oneChannelGUI Package: NGS secondary data analysis

Raffaele A Calogero, Francesca Cordero, Remo Sanges, Cristina Della Beffa

November 03, 2010

1 Introduction

OneChannelGUI was initially developed to provide a new set of functions extending the capability of affylmGUI package. oneChannelGUI was designed specifically for life scientists who are not familiar with R language but do wish to capitalize on the vast analysis opportunities of Bioconductor. oneChannelGUI offers a comprehensive microarray analysis for single channel pplatforms. Since Next Generation Sequencing (NGS) is becoming more and more used in the genomic area, Bioconductor is extending the number of tools for NGS analysis. Therefore, we are also extending the functionalities of oneChannelGUI to handle RNA-seq data with a specific focus on non-coding RNA quantitative and mRNA splicing qualitative analysis. In the developing of NGS functionalities in oneChannelGUI our attention was focused on secondary analysis, i.e. data mapped on reference genome.

2 Non-coding RNA-seq

The present goal of oneChannelGUI is to provide a graphical interface for the secondary analysis of ncRNA. Secondary analysis of ncRNA can be done on a common 64 bits laptop with at least 4 Gb RAM.

2.1 Input data

We are constantly increasing the number of outputs derived from primary mapping tools that can be loaded on oneChannleGUI At the present time oneChannelGUI allows the loading of data produced from various primary tools specifically designed for microRNA analysis, fig 1. .

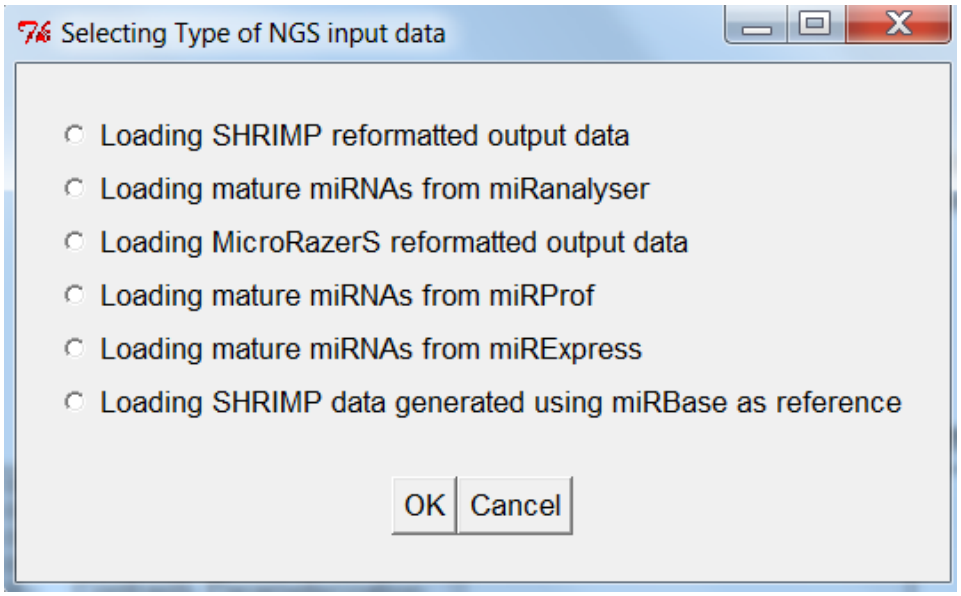


Figure 1: Primary mapping tools supported by oneChannelGUI.

2.1.1 SHRIMP

SHRIMP mapping tool information can be obtained in:

Rumble SM, Lacroute P, Dalca AV, Fiume M, Sidow A, et al.
SHRiMP: Accurate Mapping of Short Color-space Reads.
PLoS Comput Biol 2009 5(5): e1000386.
doi:10.1371/journal.pcbi.1000386

The program can be retrieved at:
http://compbio.cs.toronto.edu/shrimp/

To run SHRIMP version 2 the reference genome need to be indexed. It is possible to use a subset of the genome of interest encompassing only non-coding RNAs: it can be easily obtained using the function *"oneChannelGUI: Export non-coding RNA fasta reference file for ncRNA-seq quantitative analysis"*. The above function retrieves from oneChannelGUI a fasta file from the data of oneChannleGUI. The fasta file is generated by the standalone function *"ncScaffold"*, for its usage please refer to the standalone vignette. Primary alingment data provided by shrimp using the above reference sequences need to be reformatted using the function provided in the file menu: *"oneChannelGUI: Reformat NGS primary mapping output"*, fig. 2 . For this reformatting it is necessary to have perl or active perl installed in the computer. .

It is also possible to use as reference the full unmasked genome available on NCBI, also in this case it is necessary to reformat the aligned data using the function *"oneChannelGUI: Reformat NGS primary mapping output"*. Finally it is also possible to use as

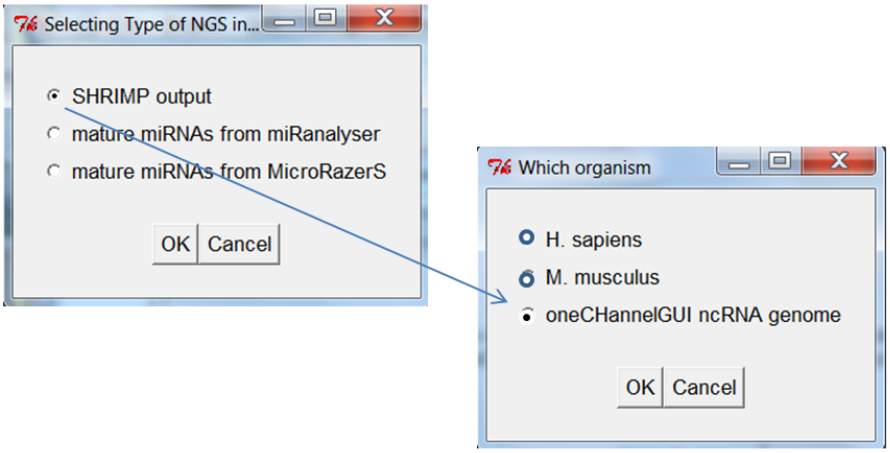


Figure 2: Reformatting data produced using SHRIMP and the nc-RNA reference set.

reference the microRNA precursors set of mirBase, actually is implemented the version 15.0 The fasta file are available in the ngsperl dir located in the oneChannelGUI path. This folder is created using the function *"oneChannelGUI: Set NGS perl scripts folder"*. In case miRbase precursors fasta file is used as reference the mapped data produced by SHRIMP can be directly loaded in oneChannelGUI without reformatting.

To run SHRIMP The reference genome needs to be indexed by SHRIMP with the following code:

```
$SHRIMP_FOLDER/utils/project-db.py \
--seed \
0011111001111111100, \
0011111110011111100, \
0011111111100111100, \
0011111111111001100, \
0011111111111110000  \
--h-flag --shrimp-mode cs /folder.where.fasta.file.is.located/ncRNAs.mm.fa
```

The flag for SOLID data is:

```
--shrimp-mode cs
```

The code to format the reference fasta file for SOLID data is:

```
$SHRIMP_FOLDER/bin/gmapper-cs -L /folder.where.fasta.file.is.located/ncRNAs.mm-cs \
-Y -V /dev/null >/dev/null
```

To run SHRIMP the code line is the following for solid data:

```
$SHRIMP_FOLDER/bin/gmapper-cs -L /folder.where.fasta.file.is.located/ncRNAs.mm-cs \
/folder.where.reads.file.is.located/myreads.file \
-N number.of.cores.used.by.shrimp -n 1 -U -o 1 -V -w 170% \
>myreads.file.out 2>myreads.file.log  &
```

The flag for Illumina data is:

```
--shrimp-mode ls
```

The code to format the reference fasta file for Illumina data is:

```
$SHRIMP_FOLDER/bin/gmapper-ls -L /folder.where.fasta.file.is.located/ncRNAs.mm-ls \
-Y -V /dev/null >/dev/null
```

To run SHRIMP the code line is the following for illumina data:

```
$SHRIMP_FOLDER/bin/gmapper-ls -L /folder.where.fasta.file.is.located/ncRNAs.mm-ls \
/folder.where.reads.file.is.located/myreads.file \
-N number.of.cores.used.by.shrimp -n 1 -U -o 1 -V -w 170% \
>myreads.file.out 2>myreads.file.log  &
```

The output of SHRIMP must be reformated to produce two files with the extension .bed and .logos. The output of SHRIMP has the following columns:

```
readname          Read tag name
contigname        Genome (Contig/Chromosome) name
strand            Genome strand ('+' or '-')
contigstart        Start of alignment in genome (beginning with 1, not 0).
contigend          End of alignment in genome (inclusive).
readstart          Start of alignment in read (beginning with 1, not 0).
readend            End of alignment in read (inclusive).
readlength        Length of the read in bases/colours.
score                    Alignment score
editstring        Edit string.
```

It is possible to copy in the oneChannelGUI path the perl scripts needed for SHRIMP data reformatting using the general menu function *oneChannelGUI: Set NGS perl scripts folder*. It is also necessary that perl is installed in the system. For windows user the best will be the installation of active perl, which is very strait forward. The function *oneChannelGUI: Reformat SHRIMP output* present in the File menu will allow data reformatting using as input a Target file, which will be also used subsequently to load the reformatted .bed and .logos files in oneChannelGUI.

The reformat routine, produces two files: one with the extension .bed and an other with the extension .logos. .bed file has three columns all represented by numbers. First

column is chromosome number. The mitocondrial genome represented by 77, X by 88 and Y by 89. Second column is strand given by 1 and -1 Third column is first position of the read mapping over the reference chromosome. An example of the .bed file structure is given in figure 3.

```
17  -1  56408593
16  -1  2205024
9 1 96938629
5 1 159912359
5 1 159912359
10  1 104196269
3 -1  52302292
10  1 104196269
3 -1  52302292
10  1 104196269
10  1 104196269
19  -1  50004042
```

Figure 3: Structure of mapping data that can be imported in oneChannelGUI.

.logos file has four columns, first is the ENSEMBL gene id second column is the description of the mapping given by Edit string:

The edit string consists of numbers, characters and the following additional symbols: '-', '(' and ')'. It is constructed as follows:
<number> = size of a matching substring
<letter> = mismatch, value is the tag letter
(<letters>) = gap in the reference, value shows the letters in the tag
- = one-base gap in the tag (i.e. insertion in the reference)
x = crossover (inserted between the appropriate two bases)

```
For example:
    A perfect match for 25-bp tags is: 25
    A SNP at the 16th base of the tag is: 15A9
    A four-base insertion in the reference: 3(TGCT)20
    A four-base deletion in the reference: 5----20
    Two sequencing errors: 4x15x6        (i.e. 25 matches with 2 crossovers)
```

Third column is the position of the beginning of the alignment on the ENSENBL gene Fourth column is the position of the first alignment on the read.

In case the mirBase precursors are used as reference there is no need of reformatting the output of SHRIMP. Output files produced by SHRIMPS are directly loaded and only alignments with one SNP or with perfect march are kept for the analysis.

2.1.2 miRanalyser

miRanalyser mapping tool information can be obtained in:

```
Nucleic Acids Res. 2009 Jul 1;37(Web Server issue):W68-76.
miRanalyzer: a microRNA detection and analysis tool for
next-generation sequencing experiments.
Hackenberg et al.

The web program can be used at:
http://web.bioinformatics.cicbiogune.es/microRNA/miRanalyser.php
```

The interesting point of this tool, figure 4 is that user need only to load the reads to be mapped as a multi-fasta file.

```
>ID 49862
GAGGTAGTAGGTTGTA
>ID 15490
ACCCGTAGAACCGACC
>ID 13762
GGAGCATCTCTCGGTC
```

This web tool facilitates the generation of primary data for unexperienced users. The output is generated in few hours and can be retrieved by the user after bookmarking each of the job pages. The output is a very complete, but we will focus only on the retrieval of the mapping data referring to the mature miRNAs, figure 5

2.1.3 miRExpress

miRExpress mapping tool information can be obtained in:

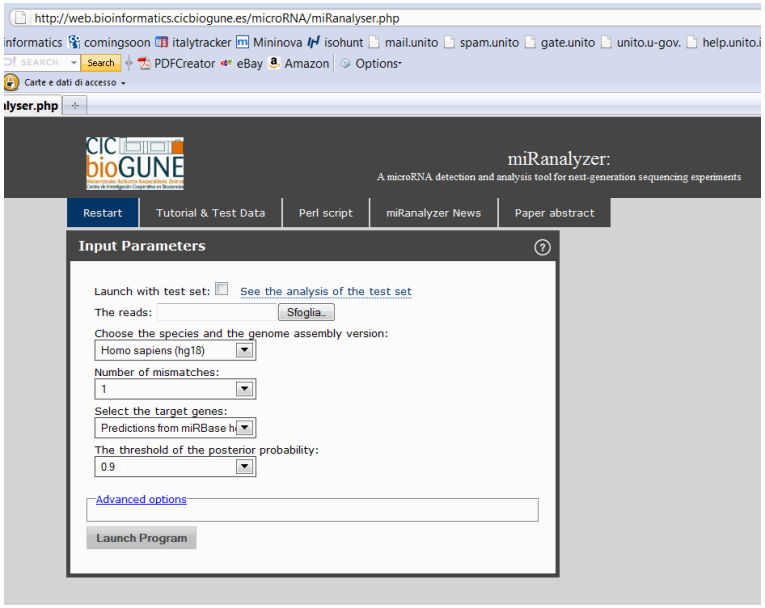
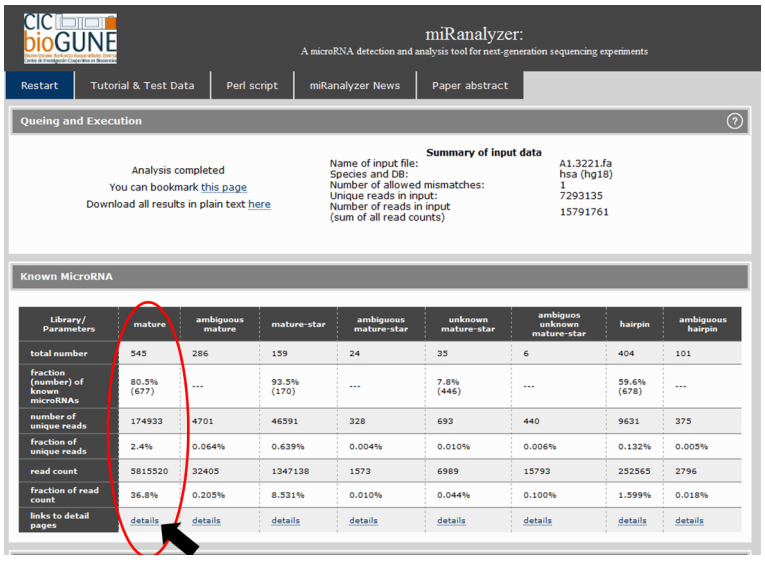


Figure 4: web interface of miRanalyser



Library/ Parameters	mature	ambiguous mature	mature-star	ambiguous mature-star	unknown mature-star	ambiguos unknown mature-star	hairpin	ambiguous hairpin
total number	545	286	159	24	35	6	404	101
fraction (number) of known microRNAs	80.5% (677)	---	93.5% (170)	---	7.8% (446)	---	59.6% (678)	---
number of unique reads	174933	4701	46591	328	693	440	9631	375
fraction of unique reads	2.4%	0.064%	0.639%	0.004%	0.010%	0.006%	0.132%	0.005%
read count	5815520	32405	1347138	1573	6989	15793	252565	2796
fraction of read count	36.8%	0.205%	8.531%	0.010%	0.044%	0.100%	1.599%	0.018%
links to detail pages	details	details	details	details	details	details	details	details

Figure 5: The arrow indicate the link that allows to retrieve the tab delimited file referring to mapping to the mature subset of microRNAs

BMC Bioinformatics 2009, 10:328.
miRExpress: Analyzing high-throughput sequencing data for
profiling microRNA expression.
Wei-Chi Wang, Feng-Mao Lin, Wen-Chi Chang, Kuan-Yu Lin,
Hsien-Da Huang and Na-Sheng Lin

The tool can be retrieved at:
\url{http://miRExpress.mbc.nctu.edu.tw}

The alignment files can be directly loaded in oneChannelGUI. Below an example of the structure of the alignment files generated with miRExpress.

```
hsa-mir-520a
CUCAGGCUGUGACCCUCCAGAGGGAAGUACUUUCUGUUGUCUGAGAGAAAAGAAAGUGCUUCCCUUUGGACUGUUUCGGUUUGA
Others*********************************************************************************
              CTCCAGAGGGAAGTACTTTCT 3
//

hsa-mir-99a
CCCAUUGGCAUAAACCCGUAGAUCCGAUCUUGUGGUGAAGUGGACCGCACAAGCUCGCUUCUAUGGGUCUGUGUCAGUGUG
Others*********************************************************************************
                                                    AACCCGTAGATCCGATCTTGTG 30
                                                    CAAGCTCGCTTCTATGGGTCTG 87
                                                    CAAGCTCGCTTCTATGGGTCT 64
//
```

2.1.4 miRProf

miRProf is web mapping tool. The web tool can be used at: `http://srna-tools.cmp.uea.ac.uk/animal/cgi-bin/srna-tools.cgi` The tab delimited files provided by miRProf can be directly loaded in oneChannelGUI.

3 File menu

In this menu selecting the option RNA-seq it is possible to load NGS data.

3.1 Data loading

Data are loaded in oneChannelGUI using the *New function* in the File Menu selecting from the menu of the available platforms NGS, figure 6. oneChannelGUI will require a target file which is a tab delimited file with three columns: Names, FileNames and Target, figure 7. The FileName column must contain the names of the files, each one

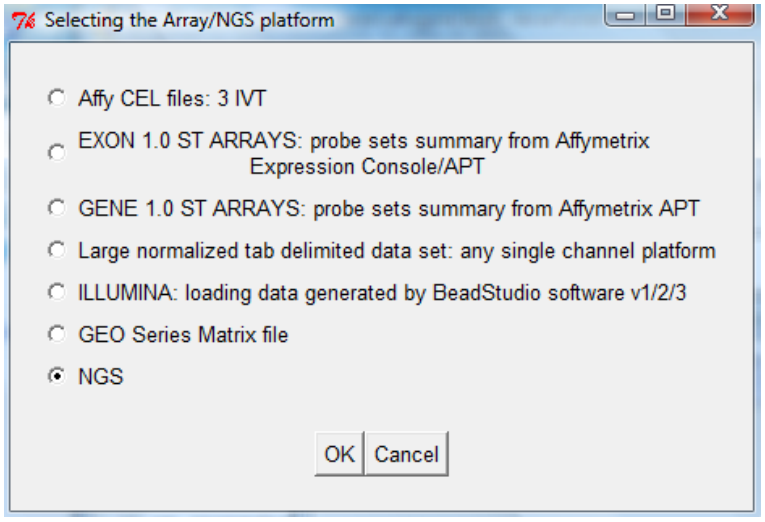


Figure 6: Data loading menu.

Name	FileName	Target
s2	sample2.mapping.filtering	s_11189125_3789945
s3	sample3.mapping.filtering	s_11189125_3980027
m2	mock2.mapping.filtering	m_9759522_2738303
m3	mock3.mapping.filtering	m_9759522_2872923

Figure 7: Target file structure.

belonging to a mapping experiment, with the structure previously indicated. IMPORTANT: target column contains, togeter with the covariate, also the total number of reads and the total number of mapped reads separated by an underscore. In case the above data are not known they need to be sostituted by NA. The name in the FileName column of the target file are those produced by the primary mapping tool. The loaded files are those produced by the reformatting tools available in the FileMenu *Reformat NGS primary mapping output*

3.2 Reformatting NGS data

The raw data derived from a NGS experiment cannot be used directly for statistical analysis. At the present time are supported the output files produced by SHRIMP, MicroRazerS, miRanalyzer, miRExpress and miRProf. When user loads the set of NGS runs, described by the target file, those data are reoganized in an ExpressionSet format, which is quite common for microarray data. The are two options, one is the use of Genominator package and the other the use of a segmentation approach based on chipseq package. A pop-up will ask the user to decide between the two approaches. The Genominator based approach allows a quick and efficient reorganization of the data based on the availability of a specific set of annotation on which the reads are mapped. In oneChannelGUI the set of annotation is dinamically generated from ENSEMBL and allow to use Genominator for microRNA, exons and TSS, i.e. Transcription Start Sites.

An alternative option to Genominator based data loading is a segmentation approach based on chipseq package. In this case to reduce RAM consumption and to allow the analysis also on conventional 32 bit computers, the raw data are loaded and stored in files on the bases of their belonging to a specific chromosome. Subsequently, the union of all counts over the all samples for a specific chromosome are used to define the peaks where the reads are clustering on the plus and minus strand. To define chromosome peaks user need to indicate the size of extension of the reads. We usually extend the reads to the real length, e.g. 35. However, extension size for ncRNA quantification values are also between 100 and 200 nts, if the size of cDNA library is considered. Library size is between 108 and 130 nts if a fractionation of small RNAs (10-40 nts) was used. In case cDNA was prepared from total RNA, which is better for quantification, the size is between 150 and 200 nts. For quantification the preparation of the library from total RNA is preferred since it is characterized by a lower inter-experiment variability. User needs also to define the number of reads that are mapped as random event, for a library between 10-20 milion 35-mer tags this value is about 8. It is important to remember that using long extension value an high number of peaks will be created since more peaks will be characterized by having a number of mapped reads greater than 8. The name of each peak is made by:

```
chr name.strand.start position-end position
e.g. chr1.plus.100000-105000
     chr2.minus.500000-500630
```

After reformatting the counts are saved in an ExpressionSet object which is stored in the affylmGUIenvironment and it is ready to be further analyzed.

In the case of miRanalyser, miRExpress, miRProf and data generated using SHRIMP, with miRBase precursors as reference, the loading procedure is much faster since the data produced by the primary mapping tools need only to be reorganized in a matrix.

4 Reformatting-Normalizing NGS data menu

The NGS data stored in the ExpressionSet can be log2 transformed. Since it is possible that some peaks are subset of the same peak, e.g. two peaks located less then 50 bases to each other, the function Refining peaks allows to merge peaks located near to each other, given a user define threshold. In figure 8

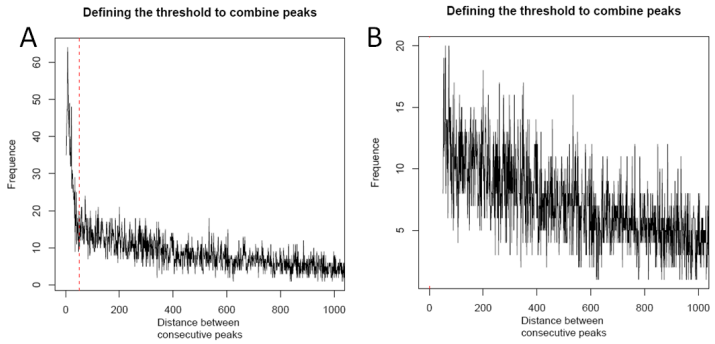


Figure 8: A) Plot of frequence of two nearby peaks versus inter-peaks distance. The dashed lines indicate a max distance defined by user, in this case 50 nts. B) Plot of the refined peaks after recursive merging of nearby peaks.

It is now possible to scale/normalize the NGS data using the *oneChannelGUI: Scale/normalize NGS data*, which used the method described by Robinson and Oshlack in Genome Biology 2010, 11:R25 and it is implemented in edgeR package. The outplot plot descrive the sample organization before and after normalization using Multidimensional scaling plot which also plot the variation in the common dispersion 9.

5 QC menu

The section QC allows to visualized the box plot of the NGS samples after reformatting in peaks as well as samples PCA and hierarchical clustering. It is also possible to visualize data using a Multidimensional scaling plot, using the function *oneChannelGUI: Multidimensional scaling plot (edgeR package)*. This plot is a variation on the usual

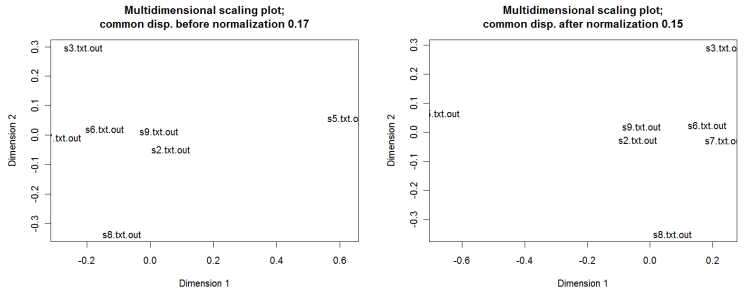


Figure 9: A) Multidimensional scaling plot before normalization. B) Multidimensional scaling plot after normalization. it is clear that there in an outlier, i.e. the sample that get in the first dimension very far away form the others.

multdimensional scaling (or principle coordinate) plot, in that a distance measure particularly appropriate for the digital gene expression (DGE) context is used. The distance between each pair of samples (columns) is the square root of the common dispersion for the top n (default is n = 500) genes which best distinguish that pair of samples. These top n genes are selected according to the tagwise dispersion of all the samples.

6 Filtering menu

The section filtering allows remove those peaks that are little informatives, i.e. those with too little counts. It also allows to filter the dataset on a list of peaks identifiers, e.g. those derived from a list of differentially expressed peaks Furthermore, tab delimited files containing the loaded counts can be exported.

7 Statistics menu

In this section are implemented interfaces to edgeR and baySeq package. Both for edgeR and baySeq the inteface uses the negative binomial distribution model to detect differential expression. P-value adjustment can be made also used.

8 Biological Interpretation menu

Under development