: 6.41 minutes.
```

*Exon level probe sets summary ended*

*Exon level probe sets summary ended*

*DABG calculation started*

*Read 6 cel files from: target3d92750*

*Opening bgp file: HuEx-1\_0-st-v2.r2.antigenomic.bgp*

*Opening clf file: HuEx-1\_0-st-v2.r2.clf*

*Opening pgf file: HuEx-1\_0-st-v2.r2.pgf*

*Expecting 1 iteration.*

*Doing iteration: 1*

*Opening clf file: HuEx-1\_0-st-v2.r2.clf*

*Opening pgf file: HuEx-1\_0-st-v2.r2.pgf*

*Loading 22011 probesets and 908532 probes.*

*Reading 6 cel files.....Done.*

*Processing Probesets.....Done.*

*Cleaning up.*

*Done.*

*Run took approximately: 3.55 minutes.*

*DABG calculation ended*

#### **4.1.4 Loading GENE ARRAYS**

This sub menu, fig. 3, allows to load gene 1.0 ST arrays starting from .CEL, taking advantage of Affymetrix APT tools (<http://www.affymetrix.com/support/developer/powertools/index.affx>). Subsequently, the user will select:

1. the organism and the subset of exon arrays to be evaluated, fig. 8,
2. a working directory,
3. a target file,
4. the type of probe set summary to be applied to gene/exon level data, fig. 9.

Probe set summary calculation and uploading will take few minutes depending on the number of .CEL to be loaded and the PC in use.

#### **4.1.5 Loading ILLUMINA BeadStudio**

This sub menu, fig. 3, allows user to load expression data starting from the output of BeadStudio software. The interface allows to load outputs generated by BeadStudio version 1, 2 and 3. The Bioconductor annotation libraries for illumina arrays are associated to the loaded data. Since output of BeadStudio is not log2 transformed, a



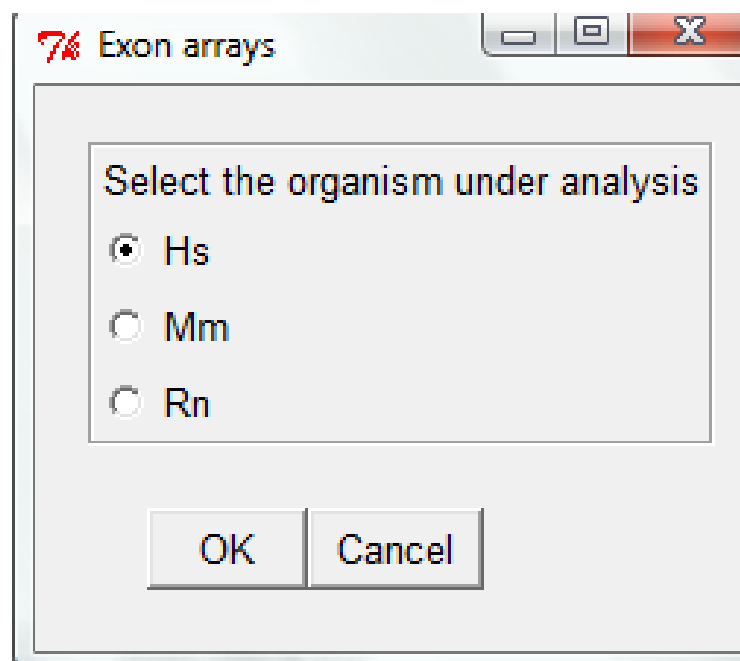


Figure 8: Sub menu to define the organism and the subset of data that will be loaded.

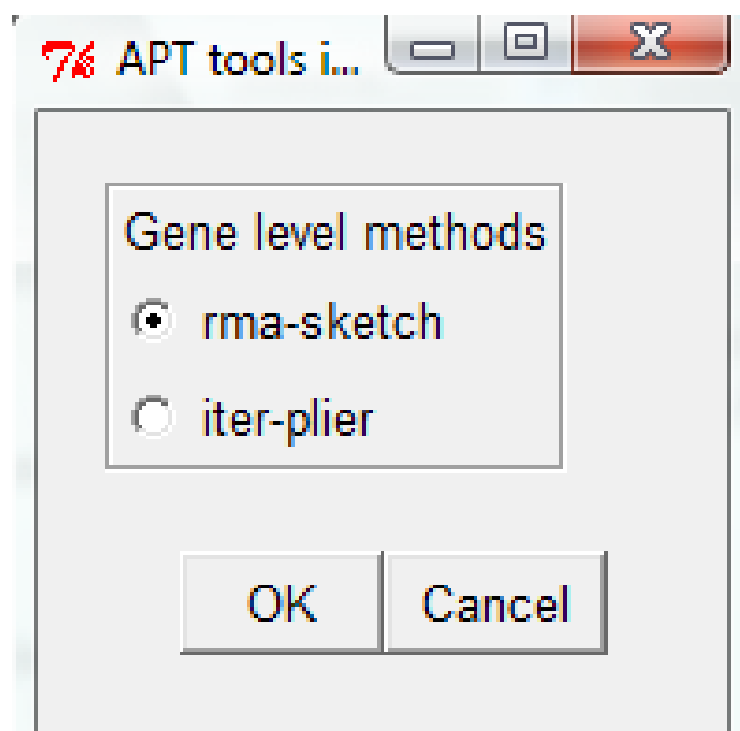


Figure 9: Sub menu to define the type of probe set summary to be applied.

popup menu will allow the data modification. Furthermore, if BeadStudio data were not normalized, user could apply various normalization procedures available in the Menu Probe set summary. It is possible to load also data from beadstudio v3. Files should be tab delimited and contain at least Average signal intensity for PROBE LEVEL this is needed to use the annotation library available in Bioconductor. Here are described the steps to generate an export file from BeadStudio v3 compatible with onechannelGUI, figs. 10, 11, 12.

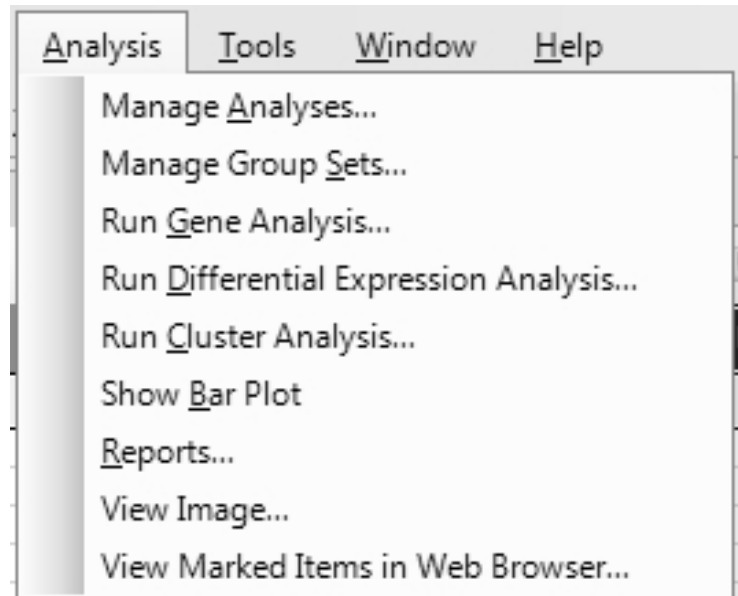


Figure 10: After having generated an analysis in BeadStudio v3 enter in the analysis menu and select reports.

*An example of genome studio exported dataset before*

*probeid\_report\_from\_genomestudio.txt*

*and after*

*probeid\_report\_from\_genomestudio\_ready\_foroneChanneGUI.txt*

*the editing is available at*

*[http://sourceforge.net/projects/ochguiextras/files/illumina\\_array\\_reformat\\_examples/](http://sourceforge.net/projects/ochguiextras/files/illumina_array_reformat_examples/)*

In general tools menu us present the function GenomeStudio reformat that reformat the genome studio output in a format suitable for oneChannelGUI.

#### 4.1.6 Loading GEO Matrix Series files

This sub menu, fig. 3, allows to load GEO Matrix Series files. To load a GEO Matrix Series file it is only necessary to locate in a specific folder a target file and the Matrix Series file downloaded from GEO database.

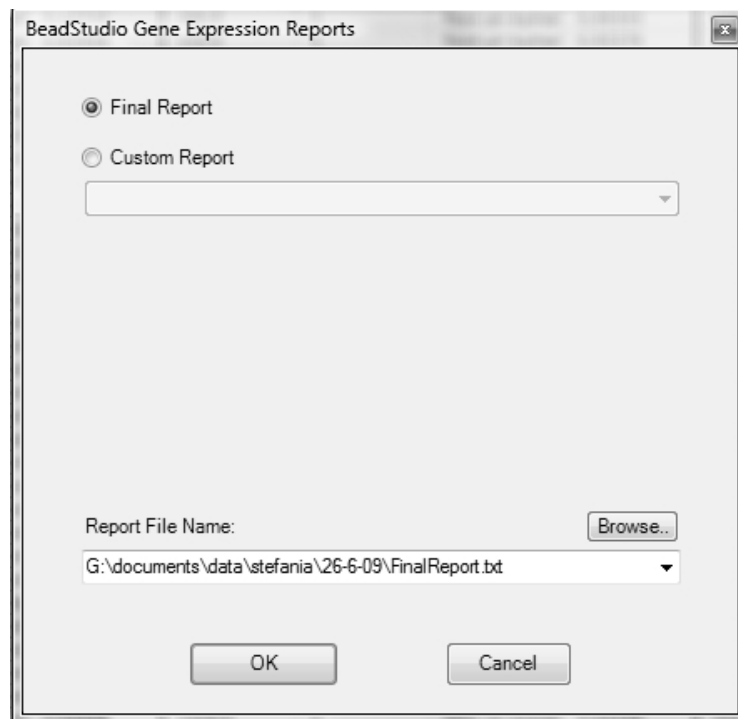


Figure 11: select the name of the final report and press OK.

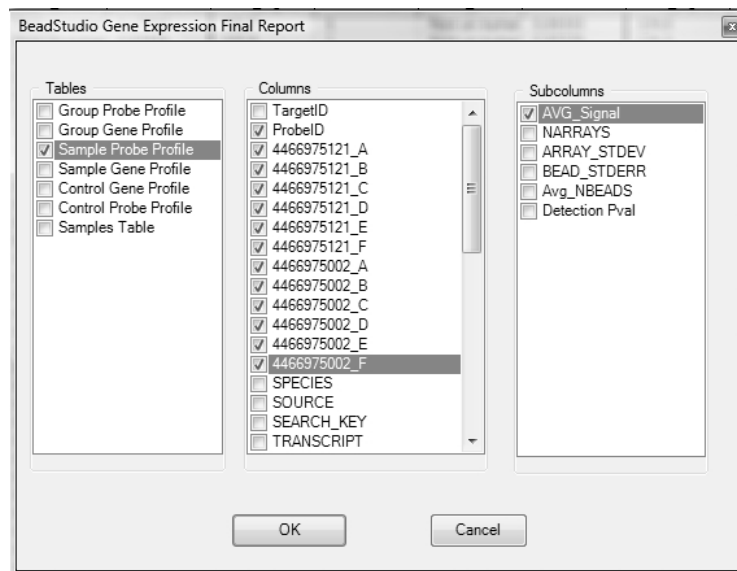


Figure 12: Select sample probe profile, select ProbeID, the array slots you want to export and AVG \_signal column and all annotation data. Subsequently edit the file with excel and exchange the column ProbeID with the column PROBE\_ID present in the annotation section. Delete all columns unless PROBE\_ID and expression data

*NB: In the target file the FileName column must contain exactly the same names present in the header below the row !series\_matrix\_table\_begin in the Matrix Series file. Instead Target column could be derived by the row !Sample\_description in the Matrix Series file.*

#### 4.1.7 Creating a Target file from GEO matrix series file

To make easier to user the creation of target file for GEO matrix series files. This function, fig. 2 opens the GEO matrix file of interest and creates a data frame with the following columns, Name, FileName, using the informations written in GEO file:

```
Name: !Sample_title
FileName: ID_REF
Target: !Sample_source_name_ch1
```

The data frame is the written in the working directory This target can be further edited and used to load the GEO matrix series file in oneChannelGUI.

*N.B. Editing of the target file is frequently neded to correctly organize the Target column,*

to fullfil the user analysis needs.

The Target file could contain a subset of the array data present in the series matrix. The GEO matrix series file present in the Target file.

#### 4.1.8 Combining GEO matrix series file

In large GEO experiments, e.g. GSE2109, the experiment is splitted in mutiple Matrix Series Files. The function *Combining GEO matrix series file* ,2, allows to combine the splitted Matrix series Files in a unique ExpressionSet to be used in oneChannelGUI. The user need to prepare a target file for each of the pieces of the experiment to be combined. The function will ask the user the number of GEO matrix series files to be combined and subsequently for each of them will ask for the Target file name and for the corresponding GEO matrix series file to be loaded.

#### 4.1.9 Loading Tab delimited files

This sub menu, fig. 3, allows to load tab delimited file containing expression data only. Also in this case the target and the expression file are the only two files needed to load these data in oneChannelGUI. In the target file the FileName column should contain exactly the same names present in the header of the tab delimited matrix file. Example of targets are available at <http://www.bioinformatica.unito.it/bioinformatics/DAGEL.II/>. Actually a specialized module to load *processed-data* derived from Array-Express database <http://www.ebi.ac.uk/arrayexpress/> is not available. However, *processed-data*, reorganized in a flat tab delimited file containing only expression values, can be loaded on oneChannelGUI.

### 4.2 Open, Save, Save as

A project can be saved using the functions *Save as* fig. 2. A microarray project can also be uploaded again in oneChannelGUI with the function *open*.

### 4.3 Exporting normalized expression values

This function, fig. 2, allows to export, as tab delimited files expression data, loaded in oneChannelGUI. This function is also located in *filtering menu* and in the *exon menu*. If exon arrays are loaded in oneChannelGUI it is possible to extract not only the gene level expression data available but also exon level expression data and any other data generated during exon array analysis: Splice Index, MiDAS p-values, RP alternative splicing data.

## 4.4 Info about the loaded data set

This function, fig. 2, gives information about the set of data loaded in oneChannelGUI and on the corresponding annotation library, if available.

## 4.5 Attaching annotation lib info

If a Bioconductor library is available this is attached to the data loaded in oneChannelGUI and it will appear in the output of *Info about the loaded data set*. Using *Attaching annotation lib info* function, after loading expression data as a tab delimited file, it is possible to attach the Bioconductor annotation library associated to it. In case gene-level Affymetrix Whole transcriptome data are loaded as tab delimited file HuEx for human, MoEx for mouse and RaEx for rat annotation need to be attached. This is needed to allow the attachment of annotation information to the data set.

### 4.5.1 Probe set annotation

The Bioconductor annotation library for IVT Affymetrix arrays or GEO Matrix Series file are directly attached. Concerning Exon 1.0 ST arrays, gene level annotation information are actually embedded in oneChannelGUI, and a stand alone function is provided to use gene-level annotation externally to oneChannelGUI. For exon arrays annotation is available at the gene level for the core subset of Hs/Mm/Rn. Info about the available Affymetrix annotation release can be found in the main R window as part of the oneChannelGUI release major changes. For EXON 1.0 ST arrays, it is possible to link GeneBank accession numbers and EG to the gene-level probe sets using the function *Attaching ACC to Probe set IDs*, present in the Biological Interpretation menu. This function also allows to link EGs to gene-level probe sets of a tab delimited file that has in the first column the probe set ids. It is also possible to extract exon-level Probe Selection Region sequences associated to a specific gene-level probeset using the function *Extracting exon-level PSR sequences associated to one gene-level probeset* in the Biological Interpretation menu.

## 5 RNA target

The first item in the menu, fig. 13, is inherited from affylmGUI and allows the visualization of the experimental structure described by the target file used to load the expression data.

The second item, fig. 13, *maSigPro create/view edesign* reorganizes the target file to extract all the information needed to analyse a time course experiment using maSigPro. For time course experiments a specific target file is needed, fig. 14.

Each row of the column named Target, in the target file, describes the array on the basis of the experimental design. Each element needed for the construction of design fro

RNA Targets	QC	Study design	Probe set summary	Filter
affyImGUI:View RNA Targets				
oneChannelGUI: maSigPro: Create/View edesign				
oneChannelGUI: Create/View classification parameters				

Figure 13: RNA target menu.

## Time Course design for maSigPro

	A	B	C
<p>The targets file for maSigPro has a peculiar structure: Each row of the column named Target describes the array on the basis of the experimental design.</p> <p>Each element describing the time course experiment is separated from the others by an underscore.</p> <p>The first three elements of the row are fixed and represent <b>Time</b>, <b>Replicate</b>, <b>Control</b>, all the other elements refer to various experimental conditions.</p> <p>In this case we have a 8, 24 48 h time course, in triplicates with two different treatments: <b>cond1</b> and <b>cond2</b></p>	Name	FileName	Target
	exp1.01	1539121008.A.CEL	8_1_1_0_0
	exp2.01	1539121006.A.CEL	8_1_1_0_0
	exp3.01	1539121005.A.CEL	8_1_1_0_0
	exp1.03	1539121008.C.CEL	24_2_1_0_0
	exp2.03	1539121006.C.CEL	24_2_1_0_0
	exp3.03	1539121005.C.CEL	24_2_1_0_0
	exp1.05	1539121008.E.CEL	48_3_1_0_0
	exp2.05	1539121006.E.CEL	48_3_1_0_0
	exp3.05	1539121005.E.CEL	48_3_1_0_0
	exp1.07	1539121020.A.CEL	8_4_0_1_0
	exp2.07	1539121009.A.CEL	8_4_0_1_0
	exp3.07	1539121021.A.CEL	8_4_0_1_0
	exp1.09	1539121020.C.CEL	24_5_0_1_0
	exp2.09	1539121009.C.CEL	24_5_0_1_0
	exp3.09	1539121021.C.CEL	24_5_0_1_0
	exp1.11	1539121020.E.CEL	48_6_0_1_0
	exp2.11	1539121009.E.CEL	48_6_0_1_0
	exp3.11	1539121021.E.CEL	48_6_0_1_0
	exp1.02	1539121008.B.CEL	8_7_0_0_1

Figure 14: Target file for time course analysis.

time course is separated from the others by an underscore. The first three elements of the row are fixed and represent `Time Replicate Control`, all separated by an underscore:

*Time\_Replicate\_Control*

All the other elements refer to various experimental conditions.

Considering two different conditions to be evaluated each row is made of 5 elements:

*Time\_Replicate\_Control\_cond1\_cond2 all separated by an underscore.*

Having an experiment made of 9 arrays, with two time points, 0h and 24h, in triplicate, and two different experimental conditions to be evaluated, the target file will look like:

<i>Name</i>	<i>FileName</i>	<i>Target</i>
<i>mC1</i>	<i>M1.CEL</i>	<i>0_1_1_0_0</i>
<i>mC2</i>	<i>M4.CEL</i>	<i>0_1_1_0_0</i>
<i>mC3</i>	<i>M7.CEL</i>	<i>0_1_1_0_0</i>
<i>mE1</i>	<i>M3.CEL</i>	<i>24_2_0_1_0</i>
<i>mE2</i>	<i>M6.CEL</i>	<i>24_2_0_1_0</i>
<i>mE3</i>	<i>M9.CEL</i>	<i>24_2_0_1_0</i>
<i>mI1</i>	<i>M2.CEL</i>	<i>24_3_0_0_1</i>
<i>mI2</i>	<i>M5.CEL</i>	<i>24_3_0_0_1</i>
<i>mI3</i>	<i>M8.CEL</i>	<i>24_3_0_0_1</i>

The third item, fig. 13, instead refers to the reorganization of a target file containing the information related to clinical parameters to be used for classification purposes. In this case each clinical parameter is separated from the others by an underscore as in the case of the time course. The absence of a parameter **NEEDS** to be indicated in the Target file by NA. Having an experiment made of 9 arrays with 4 different experimental/clinical parameters the target file will look like:

<i>Name</i>	<i>FileName</i>	<i>Target</i>
<i>mC1</i>	<i>M1.CEL</i>	<i>0_1_pos_0_NA</i>
<i>mC2</i>	<i>M4.CEL</i>	<i>0_1_pos_0_yes</i>
<i>mC3</i>	<i>M7.CEL</i>	<i>0_1_neg_0_no</i>
<i>mE1</i>	<i>M3.CEL</i>	<i>24_2_neg_1_NA</i>
<i>mE2</i>	<i>M6.CEL</i>	<i>24_2_NA_1_yes</i>
<i>mE3</i>	<i>M9.CEL</i>	<i>24_2_neg_1_yes</i>
<i>mI1</i>	<i>M2.CEL</i>	<i>12_3_0_pos_yes</i>
<i>mI2</i>	<i>M5.CEL</i>	<i>12_3_0_pos_no</i>
<i>mI3</i>	<i>M8.CEL</i>	<i>12_3_0_pos_no</i>

Once the target file is reorganized by *create/view classification parameters* function, the user will be requested to selected an external file containing the description of the experimental/clinical parameters. In this file, the description of each parameter is separated from the others by a carriage return.



*Drug treatment time*  
*Tumor grade*  
*IHC ER*  
*Metastasis within 5 years*  
*Positive lymphonode*

This information will be used to selected a specific clinical parameter for classification analysis.

## 6 QC

The functions available in this menu are specific of the type of microarray data set loaded

### 6.1 QC for IVT arrays loaded starting from .CEL files

This menu, fig. 15, inherits all affyImGUI probe/probe set level quality controls, please refer to affyImGUI for their usage.

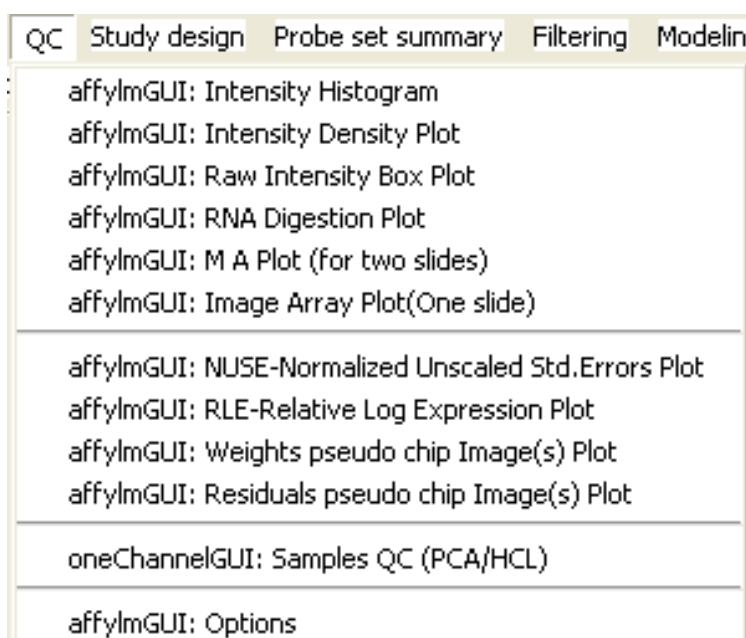


Figure 15: QC for IVT arrays.

Furthermore, after probe set summary is calculated, samples similarities can be visualized using the *Sample QC: PCA/HCL* function, producing a 2D PCA plot and a hierachical clustering of the samples, fig. 16.

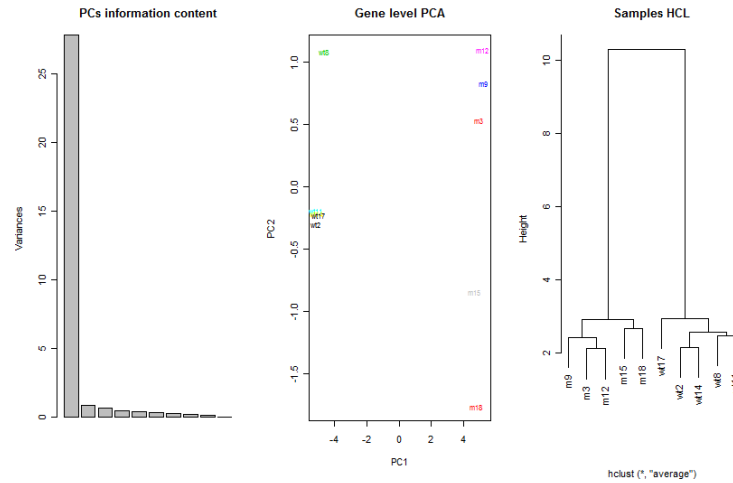


Figure 16: Sample QC: PCA/HCL output for IVT arrays. First panel on the left side contains the information content described by each of the PCs. Panel in the middle PC2 versus PC1 plot. Last panel on the right HCL.

If exon data are loaded the function *Gene/Exon PCA/HCL* results could be visualized both at gene or exon level. Furthermore, the function *Gene/Exon Intensity Histogram* will show the density plot of the normalized intensities both at gene and at exon level.

## 6.2 QC for GEO/flat tab delimited files

Ones probe set expression data derived by GEO Matrix Series file or an expression tab delimited file *Sample QC: PCA/HCL* function is available as QC. There is also the function *Box plot of normalized data* which show the array distribution as box plot 17.

## 6.3 QC for exon arrays

In the case of exon array the QC menu is slightly different, as shown in fig. 18

Two functions are available:

**Sample QC: PCA/HCL** This function will produce a PCA/HCL for both gene/exon level data.

**Gene/Exon intensity histogram** This function will produce a density histogram for gene or exon expression levels.

**Controls raw intensity histogram** This function will produce a box plot for exon, positive controls, and introns, negative controls, for housekeeping genes. Probe level data are directly extracted from CEL files using APT tools.

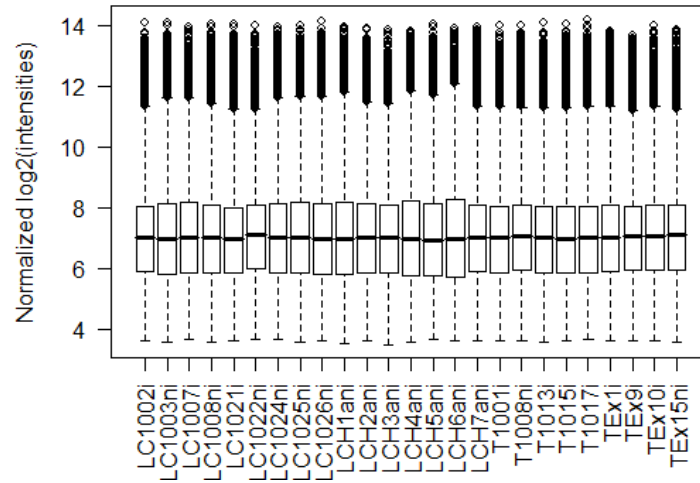


Figure 17: Box plot of normalized data.

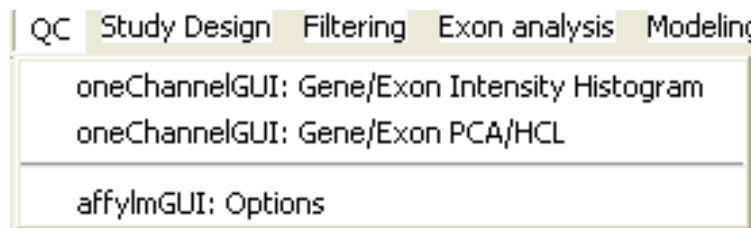


Figure 18: QC menu for exon arrays.

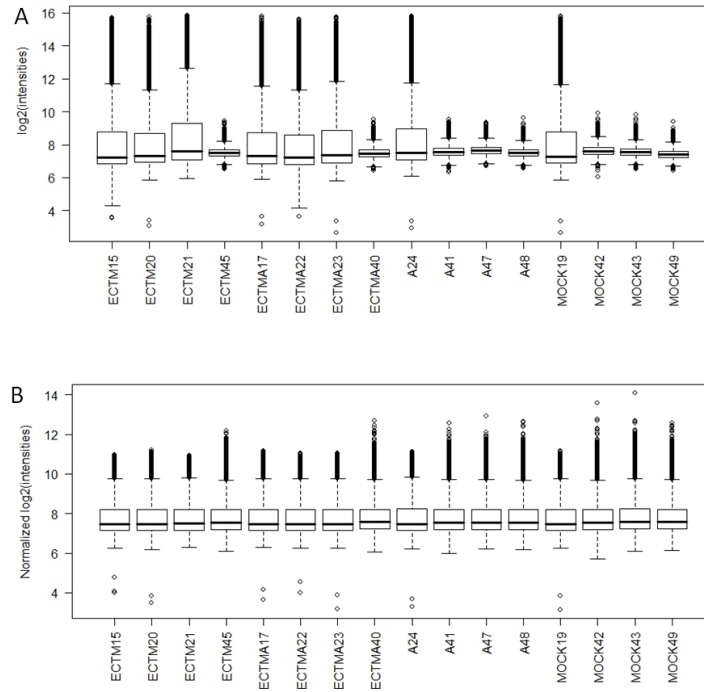


Figure 19: A set of Illumina arrays before and after data normalization.

It useful, as quality control, to check intensities before normalization.

As it can be seen in fig. 19 normalization masks the fact that a sub set of arrays, i.e. those with a very narrow boxplot 19A, had something wrong in hybridization. This problem is completely masked in the normalized data 19B. For this reason *Controls raw intensity histogram* was written for exon array data since probe sets data are directly uploaded as normalized in oneChannelGUI, via APT tools. This function produce a box plot for exon, positive controls, and introns, negative controls, for housekeeping genes. This box plot gives an idea of signals both at high and low intensity range.

## 7 Study design

This menu allows to investigate the statistical quality of a microarray study, fig. 20.

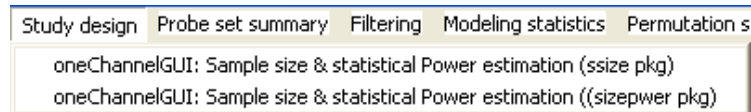


Figure 20: Study design menu.

This menu gives access to two functions, which are graphical implementations of the sizepower Bioconductor library. These functions allow user to determine how many samples are needed to achieve a specified power for a test of whether a gene is differentially expressed or, in reverse, to determine the power of a given sample size.

## 8 Probe set summary

This menu inherits the affylnGUI probe set summary methods for IVT arrays. Furthermore, the *expresso* function, which allows the integration of different methods for background correction, normalization, probe specific correction, and summary value computation, is added. This menu is also available for GEO and tab delimited expression data files and it allows to perform the following normalization procedures if a data set without normalization is loaded:

1. Cyclic LOESS.
2. QUANTILE.
3. QSPLINE.

## 9 Filtering

A central problem in microarray data analysis is the high dimensionality of gene expression space, which prohibits a comprehensive statistical analysis without focusing on particular aspects of the joint distribution of the gene expression levels. Possible strategies are to perform data-driven nonspecific filtering of genes (von Heydebreck, 2004) before the actual statistical analysis or to filter, making use of biologically relevant a priori knowledge. This menu allows user to apply a variety of filtering procedures, fig. 21

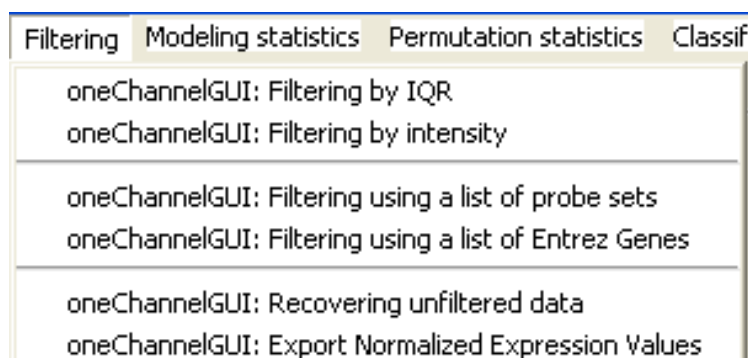


Figure 21: Filtering menu for GEO/Affy IVT arrays.

## 9.1 Filtering by IQR

The IQR filter will select only those probe sets characterized by a relative large signal distribution. The way the IQR filter is shown in fig. 22

### How filtering by IQR works?

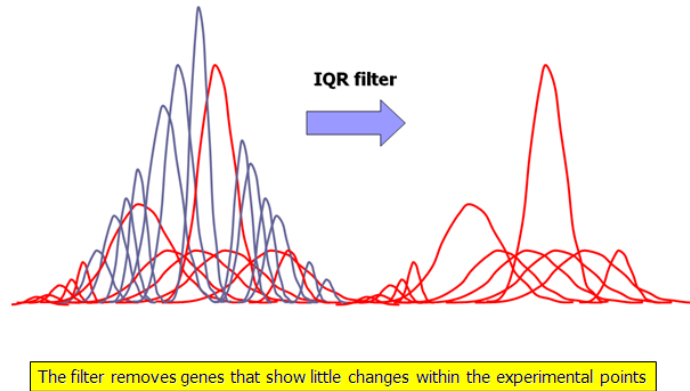


Figure 22: IQR filtering: The distributions of the various probe sets belonging to a data set are shown in red, if they are wide and they are retained by the filter, and in blue, if they are narrow and they are discarded by the filter.

In oneChannelGUI it is possible to select three filtering values:

1. IQR 0.1, weak filter, i.e. only the extreme unchanging probe sets are removed.
2. IQR 0.25, intermediate filter.
3. IQR 0.5, strong filter, i.e. the majority of the unchanged probe sets are removed.

This filtering procedure can be applied to any kind of loaded arrays. However, it seems not to be very effective when it is used to gene level expression data calculated with iterPlier.

## 9.2 Filtering by intensity

For IVT/GEO/tab delimited expression data files it is also possible to apply a filtering procedure based on intensity signals, the graphical interface to do it is shown in fig. 23.

This filtering approach is quite useful to remove probe sets having very low intensity values.

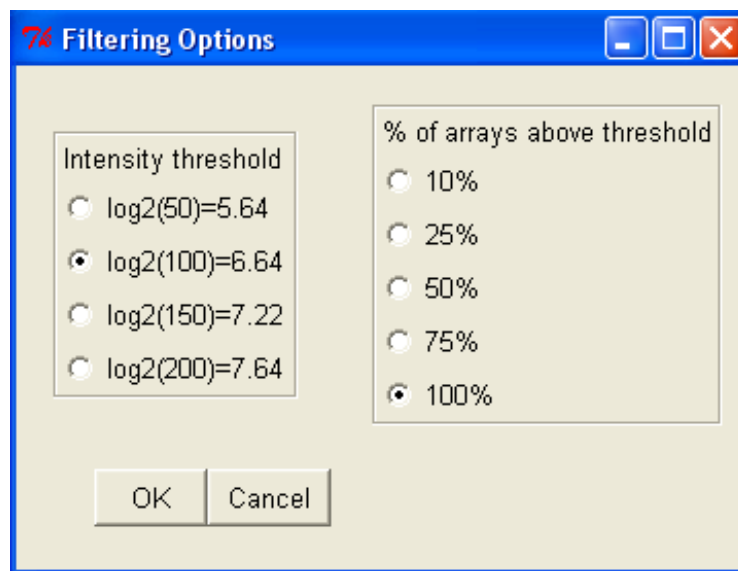


Figure 23: Intensity filtering: This filter will retain a probe set only if a certain fraction of the samples are characterized by an intensity value over a certain user defined threshold.

### 9.3 Filtering by list of probe sets/EG ids

It is also possible to filter expression data using a text file containing a list of probe set ids separated by carriage return. If the data set is associated to a Bioconductor annotation library the filtering procedure can be also done using a text file containing a list of Entrez gene identifiers separated by carriage return.

### 9.4 Recovering unfiltered data

It is possible to recover the data before the last filtering using the *Recovering unfiltered data* function.

## 10 Modelling statistics

This menu allows to perform limma differential expression analysis as well as time course analysis using the maSigPro package, fig. 24.

### 10.1 limma

The affyGUI interface to limma is fully inherited, see limma and affyGUI vignettes for usage. The function *raw p-value distribution* is implemented to evaluate if the BH/BY

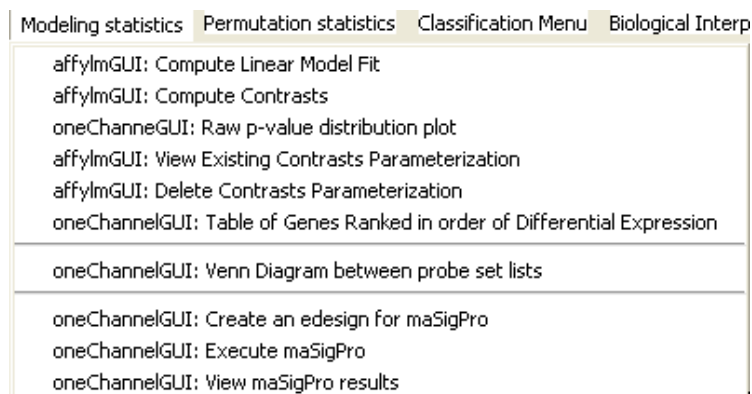


Figure 24: Modelling statistics

type I error correction methods could be used. To apply BH correction two conditions should be satisfied:

1. The gene expressions are independent from each other.
2. The raw distribution of p-values should be uniform in the non significant range.

Instead if BY correction is used it is sufficient only the second one, fig. 25.

The `affyGUI` function *Table of genes ranked in order of differential expression* is a modified version of the original found in `affyGUI` to allow users to check with MA/Volcano plots the set of differentially expressed probe sets before saving the table, fig. 26.

The `limma` analysis, implemented in `affyGUI`, does not allow the inclusion in the model of batch effect. A new function *Two groups Linear Model with batch effect*, in the Modelling menu, integrates in linear model batch effect. This functionality is only available for two sample groups. For this analysis the different origins of the samples are inserted in the Target column of the target file as shown below in the Rank Product section.

## 10.2 Venn diagrams between probe set list

This function is modified with respect to the original one presents in `affyGUI` to allow Venn diagrams using lists of probe sets, saved in text files where each id is separated by the others by carriage return, derived by any of the available statistical methods implemented in `oneChannelGUI`. Furthermore, if a Bioconductor annotation library is linked to the loaded data set, Venn diagrams can be generated using the Entrez Gene ids associated to the probe sets, removing probe sets redundancy.



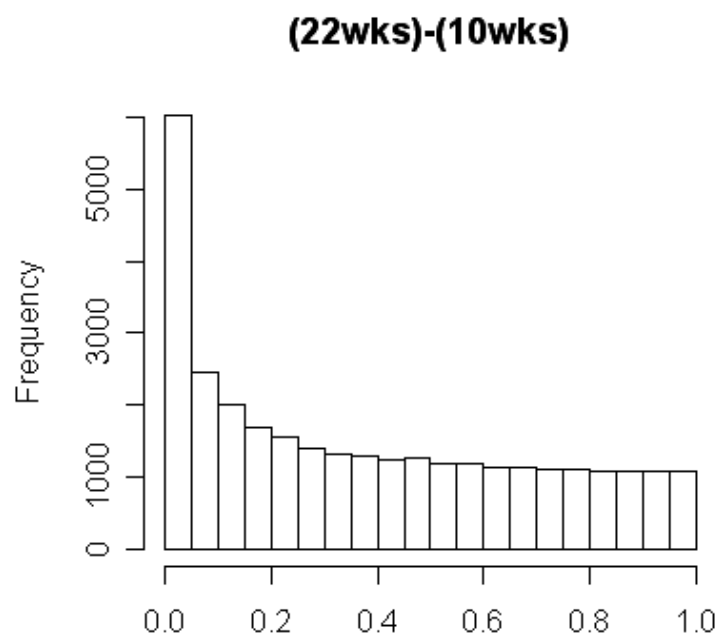


Figure 25: Output of *raw p-value distribution*: The raw distribution of p-values is uniform in the non significant range.

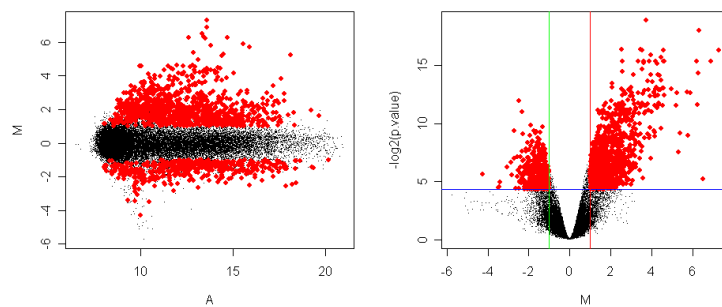


Figure 26: MA and Volcano plots for differentially expressed probe sets, red dots, detected by limma method.

## 10.3 Time course analysis

Time course analysis can be performed on oneChannelGUI using maSigPro package, fig. 24.

maSigPro is a R package for the analysis of single and multiseriess time course microarray experiments. maSigPro follows a two steps regression strategy to find genes with significant temporal expression changes and significant differences between experimental groups. To handle the batch effect the target file should be organized as the target file used in the RankProduct section.

The first step, to run maSigPro analysis, is to reorganize the target file using the function *create an edesign for maSigPro*, see also target file paragraph for time course experiment requirements. Using the function *Execute maSigPro* user will select the parameters needed for maSigPro, fig. 27.

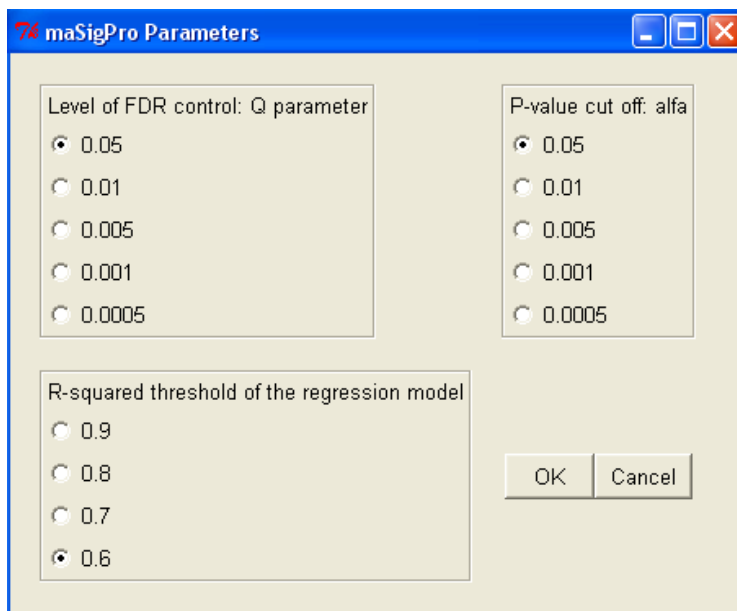


Figure 27: maSigPro parameters setup.

### 10.3.1 Lever of FDR control: Q parameter

The first step is to compute a regression fit for each gene. The p-value associated to the F-Statistic of the model are computed and they are subsequently used to select significant genes. maSigPro corrects this p-value for multiple comparisons by applying false discovery rate (FDR) procedures. The level of FDR control is given by the function parameter Q, fig. 27.

### 10.3.2 P-value cut off: $\alpha$

maSigPro applies, as second step, a variable selection procedure to find significant variables for each gene. This will ultimately be used to find which are the profile differences between experimental groups. At each regression step the p-value of each variable is computed and variables get in/out the model when this p-value is lower or higher than the given cut-off value  $\alpha$ , fig. 27.

### 10.3.3 R-squared threshold of the regression model

The last step in maSigPro analysis is to generate a lists of significant genes. As filtering maSigPro uses the R-squared of the regression model, fig. 27.

maSigPro calculation steps can be followed on the main R window. The end of the maSigPro analysis will be given by a popup message, fig. 28.

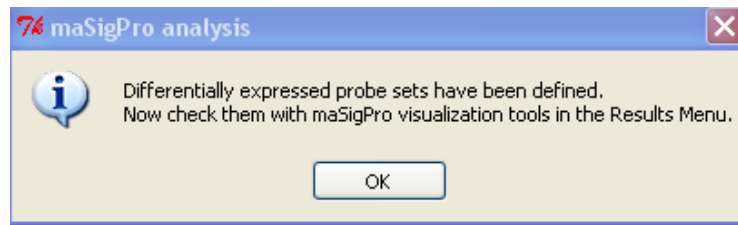


Figure 28: End of maSigPro calculation.

N.B: Multiple test problem is also present in maSigPro analysis. Therefore, before running maSigPro, remember to perform some filter based on functional information or samples distribution.

### 10.3.4 View maSigPro results

The coefficients obtained in the second regression model will be useful to cluster together significant genes with similar expression patterns and to visualize results. Various visualization options are available:

1. Venn diagrams, fig. 29 .
2. Expression profiles saved in a pdf file, figs. 30, 31.
3. Tab delimited files with the probe sets found differentially expressed in each of the experimental conditions.

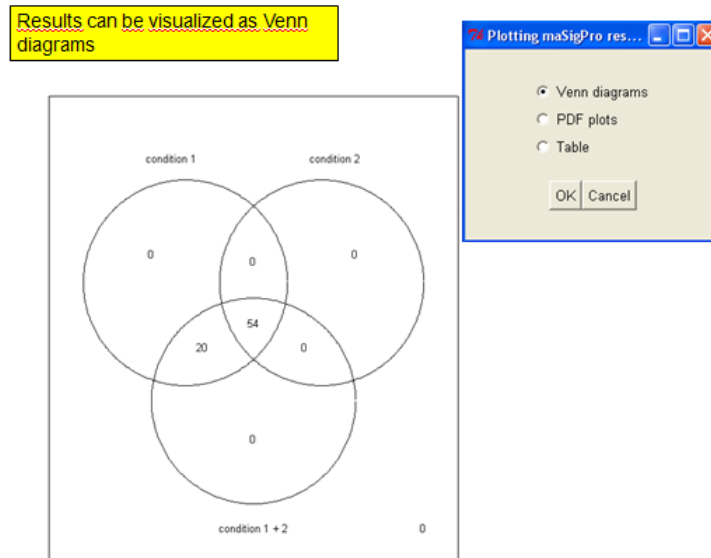


Figure 29: maSigPro Venn diagrams output.

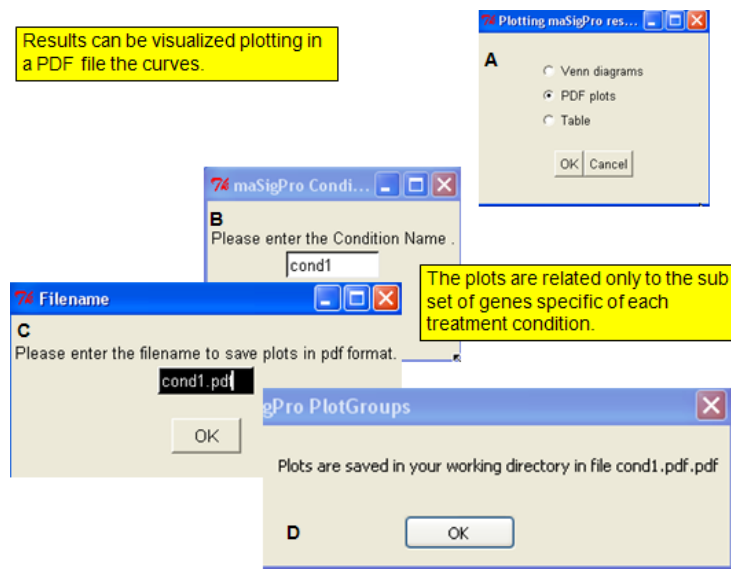


Figure 30: Selecting the experimental condition to be used to profiles plotting.

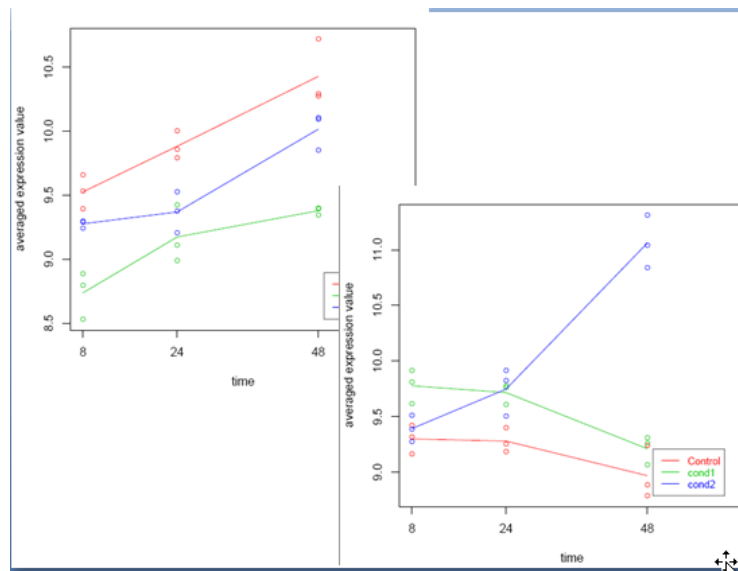


Figure 31: An example of profiles plotting.

## 11 Permutation statistics

The permutation statistics menu, fig. 32, allows to run two class unpaired SAM analysis implemented in the siggenes package and two class samples analysis using the rank product method implemented in RankProd package.

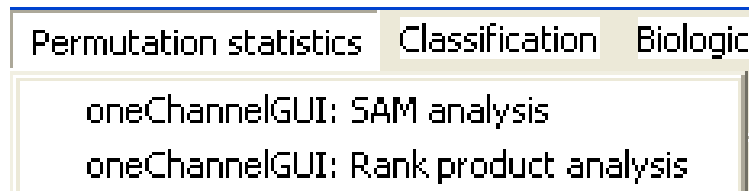


Figure 32: Permutation statistics menu.

### 11.1 SAM analysis

The module recognizes if a two class unpaired analysis can be performed. Subsequently, a table with DELTA values and FDRs will be shown to the user. Furthermore, user need to select a delta threshold to continue the analysis, fig. 33.

Siggenes output for differentially expressed genes, given the selected DELTA value, will be shown in the main R window, fig. 34, together with a absolute  $\log_2(\text{FC})$  selection module, fig. 34.

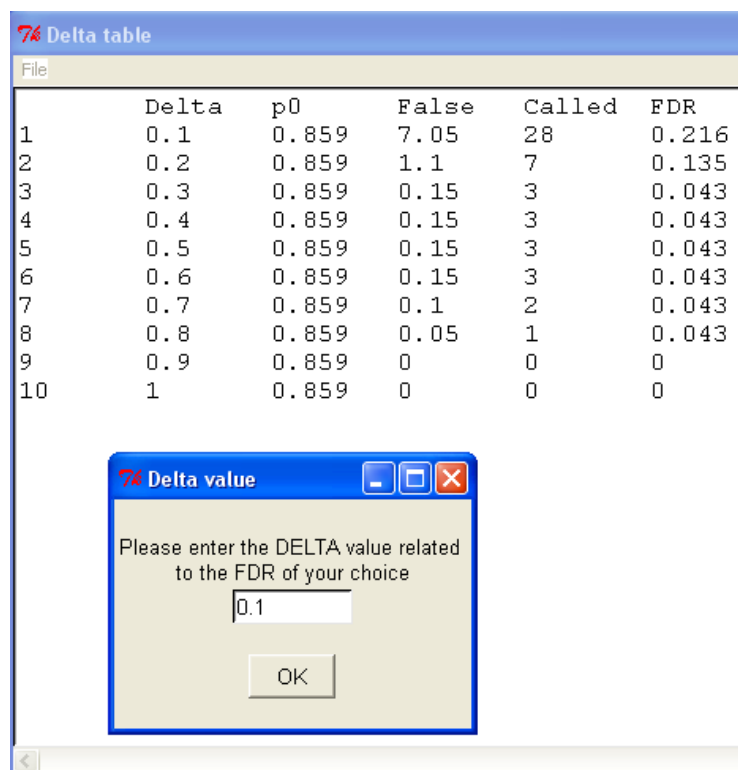


Figure 33: DELTA table and DELTA value selection module.

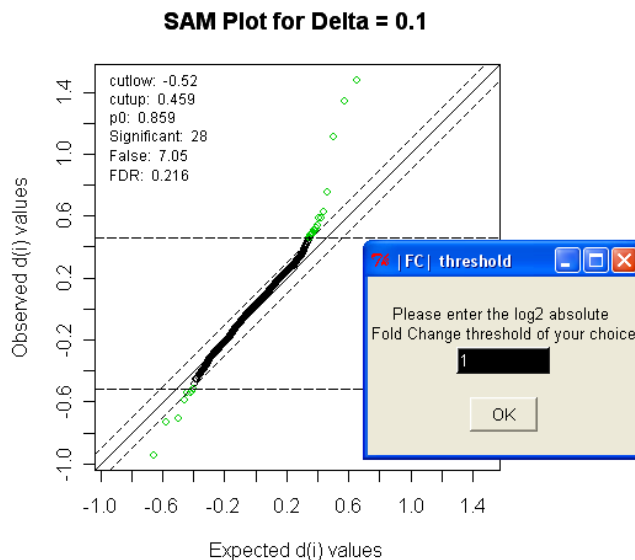


Figure 34: SAM results given at specific user defined DELTA value and the absolute  $\log_2(fc)$  selection mask.

The fold change filters allows the selection, within the SAM significant probe sets, of those greater than a user defined threshold. Subsequently, the differentially expressed genes will be shown, fig. 35, and the user will decide if they should be saved.

10 genes were found differentially expressed using a delta= 0.1 and a |FC| threshold= 1

AffyID	EG	Symbol	d.value	stdev	rawp	q.value	log2.R.fold
238733_at	1368	CPM	1.5	0.4	1.6e-05	0.086	2.7
211834_s_at	8626	TP73L	1.3	0.15	3.3e-05	0.086	2.1
221577_x_at	9518	GDF15	1.1	0.18	4.9e-05	0.086	1.7
1565483_at	1956	EGFR	-0.94	0.36	0.00011	0.15	-1.7
202284_s_at	1026	CDKN1A	0.75	0.084	0.00015	0.15	1.1
1565484_x_at	1956	EGFR	-0.73	0.44	2e-04	0.16	-1.4
1552701_a_at	114769	COP1	-0.71	0.44	0.00021	0.16	-1.4
1554400_at	6991	TCTE3	-0.59	0.27	0.00041	0.21	-1
1555786_s_at	NA	NA	-0.54	0.36	0.00054	0.21	-1
228697_at	135114	HINT3	0.54	1	0.00057	0.21	2

Figure 35: Differentially expressed probe sets to be saved.

## 11.2 Rank product analysis

The RankProd module is a graphical interface to the RankProd package functions for the analysis of gene expression microarray data. RankProduct package allows the identification of differentially expressed genes using the so called rank product non-parametric

method (Breitling et al., 2004, FEBS Letters 573:83) to identify up-regulated or down-regulated genes under one condition against another condition, e.g. two different treatments, two different tissue types, etc. The user needs only to define the pfp (percentage of false prediction) threshold and the number of permutations to be applied, fig. 36.

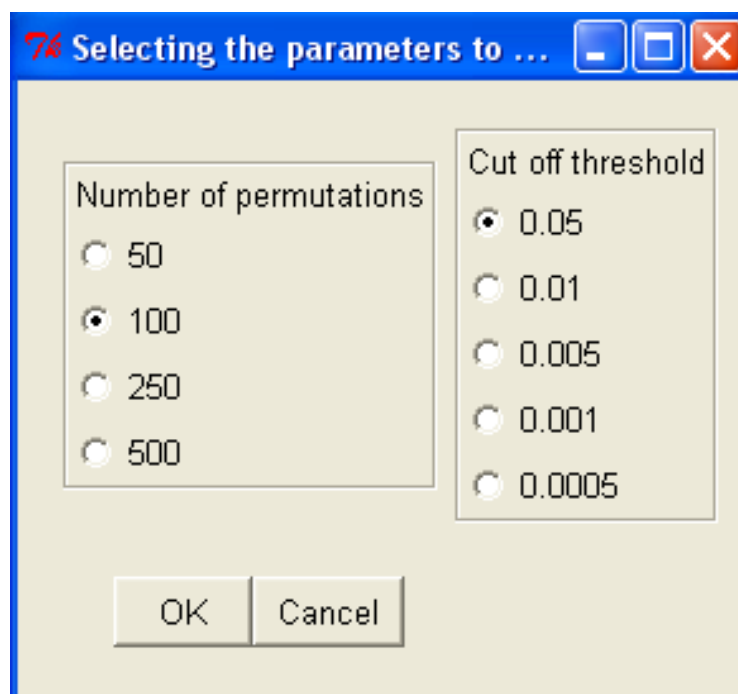


Figure 36: RankProd selection parameters mask.

At the end of the analysis the user will decide if he would like to save the differentially expressed probe sets in a tab delimited file. If a Bioconductor annotation library is available Entrez Gene identifier and Symbols will be added to the saved output.

### 11.2.1 Target structure

In a rank product analysis for data sets from different origin the structure of the Target column of the target file can contain also an integer describing the data origin.

<i>Name</i>	<i>FileName</i>	<i>Target</i>
<i>mC1</i>	<i>M1.CEL</i>	<i>0_1</i>
<i>mC2</i>	<i>M4.CEL</i>	<i>0_1</i>
<i>mC3</i>	<i>M7.CEL</i>	<i>0_1</i>
<i>mE1</i>	<i>M3.CEL</i>	<i>0_2</i>
<i>mE2</i>	<i>M6.CEL</i>	<i>0_2</i>
<i>mE3</i>	<i>M9.CEL</i>	<i>1_1</i>



<i>mI1</i>	<i>M2.CEL</i>	<i>1_1</i>
<i>mI2</i>	<i>M5.CEL</i>	<i>1_2</i>
<i>mI3</i>	<i>M8.CEL</i>	<i>1_2</i>

The oneChannelGUI module will select the RankProd method on the basis of the Target structure.

## 12 Classification

This module, fig. 37, provides a link to the pamr and pdmclass packages designed to carry out sample classification from gene expression data, respectively by the method of nearest shrunken centroids (Tibshirani, et al., 2002) and by penalized discriminant methods.

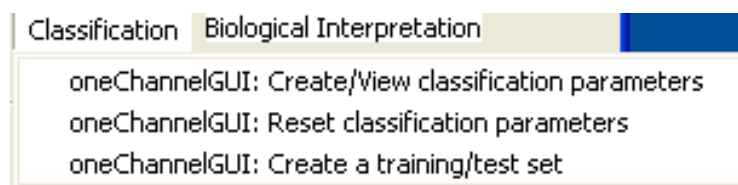


Figure 37: Classification menu.

### 12.0.2 Create/view/reset classification parameters

The *Create/view classification parameters* function reorganizes the Target columns separating the experimental/clinical parameters. The *Reset classification parameters* function deletes the Targets reorganization and the association to the file containing the names of the parameters present in the Target column of the target file.

### 12.0.3 Create a training/test set

The first step of this module is the definition of the covariate to be used for the classification analysis. The user will be requested to select, from a table, listing the names clinical parameters, i.e. phenoData covariate names, one of them indicating its row number, fig. 38. Subsequently, the user could decide to divide the data set in a training (2/3) and a test (1/3) set or use the full data set as training set. All arrays, which are not linked to any of clinical/experimental params, i.e. those marked as NA, will be discarded from the following analyses.

The *Create a training/test set* function then allows the access to PAMR/PDMCLASS classification tools and to a PCA visualization module, fig. 39.

It is also possible to evaluate how the classification performance can be associated to a single subset of probesets or if it is a general characteristic of the data set. Ideally

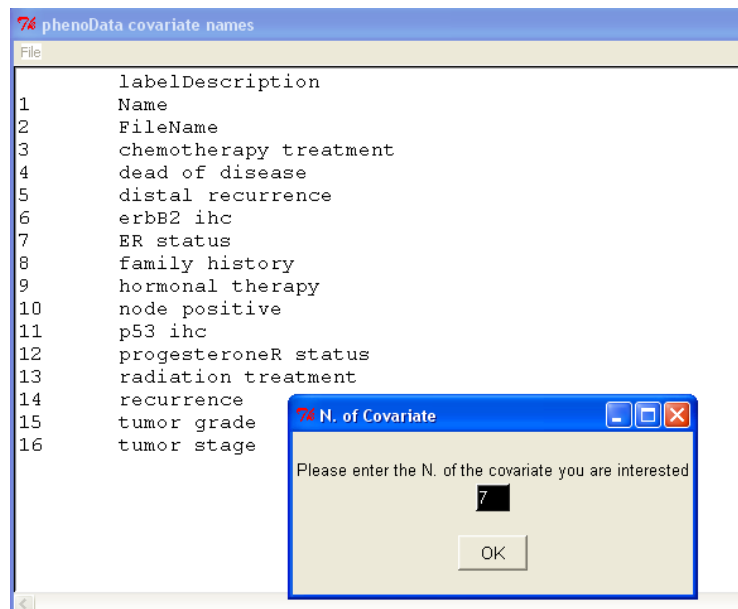


Figure 38: Selecting the classification parameter.

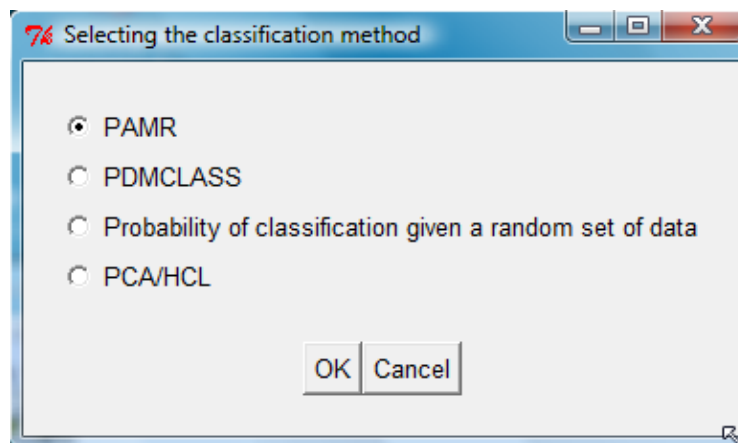


Figure 39: Classification methods selection mask.

we expect that only a small subset of probe sets should be able to discriminate between groups, if the full data set is able to discriminate in a way independent by the subset of probe sets considered for the classification it might be an indication of some bias that could not be necessarily associated to the biological event under investigation, eg. it could be due to some experimental bias. This functionality is provided if the *Probability of classification given a random set of data* option is selected, see fig. 39 Using this function user can select the size of the dandom set to be used for classification, e.g. 100 probesets, and the number of samples retrived form the total data set used to perform this analysis, eg. 1000. Subsequently the user will select a classification method, e.g. PAMR, and the function will calculate the frequence of correct classification of the random set size defined by the user. This functin is particularly useful if user is interested to evaluate the specificity of a specific transcription signature.

#### 12.0.4 PAMR

If PAMR method is selected, 2-3 steps are performed and pop-up info messages allow to check the resulting plots. Initially the cross-validated misclassification error curves are calculated, fig. 40, and shown in the main R window. Then, user defines a shrinking

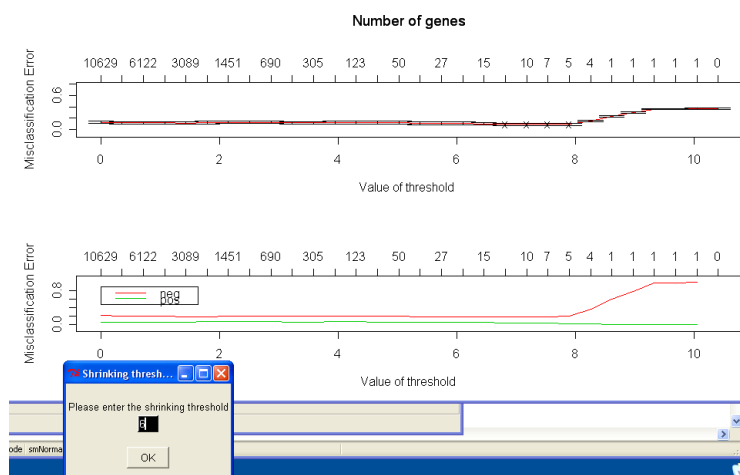


Figure 40: Cross-validated misclassification error curves

threshold and if the number of selected probe sets is below 50 the centroids will be plotted, fig. 41.

Subsequently the classification performance of the selected sub group of probe sets will be shown as plot and as text in the R window , fig. 42.

Results are also available as numerica values in the R window:

	<i>neg</i>	<i>pos</i>	<i>Class</i>	<i>Error rate</i>
<i>neg</i>	23	5		0.1785714
<i>pos</i>	2	48		0.0400000

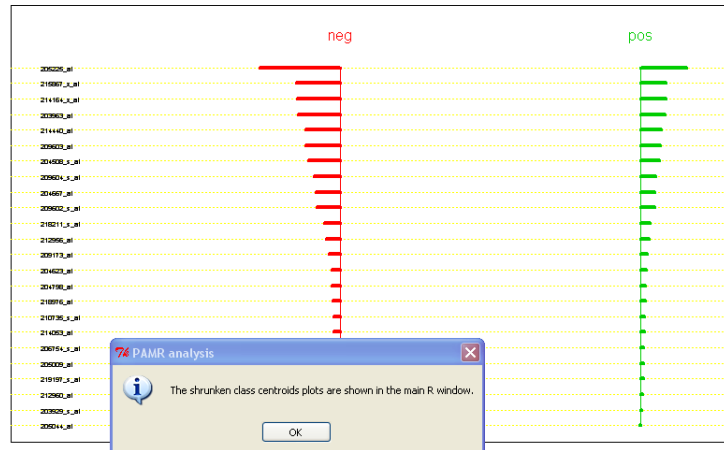


Figure 41: Shrunken class centroids.

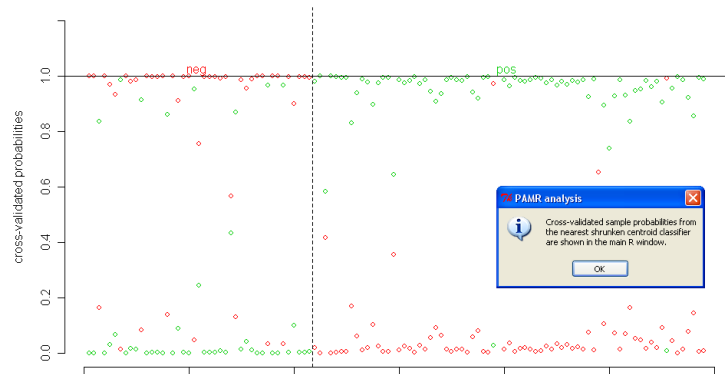


Figure 42: Cross-validated sample probabilities.

If the results are satisfying user can save the probe sets defined by this analysis, fig. 43.

74 List of genes that survive the thresholding			
	id	neg-score	pos-score
1	205225_at	-0.5366	0.3077
2	215867_x_at	-0.2947	0.169
3	214164_x_at	-0.2878	0.165
4	203963_at	-0.2822	0.1618
5	214440_at	-0.2335	0.1339
6	209603_at	-0.2302	0.132
7	204508_s_at	-0.2119	0.1215
8	209604_s_at	-0.176	0.1009
9	204667_at	-0.1628	0.0933
10	209602_s_at	-0.1607	0.0921
11	218211_s_at	-0.1075	0.0616
12	212956_at	-0.0968	0.0555
13	209173_at	-0.0799	0.0458
14	204623_at	-0.0595	0.0341
15	204798_at	-0.0567	0.0325
16	218976_at	-0.0488	0.028
17	210735_s at	-0.047	0.027

Figure 43: Probe sets to be use as classifier.

Furthermore, if the test set was created it will be possible to check the ability of the selected sub set of genes to separate the classes under analysis using a hierarchical clustering, fig. 44.

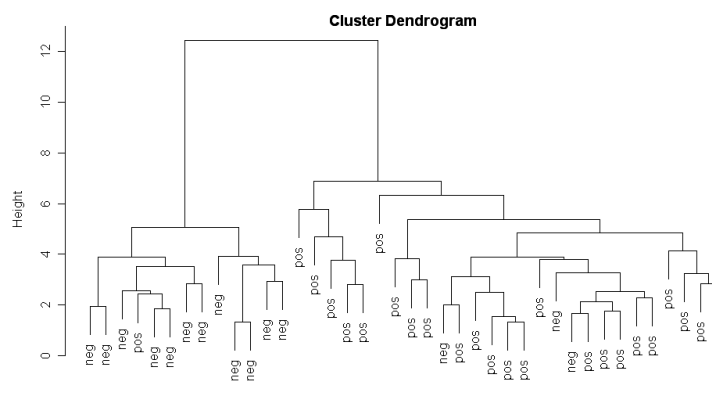


Figure 44: Testing the efficacy of the classifier on the test set by HCL.

## 12.0.5 PCA

The PCA visualization method offers the possibility to see how the data set can be grouped on the basis of the used clinical/experimental parameter under analysis, fig. 45.

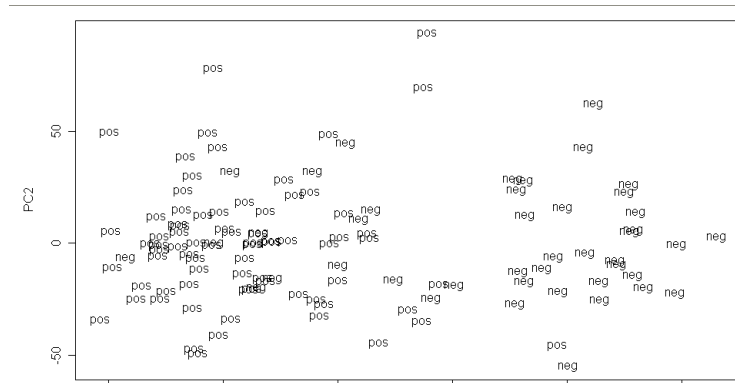


Figure 45: 1st and 2nd principal components space.

### 12.0.6 PDMCLASS

The PDMCLASS module allows the selection of different type of classification procedures, fig. 46.

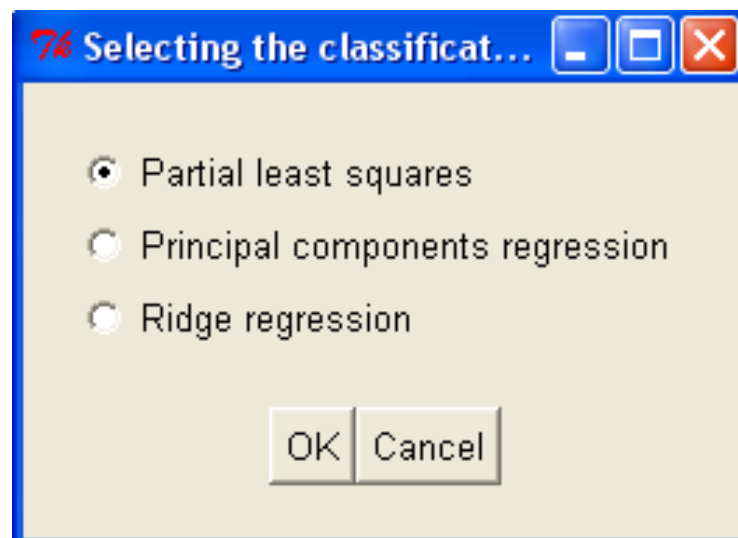


Figure 46: PDMCLASS methods selection mask.

The analysis will produce a numerical output of the efficacy of the dataset as classifier:

```
object neg pos
neg    21   1
pos    22  74
```

```
attr("error")
[1] 0.1949153
```

It is notable that this part of the analysis could take some minutes depending on the data set under analysis and the machine used for the analysis. Subsequently it will be possible to select the probe sets that have the greatest influence in differentiating sample classes. To do it, user will be requested to select the number of top ranked probe sets and the number of permutations to be used for the cross-validation. Probe set will be shown in a TK/TCL table with their probabilities to be able to discriminate between classes:

	<i>pos vs neg</i>
209604_s_at	1
202088_at	0.92
218807_at	0.8
211430_s_at	0.56
205081_at	0.48
213693_s_at	0.4
209138_x_at	0.44
200670_at	0.32
212099_at	0.44
208682_s_at	0.28

These results could be saved as a tab delimited file.

## 13 Biological Interpretation

This section gives a graphical interface to the GOstats package fig. 47. It also allows very basic meta-analysis using the metaArray package.

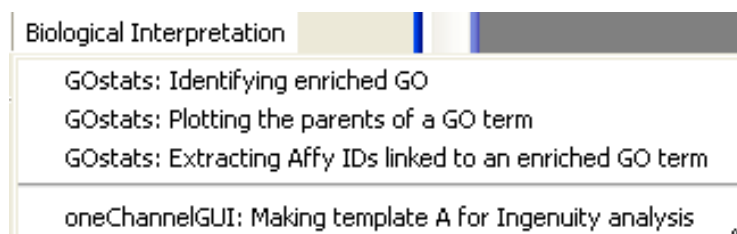


Figure 47: Biological interpretation menu.

### 13.1 Identifying enriched GO terms and related issues

This function is also available for gene level exon array analysis. Specific annotation libraries are not available for exon arrays, yet. Therefore, to perform this analysis we use the annotation informations embedded in oneChannelGUI and link the accession ids available in this annotation to Entrez Gene ids using the humanLLMappings, mouseLLMappings and ratLLMappings available in Bioconductor. The function *oneChannelGUI: Identifying enriched GO terms* searches for the presence of enriched GO terms within a set of differentially expressed probe sets, given a certain probe set universe, i.e. the array data available in Normalized Affy Data. For more information about GO enrichment please refer to the GOSTats vignette in the oneChannelGUI help menu. The user needs to select some parameters using a selection mask, fig. 48.

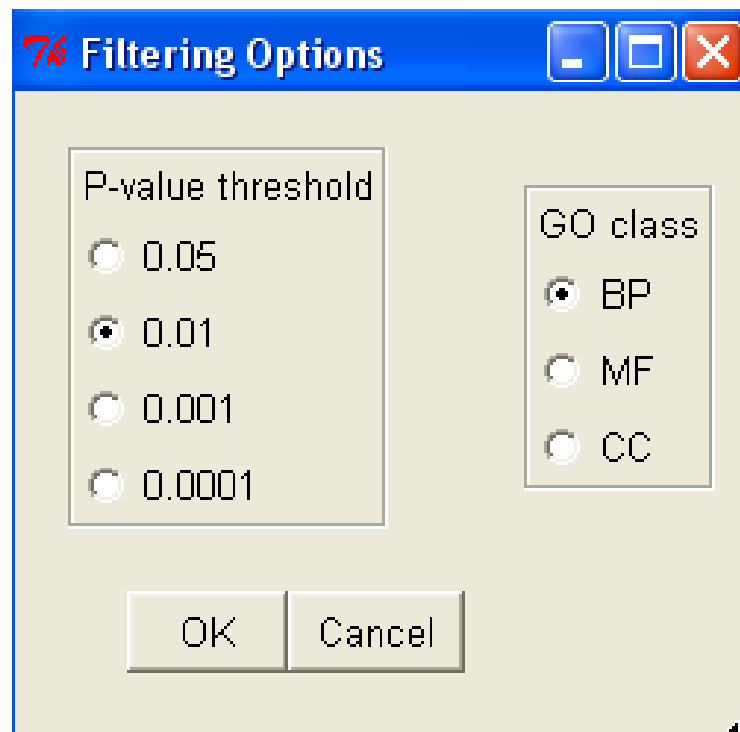


Figure 48: GO terms enrichment parameters selection mask.

Subsequently, the user will be requested to select a list of differentially expressed probe sets, saved in a txt file. The file should contain only a list of probe set separated by carriage return, without header:

```
1452968_at
1448228_at
1418028_at
1439113_at
```



1424338\_at  
1416503\_at  
1416371\_at  
1437165\_a\_at  
1451047\_at  
1434005\_at  
1421916\_at  
1457012\_at  
1443823\_s\_at  
1429379\_at  
1416168\_at  
1429974\_at  
1416121\_at  
1421917\_at  
1416405\_at

The analysis could require quite a lot of RAM and when it is finished a message summarizing the results pops up, fig. 49.

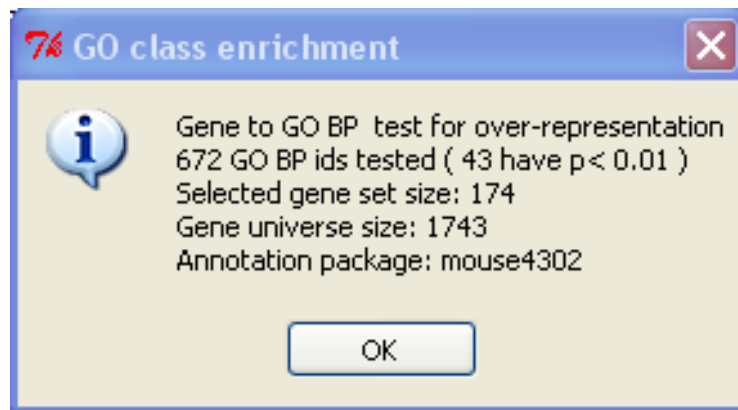


Figure 49: GO enrichment results summary message.

A table with the enriched GO terms will be then shown and it could be saved as tab delimited file, fig. 50.

In the main R window it will be possible to see a plot summarizing the GO terms relations existing between the enriched GO terms, fig. 51. Plotting results depends on the availability in your system of graphviz software and Rgraphviz package. Their installation can be quite critical for windows users, please contact the oneChannelGUI mantainer if you get problem in their installation or search over the Bioconductor mailing list.

It is also possible to highlight parents of a specific GO term using the function *Plotting parents of a GO term* In this case a dialog will be used to pass to the function the GO

74 GO enriched classes using p-value= 0.01 annotation lib= mouse4302 GO class= BP

GOBPID	Pvalue	OddsRatio	ExpCount	Coun	
GO:0006817	GO:0006817	1e-09	21	1.9	13
GO:0006820	GO:0006820	2.2e-08	9.8	3	15
GO:0015698	GO:0015698	1.2e-07	9.1	2.9	14
GO:0007155	GO:0007155	7.8e-07	3.4	11	29
GO:0048513	GO:0048513	2.6e-05	2.5	18	35
GO:0006811	GO:0006811	2.8e-05	3.2	8.9	22
GO:0001568	GO:0001568	0.00012	4.3	4.2	13
GO:0009607	GO:0009607	0.00014	3.7	5.4	15
GO:0001944	GO:0001944	0.00016	4.1	4.3	13
GO:0006952	GO:0006952	0.00034	3.5	5.2	14
GO:0051707	GO:0051707				
GO:0001525	GO:0001525				
GO:0030334	GO:0030334				
GO:0045765	GO:0045765				
GO:0009611	GO:0009611				
GO:0048514	GO:0048514				
GO:0006955	GO:0006955				
GO:0007596	GO:0007596	0.0016	9.3	1	5
GO:0007599	GO:0007599	0.0016	9.3	1	5
GO:0050878	GO:0050878	0.0016	9.3	1	5
GO:0050817	GO:0050817	0.0016	9.3	1	5
GO:0051270	GO:0051270	0.0016	5.4	1.9	7

Plotting GO parents

Plot of GO term parents will be displayed on the main R window

OK

Figure 50: Enriched GO terms table.

term, e.g. GO:0001525. Subsequently after selecting the GO class, i.e. BP, MF or CC, the results will be available in the main R window, fig. 52.

It is also possible to annotate and save, in an html file, the subset of differentially expressed probe sets associated to a specific enriched GO term using the function *oneChannelGUI: Extracting Affy IDs linked to an enriched GO term*. In the case exon arrays are used with *oneChannelGUI: Extracting Affy IDs linked to an enriched GO term* function the output file is a tab delimited file with the available annotations instead of an HTML file. The user will be requested to select the GO term of interest, fig. 51, and subsequently to open the file list of differentially expressed probe sets used for the GO enrichment analysis. A pop-up message will indicate when the annotation table will be ready to be saved in an HTML file, fig. 53. The output for exon arrays will be instead a tab delimited file.

## 13.2 meta-analysis

This menu gives also access to some meta-analysis tools, fig. 54.

It is possible to merge to the NormalizedAffyData up to 3 other data sets characterized by having the same ids and the same order of the NormalizedAffyData ids. To merge the data sets it is necessary a tab delimited file and a target for each data set. Integrative correlation (Parmignani et al. 2004), implemented in the metaArray package, can be accessed with the function *Mining similarities/dissimilarities between merged data sets (IC)*. The function produces an histogram of the various comparisons and it saves, in tab delimited file, the IC values for the various comparisons.

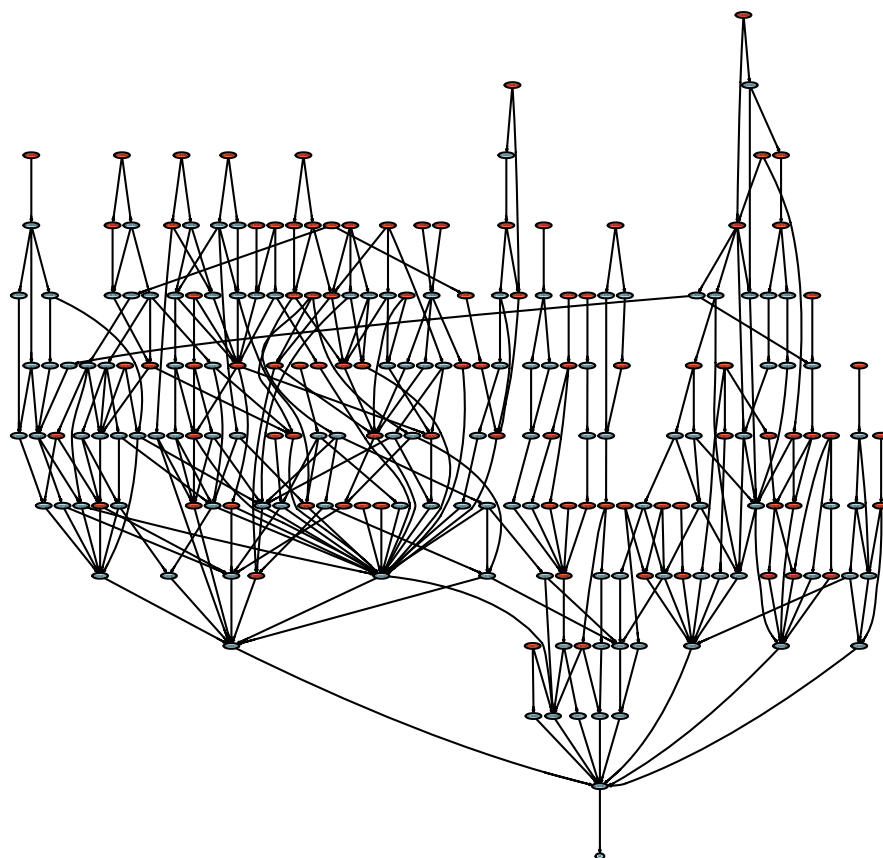


Figure 51: Relations between enriched GO terms. Enriched GO terms, red, others, light blue.

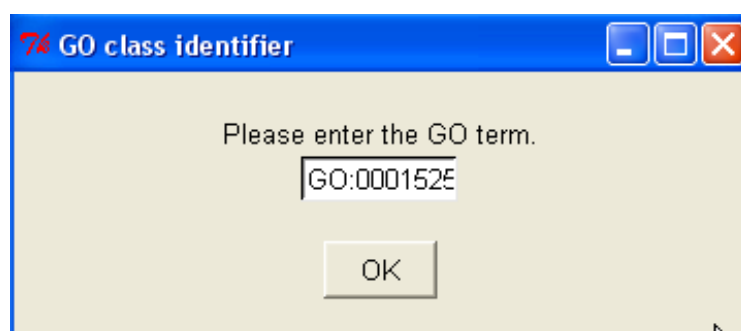


Figure 52: Plotting GO term parents.

Differentially expressed probe sets in GO:0001525

Probe	Symbol	Description	Function	Chromosome	Chromosome Location	GenBank	LocusLink	Cytoband	UniGene	PubMed	Gene Ontology
<a href="#">1416238_at</a>	Tie1	tyrosine kinase receptor 1		4	-117969122	<a href="#">NM_011587</a>	<a href="#">21846</a>	<a href="#">4 D2.1 4 50.0 cM</a>	<a href="#">Mm.4345</a>	62	<a href="#">nucleotide binding</a> <a href="#">protein kinase activity</a> <a href="#">protein serine/threonine kinase activity</a> <a href="#">protein-tyrosine kinase activity</a> <a href="#">receptor activity</a> <a href="#">ATP binding</a> <a href="#">extracellular space</a> <a href="#">protein amino acid phosphorylation</a> <a href="#">membrane integral to membrane</a> <a href="#">kinase activity</a> <a href="#">negative regulation of angiogenesis</a> <a href="#">transferase activity</a> <a href="#">negative regulation of cell migration</a>

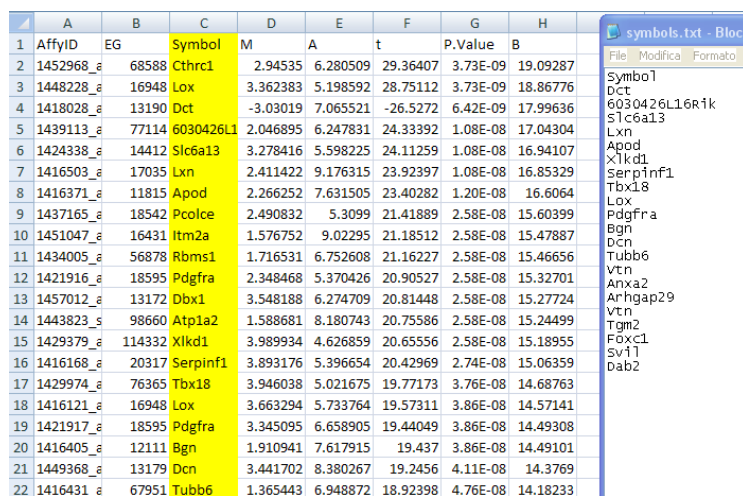
Figure 53: Annotation file for a subset of differentially expressed probe sets linked to GO:0001525 BP enriched term.

Biological Interpretation
oneChannelGUI: Identifying enriched GO terms
oneChannelGUI: Plotting the parents of a GO term
oneChannelGUI: Extracting Affy IDs linked to an enriched GO term
oneChannelGUI: Making template A for Ingenuity analysis
oneChannelGUI: Merging the same set of probe sets from different data sets
oneChannelGUI: Mining similarities/dissimilarities between merged data sets (IC)

Figure 54: Biological Interpretation DEVEL version.

## 14 General tools

This section allows the use of some functions which are not part of a specific Bioconductor package but could be of general use. The function *oneChannelGUI: Update all Bioconductor libraries* allows to run a on-line update of all the Bioconductor libraries present in the system. This function is very useful to keep updated for bugs correction during the 6 months life of a Bioconductor release. The function *oneChannelGUI: Extract a column from a tab delimited file* allows the extraction of any of the columns of a tab delimited file. This function is particularly useful to generate probeset ids list to be used for Venn diagram representation. The function *oneChannelGUI: Filtering a tab delimited file* allows to subset a tab delimited file given a list of values, e.g. values, symbols, probe sets, etc., present in a file where each value is separated from the others by carriage return. The tab delimited file subsetting is performed on the basis of the column, fig. 55 yellow, sharing the same header of the list of values, fig. 55.



	A	B	C	D	E	F	G	H
1	AffyID	EG	Symbol	M	A	t	P.Value	B
2	1452968_e	68588	Cthrc1	2.94535	6.280509	29.36407	3.73E-09	19.09287
3	1448228_e	16948	Lox	3.362383	5.198592	28.75112	3.73E-09	18.86776
4	1418028_e	13190	Dct	-3.03019	7.065521	-26.5272	6.42E-09	17.99636
5	1439113_e	77114	6030426L1	2.046895	6.247831	24.33392	1.08E-08	17.04304
6	1424338_e	14412	Slc6a13	3.278416	5.598225	24.11259	1.08E-08	16.94107
7	1416503_e	17035	Lxn	2.411422	9.176315	23.92397	1.08E-08	16.85329
8	1416371_e	11815	Apod	2.266252	7.631505	23.40282	1.20E-08	16.6064
9	1437165_e	18542	Pcolce	2.490832	5.3099	21.41889	2.58E-08	15.60399
10	1451047_e	16431	Itm2a	1.576752	9.02295	21.18512	2.58E-08	15.47887
11	1434005_e	56878	Rbms1	1.716531	6.752608	21.16227	2.58E-08	15.46656
12	1421916_e	18595	Pdgfra	2.348468	5.370426	20.90527	2.58E-08	15.32701
13	1457012_e	13172	Dbx1	3.548188	6.274709	20.81448	2.58E-08	15.27724
14	1443823_e	98660	Atp1a2	1.588681	8.180743	20.75586	2.58E-08	15.24499
15	1429379_e	114332	Xlkd1	3.989934	4.626859	20.65556	2.58E-08	15.18955
16	1416168_e	20317	Serpinf1	3.893176	5.396654	20.42969	2.74E-08	15.06359
17	1429974_e	76365	Tbx18	3.946038	5.021675	19.77173	3.76E-08	14.68763
18	1416121_e	16948	Lox	3.663294	5.733764	19.57311	3.86E-08	14.57141
19	1421917_e	18595	Pdgfra	3.345095	6.658905	19.44049	3.86E-08	14.49308
20	1416405_e	12111	Bgn	1.910941	7.617915	19.437	3.86E-08	14.49101
21	1449368_e	13179	Dcn	3.441702	8.380267	19.2456	4.11E-08	14.3769
22	1416431_e	67951	Tubb6	1.365443	6.948872	18.92398	4.76E-08	14.18233

symbols.txt - Bloc...

File Modifica Formato

Symbol  
Dct  
6030426L16R1k  
Slc6a13  
Lxn  
Apod  
Xlkd1  
Serpinf1  
Tbx18  
Lox  
Pdgfra  
Bgn  
Dcn  
Tubb6  
Vtn  
Anxa2  
Arhgap29  
Vtn  
Tgm2  
Foxc1  
sv11  
Dab2

Figure 55: Sub setting a tab delimited file by a list of symbols.

The function *oneChannelGUI: Downloading Gene/Exon library files* allows to download all the library files needed to use APT tools for probe set summaries for Gene and Exon 1.0 ST arrays. The function *oneChannelGUI: Set Affymetrix apt tools folder and download Reference Sequences* allows the user to define a folder where apt tools were installed and to download in the subdir blast of the apt dir the reference sequences from NCBI repository. Those file are compressed and have the gz extension. They can be unpacked manually by the user or via oneChannelGUI. This option is available at the end of the downloading but takes quite a long time.

The function *oneChannelGUI: deleteLocalData* will reset the folders defined by *oneChannelGUI: Downloading Gene/Exon library files* and *oneChannelGUI: Downloading Gene/Exon library files*. Data present in the two folders will not be deleted! The function *oneChan-*

*oneChannelGUI: buildingLocalAnnotation* allows to update the internal oneChannelGUI gene/exon-level annotations quiering netaffx database using the affyCompatible library. Annotation files are saved in .rda format in the subdir data in located in the oneChannelGUI folder. Windows users need to drag those .rda files in the Rdata.zip file present in the data dir. A file called netaffxUpdates.txt in the etc subdir kept tracks of annotation file updating. The function *oneChannelGUI: variantExons* allows to update the internal oneChannelGUI gene/exon-level annotations that links exon-level probesets to exons which are associated only to a subset of transcript isoforms. The table is generated integrating the exon-level annotation derived by the core subset of Affymetrix exon level annotation and a USCS table mapping the location of variant exons on the genome. The function *Installing meV clustering software* will install the meV clustering tool. A java application very powerful for microarray data clustering. meV software can be started using the function *Starting meV clustering software* in the Biological Interpretation menu. The function *Install Bowtie and Picard tools* allows the installation of bowtie and picard tools for primary mapping of short reads. The above mentioned function also allow to download/build Bowtie miRNAs precursors and exons reference sequences for human, mouse, rat, bovine. The function *Information on external software and data connected to oneChannelGUI* provides information on external software installed and connected to oneChannelGUI

## 15 Help

This menu allows to access to the vignettes of the Bioconductor packages implemented in oneChannelGUI and to this oneChannelGUI vignette.