oneChannelGUI Package Vignette

Raffaele A Calogero, Francesca Cordero, Remo Sanges

August 31, 2008

1 Introduction

This package is an add-on of affylmGUI 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 affylmGUI capabilities, rearranging and extending the affylmGUI menus.

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

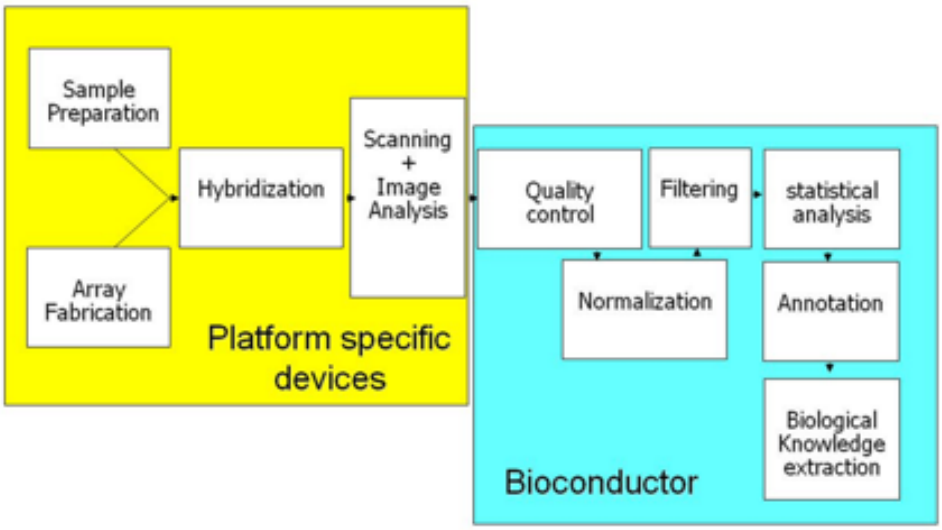


Figure 1: Microarray analysis pipe-line.

This vignette gives a general overview of the available graphical tools present in oneChannelGUI.

N.B:

All the oneChannelGUI graphical outputs are visualized in the R main window, to reduce RAM usage, which is a critical issue when new generation 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 and 2.

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 software version 1 or 2. In this menu, fig. 2, are given the main functionalities to handle a microarray analysis project.

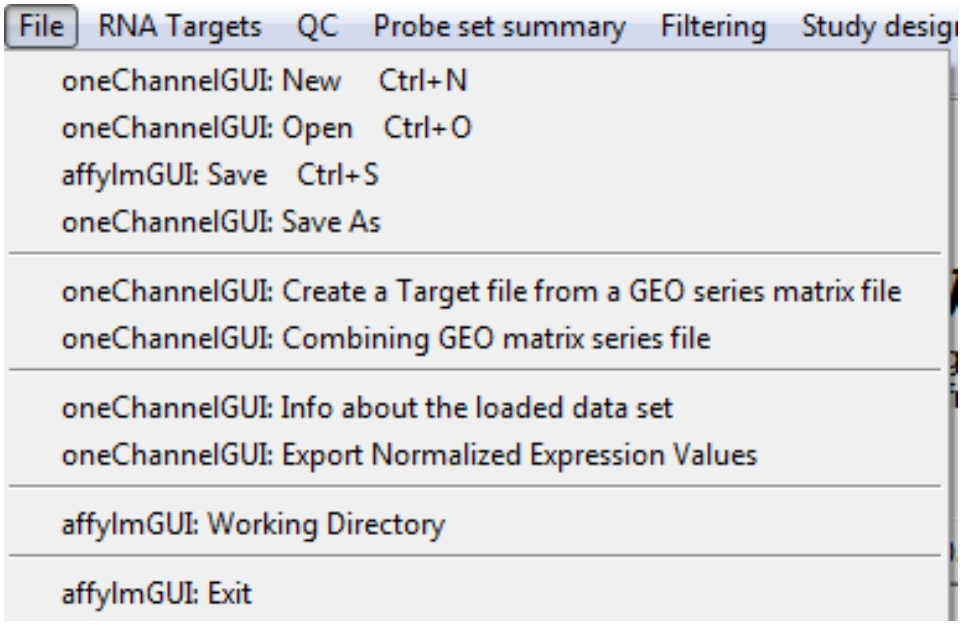


Figure 2: File menu.

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 ;,:_-|\!?+^()[]{}*

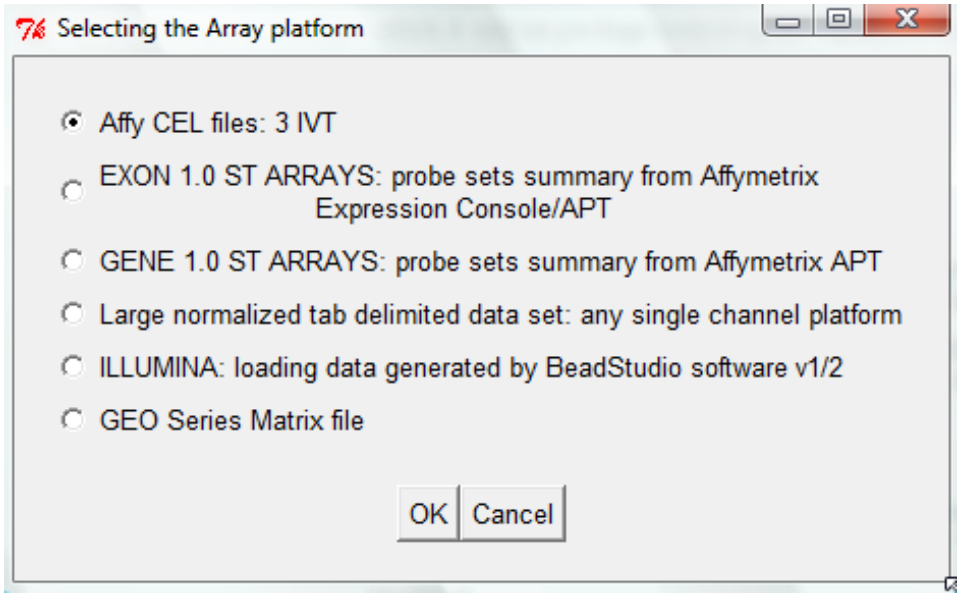
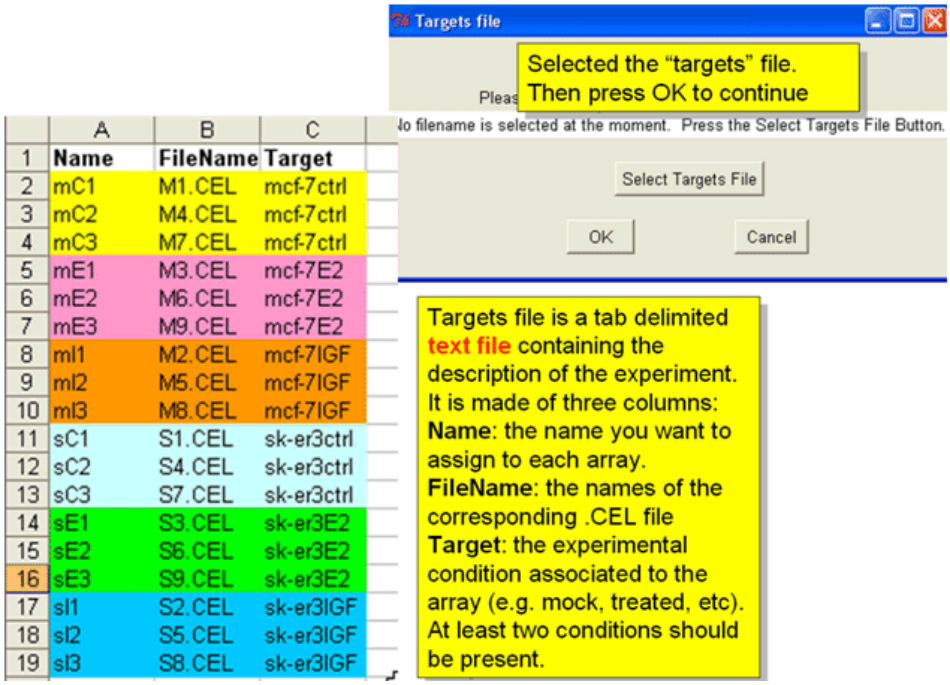


Figure 3: New: array type selection menu.



	A	B	C
1	**Name**	**FileName**	**Target**
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	mI1	M2.CEL	mcf-7IGF
9	mI2	M5.CEL	mcf-7IGF
10	mI3	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	sI1	S2.CEL	sk-er3IGF
18	sI2	S5.CEL	sk-er3IGF
19	sI3	S8.CEL	sk-er3IGF

Figure 4: Target file structure.

4.1.2 Loading Affy .CEL files

This sub menu, fig. 3, is enterily 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`). 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 IT IS NEEDED TO DOWNLOAD THE GENE/EXON LIBRARY FILES. THIS CAN BE DONE WITH THE FUNCTION
oeChannelGUI: Set library folder and install Affy gene/Exon library files
LOCATED IN THE GENERAL TOOLS MENU

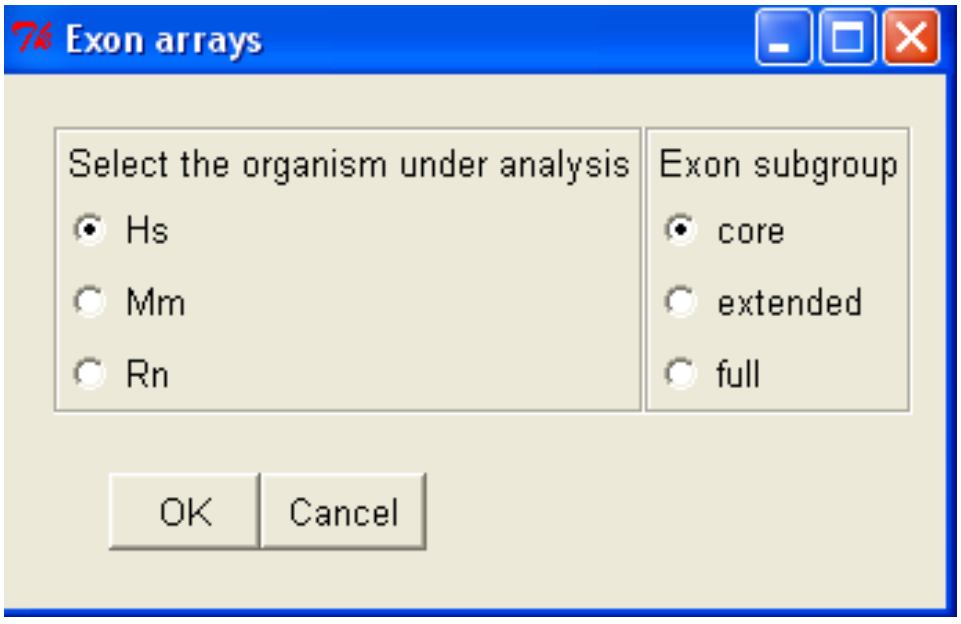


Figure 5: Sub menu to define the organism and the subset of exon data that will be loaded.

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.

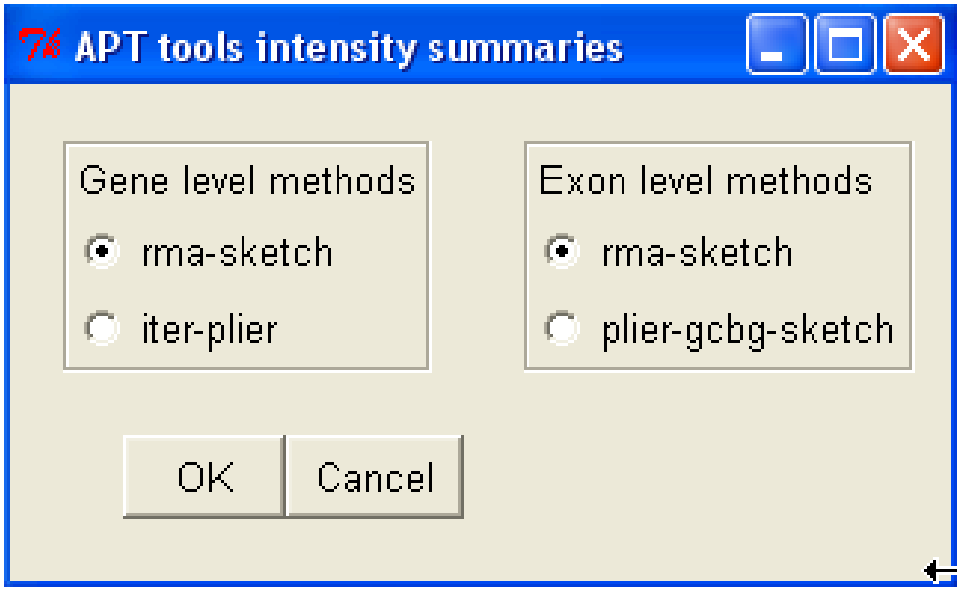


Figure 6: Sub menu to define the type of probe set summary to be applied.

Concerning probe set summary options, fig. 6, since PLIER/RMA are model-based algorithms, exons that are alternatively spliced in the samples, therefore exhibiting different expression patterns compared to the constitutive exons, will have down-weighted effect in overall gene-level target response values. A better estimation of gene-level signal could be obtained using IterPLIER, which is a variation of PLIER that iteratively discards features (probes) that do not correlate well with the overall gene-level signal and then recalculates the signal estimate to derive a robust estimation of the gene expression value primarily based on the expression levels of the constitutive exons. Concerning exon level expression estimation, most probe sets only have four probes, which is too limited

to be useful with IterPLIER at the individual exon level, therfore il will be better to use PLIER/RMA.

Probe set summary calculation and uploading will take few minutes depending on the number of .CEL to be loaded and the PC in use. Ones probe set summary has been calculated, using APT tool, it is also possibile to calculate DABG p-values, fig. 7.

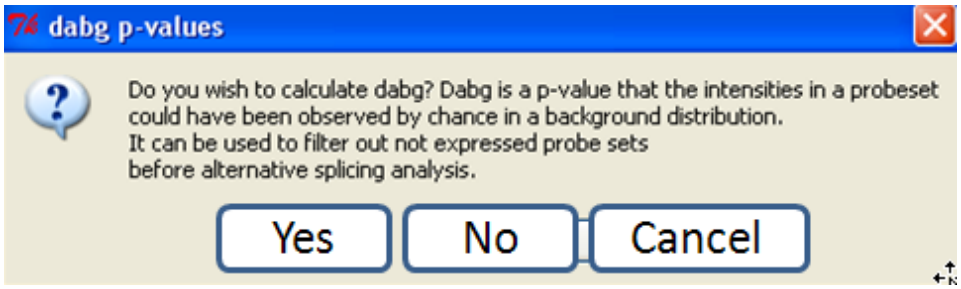


Figure 7: Selecting DABG p-value calculation.

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.

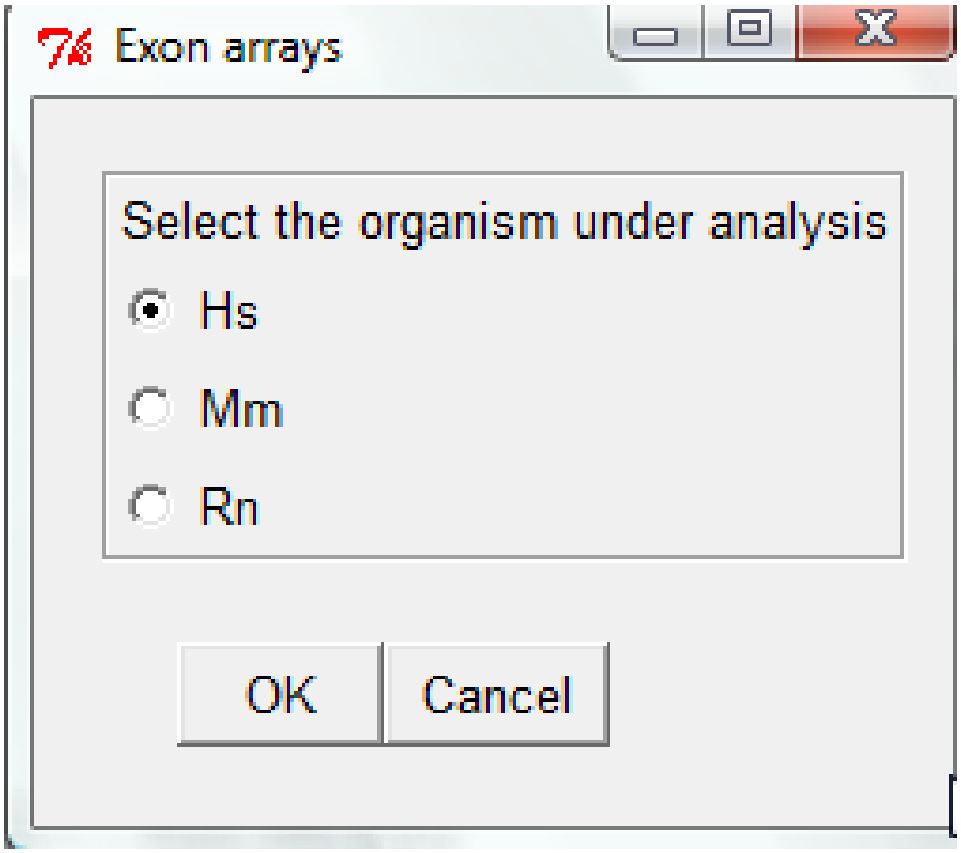


Figure 8: Sub menu to define the organism and the subset of data that will be loaded.

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

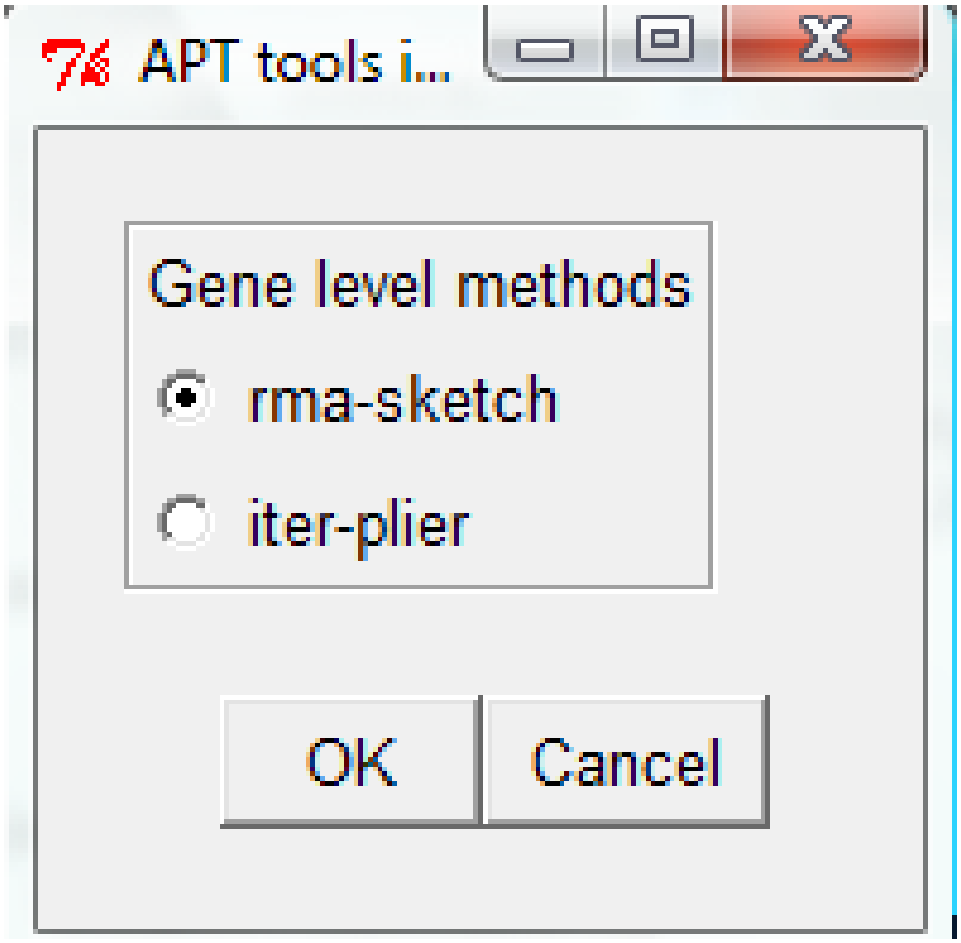


Figure 9: Sub menu to define the type of probe set summary to be applied.

4.1.5 Loading ILLUMINA BeadStudio

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

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.

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 matri
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 ArrayExpress 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* or *Save*, 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 in Normalize Affy Data but also exon level expression data. Furthermore, if already calculated it is possible to export 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.

4.5.1 Probe set annotation

The Bioconductor annotation library for IVT Affymetrix arrays or GEO Matrix Series file are directly attached. Concerning Gene and Exon 1.0 ST arrays, annotation information are actually embedded in oneChannelGUI. For exon arrays annotation is available at the gene level for the core subset of Hs/Mm/Rn. As soon as Bioconductor annotation libraries will be available for exon arrays the oneChannelGUI annotation will use them for annotation. 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 of data present in Normalized Affy Data using the function *Attaching ACC to Probe set IDs*, present in the Biological Interpretation menu. This funciton also allows to ling EGs to glevel probe sets of a tab delimited fine that has in the first column the probe set ids.

5 RNA target

The first item in the menu, fig. 10, is inherited from affylmGUI and allows the visualization of the experimental structure described by the target file used to load the expression data.

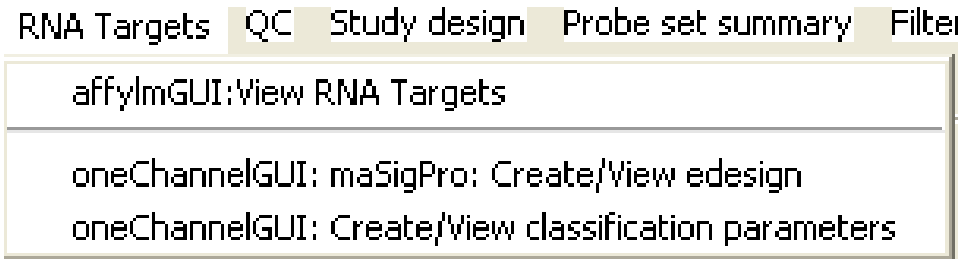


Figure 10: RNA target menu.

The second item, fig. 10, *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. 11.

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 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:

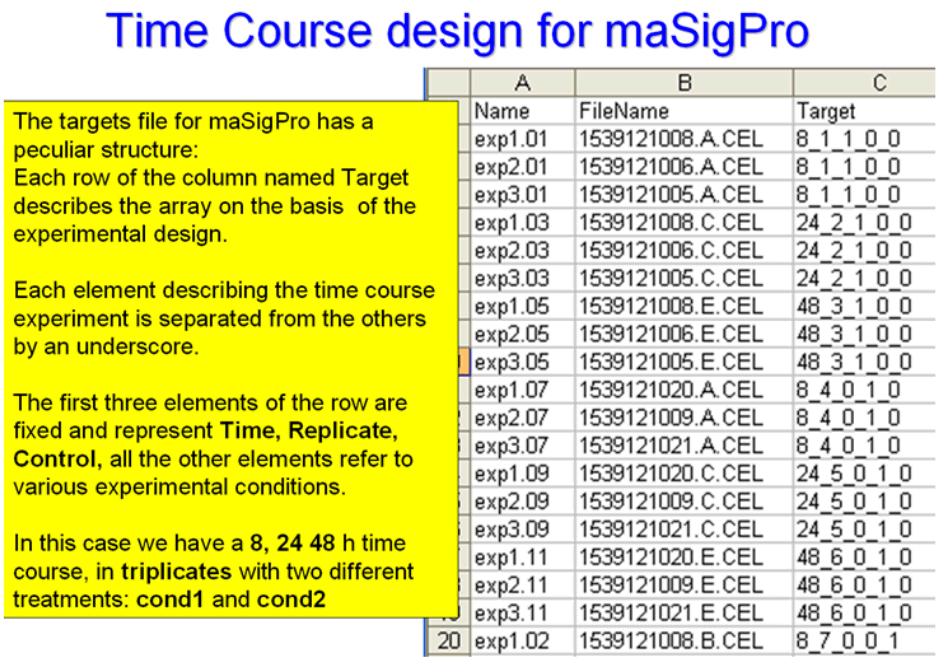


Figure 11: Target file for time course analysis.

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:

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

The third item, fig. 10, instead refers to the reorganization of a target file containing the information related to clinical parameters to be used for classification pourposes. 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:

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

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

This menu is specialized depending on the type of microarray data set loaded

6.1 QC for IVT arrays loaded starting from .CEL files

This menu, fig. 12, inherits all affylmGUI probe/probe set level quality controls, refer to affylmGUI for their usage.

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. 13.

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.

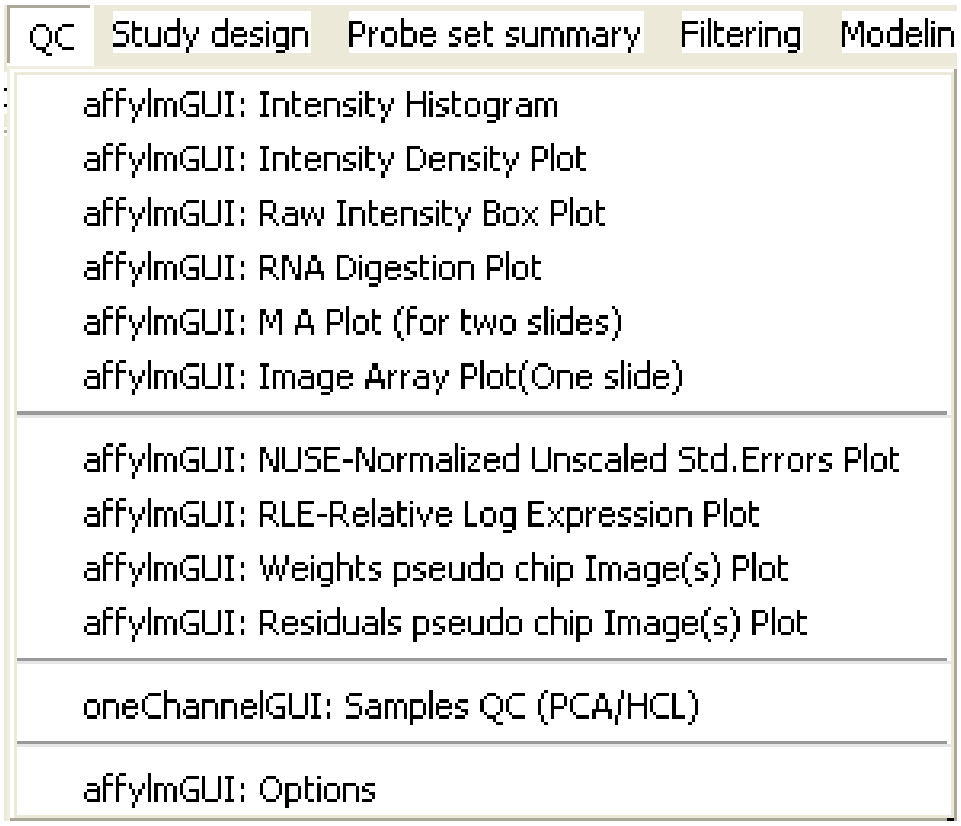


Figure 12: QC for IVT arrays.

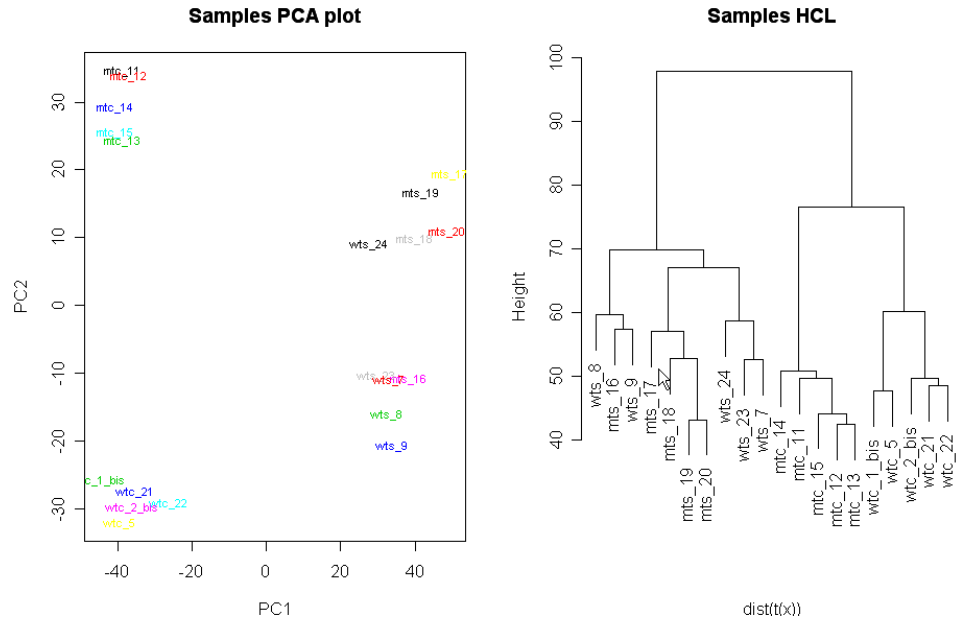


Figure 13: Sample QC: PCA/HCL output for IVT arrays.

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 14.

6.3 QC for exon arrays

In the case of exon array the QC menu is slighly different, as shown in fig. 15
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.

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

As it can be seen in fig. 16 normalization masks the fact that a sub set of arrays, i.e. those with a very narrow boxplot 16A, had something wrong in hybridization. This

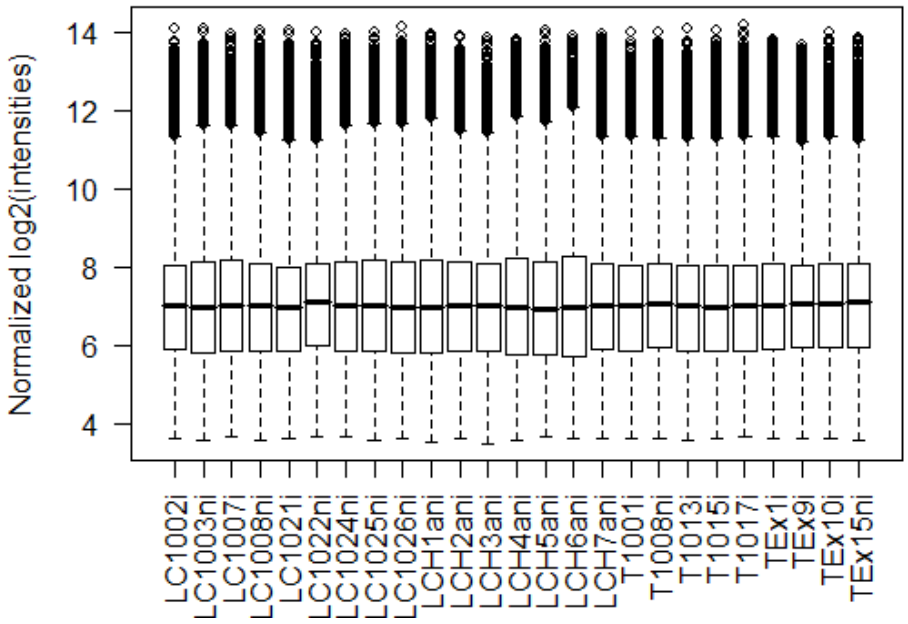


Figure 14: Box plot of normalized data.

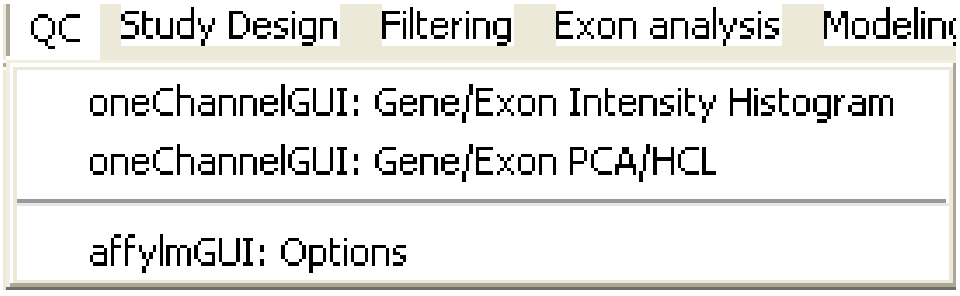


Figure 15: QC menu for exon arrays.

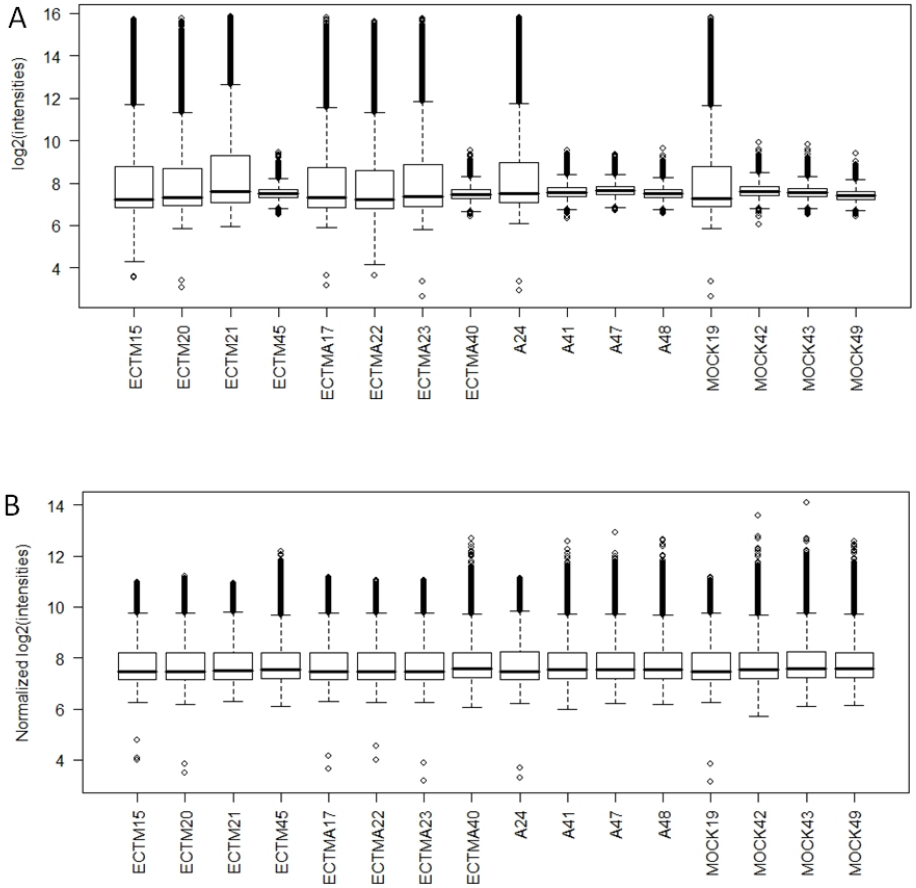


Figure 16: A set of Illumina arrays before and after data normalization.

problem is completely masked in the normalized data 16B. 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. 17.

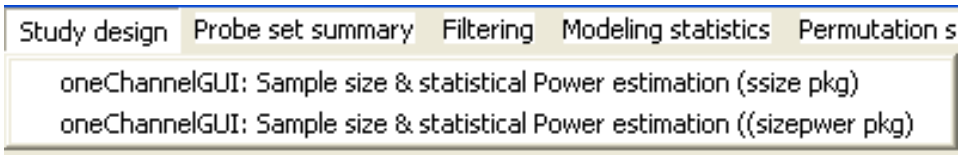


Figure 17: Study design menu.

This menu gives access to two functions, which are graphical implementations of the ssize and sizepower Bioconductor libraries. 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 affylmGUI 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.

In the case of exon arrays this menu is not available since expression data, for exon arrays are calculated by APT tools using the oneChannelGUI interface or they are loaded as tab delimited files exported by Affymetrix Expression Console.

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. 18

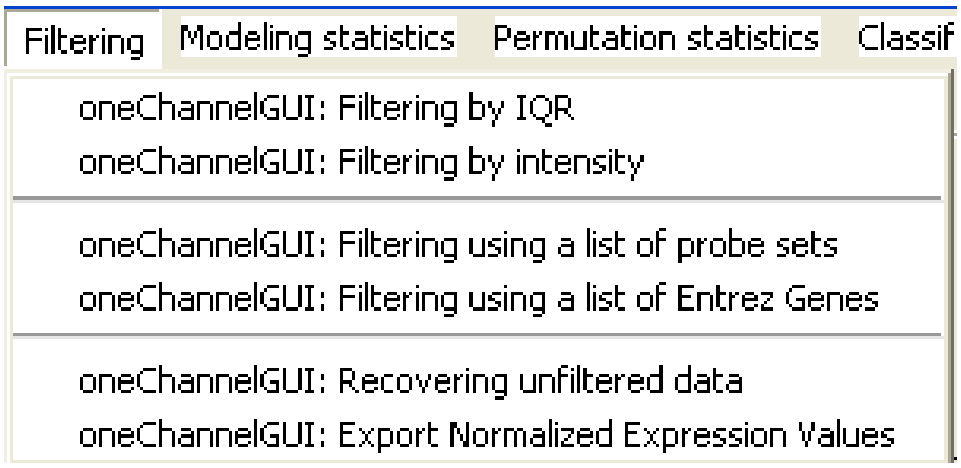


Figure 18: Filtering menu for GEO/Affy IVT arrays.

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. 19

In oneChannelGUI it is possible to select three filtering values:

1. IQR 0.1, weak filter, i.e. only the estreme 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.

More informations about the efficacy of the filtering procedure can be seem in: `http://www.bioinformatica.unito.it/oneChannelGUI/diaset.1.usa.ppt`

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.

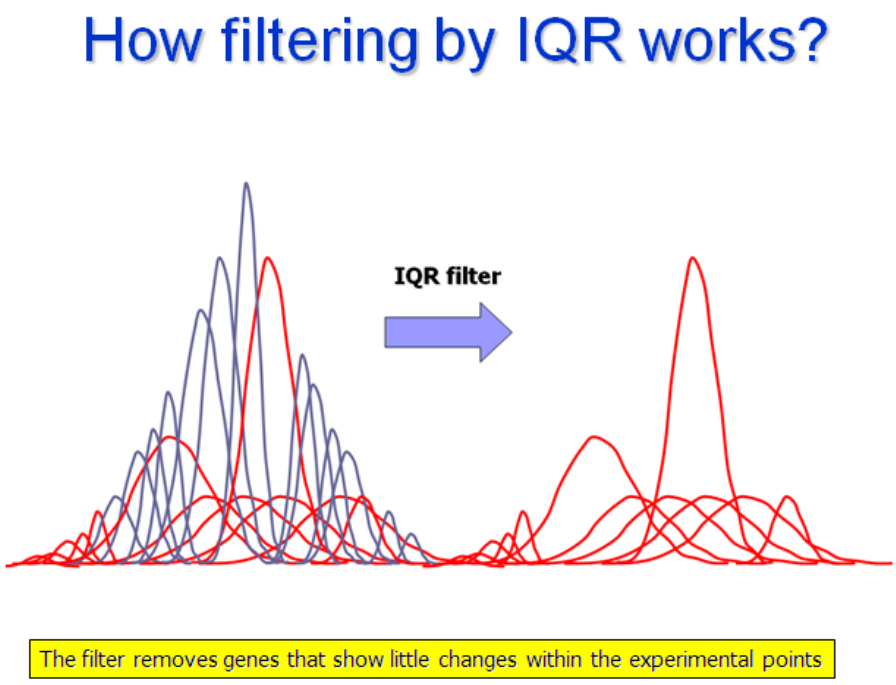


Figure 19: IQR filtering: The distributions of the various probe sets beloging to a data set are shown in red, if they are wide and they are retained by the filter, and in blue, if they are narrow and they are discarded by the filter.

9.2 Filtering by intensity

For IVT/GEO/tab delimited expression data files it is also possibile to apply a filtering procedure based on intensity signals, the graphical interface to do it is shown in fig. 20.

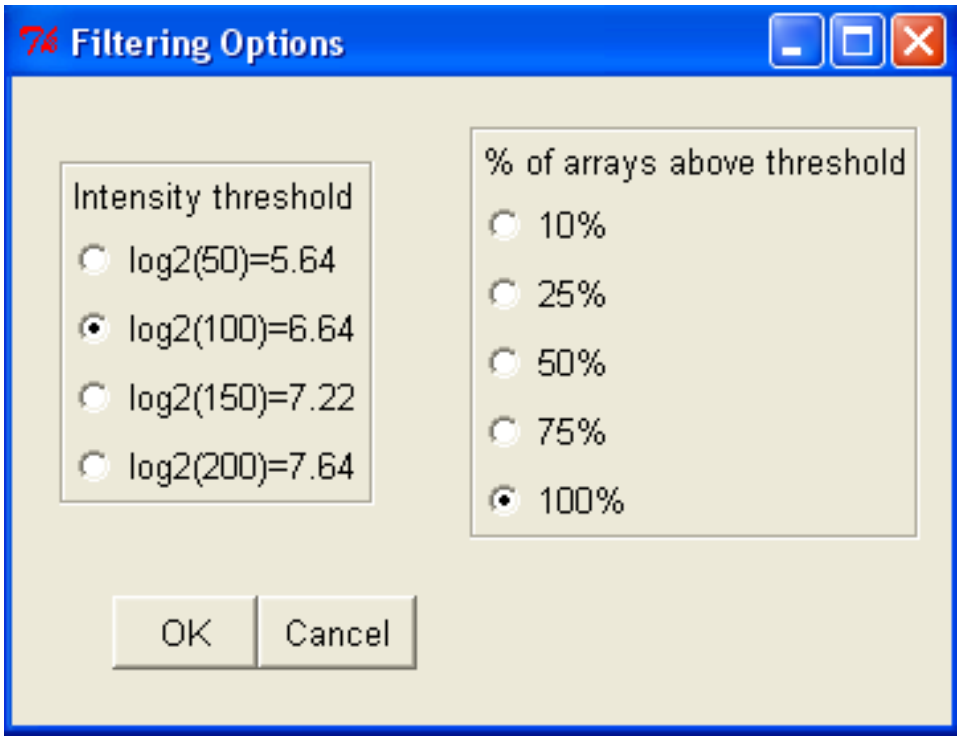


Figure 20: Intensity fitering: This filter will retain a probe set only if a certain fraction of the samples are characterized by an intensity value over a certain user defined threshold.

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

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.

9.5 Filtering menu: exon data

If exon data are loaded the filtering menu appear slighly different, fig. 21.

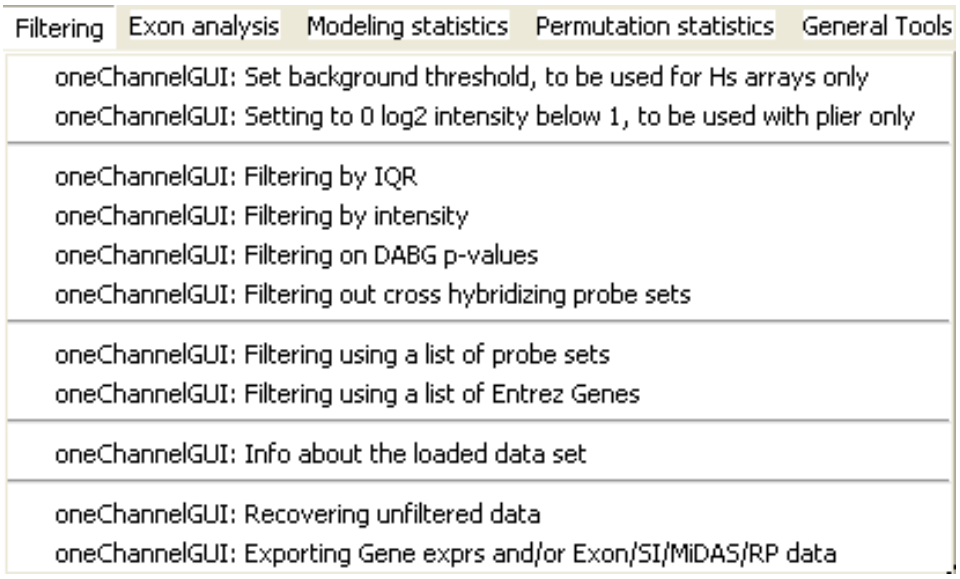


Figure 21: Filtering menu for exon data.

In particular, the function *Set background threshold* collects the exon/intron expression values for a set of housekeeping genes present in the chip within chip quality controls and it offers the opportunity to set a background intensity threshold on the basis of the desidered level of intersection between the expression of exons versus introns. RMA intensity calculation is preferred, fig. 22, since, if probe set summaries are calculated with Plier or iterPlier, the differences in expression distribution between exon and introns are not enought wide, fig. 23.

Setting a background threshold using exon/intro distributions for HK genes, it is possible to apply to the full data set an intensity filtering that will remove gene and the corresponding exons on the basis of the selected threshold. The intensity filter for exon arrays works exactly as that for IVT arrays but using a fixed threshold defined as described.

An other filter that allows the removal of low intensity probe sets is based on the DABG p-values. Using the function *Filtering on DABG p-values* it is possible to select the desidered level of filtering using a mask, fig. 24.

A threshold of 50% means that only probe sets where in half of the samples over the selected DABG p-value threshold will kept. As can be seen in fig. 25 this filtering also removes low intensity signals very near to zero.

N.B. Recovering the data prior filtering is not implemented for DABG p-value filtering, yet.

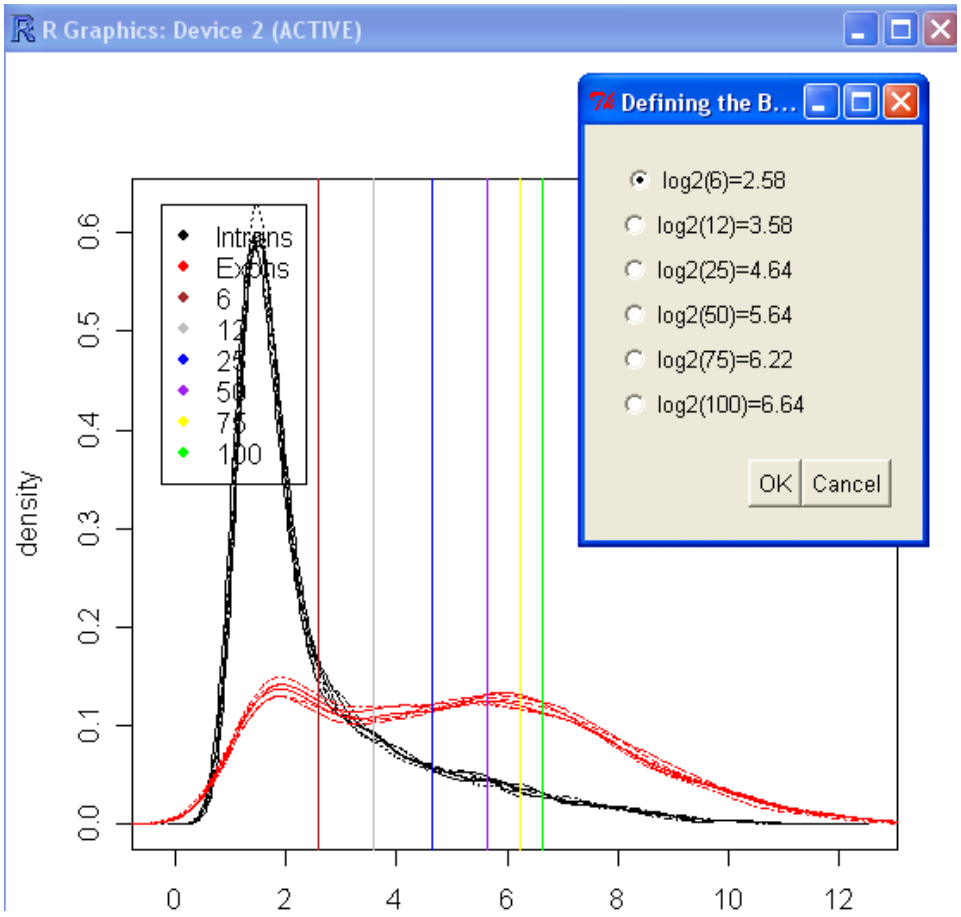


Figure 22: Human exon arrays, probe set summaries were calculated with RMA, exon/intron distribution of HK present in the chip as quality controls.

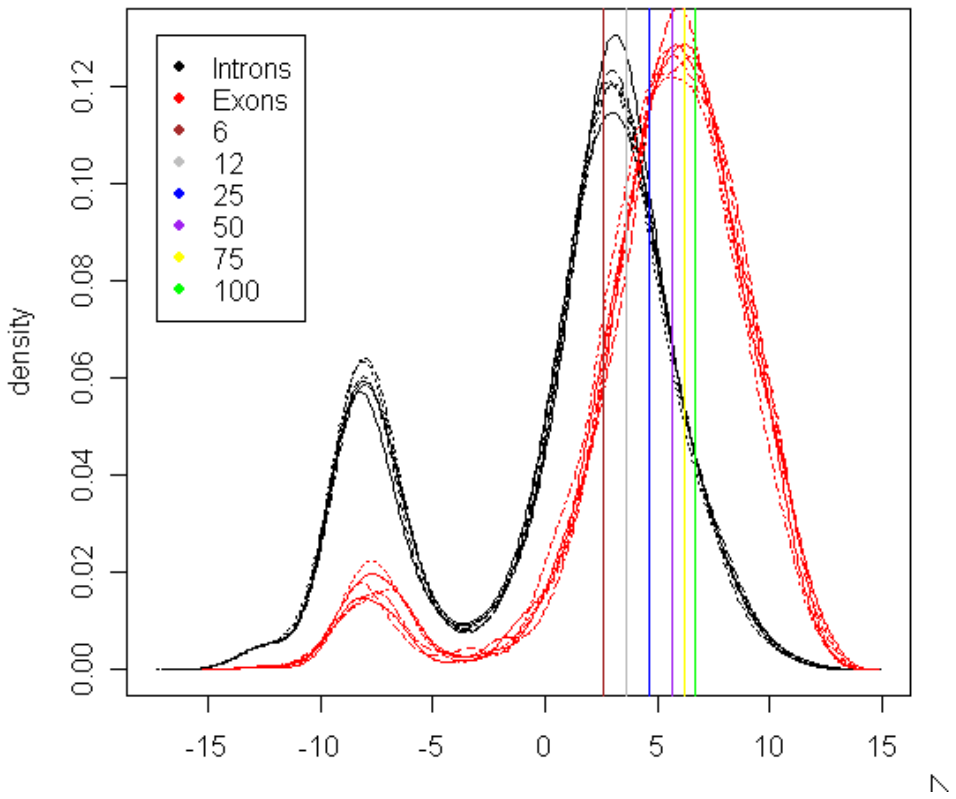


Figure 23: Human exon arrays, probe set summaries were calculated with iterPlier (gene level) and Plier (exon level), exon/intron distribution of HK present in the chip as quality controls.

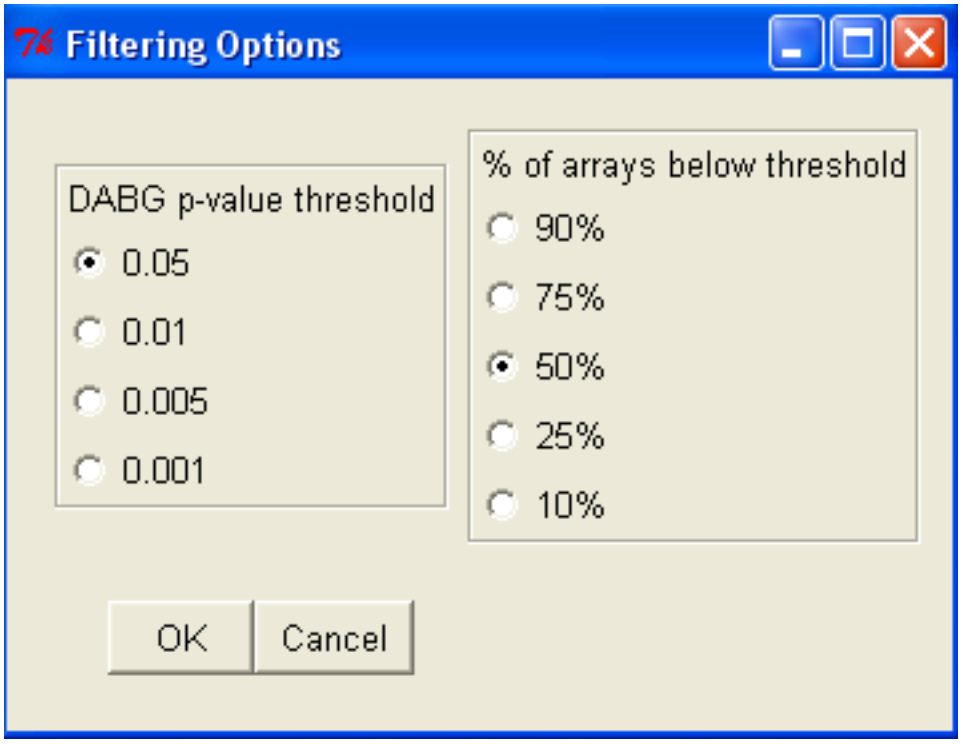


Figure 24: DABG p-value based filtering selection mask.

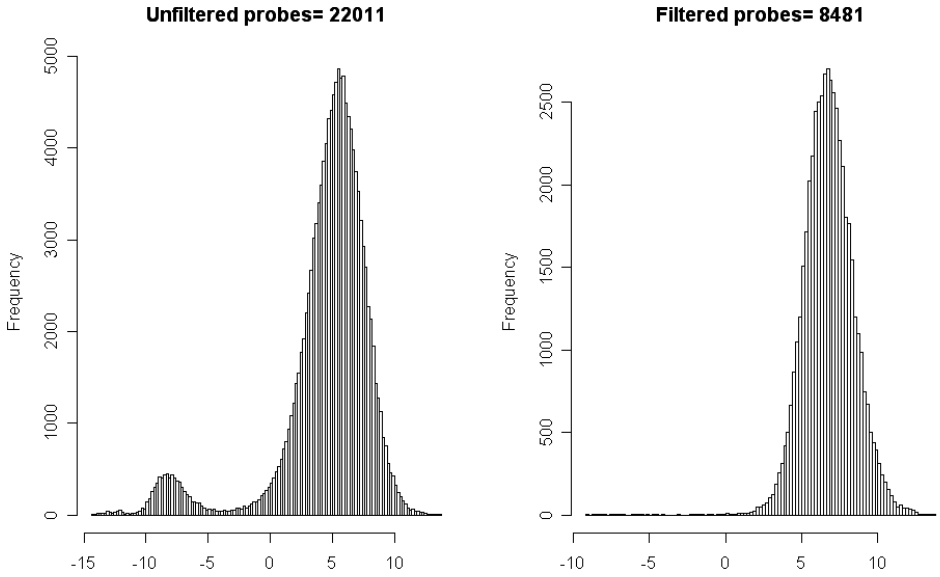


Figure 25: DABG p-value filtering results with parameters: DABG p-value threshold 0.05 and 50% of arrays over the threshold.

Reguarding very low intensity probe sets present if iterPlier/Plier are used, the function *Setting to 0 log2 intensity below 1, to be used with plier only* will set them zero.

A filter to eliminate cross hybridizing probe sets is based on Affymetrix XHYB and CROSSHYB annotations, which are part of the data embedded in oneChannelGUI. XHYB field is mainly an indicator of weak assignment between a transcript cluster and the assigned mRNA, suggesting a potential crosshyb. CROSSHYB is a measure of the promiscuity of the probes within a probe set among transcribed sequences.

1. unique. All probes in the probe set perfectly match only one sequence in the putatively transcribed array design content. The vast majority of probe sets are unique.

2. similar. All the probes in the probe set perfectly match more than one sequence in the putatively transcribed array design content.

3. mixed. The probes in the probe set either perfectly match or partially match more than one sequence in the putatively transcribed array design content.

XHYB and CROSSHYB are used to remove probe sets characterized by multiple hybridization of exon probes Cross-hybridization potential of the probe set determined at the time of array design. The function *Filtering out cross hybridizing probe sets* allows to remove all gene level probe sets, and the corresponding exon data, associated to exon level probe sets mapped as XHYB or CROSSHYB. The filtering option mask is shown in fig. 26.

The function *Selecting only probe sets with multiple mRNA association in ensembl* it is very useful when alternative splicing events are investigated, if the researcher is interested to investigate only those probe sets associated to multiple transcripts annotated on ensembl database. We strongly suggest to apply this filter at least to get an overview of the possibile known alternative splicing events that could be collected within the annotated ensembl data. This filter will reduce both the computational time to calculate splice index and type I statistical error, at the level of statistical analysis for alternative splicing detection.

Specifically, this function select at gene-level only those probe sets which are associated to multiple entries on ensembl data base. The filter uses the biomaRt package to collect this information from ensembl database.

The function oneChannelGUI: *Exporting Gene-level probe set ids* is useful to extract the list of probe set ids associated to the gene-level data set loaded on oneChannelGUI.

10 Modelling statistics

This menu allows to perform limma differential expression analysis as well as time course analysis using the maSigPro package, fig. 27.

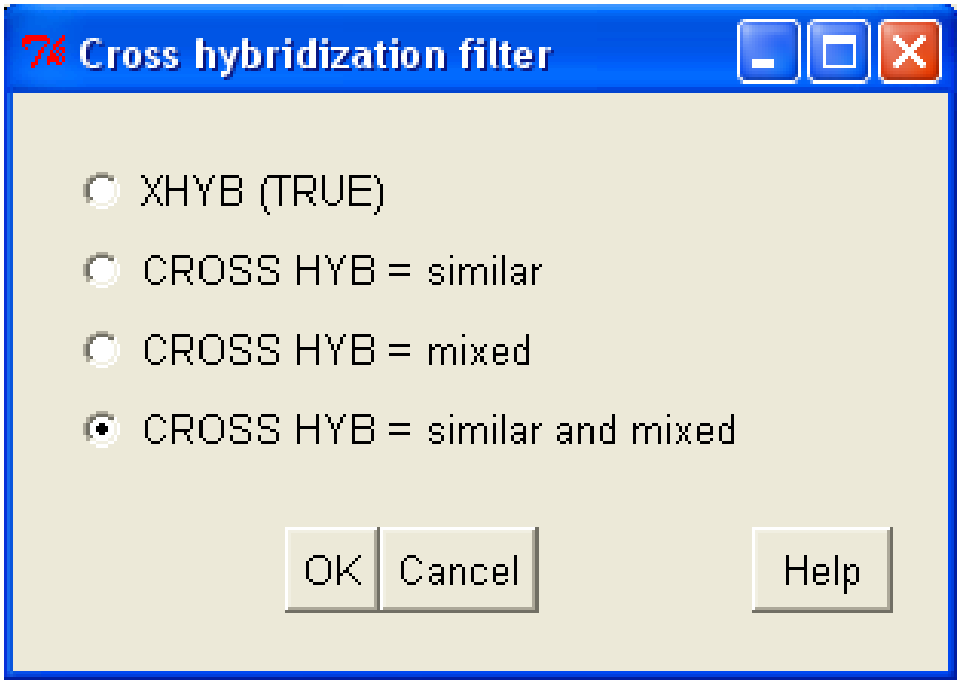


Figure 26: Cross hybridization filtering options mask.

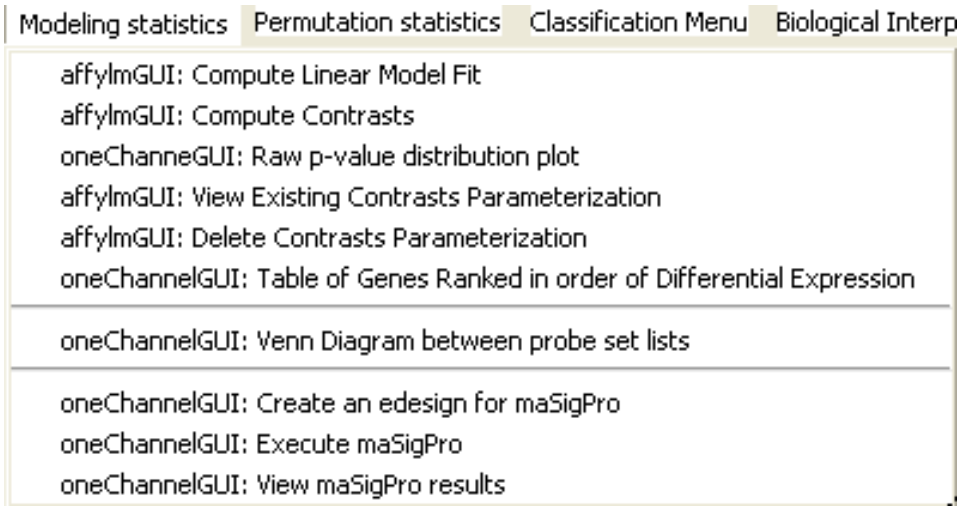


Figure 27: Modelling statistics

10.1 limma

The affylmGUI interface to limma is fully inherited, see limma and affylmGUI vignettes for usage. The function *raw p-value distribution* is implemented to evaluate if the BH/BY 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. 28.

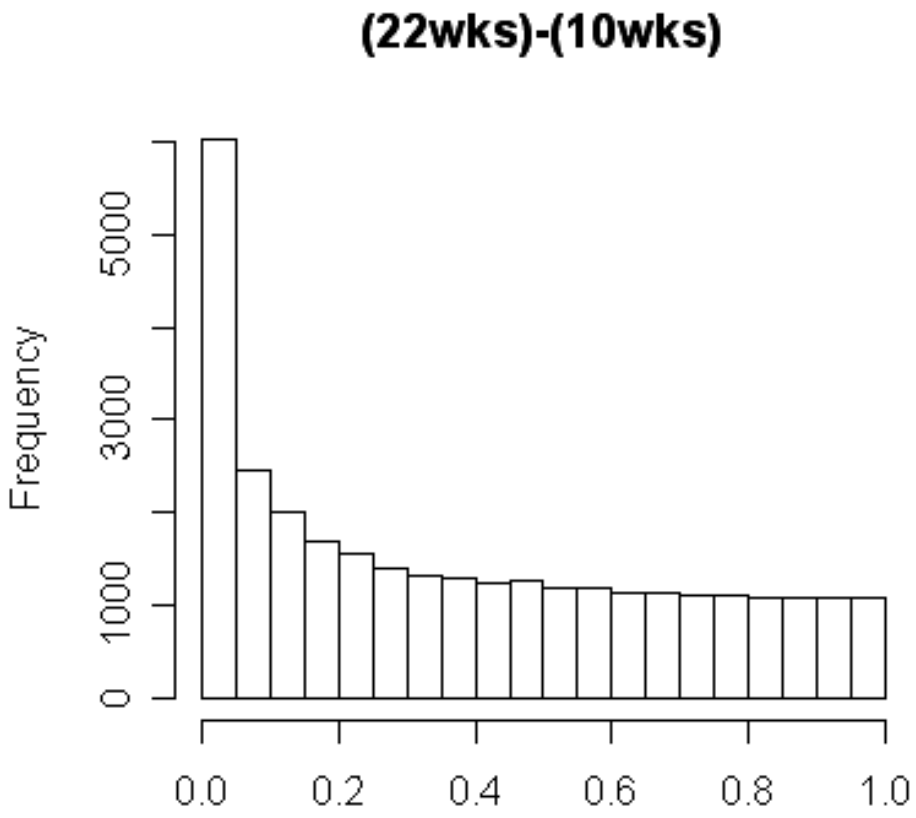


Figure 28: Output of *raw p-value distribution*: The raw distribution of p-values is uniform in the non significant range.

The affylmGUI function *Table of genes ranked in order of differential expression* is a modified version of the original found in affylmGUI to allow users to check with

MA/Volcano plots the set of differentially expressed probe sets before saving the table, fig. 29.

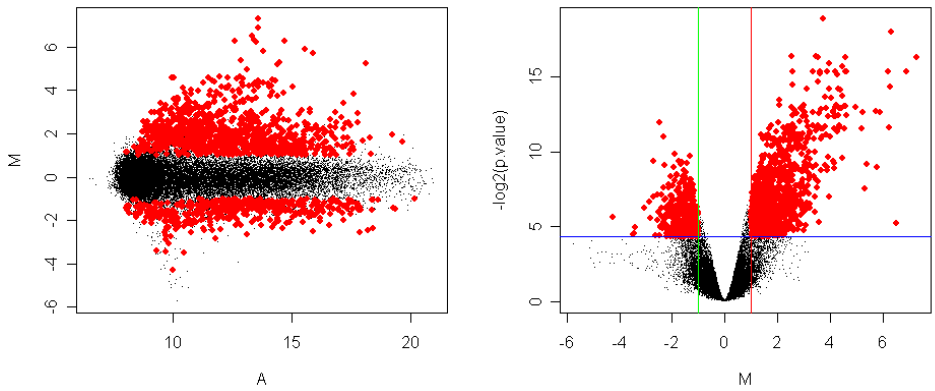


Figure 29: MA and Volcano plots for differentially expressed probe sets, red dots, detected by limma method.

10.2 Venn diagrams between probe set list

This function is modified with respect to the original one presents in affylmGUI to allow Venn diagrams using lists of probe sets, saved in text files where each id is separed 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 retundancy.

10.3 Time course analysis

Time course analysis can be performed on oneChannelGUI using maSigPro package, fig. 27.

maSigPro is a R package for the analysis of single and multiseries 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.

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. 30.

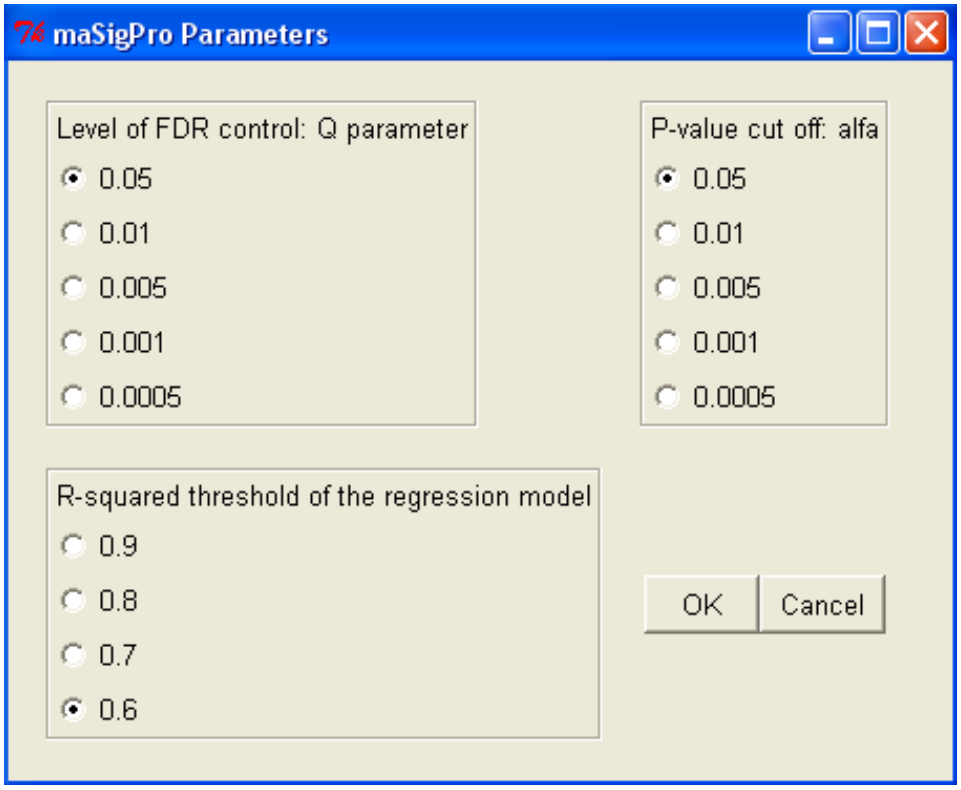


Figure 30: maSipPro parameters setup.

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. 30.

10.3.2 P-value cut off: alfa

maSigPro applies, as second step, a variable selection procedure to find significant variables for each gene. This will ultimatelly 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 alfa, fig. 30.

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. 30.

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. 31.

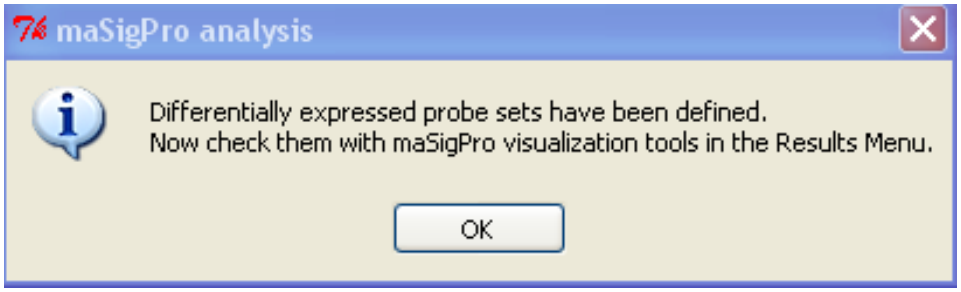


Figure 31: End of maSigPro calculation.

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 maSipPro 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. 32 .
2. Expression profiles saved in a pdf file, figs. 33, 34.
3. Tab delimited files with the probe sets found differentially expressed in each of the experimental conditions.

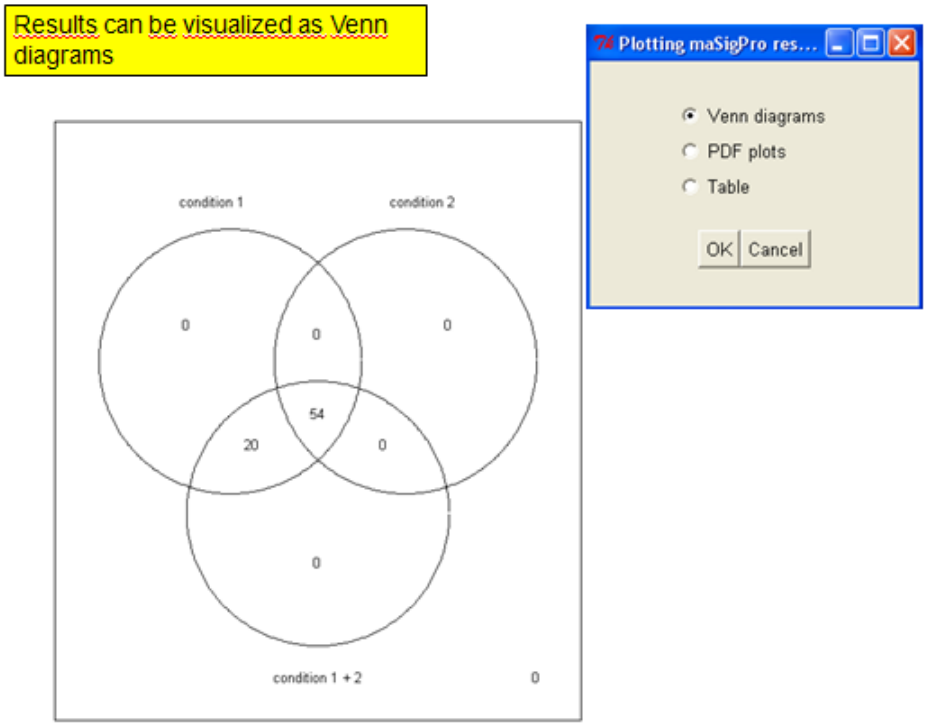


Figure 32: maSigPro Venn diagrams output.

11 Permutation statistics

The permutation statistics menu, fig. 35, 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.

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. 36.

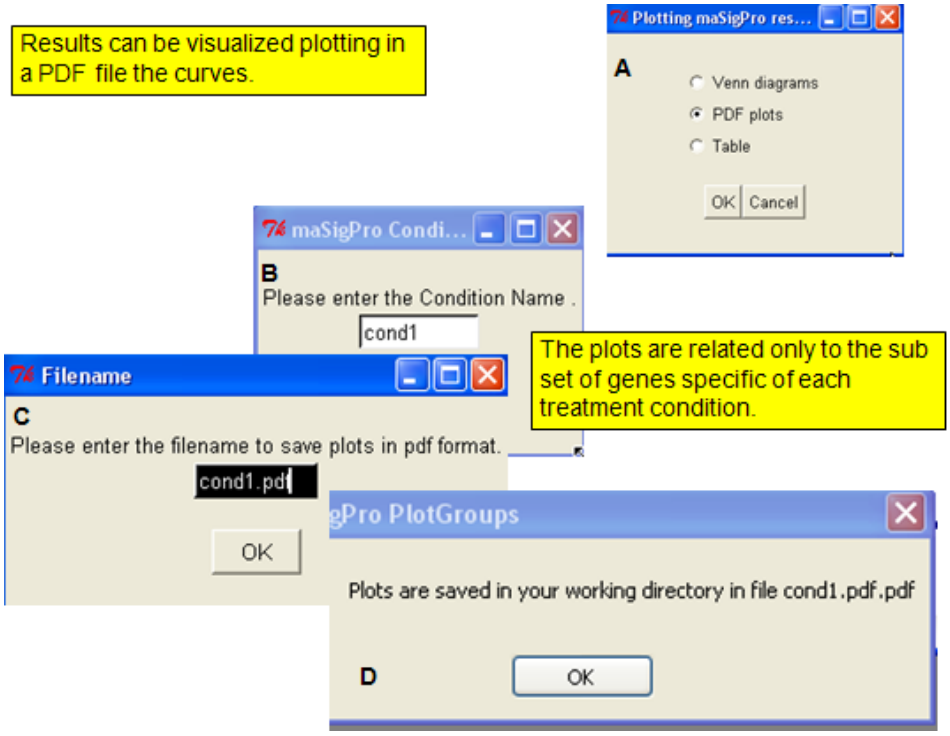


Figure 33: Selecting the experimental condition to be used to profiles plotting.

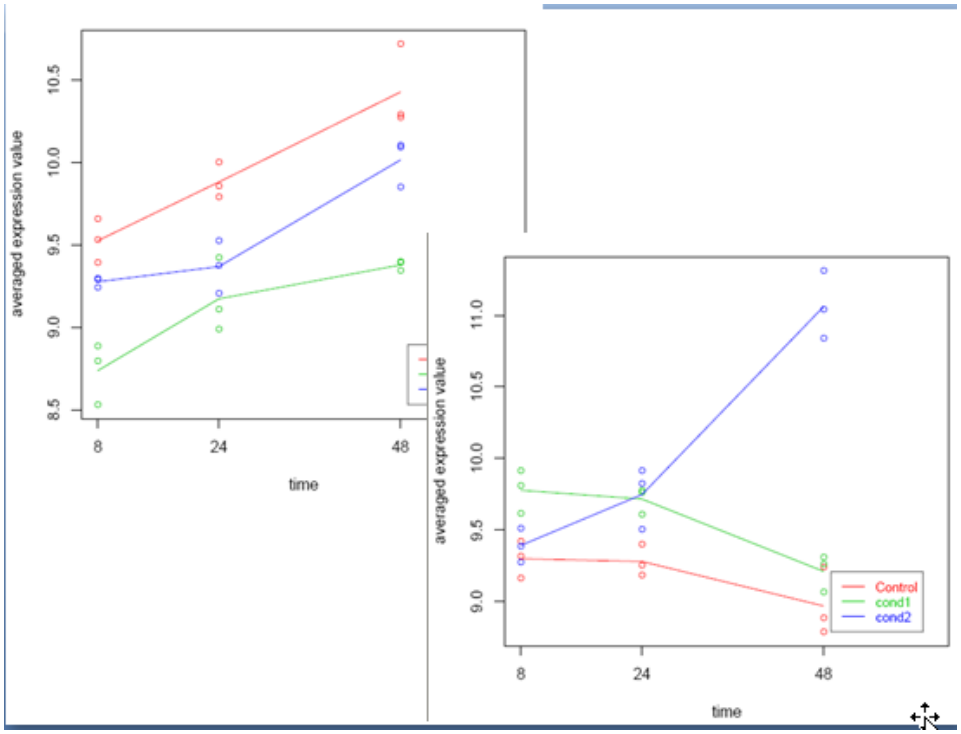


Figure 34: An example of profiles plotting.

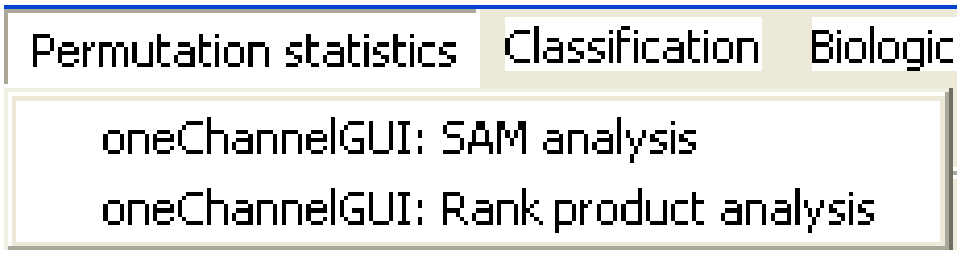
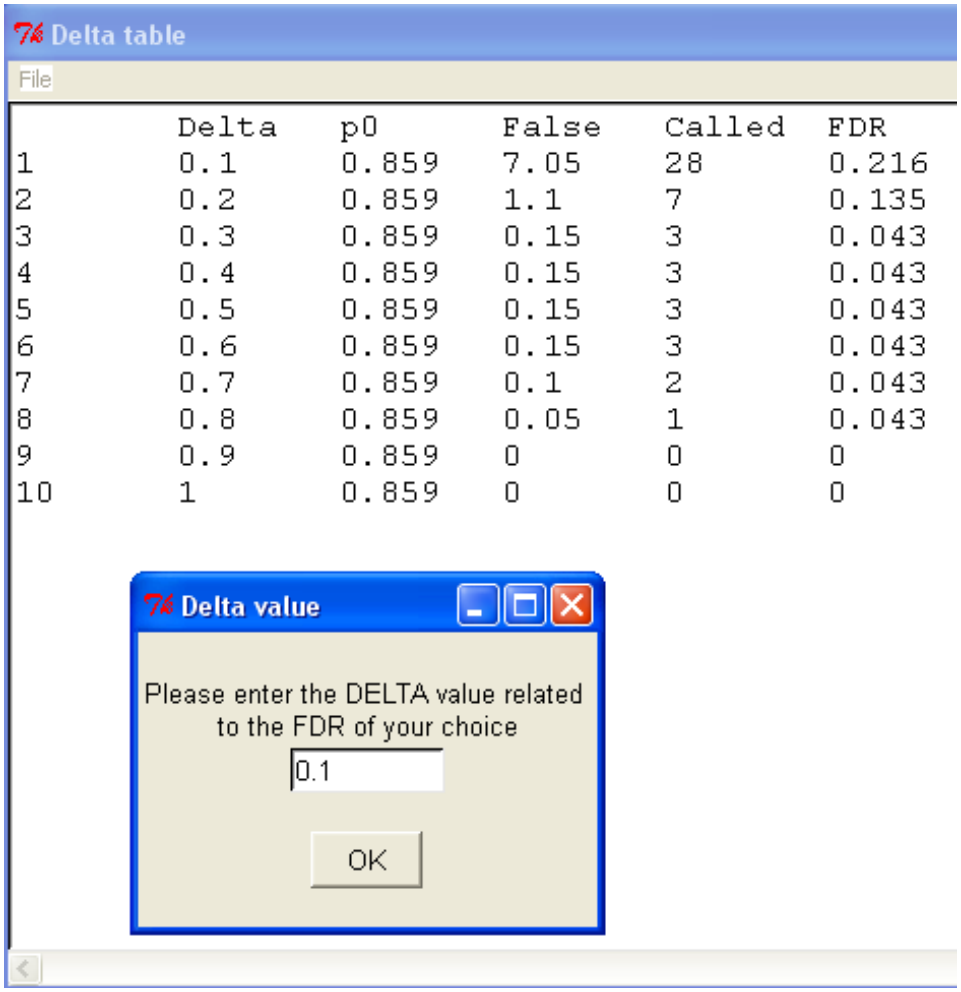


Figure 35: Permutation statistics menu.



	Delta	p0	False	Called	FDR
1	0.1	0.859	7.05	28	0.216
2	0.2	0.859	1.1	7	0.135
3	0.3	0.859	0.15	3	0.043
4	0.4	0.859	0.15	3	0.043
5	0.5	0.859	0.15	3	0.043
6	0.6	0.859	0.15	3	0.043
7	0.7	0.859	0.1	2	0.043
8	0.8	0.859	0.05	1	0.043
9	0.9	0.859	0	0	0
10	1	0.859	0	0	0

Figure 36: DELTA table and DELTA value selection module.

Siggenes output for differentially expressed genes, given the selected DELTA value, will be shown in the main R window, fig. 37, togheter with a absolute log2(FC) selection module, fig. 37.

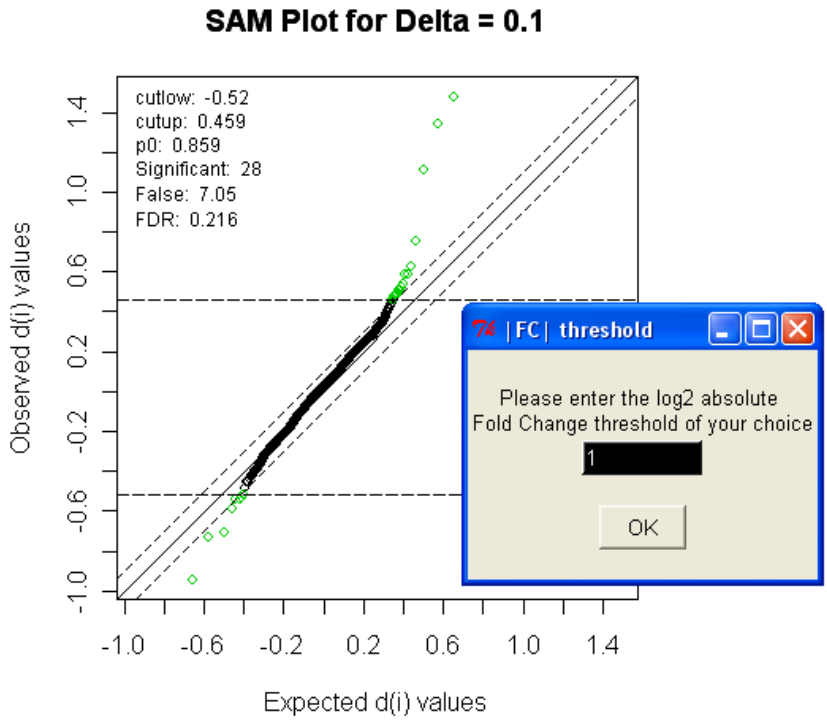


Figure 37: SAM results given at specific user defined DELTA value and the absolute log2(fc) selection mask.

The fold change filters allows the selection, within the SAM significant probe sets, of those grater than a user defined threshold. Subsequently, the differentially expressed genes will be shown, fig. 38, and the user will decide if they should 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. 39.

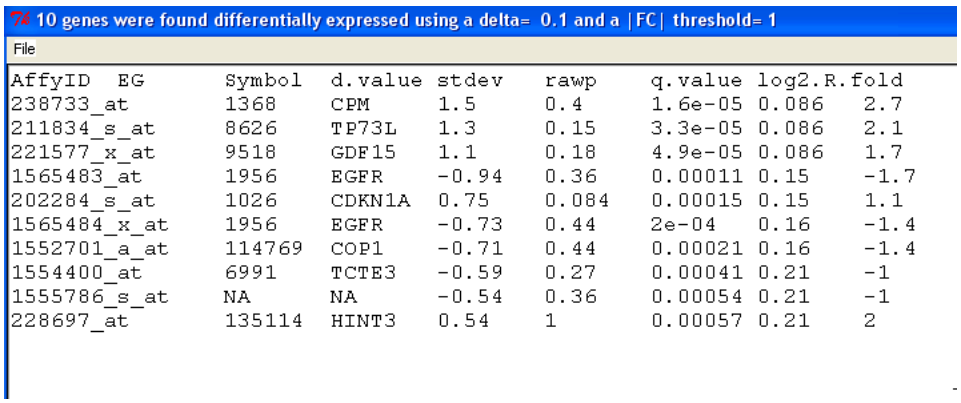


Figure 38: Differentially expressed probe sets to be saved.

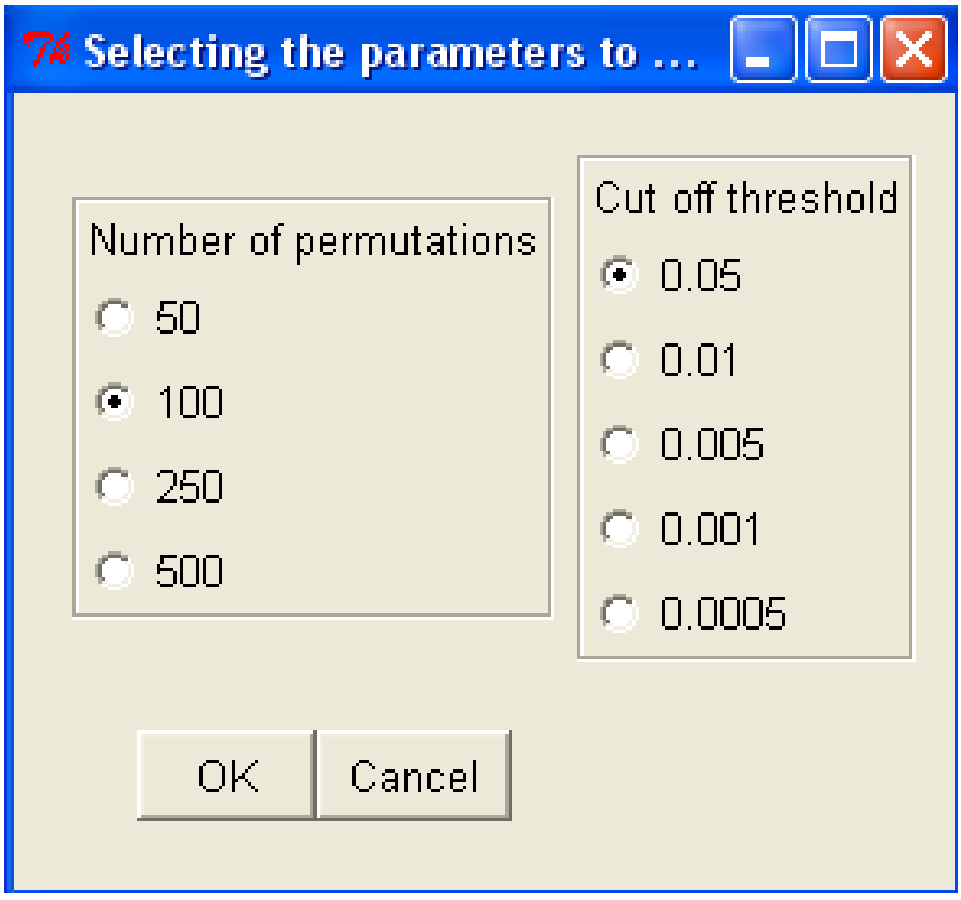


Figure 39: RankProd selection parameters mask.

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.

Name	*FileName*	*Target*
mC1	*M1.CEL*	*0_1*
mC2	*M4.CEL*	*0_1*
mC3	*M7.CEL*	*0_1*
mE1	*M3.CEL*	*0_2*
mE2	*M6.CEL*	*0_2*
mE3	*M9.CEL*	*1_1*
mI1	*M2.CEL*	*1_1*
mI2	*M5.CEL*	*1_2*
mI3	*M8.CEL*	*1_2*

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

12 Classification

This module, fig. 40, 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.

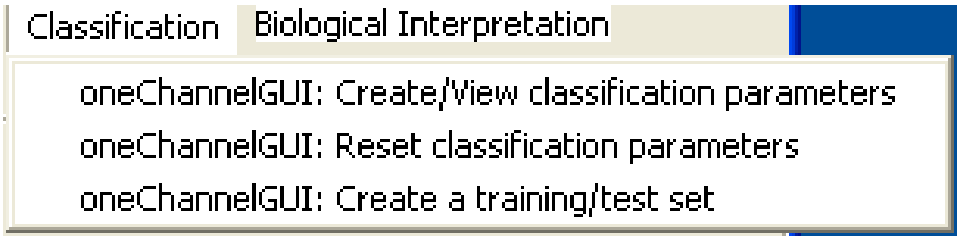


Figure 40: Classification menu.

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. 41. Subsequently, the user could decide to divide the data set in a training (2/3) and a

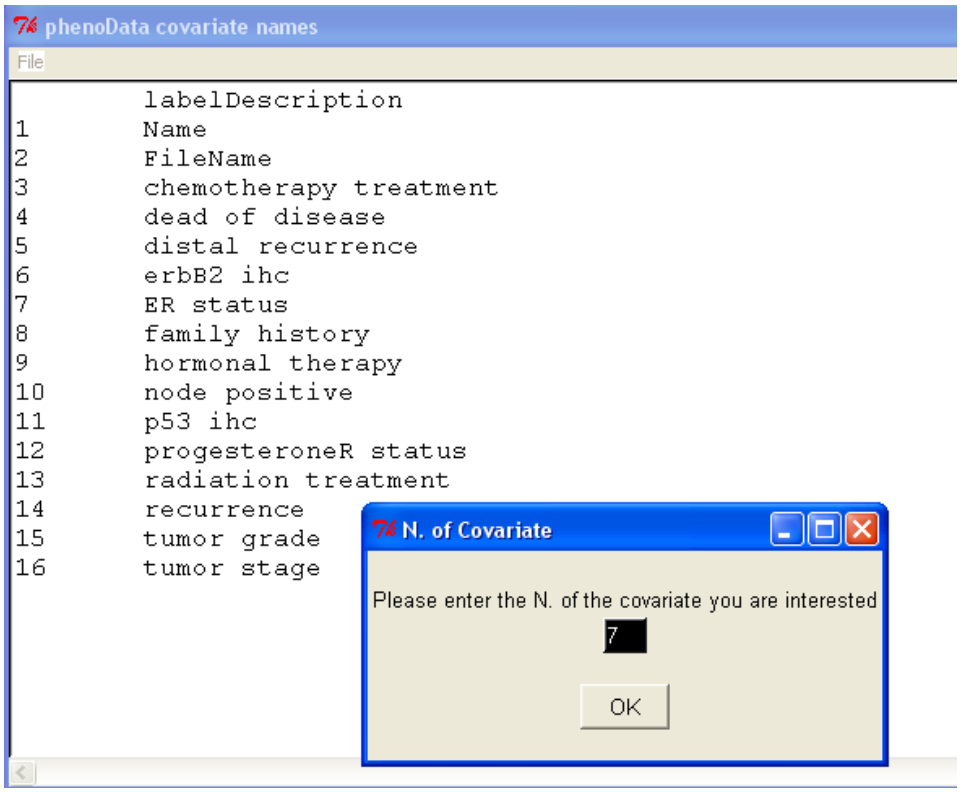


Figure 41: Selecting the classification parameter.

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. 42.

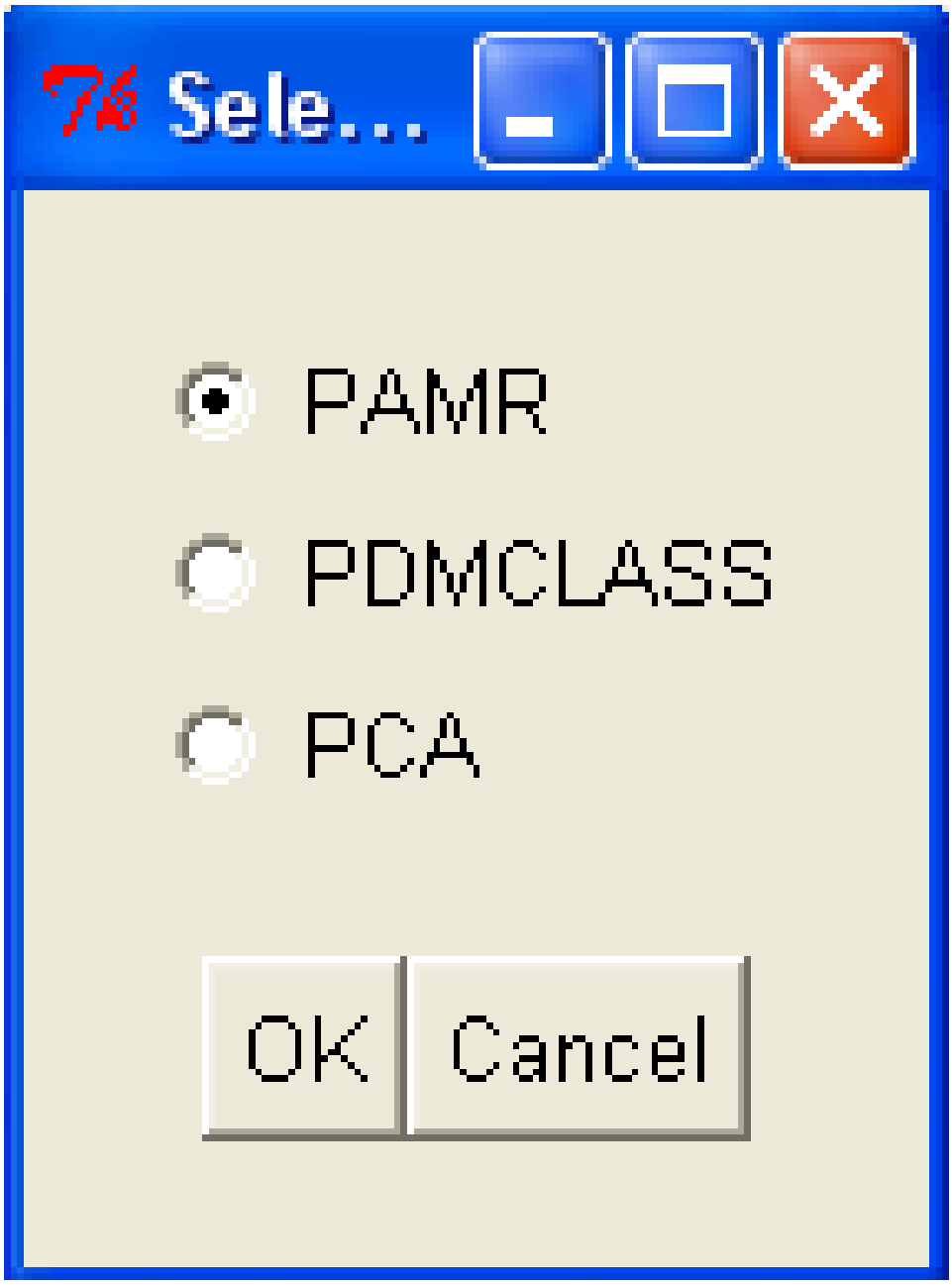


Figure 42: Classification methods selection mask.

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. 43, and shown in the main R window. Then, user defines a shrinking

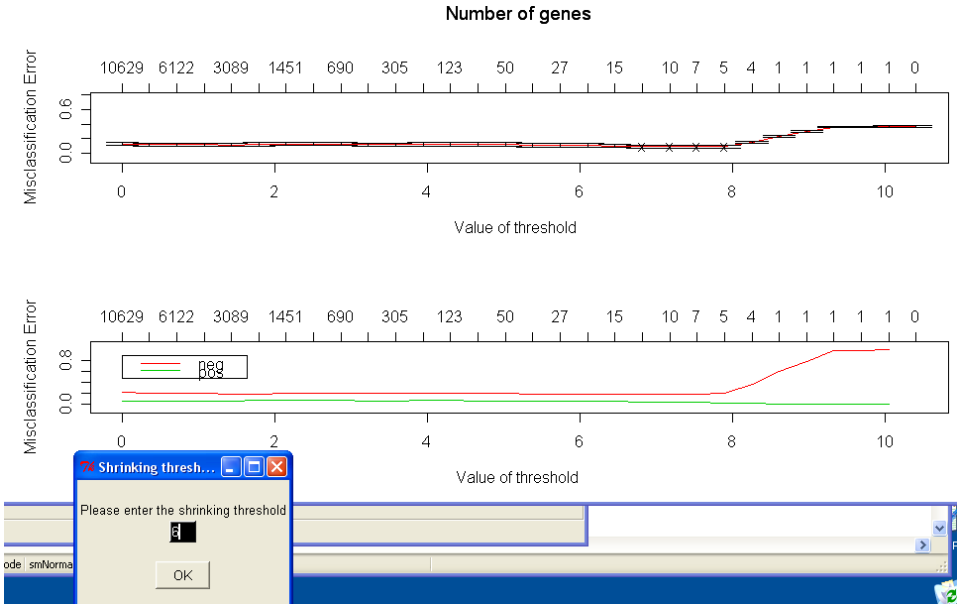


Figure 43: Cross-validated misclassification error curves

threshold and if the number of selected probe sets is below 50 the centroids will be plotted, fig. 44.

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. 45.

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

```
     neg pos Class Error rate
neg   23   5         0.1785714
pos    2  48         0.0400000
```

If the results are satisfying user can save the probe sets defined by this analysis, fig. 46.

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. 47.

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. 48.

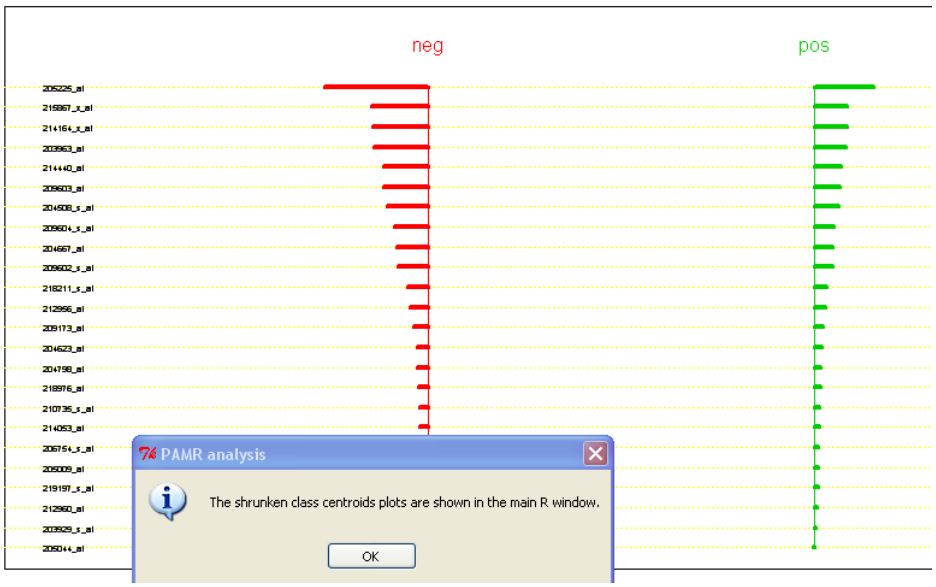


Figure 44: Shrunken class centroids.

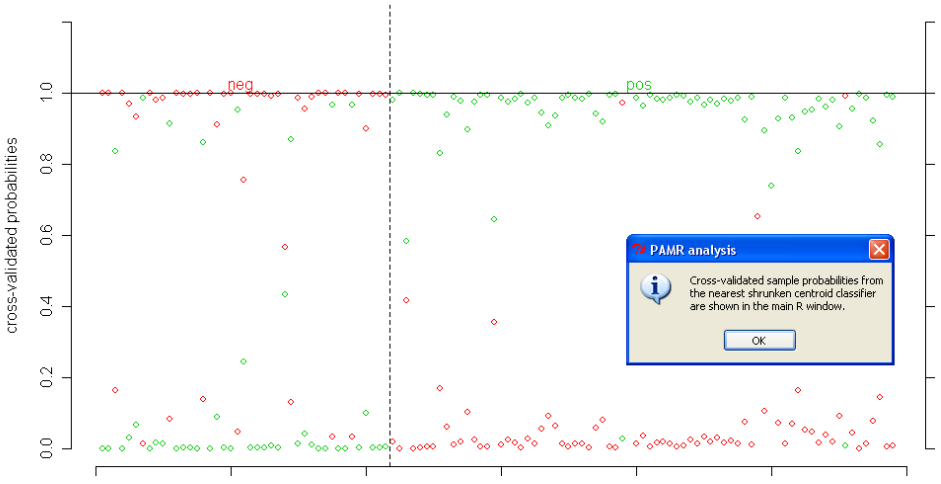


Figure 45: Cross-validated sample probabilities.

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 46: Probe sets to be use as classifier.

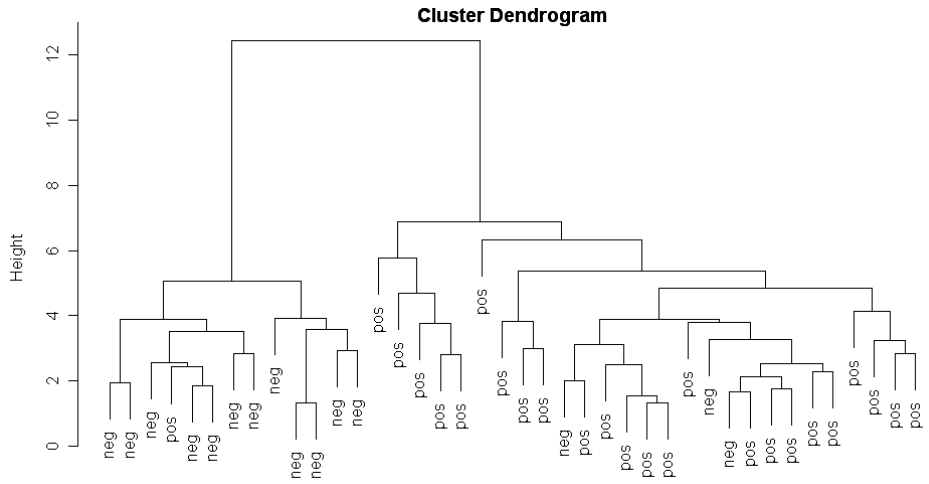


Figure 47: Testing the efficacy of the classifier on the test set by HCL.

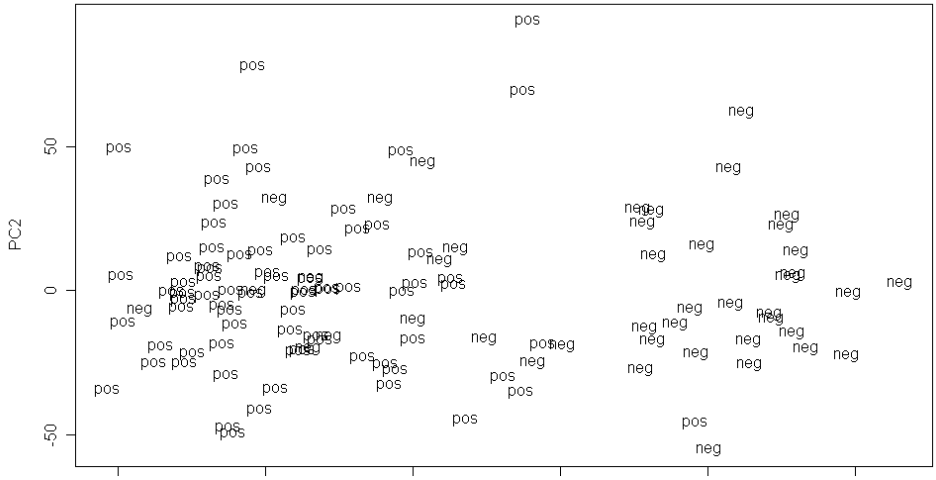


Figure 48: 1st and 2nd principal components space.

12.0.6 PDMCLASS

The PDMCLASS module allows the selection of different type of classification procedures, fig. 49.

The analysis will produce a numerical output of the effecacy 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 gratest 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:

```
        pos vs neg
209604_s_at        1
202088_at       0.92
218807_at       0.8
211430_s_at       0.56
205081_at       0.48
```

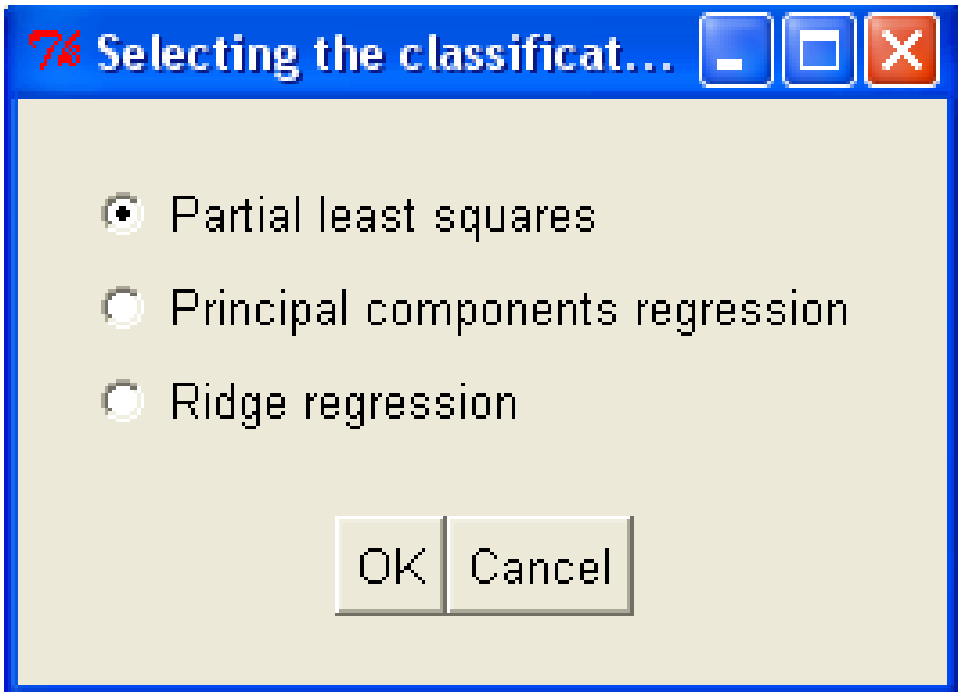


Figure 49: PDMCLASS methods selection mask.

```
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. Testing the efficacy of the selected probe sets in the test set it is not implemented, yet.

13 Biological Interpretation

This section gives a graphical interface to the GOstats package and it allows the preparation of template A for IPA analysis on `http://www.ingenuity.com`, fig. 50. It also allows very basic meta-analysis using the metaArray package.

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

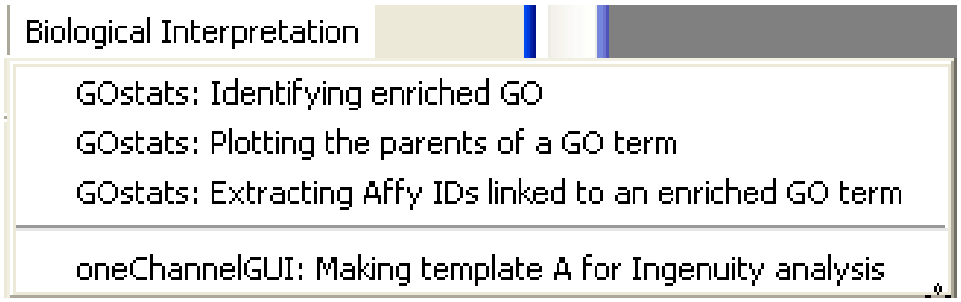


Figure 50: Biological interpretation menu.

we use the annotation informations embedded in oneChannelGUI and link the accession ids available in this annotation to Entrez Gene ids using the humanLLMappings, mouseLLMapppings 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. 51.

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
```

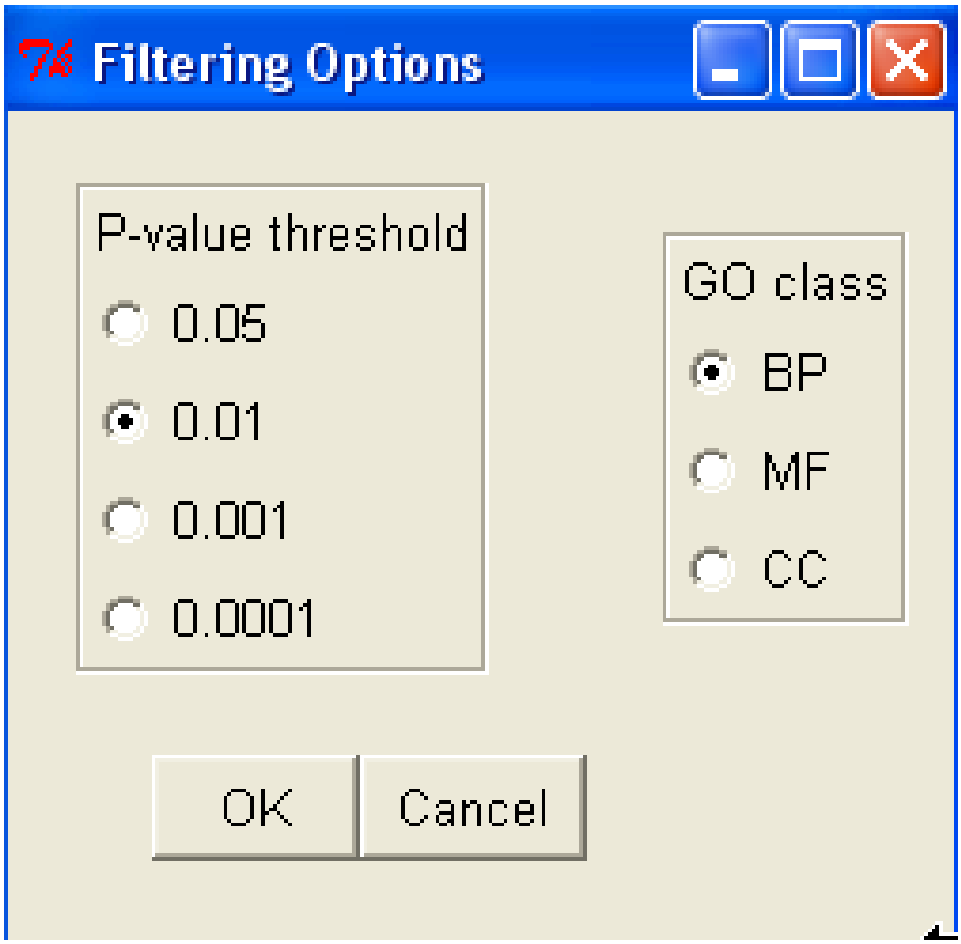


Figure 51: GO terms enrichment parameters selection mask.

The analysis could require quite a lot of RAM and when it is finished a message summarizing the results pops up, fig. 52.

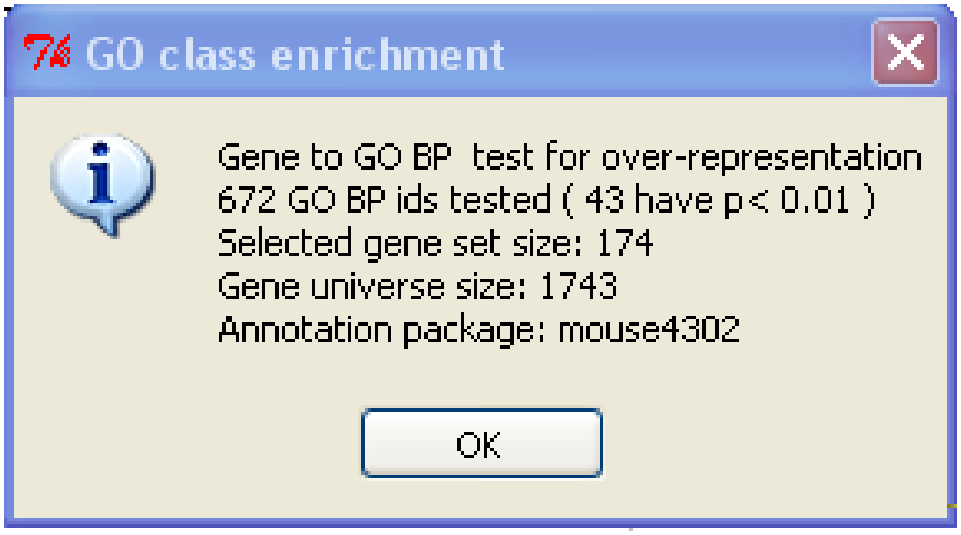


Figure 52: GO enrichment results summary message.

A table with the enriched GO terms will be then shown and it could be saved as tab delimited file, fig. 53.

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. 54.

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 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. 55.

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. 54, 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. 56. The output for exon arrays will be instead a tab delimited file.

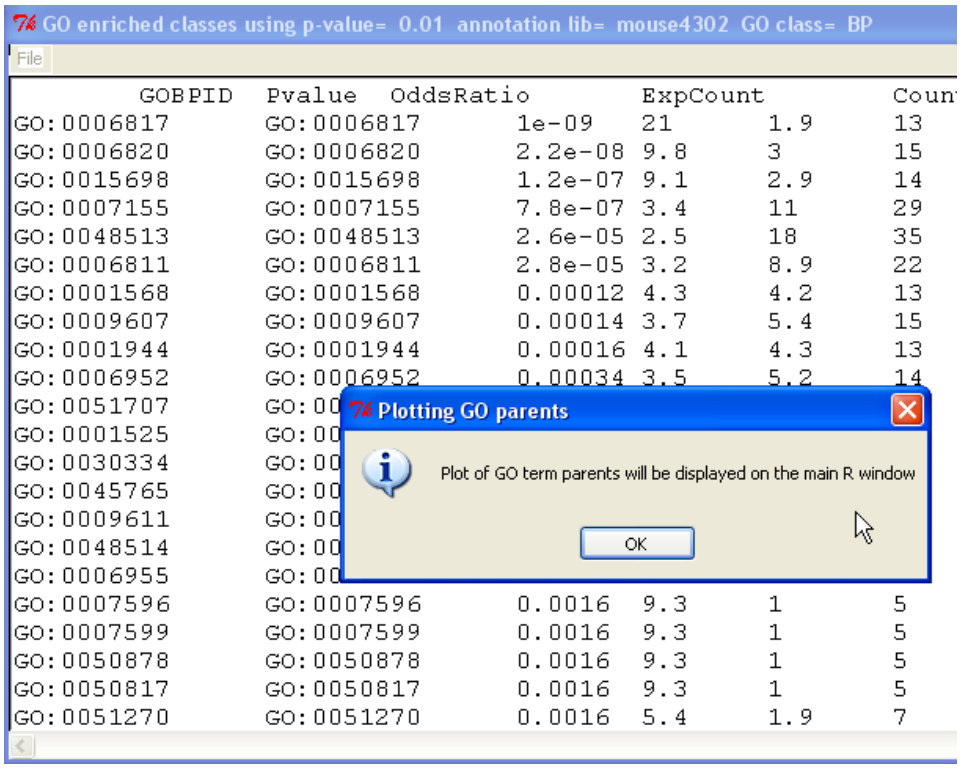


Figure 53: Enriched GO terms table.

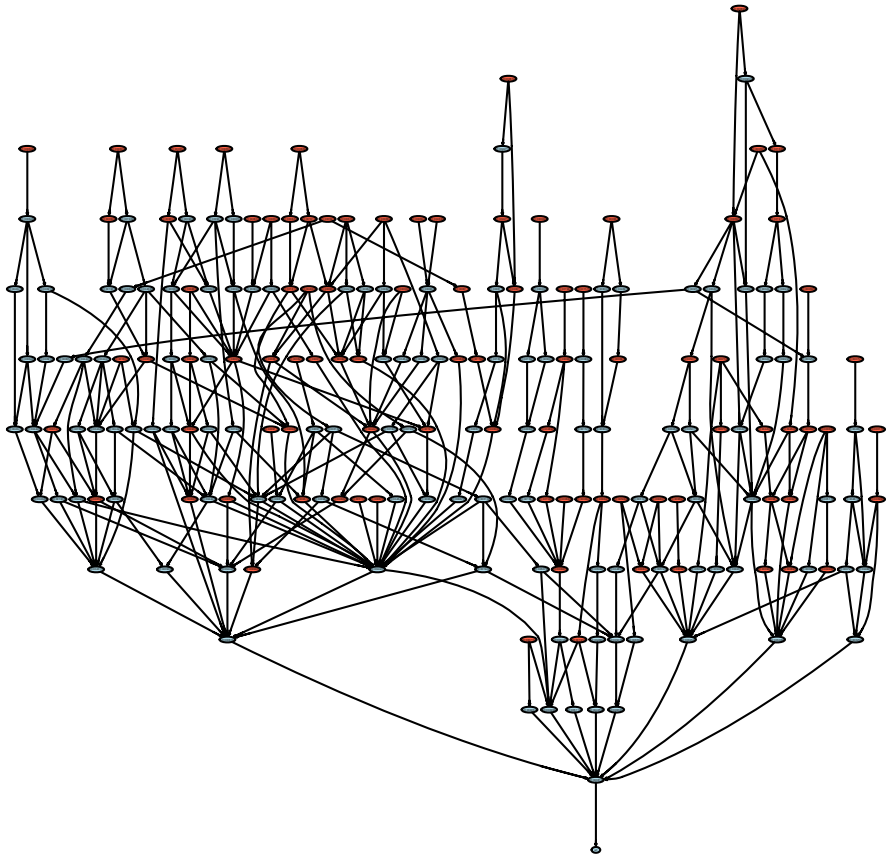


Figure 54: Relations between enriched GO terms. Enriched GO terms, red, others, light blue.

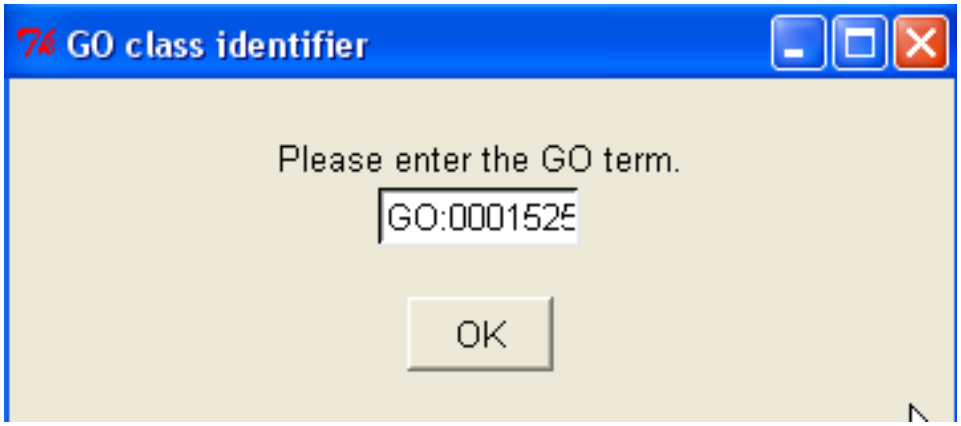


Figure 55: Plotting GO term parents.

Differentially expressed probe sets in GO:0001525

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

Figure 56: Annotation file for a subset of differentially expressed probe sets linked to GO:0001525 BP enriched term.

13.2 oneChannelGUI: Making template A for Ingenuity analysis

This function reorganizes the output derived by any of the tables generated by limma/siggenes/RankProd to generate a template A to be uploaded to Ingenuity database. The function initially requests to select the type of top table to be used, fig. 57.

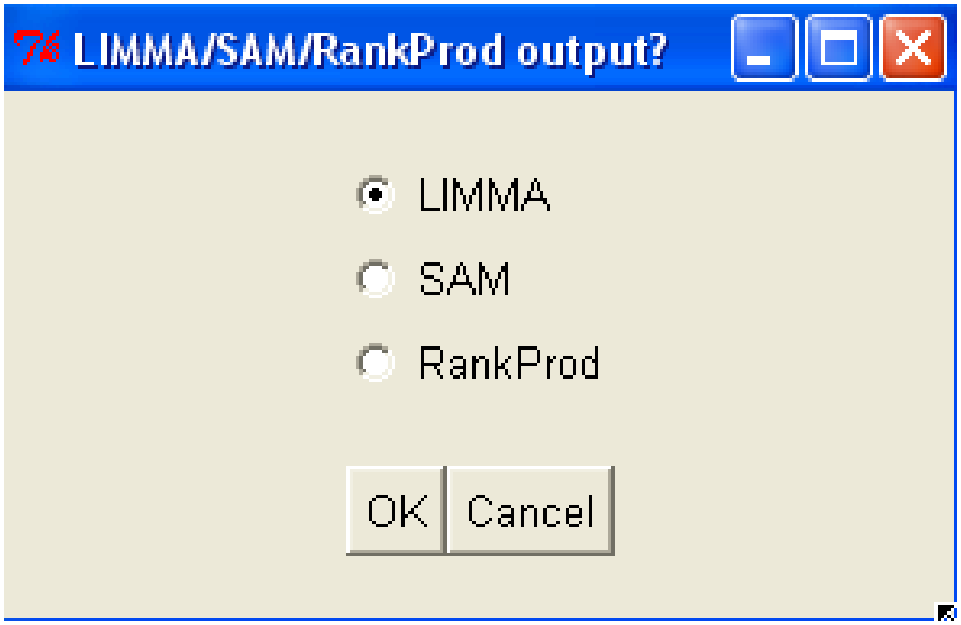


Figure 57: Differentially expressed probe sets tables selection mask.

The output templateA table will have the following structure, fig. 58.

13.3 Biological Interpretation for EXON 1.0 ST arrays

In case EXON 1.0 arrays are loaded inte Biological Interpretation menu contains the following functions: *oneChannelGUI: Attaching ACC and Entrez Gene IDs to Probe set IDs (EXON 1.0 ST)* which allows to associate EG ids to gene-level probe sets. *oneChannelGUI: Mapping exon level prober sets to EG exons* which associates the statistical and expression data produced by a oneChannleGUI exon-level analysis to the exonic structure of Entrez Gene ID. This function uses biomaRt to retrieve the sequence of EG exons. RRE database (`http://www6.unito.it/RRE/EN/`) is instead used to retrieve the exon-level target sequences. Any exon-level probe set id to be associated to the EG exonic sequence need to be a perfect matching substring of the exon. In the otehr case no exon is associated to the probe set. Furthermore, the conservation of each exon over the various isoform is defined. The data frame containing these information can be saved using the function *oneChannelGUI: Exporting Gene exprs and/or Exon/SI/MiDAS/RP*

	A	B	C	D	E	F
1	Gene/Protein ID	Fold change	P value for subsetting	True P value	Absent	Override
2	1435436_at	-4.4780142	0	0.00056044		
3	1428535_at	-4.3628274	0	0.000165244		
4	1415806_at	-3.0301886	0	6.42E-09		
5	1419271_at	-1.878537	0	2.45E-07		
6	1440830_at	-1.6899624	0	0.001729493		
7	1457012_at	-1.6839344	0	0.023130807		
8	1427320_at	-1.6138972	0	1.43E-06		
9	1435092_at	-1.5109728	0	0.000523988		
10	1454709_at	-1.509256	0	0.002709374		
11	1421840_at	-1.4068938	0	3.04E-07		
288	1454745_at	4.4606454	0	0.003208596		
289	1454866_s_at	4.4617282	0	0.003559589		
290	1438363_at	4.6634546	0	0.001295888		
291	1434893_at	4.6665508	0	1.70E-06		
292	1454830_at	4.7422502	0	0.00333454		
293	1436996_x_at	4.9635932	0	1.82E-06		
294	1452163_at	5.4465946	0	2.38E-05		
295	1415685_at		1	1		
296	1415689_s_at		1	1		
297	1415728_at		1	1		
298	1415729_at		1	1		
299	1415746_at		1	1		
300	1415757_at		1	1		
301	1415784_at		1	1		
302	1415793_at		1	1		

Figure 58: Template A structure, The first column contains the Gene/Protein IDs the second column log2(fc) only for the set of probe sets considered differentially expressed. The third column has 0 for all the differentially expressed probe sets and 1 for the rest. The 4th column has the true p-value only for the differentially expressed probe set, the rest is set to 1.

data/elevel IDs to exon EGs The column referring to exon conservation is called conserved.exons. If the value in this column is 1 the exon is conserved over all isoforms. If it is lower than 1 is conserved only in some of the isoforms, fig. 59.

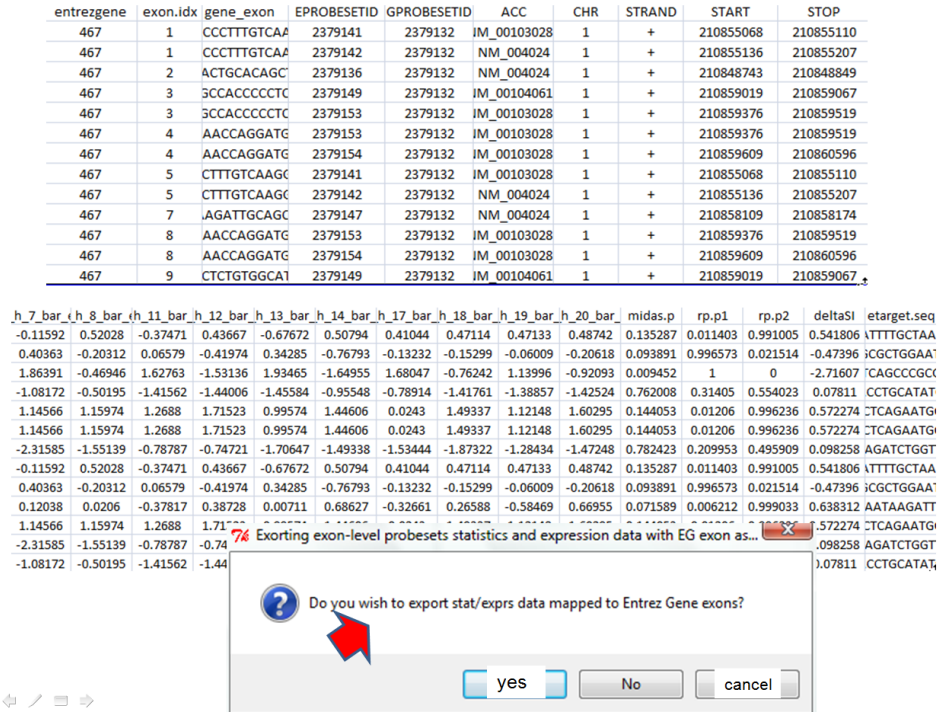


Figure 59: Statistical and expression data malled on EG exons.

The previous function also export a file where the structure of alternative spliced isoforms is described in a tab delimite format, fig. 60.

14 Biological Interpretation DEVEL ONLY

This menu, in the devel version, gives also access to some meta-analysis tools, fig. 61.

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.

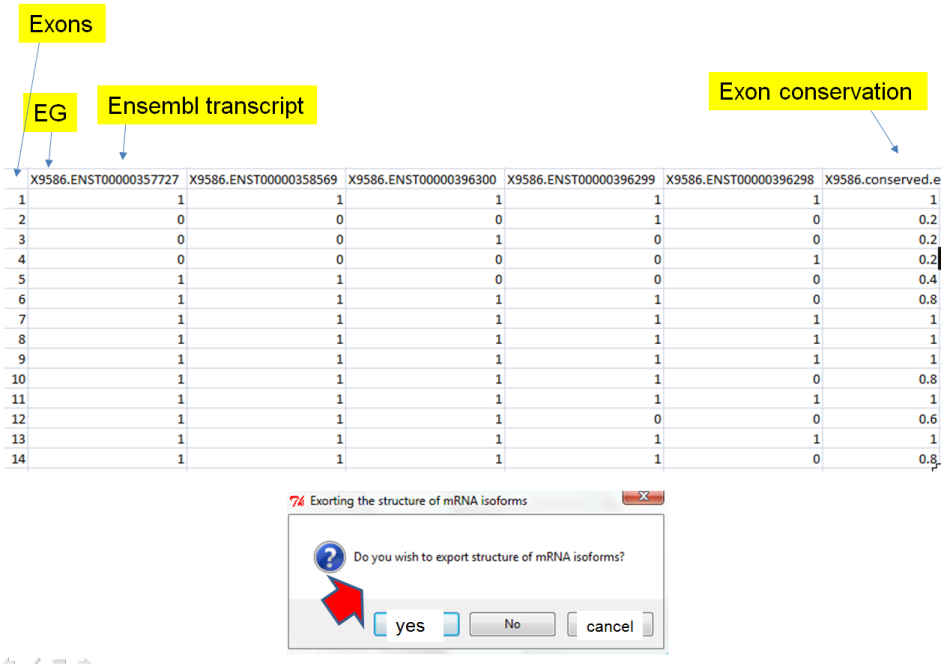


Figure 60: Structure of alternative spliced isoforms.

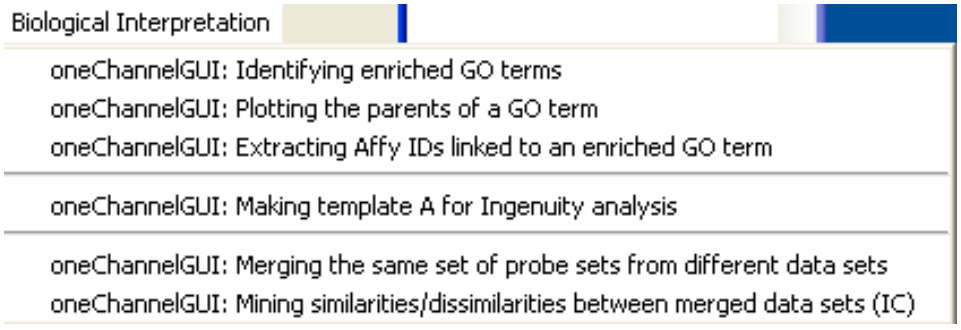


Figure 61: Biological Interpretation DEVEL version.

15 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: Extract a column from a tab delimited file* allows the extraction of any of the colums of a tab delimited file. This function is particularly useful to generate probeset ids list to be used for Venn diagram rapresentation. 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. 62 yellow, sharing the same header of the list of values, fig. 62.

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

symbols.txt - Blocc

File Modifica Formato

```
Symbol
Dct
6030426L16Rik
Slc6a13
Lxn
Apod
Xlkd1
Serpinf1
Tbx18
Lox
Pdgfra
Bgn
Dcn
Tubb6
Vtn
Anxa2
Arhgap29
Vtn
Tgm2
Foxc1
Svil
Dab2
```

Figure 62: 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* allows the user to define a folder where apt tools were installed. 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 *oneChannelGUI: buildingLocalAnnotation* allows to update the internal oneChannelGUI gene-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 keept tracks of annotaiton file updating.

16 Help

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

17 Exon analysis and data mining

Once exon data are loaded the filtering menu appear slighly different, fig. 21.

This menu allows a certain number of functions to identify and visualize alternative splicing events. User should remember that exon arrays are a relatively new technology and very little is still known on their analysis. Furthermore, benchmark experiments to test the efficacy of statistical methods for alternative splicing detection are not available, yet. Therefore, this module will be subjected to various upgrading and improvement during the next years. The part related to loading gene/exon level data is described in the File menu chapter. If APT tools are used to calculate probe set intensities in oneChannelGUI will be available gene level expression data in Normalized Affy Data, exon level expression data in Normalized Exon data and, if selected, DABG p-values. The functions actually available for exon analysis are summarised in fig. 63.

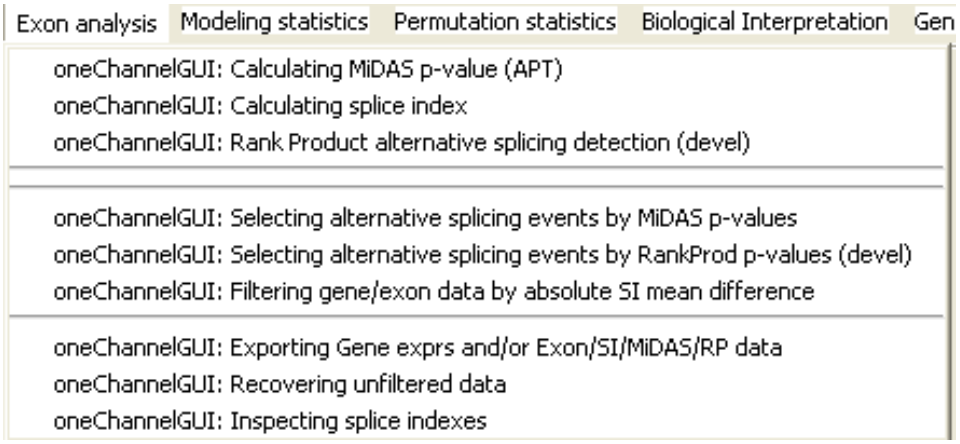


Figure 63: Exon menu.

Splice Index (SI), which represents the exon expression normalized with respect to the transcript expression, can be calculated with *oneChannelGUI: Calculating splice index*. For a two group experiment the function *oneChannelGUI: Calculating MiDAS p-value (APT)* uses APT tools to calculate MiDAS p-values for the difference between SIs in the two conditions, i.e. alternative splicing events. It is possible to subset gene/exon

level data on the basis of a MiDAS p-value threshold using the function *oneChannelGUI: Selecting alternative splicing events by MiDAS p-values*. In the devel version of oneChannelGUI we have also applied the rank product method (RankProd package) *oneChannelGUI: Rank Product alternative splicing detection (devel)* to detect significant differences between SI or exon-level log2(intensities) in two experimental conditions, i.e. alternative splicing events. Rank Product is a non-parametric statistic that detects items that are consistently highly ranked in a number of lists. It is based on the assumption that under the null hypothesis that the order of all items is random the probability of finding a specific item among the top r of n items in a list is $p = \frac{r}{n}$. Multiplying these probabilities leads to the definition of the rank product $RP = \prod_i \frac{r_i}{n_i}$, where r_i is the rank of the item in the i-th list and n_i is the total number of items in the i-th list. The smaller the RP value, the smaller the probability that the observed placement of the item at the top of the lists is due to chance. Due to performance reasons on windows based computers, the number of random permutations is fixed to 100, a menu to select the number of permutations will be implemented soon. At the end of the analysis p-values of class 1 < class2 and p-values of class 1 > class2 and average SI difference histograms are shown in the main R window.

It is possible to subset gene/exon level data on the basis of rank product results using the function *oneChannelGUI: Selecting alternative splicing events by RankProd p-values (devel)*. Since, benchmark experiments to test the efficacy of alternative splicing events are not yet available, we cannot indicate how effectives are, the methods actually implemented in oneChannelGUI, for the detection of alternative splicing events. It is also possible to filter data on the basis of the average mean or min SI difference with the function *oneChannelGUI: Filtering gene/exon data by absolute SI mean or min difference*, and to inspect a sub set of putative alternative splicing events, fig. 64, with the function *oneChannelGUI: Inspecting splice indexes*, fig. 63.

It is also possible to filter exon data integrating midas p-values with RP p-values and average mean SI difference. This option is given by the function *oneChannelGUI: Selecting alternative splicing by RP/MiDAS p-values/average mean SI difference (devel)* Sin no correction for statistical type I error is given for MiDAS we decided to use the integration of two statistical tests based on different approaches to reduce statistical type I errors. Furthermore, in this filter is integrated also the possibility to subset data on the basis of a average mean SI difference threshold. Visualization of the splicing events using one gene-level probe set at a time is possible with the function *oneChannelGUI: Inspecting splice indexes of one glevel probe set*. The output, fig. 64, represents a plot of -log2(MiDAS p-values) plotted versus the exon-level probe set index, i.e. 1 for exon 1, 2 for exon 2 etc.. A plot of the -log2(p-value) consistency between MiDAS and RP, i.e. consistency is given by the fact that both methods give a p-value < 0.05. A plot where SI or exon-level log2(intensity) are plotted versus the exon-level probe set index. The consistent splicing events are indicated by a yellow bar.

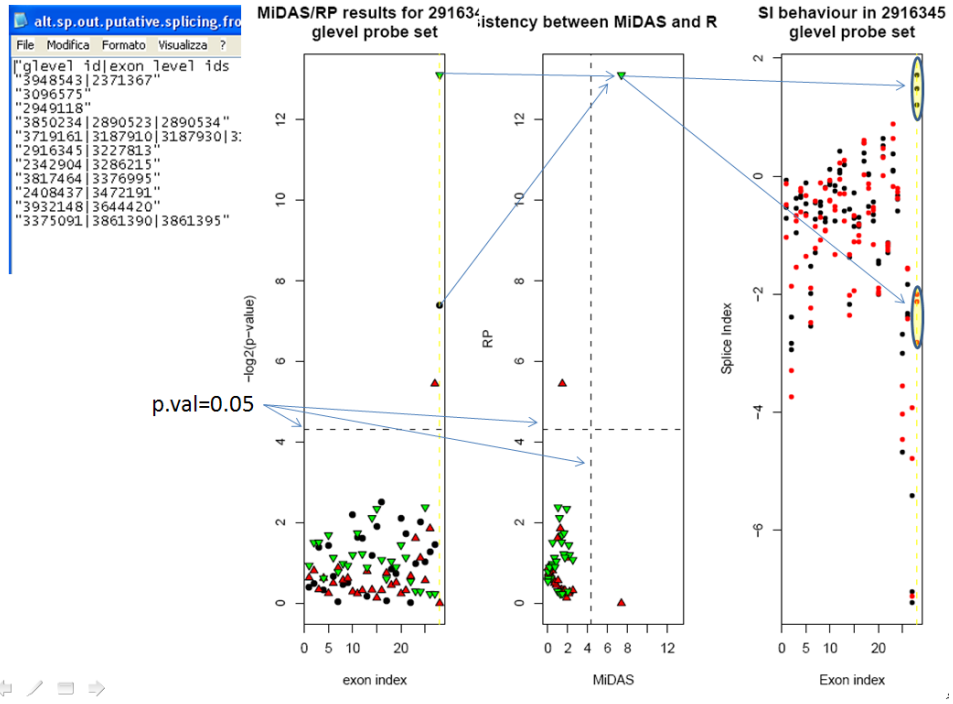


Figure 64: Example of the output of the putative alternative splicing inspection. The output is made of a tab delimited file where glevel probe sets associated to elevel probe sets and of a pdf file where each page is made of a plot of MiDAS/RP p-values with respect to exon index (black dot MiDAS, red triangle and green triangle RP). The horizontal black dashed line indicates a p-value of 0.05. The vertical yellow dashed line indicates a condition in which both MiDAS and RP p-values are below 0.05 value. In the second plot, p-values of RP are plotted versus MiDAS p-values. Those p-values that below 0.05 both in RP and MiDAS will appear in the upper right rectangle. In the third/fourth plot, it is shown the behaviour of splice indexes or exon-level log2(intensity) with respect to exon indexes. The vertical yellow dashed lines indicate those exon-level log2(intensity)/SI associated to MiDAS and RP p-values below 0.05 value.

18 Example of Exon array analysis

Data described in this example are produced using the data human set: `http://www.bioinformatica.unito.it/downloads/exon.zip`. As first step the function *New* calculates gene/exon level probe set summaries. In this example the 3 liver and 3 heart .CEL files of the Affymetrix tissue human set were used. To detect alternative splicing a certain number of prefiltering steps are needed, fig. 65.

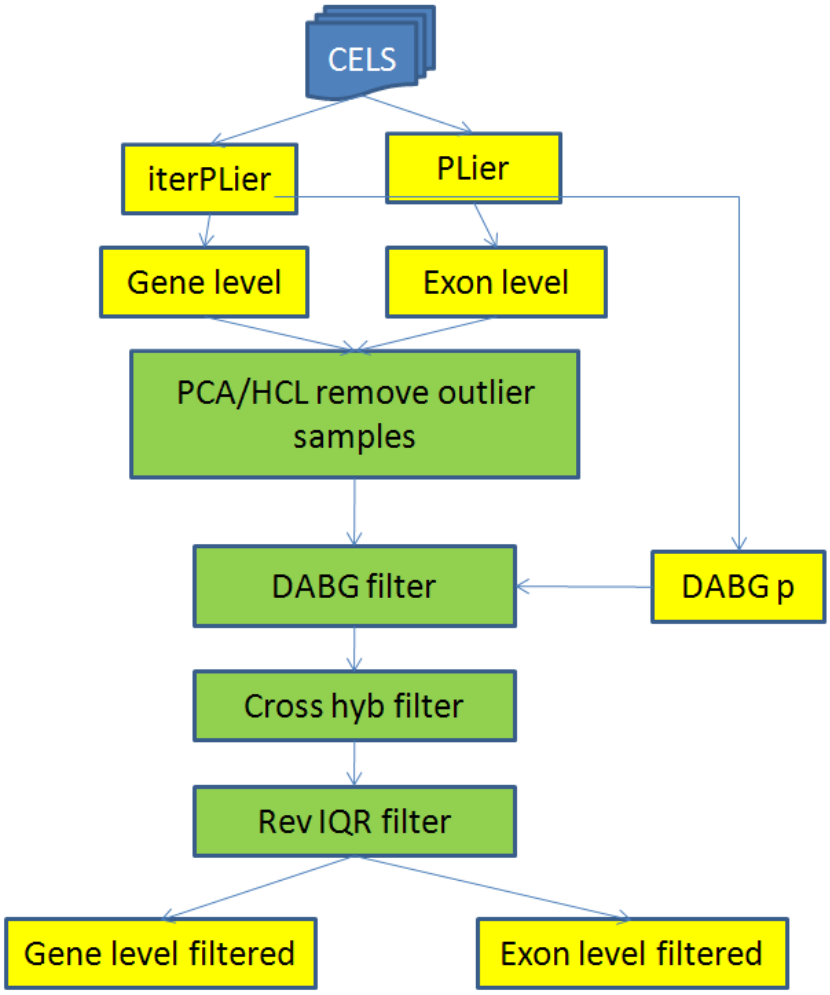


Figure 65: prefiltering steps for alternative splicing analysis.

In this example gene level probe sets are calculated with iterplier and exon level probe sets with plier, as suggested by Affymetrix. The density plot of the gene and exon level data can be evaluated with the function *Gene/Exon Intensity Histogram* To see both plots at the same time, as in fig. 66, you need to type in the main R window:

```
> par(mfrow=c(1,2))
```

and subsequently apply the *Gene/Exon Intensity Histogram* command.

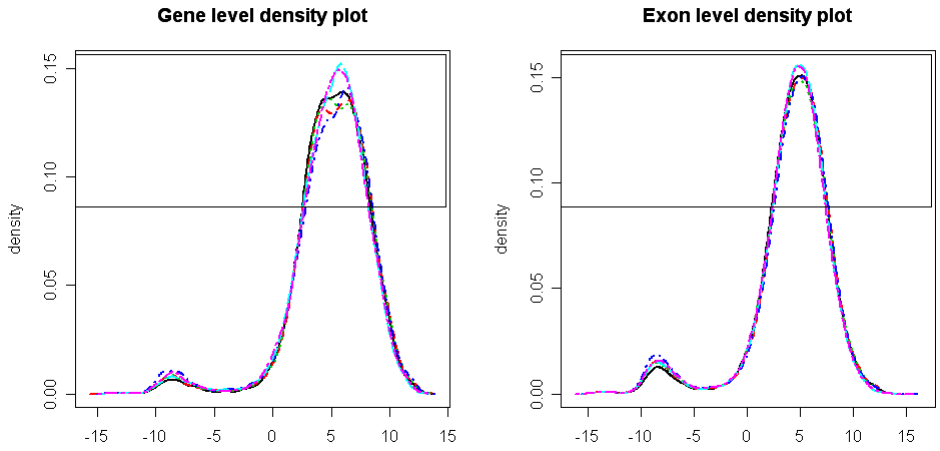


Figure 66: Iterplier gene level and plier exon level probe set distributions.

A first cleanup is part of the QC checks, since samples outliers can be detected by the function *Gene/Exon PCA/HCL*, fig. 67, which gives an idea on the homogeneity of the experimental replicates and the level of separation between the experimental groups.

If Plier/iterPLier was used for expression summaries we strongly suggest to use the function *Setting to 0 log2 intensity below 1, to be used with plier only* that brings the negative log2 values, i.e. values near 0 as intensities, to 0.

An other cleanup is made removing low intensity probe sets using the function *Filtering on DABG p-values*. The filtering threshold is a user decision. In this example we remove all probe sets that show a low intensity expression in at least half of the experiments, fig. 68.

An other clean up step is the removal of probe sets characterized by cross-hybridization. This is actually done using the crosshyb and xhyb annotations available in the Affymetrix annotation files. This filter is only available for the core exon subset, fig. 69. A description of this filter options is present in this vignette in the filtering section.

The resulting data set is made of 14630 gene level probe sets and 211038 exon level probe sets, these values can be visualized using the function *Info about the loaded data set* available in the file menu and in the filtering menu.

In our opinion it is better to keep separated the detection of gene-level differential expression with respect to exon level alternative splicing detection. Therefore, we use a prefiltering step to remove, before exon splicing analysis starts, all the transcripts which might be differentially expressed at gene-level and get back to them in a separate analysis. This is done removing probe sets characterized, at gene level, by a broad variation over samples that could be due to gene level differential expression. This filter

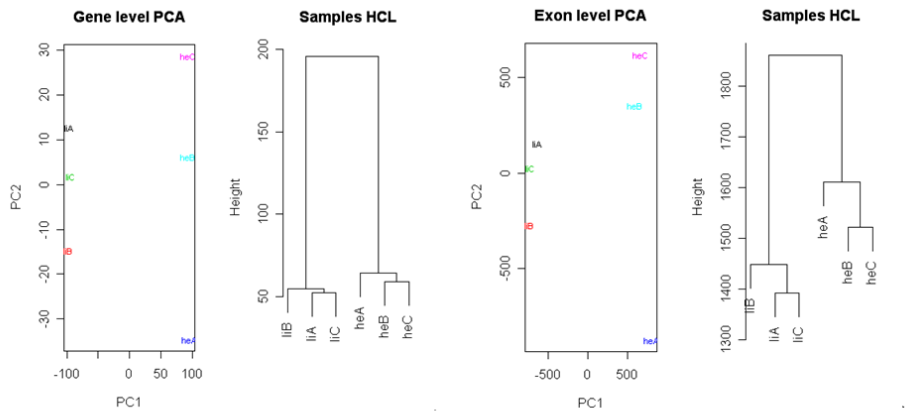


Figure 67: Gene and exon level PCA and hierarchical clustering.

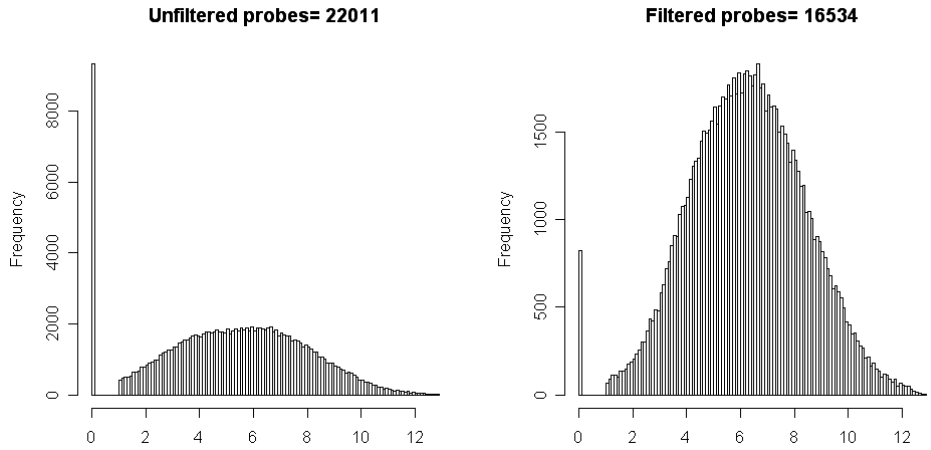


Figure 68: DABG filter half of the data set should have a DABG p-value different from 0.05.

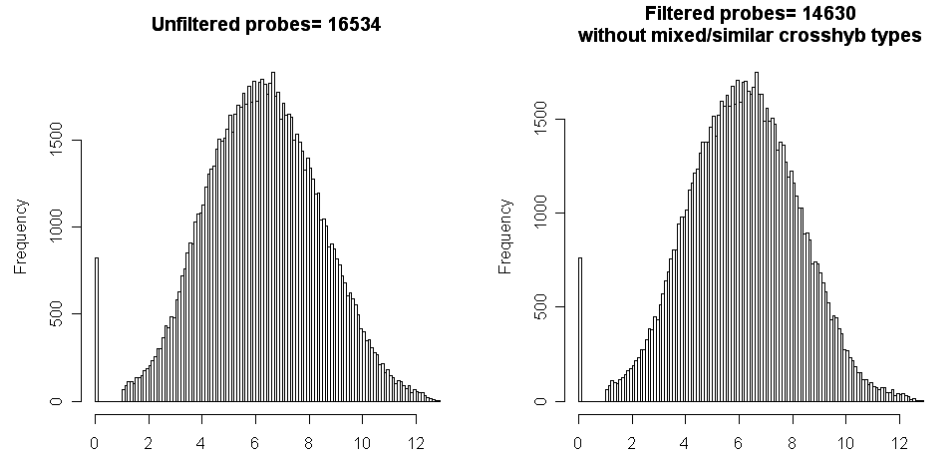


Figure 69: Cross bybridizing probe set removal for similar/mixed crosshyb type.

is a reverse implementation of the IQR filter, function *Filtering by reverse IQR (for alternative splicing analysis only!)*, fig 70.

In this example, the probe sets that are characterized by strong changes are removed using a reverse filter at IQR 0.8, fig 71.

From this filtering we reduce the gene-level probe set to 7086 probe sets and at exon level to 95091. After these filters data are ready for the calculation of putative alternative splicing p-value calculation by mean of MiDAS or RankProd.

MiDAS p-values can be calculated for the data available after the preprocessing using the function *Calculating MiDAS p-value (APT)* available in the *Exon menu*. The histogram of the p-values distribution is shown in the R window, fig. 72.

Subsequeltly also the p-value calculated with the rank product can be produced using the function *Rank Product alternative splicing detection*, this function is actually only available in the devel version of oneChannelGUI. The histograms results are shown in the main R window, fig. 73. To perform RP p-values calculation is necessary to calculate the Splice Indexes using the function *oneChannelGUI: Calculating splice index*.

The reason to use two different statistical approaches is due to the actuall lack of benchmark experiments allowing to evaluate alternative splice index method performances. In principle if an alternative splicing event is sufficiently robust it should be indentified independently by different methods and different intensity summary methods used to calculate splice index. Furthermore, the intersection of results coming from two different methods will reduce the number of type I errors. In this example we have used a weak filtering threshold for both p-values, i.e. 0.05, and we have applied it using the functions *Selecting alternative splicing events by MiDAS p-values* and *Selecting alternative splicing events by RankProd p-values (devel)*. The results can be exported using the function *Exporting Gene exprs and/or Exon/SI/MiDAS/RP data*. The intersection between the two methods at gene and exon level is shown in fig. 74.

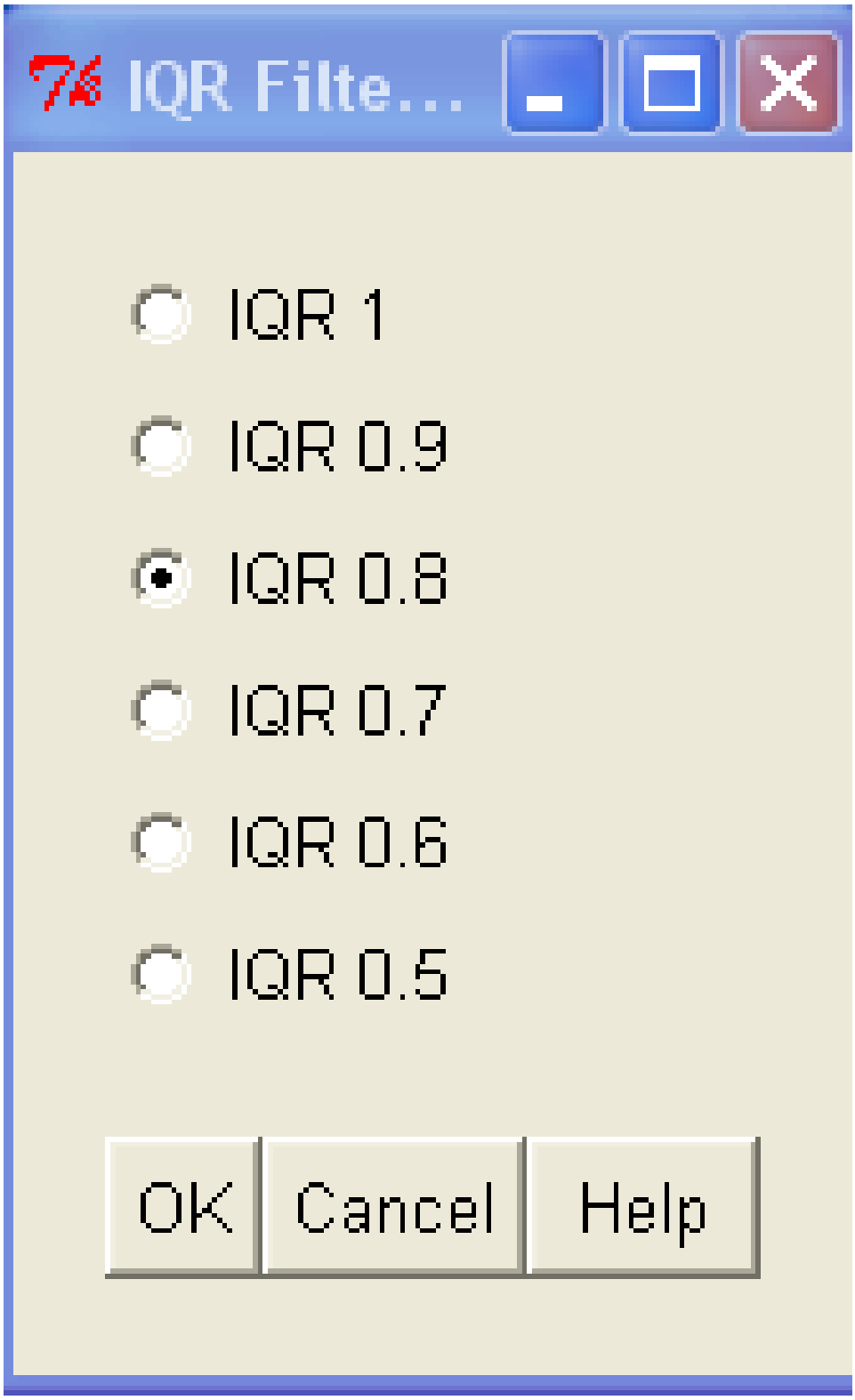


Figure 70: Reverse IQR mask.

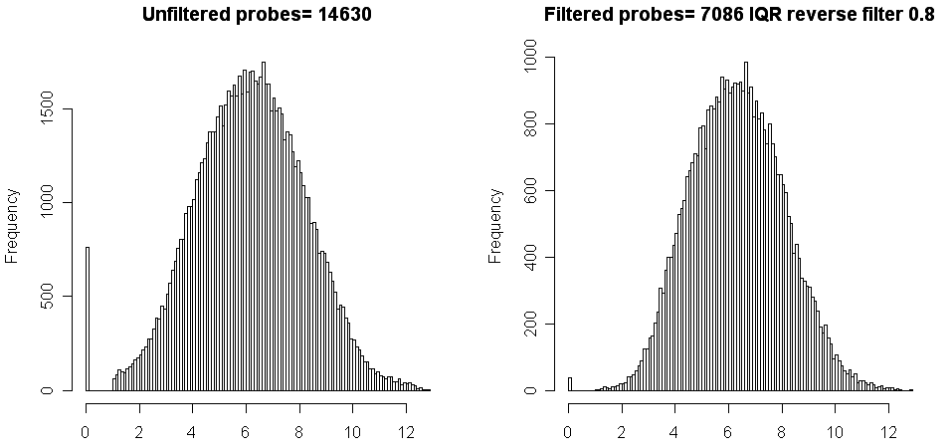


Figure 71: Reverse IQR filtering 0.8.

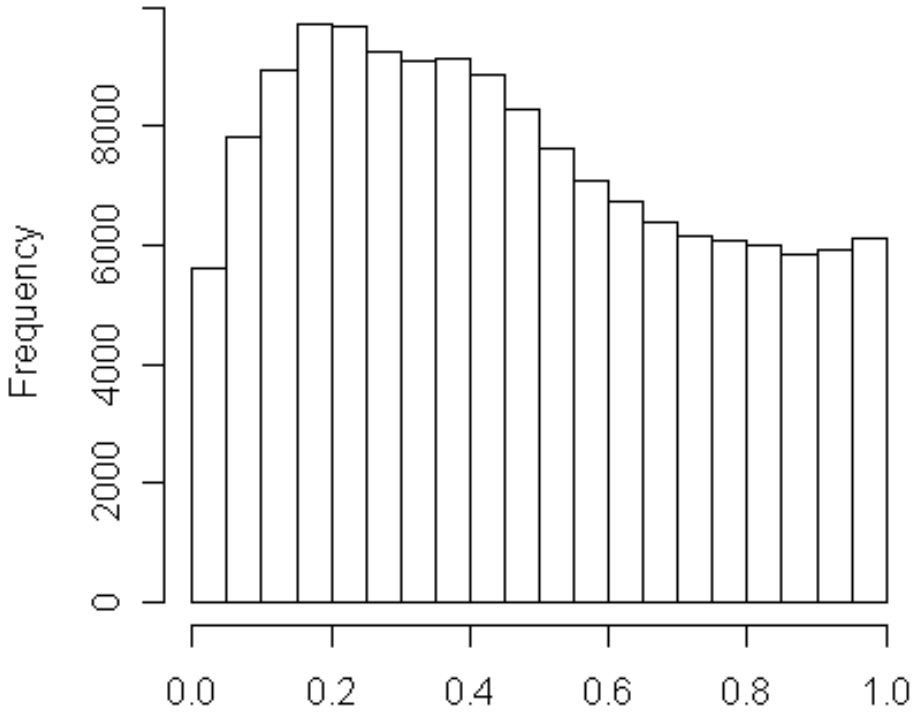


Figure 72: MiDAS p-values distribution. It is clear, from the p-values distribution, that type I error correction methods like as BH and BY cannot be applied due to the lack of a uniform distribution of p-values in the non- significant range.

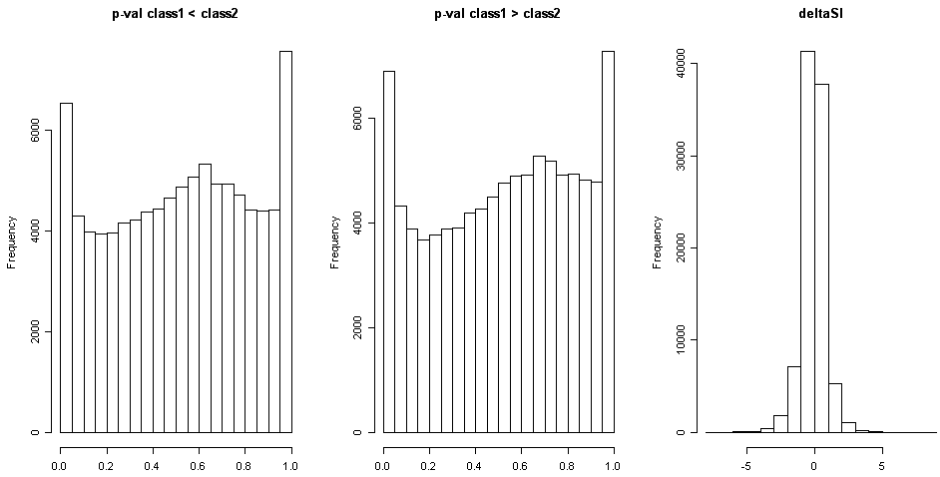


Figure 73: RankProd histograms of the p-values and average SI differences

At this point it is necessary to sub set the probe sets on the basis of a SI mean difference that user considers to be significant. The range of SI mean difference values can be seen by the output of the RP p-values results and filtering on gene/exon expression data can be done using the function *Filtering by mean of absolute SI mean difference*. In this example we use a threshold of 2 for absolute SI mean difference. This filter yields a total of 1743 glevel probe sets and 34033 elevel probe sets.

An important issue at this point of the analysis is to rank the alternative splicing events in order to start studing those more related to the biogical event under study. Our suggestion is to approach the problem at two levels:

1. Search for GO enriched terms within the set of putative alternatively spliced genes.

2. Integrate alternative splicing analysis with information that can be depicted by conventional differential expression analysis at gene level.

18.1 Search for GO enriched terms within the set of putative alternatively spliced genes.

The reason to search of enriched GO terms is due to the hypothesis that alternative splicing events are linked to the biological problem under investigation as happen for differential expression analysis. To do that we can use the function *Identifying enriched GO terms*. In this case we load the data after DABG, cross hyb and revIQR filters and we search for the presence of enriched GO terms in the sub set of putative alternative splcing events. In our example searching for enriched GO terms using the sub set of putative alternative splicing event we could identify a sub set of enriched GO terms, fig. fig. 75. At this point, it will be necessary to identify the most interesting GO terms

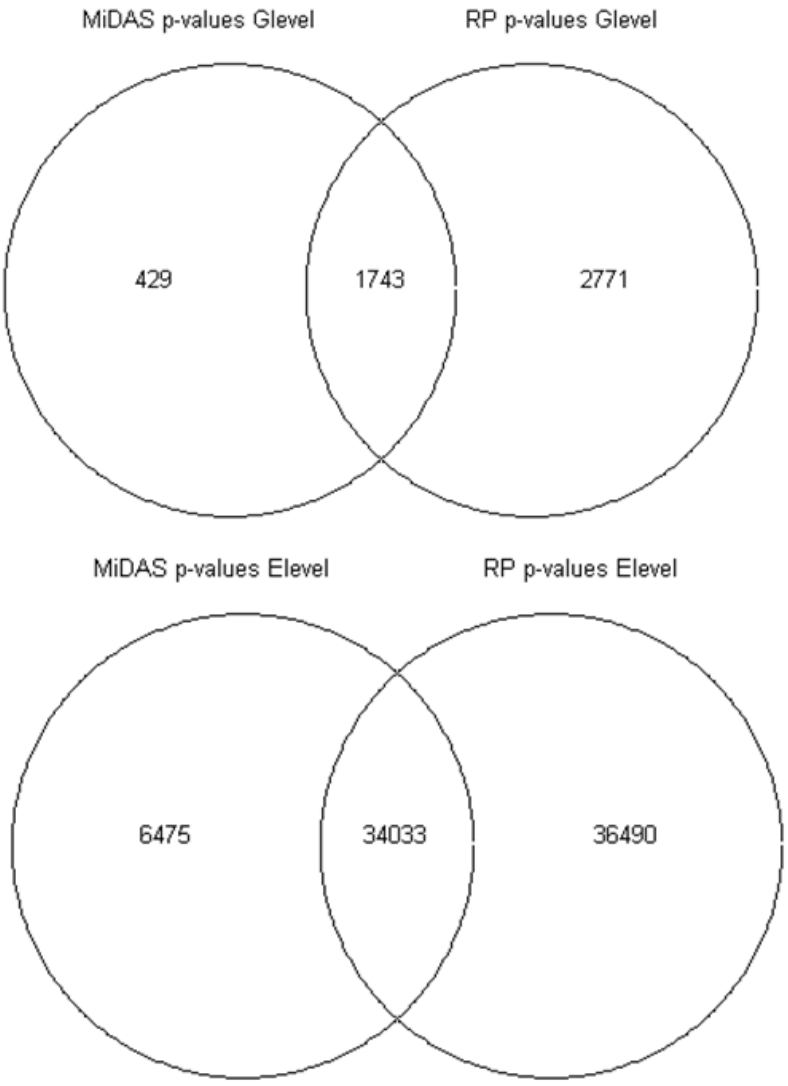


Figure 74: Venn diagrams of the intersection between MiDAS p-values and RP p-values. Clearly RP is less stringent that MiDAS.

on the basis of the user biological knowledge, extract those putative alternative splicing events using the function *Extracting Affy IDs linked to an enriched GO term* and inspect their splicing using the function *Inspecting splice indexes*.

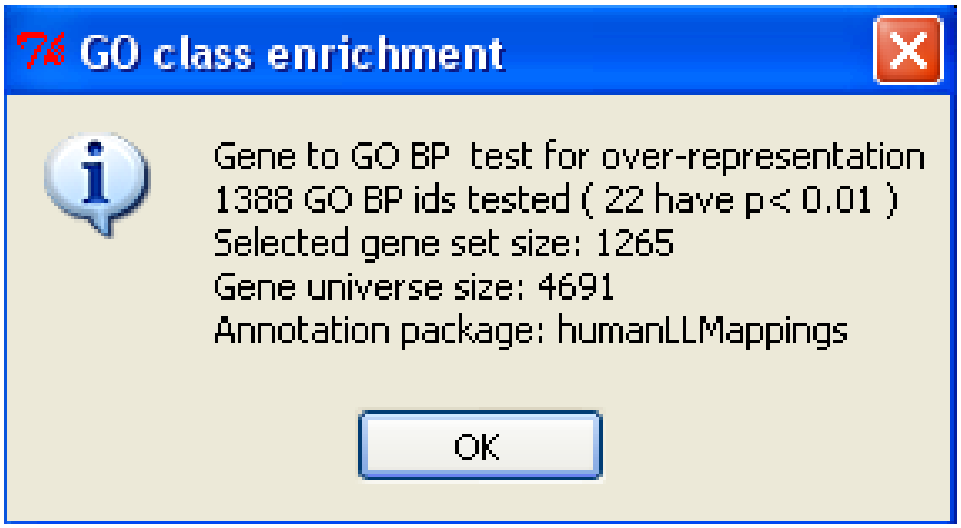


Figure 75: Sub set of enriched GO BP terms linked to putative alternative splicing events observed between heart and liver human tissues.

18.2 Integrate alternative splicing analysis with information that can be depicted by conventional differential expression analysis at gene level.

We save the data set at this point of the filtering exon level analysis, with *Save As* function, we export gene and exon level data with *Exporting Gene exprs and/or Exon/SI/MiDAS/RP data* function available in the filtering menu, and we get back to a conventional glevel differential expression analysis.

The analysis procedure is described in fig. 76.

Therefore, after DABG filter and cross hybridizing probe set removal we will apply a IQR filter to remove non-variant probe sets at gene level. Clearly as in the case on the exon analysis any filter applied to the gene level data will produce the corresponding sub set of exon level data.

The results of the IQR filter are shown in fig. 77.

At this point, since our exercise is based on two groups, it is possible to apply linear model analysis based on limma or a permutation based approach as SAM or rank product. To be conservative we use limma for two groups analysis. Checking the raw p-value distribution, function *Raw p-value distribution plot*, obtained for heart versus liver differential expression we can confirm that the BH or BY type I error correction can be

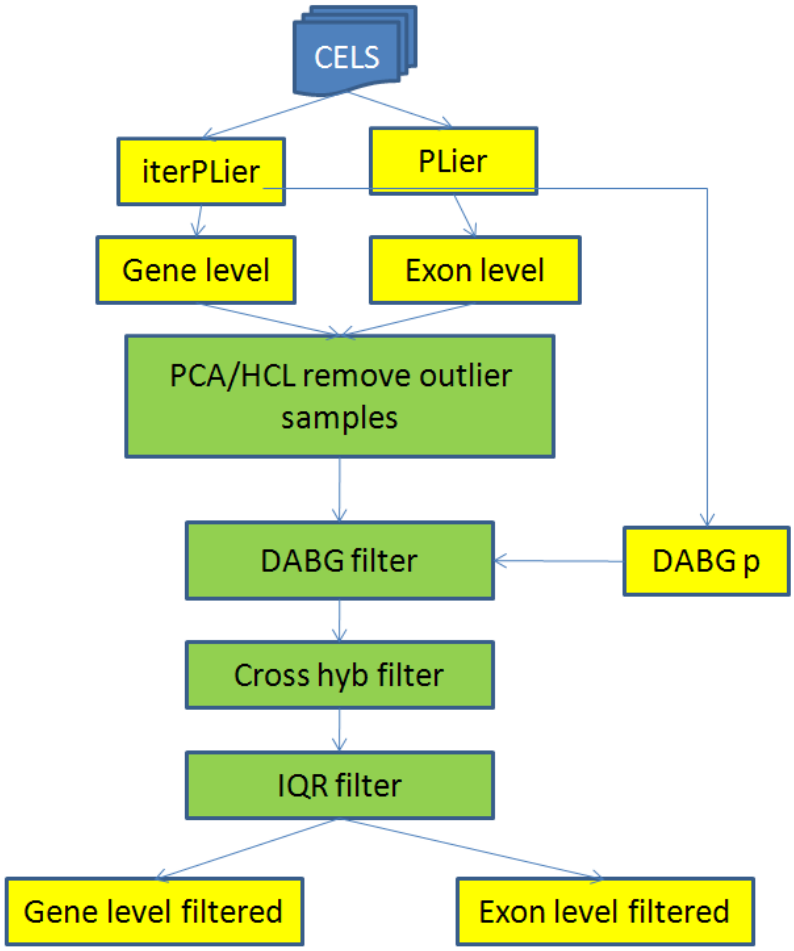


Figure 76: Gene level preprocessing.

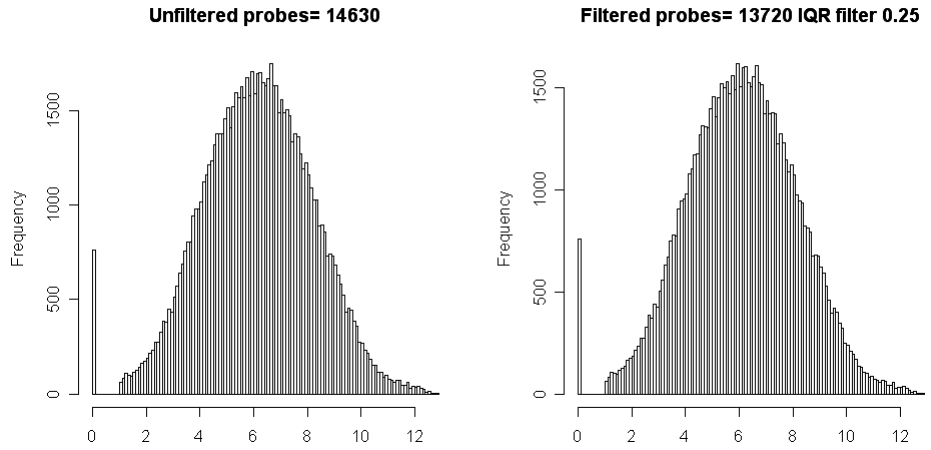


Figure 77: IQR filter at 0.25.

applied since the distribution in the non significant range is uniform. The differential expression selection is shown in fig. 78.

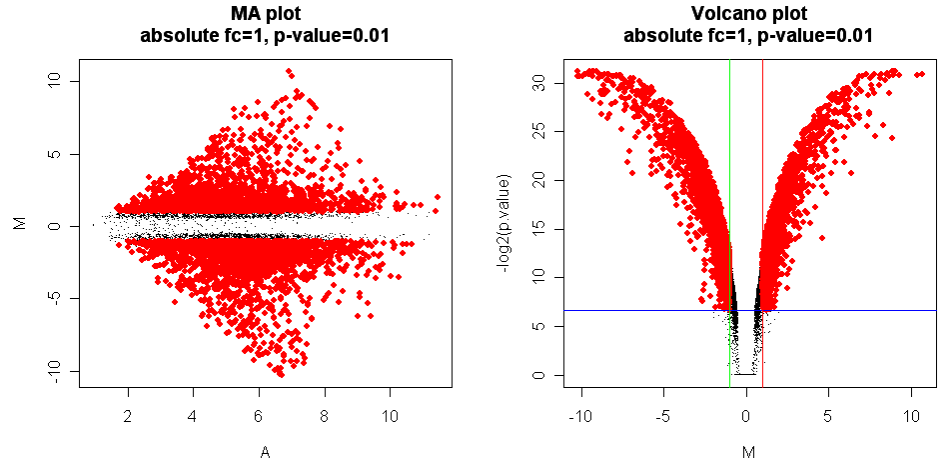


Figure 78: BY correction, absolute log2(fc)>1 and adjusted p-value < 0.01

For this sub set we search for enriched GO terms and we save the list of enriched classed. Since the putative alternative splicing events we have identified previously are too many to be experimentally investigated we in this case we will integrate GO terms found enriched within alternative splicing events, see previous sub section, and those found within the differential expression set. The reason of such intersection is due to the assumption that both alternative splcing events as well as differential expression should be linked to the biological problem under investigation. Using the two list of enriched

GO terms and the Venn diagram function available in the modelling stats menu we can detect those GO enriched terms in common between the two analysis, fig. 79.

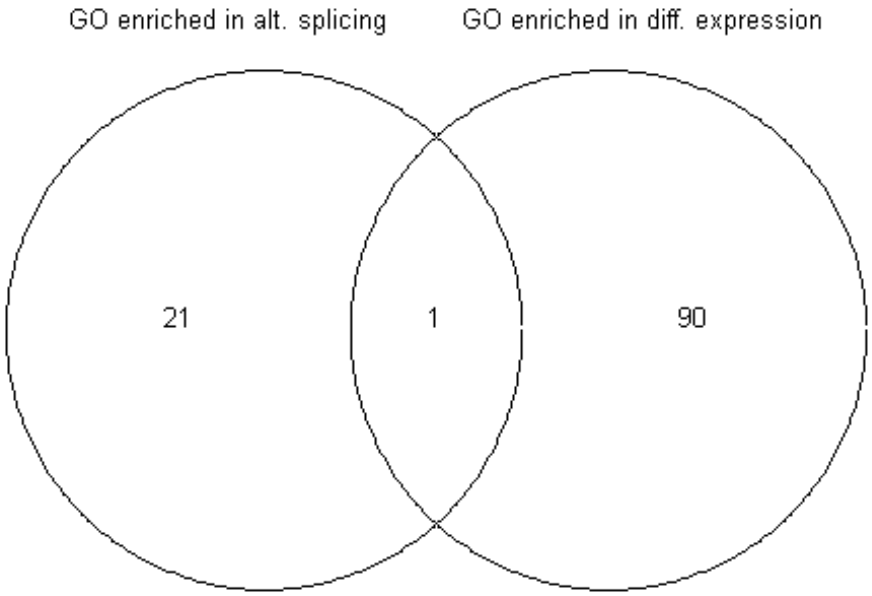


Figure 79: Intersection between the GO terms found enriched in exon level alternative splicing analysis and gene level differential expression analysis.

From this analysis we identified 1 GO term:

GO:0007155

At this point, it will be necessary to extract those putative alternative splicing events using the function *Extracting Affy IDs linked to an enriched GO term* and to inspect their splicing using the function *Inspecting splice indexes*.