oneChannelGUI Package: NGS secondary data analysis

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

August 27, 2011

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 Second 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, fig. 1.

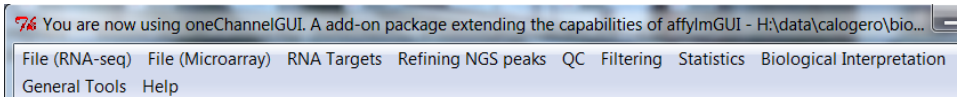

Figure 1: NGS Menu.

2 General Tools

In this menu there are all the functions necessary to install the external software required to analyse RNA-seq data. Perl or active perl should be already installed by the user. The function *Installing all external software needed for RNA-seq* installs all the external software needed for RNA-seq data analysis. It is also possible to install part of the software using the functions:

1. *Set NGS perl scripts folder* installs the perl scripts needed for secondary analysis of data produced with external software.

2. *Install Bowtie and Picard tools* allows the installation of bowtie and picard tools, as well as a set of precompiled reference sequences to be used to map short-reads. In the present implementation miRNA precursors for human, mouse rat and bovine are available.

3 File (RNA-seq)

The present goal of oneChannelGUI is to provide a graphical interface for the analysis of RNAseq, fig. 2. Analysis can be done on a common 64 bits laptop with at least 4 Gb RAM.

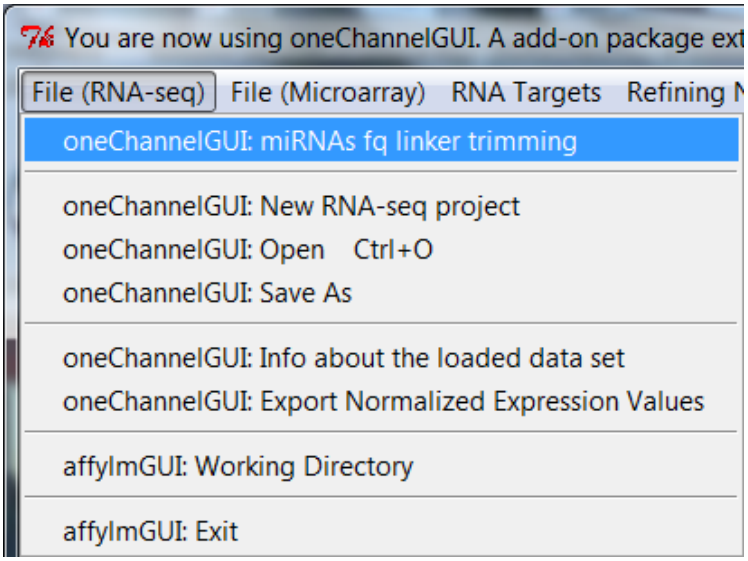

Figure 2: File (RNA-seq) Menu.

4 New RNA-seq project

In this menu selecting the option RNA-seq it is possible to load NGS data. oneChannelGUI will require a target file which is a tab delimited file with three columns: Names, FileNames and Target, figure 3. The FileName column must contain the names of the files, each one 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 total number of reads and the total number of mapped reads are not known, they will be defined on the basis of the mapped reads in the final counts data loaded in oneChannelGUI. The name in the FileName column of the target file are those produced

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 3: Target file structure.

by the primary mapping tool. The loadable files are those produced by the reformatting tools shown in figure 7.

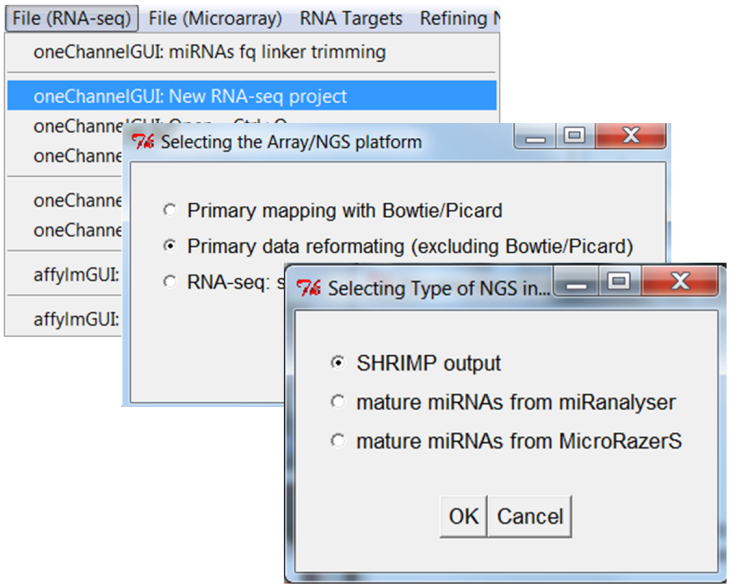

Figure 4: Reformatting functions available.

4.1 Primary mapping with Bowtie/Picard

In *New RNA-seq project menu* it is available the function for the primary mapping of short reads: *Primary mapping with Bowtie/Picard*. oneChannelGUI uses bowtie for primary mapping of Illumina short-reads, figure 5. The output of bowtie is a SAM file subsequently converted in a BAM file using picard java tools. Data are loaded using a oneChannelGUI target file, e.g. mytarget.txt:

```
Name        FileName        Target
C1        C1.fq       ctrl
```

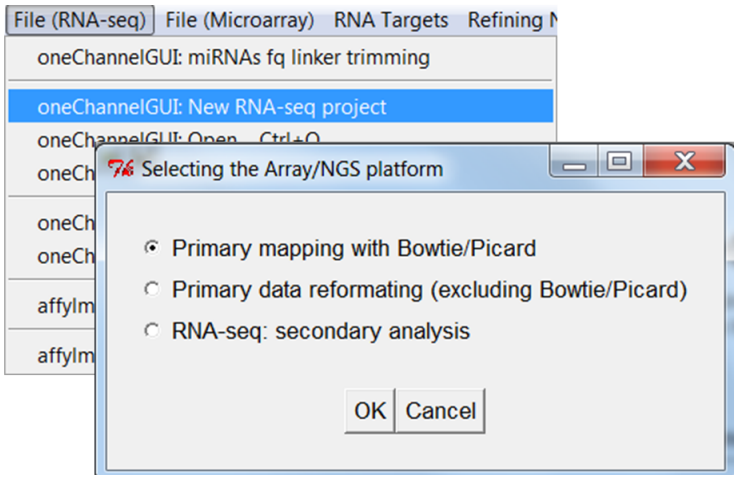

Figure 5: Primary mapping with Bowtie

```
C2          C2.fq        ctrl
T1          T1.fq        trt
T2          T1.fq        trt
```

Where filenames are the short reads generated by an Illumina machine in fastq format. Upon primary mapping and conversion of output to BAM file a new target file is produced, e.g. mytarget.bam.txt: Name FileName Target C1 C1.bam ctrl C2 C2.bam ctrl T1 T1.bam trt T2 T1.bam trt BAM files are subsequently imported in oneChannelGUI using Rsamtools and converted in an ExpressionSet for further statistical analyses. The function to load bam files in oneChannelGUI is available in the File menu as one of the *new* function options.

4.2 miRNAs fq linker trimming

The function *miRNAs fq linker trimming* is used to remove the 5 and 3 prime end linkers from single-end (SE) FASTQ formated illumina data using a perl script. The script need still some optimizations. This trimming is strongly suggested in case Bowtie mapping is performed. The scripts trims one milion reads in aprox. 5 minutes. The function loads the fastq using a conventional target file:

```
Name          FileName          Target
C1          C1.fq        ctrl
C2          C2.fq        ctrl
T1          T1.fq        trt
T2          T1.fq        trt
```

The output of the function are files with the addition of trim.fq to the initial filename. A new target file is also generated:

Name	FileName	Target
C1	C1.fqtrim.fq	ctrl
C2	C2.fqtrim.fq	ctrl
T1	T1.fqtrim.fq	trt
T2	T1.fqtrim.fq	trt

Furthermore, a pdf file with statistics of the trimmed data is produced. Each page of the pdf contains two histograms with the statistics of 3 and 5 prime end trimming, fig. 6.

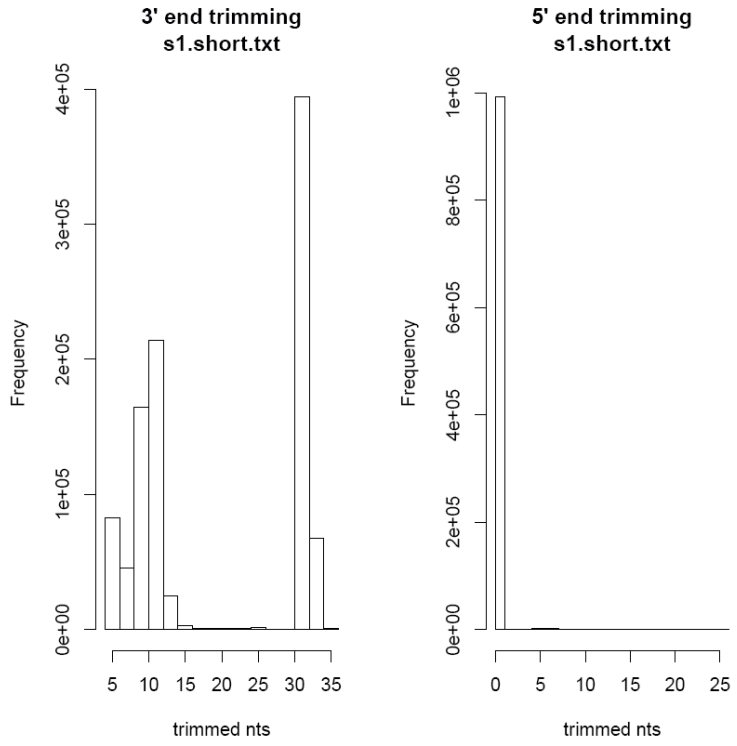

Figure 6: Trimming statistics.

4.3 Primary data reformatting (excluding Bowtie/Picard)

At the present time the output files produced by SHRIMP, MicroRazerS, miRanalyzer, miRExpress and miRProf are supported as input in oneChannelGUI. However, the output generated by SHRIMP, MicroRazerS, miRanalyzer need to be reformatted before loading: figure 7.

In the case of miRExpress and miRProf the loading procedure is much faster since the data can be directly loaded, see next subsection.

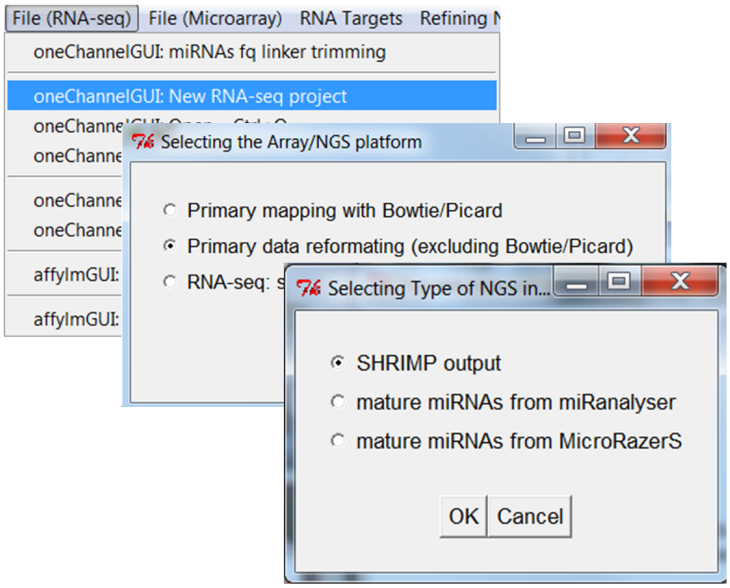

Figure 7: Reformatting tools available.

4.4 Secondary analysis: loading primary mapped 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 8. .

4.4.1 Loading BAM files generated using oneChannelGUI primary mapping of miRNAs fastq on miRBase precursors

The function *Loading BAM files generated using oneChannelGUI primary mapping of miRNAs fastq on miRBase precursors* is used to handle the BAM files generated by the Bowtie primary mapping of miRNA sequencing data, using one of the following reference sets:

1. miRBase human precursor set: hs.
2. miRBase mouse precursor set: mm.
3. miRBase rat precursor set: rn.
4. miRBase bovine precursor set: bo.
5. NB: other reference sets can be requested to oneChannelGUI developers for implementation

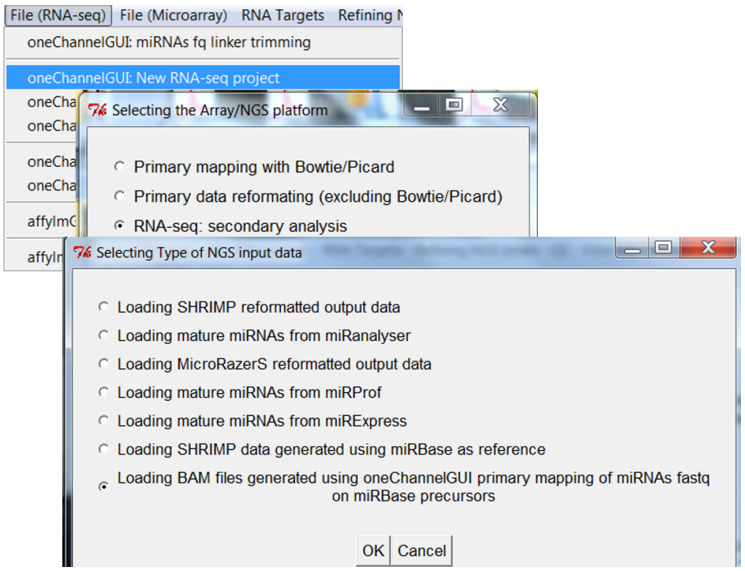

Figure 8: Primary mapping tools supported by oneChannelGUI.

With the above mentioned function BAM data are loaded in oneChannelGUI as an expression set. Subsequently data can be filtered and/or analysed to detect differentially expressed genes with the other functions implemented in oneChannelGUI.

4.4.2 Loading SHRIMP reformatted output data and Loading SHRIMP data generated using miRBase as reference

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. 9 . For this reformatting it is necessary to have perl or active perl installed in the computer. .

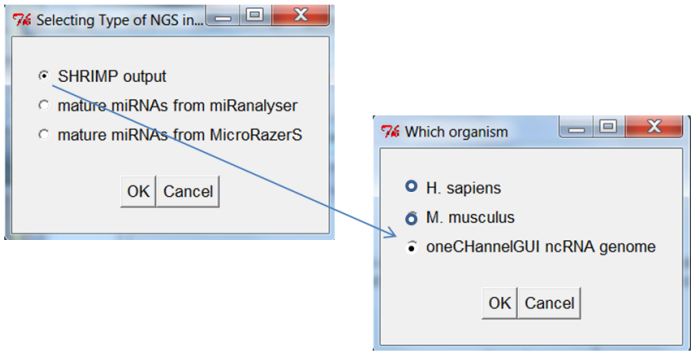

Figure 9: Reformatting data produced using SHRIMP and the nc-RNA reference set.

It is 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"*. It is also possible to use as reference the microRNA precursors set of mirBase, actually is implemented the version 17.0 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 10.

```
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 10: 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.

4.4.3 Loading mature miRNAs form 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 11 is that user need only to load the reads to be mapped as a multi-fasta file.

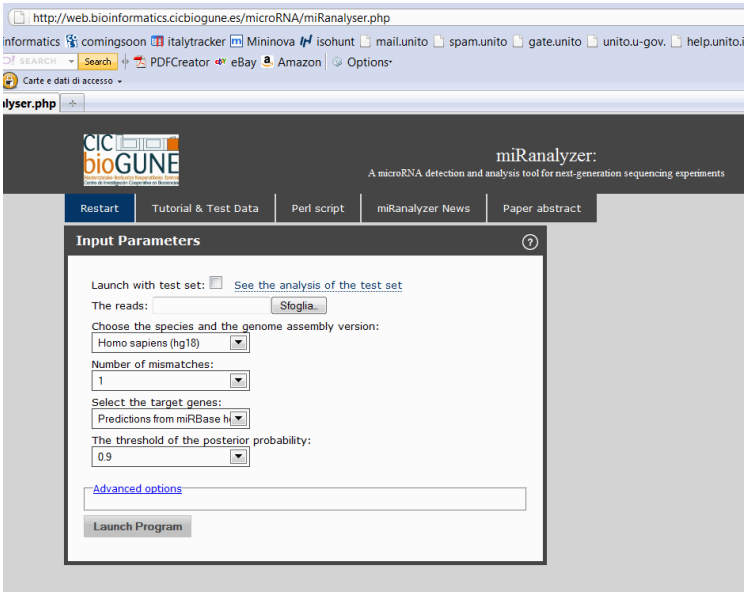

Figure 11: web interface of miRanalyser

>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 12

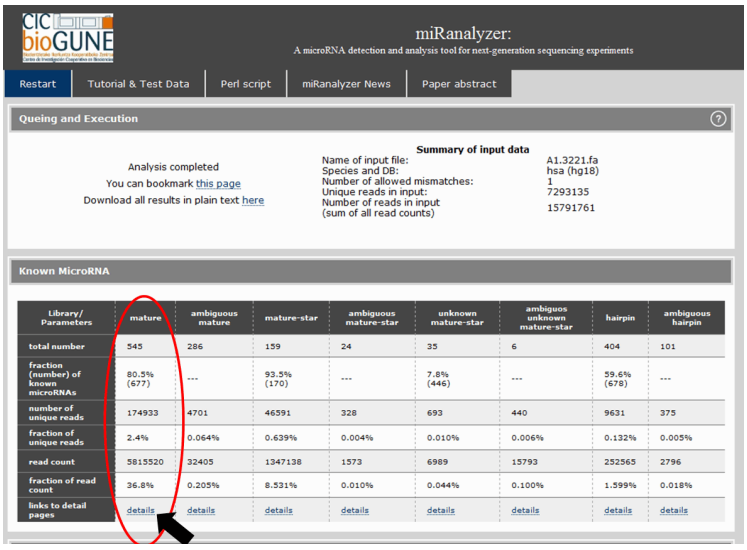

Figure 12: The arrow indicate the link that allows to retrieve the tab delimited file referring to mapping to the mature subset of microRNAs

4.4.4 Loading mature miRNAs from miRExpress

miRExpress mapping tool information can be obtained in:

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
//
```

4.4.5 Loading mature miRNAs from 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.

5 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 13

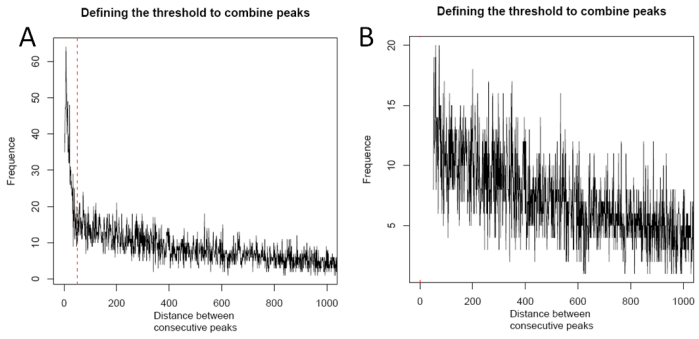

Figure 13: A) Plot of frequence of two nearby peaks versus inter-peaks distance. The dashed lines indicate a max distance defined by user, in this case 50 nts. B) Plot of the refined peaks after recursive merging of nearby peaks.

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 14.

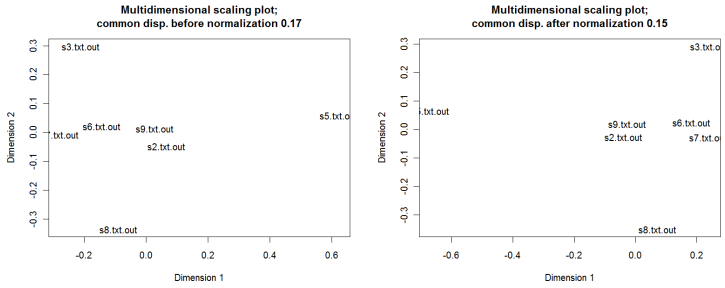

Figure 14: A) Multidimensional scaling plot before normalization. B) Multidimensional scaling plot after normalization. it is clear that there in an outlier, i.e. the sample that get in the first dimension very far away form the others.

6 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 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.

7 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.

8 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.

9 Biological Interpretation menu

Under development