miARma-Seq: **mi**RNA-Seq **A**nd **R**NA-Seq **M**ultiprocess **A**nalysis tool.

miRNA-Seq Data Analysis User's Guide

Eduardo Andrés-León, Rocío Núñez-Torres and Ana M Rojas.

Instituto de Biomedicina de Sevilla (IBIS), Hospital Universitario Virgen del Rocío/CSIC/Universidad de Sevilla, Computational Biology and Bioinformatics Group, Seville, Spain.

First edition 1.1 March 2016

Table of Contents

1. INTRODUCTION

miARma-Seq is a comprehensive pipeline analysis for RNA-Seq and miRNA-Seq data suited for mRNA, miRNA and circRNA identification and differential expression analysis of any organism with a sequenced genome. Briefly miARma-Seq integrates quality-control analysis of raw data (fastqc), trimming of the reads, with adapter sequence prediction if necessary, alignment of the reads with the correspondent genome reference, entities quantification, differential expression analysis, miRNA-mRNA target prediction, miRNA-mRNA inverse expression pattern analysis and functional analysis to detect the enrichment of metabolic pathways and gene ontologies for mRNAs. All these steps can be executed as a whole pipeline or as separated steps. To make easier the execution of single steps, miARma-Seq has been implemented with a Perl based module structure.

This guide gives a tutorial-style introduction for the practical use of miARma-Seq but does not describe every feature of the pipeline. A full description of every feature is given by the individual function help documents available in our website (http://miarmaseq.cbbio.es/Documentation/). It includes explanations of command-line options for each type of analyses to give an idea of basic usage. Input and output file formats are also detailed. Also, many examples of use are given.

This document does not try to explain the underlying algorithms or data-structures used in miARma-Seq. For these issues, we recommend to consult the information available in the webpages of the software integrated in miARma-Seq.

2. PRELIMINARIES

2.1. Pre-requisites

miARma-Seq is a tool that provides an easy and common interface to various analysis softwares. It also intends to reduce to the minimum the number of dependencies. Nevertheless, some basic programs listed below must be correctly installed:

1. Perl v5.6.0 or higher**.** http://www.cpan.org/src/5.0/perl-5.6.1.tar.gz
2. R environment v.3.0 or higher. http://www.r-project.org/
3. Java v.1.6. or higher. http://www.java.com/.
4. Bioconductor v.1.3 or higher. http://www.bioconductor.org/install/
5. Compilers:
    a. Xcode for Mac: https://itunes.apple.com/es/app/xcode/id497799835?l=en&mt=12
    b. For Linux:
        i. Gcc: https://ftp.gnu.org/gnu/gcc/
        ii. make: http://ftp.gnu.org/gnu/make/

2.2. How to get help

This user guide will hopefully answer most questions about miARma-Seq. Note that each module in miARma-Seq has its own help page (http://miarmaseq.cbbio.es/Documentation). If you have a question about any particular function, reading the module's help page will often answer the question very quickly. Nevertheless, if you've run into a question, which isn't addressed by the documentation, or you've found a conflict between the documentation and software itself, then you can visit our help & contact web page at http://miarmaseq.cbbio.es/help.

In addition, the authors of miARma-Seq always appreciate receiving reports of bugs in the pipeline modules or in the documentation. The same goes for well-considered suggestions for improvements. For these issues please contact at: miARma-devel@cbbio.es.

3. QUICK START

3.1. miARma installation instructions

Latest installation instruction for Linux, Mac and Windows, can be found in our web page at http://miarmaseq.cbbio.es/installation. If you are using a Unix system, the recommended procedure is the following:

1. Create a directory to install miARma, (eg : NGS) and download the software :

```
$> mkdir NGS
$> cd NGS/
NGS> curl -L -O https://bitbucket.org/cbbio/miarma/get/master.tar.bz2
```

2. Extract miARma binaries and libraries:

```
NGS>tar -xjf master.tar.bz2
NGS>cd cbbio-miARma-*
cbbio-miarma>ls -l
      Examples
      README.md
      bin
      lib
      miARma
```

3.2. miRNA examples installation instructions

More detail about examples and cases of use can be found at http://miarmaseq.cbbio.es/examples.

1. Inside miARma folder, download the data:

```
miARma>  curl -L -O
https://sourceforge.net/projects/miarma/files/Examples/Examples_miARma_miRNAs.tar.bz2
```

2. Uncompress it:

```
miARma>tar -xjf Examples_miARma_miRNAs.tar.bz2
```

3. Check the parameters (optional step):

```
miARma>perl miARma Examples/basic_examples/miRNAs/Known_miRNAs/1.Quality/1.Quality.ini --check
```

4. Execute the examples:

```
miARma>perl miARma Examples/basic_examples/miRNAs/Known_miRNAs/1.Quality/1.Quality.ini
```

3.3. Other needed data for miARma execution

miARma uses Bowtie1, Bowtie2 and miRDeep for read alignment in miRNAs analysis. Each tool has their special requirements. For the examples included in miARma, human hg19 genome and the correspondent genome index are provided but miARma includes examples to create your own index from any sequenced organism.

3.3.1. Human genome (hg19) installation:

1. Downloading from miARma folder:

```
miARma>curl -L -O http://miarmaseq.cbbio.es/download/Genome/hg19_genome.tar.bz2
```

2. Extracting:

```
miARma> tar –xjf hg19_genome.tar.bz2
```

3.3.2. Bowtie1 index installation:

1. Downloading from miARma folder:

```
miARma>  curl -L -O http://miarmaseq.cbbio.es/download/Genome/Index_bowtie1_hg19.tar.bz2
```

2. Extracting:

```
miARma>tar –xjf Index_bowtie1_hg19.tar.bz2
```

3.3.3. Bowtie2 index installation:

1. Downloading from miARma folder:

```
miARma>  curl -L -O http://miarmaseq.cbbio.es/download/Genome/Index_bowtie2_hg19.tar.bz2
```

2. Extracting

```
miARma>tar –xjf Index_bowtie2_hg19.tar.bz2
```

3.3.4. miRDeep index installation:

1. Downloading from miARma folder:

```
miARma>  curl -L -O http://miarmaseq.cbbio.es/download/Genome/Index_bowtie1_hg19.tar.bz2
```

2. Extracting :

```
miARma>tar –xjf Index_bowtie1_hg19.tar.bz2
```

4. miRNA-SEQ ANALYSIS

miARma-Seq presents a highly flexible modular structure to perform the different stages of the miRNA analysis. In this section, each module will be extensively described, including the description of

the input and output files, the different parameters for the analysis and the creation of the configuration file to execute it.

miARma-Seq has implemented two alternative analysis for miRNA analysis according to the identification or not of de novo miRNAs. In order to better explain this module, data from GEO (GEO code: GSE47602) will be used (this data can be downloaded from GEO). For testing purposes, miARma provides a reduced version of raw files from this experiment in order to illustrate how it works. Briefly, this experiment contains miRNA-Seq data obtained from a time course experiment performed in the MCF7 cell line in hypoxic conditions (2 replicates in normal conditions and 2 replicates in hypoxic conditions during 16, 32 and 48 hours respectively). In the differential expression analysis, DE miRNAs at 16, 32 and 48 hours will be identified. Examples installation is described in section 3.2.

4.1. Known miRNAs analysis

This analysis allows to the user, the identification of differentially expressed known miRNAs from high throughput sequencing data. A complete example of the pipeline can be executed using:

```
miARma>perl miARma Examples/basic_examples/miRNAs/Known_miRNAs/Known_miRNAs_pipeline.ini
```

4.1.1 General features

4.1.1.1. Configuration file

In order to execute miARma-Seq, a configuration file in INI format is mandatory with information about your experiment setup. General information must be provided by the heading **[General]** at the beginning of the configuration file. Although general section is required for any analysis with miARma-Seq a configuration file only with this section will not perform any analysis. See below a detailed explanation in order to configure the different steps of the analysis. This information is mainly oriented to the path of input files and output directories.

The parameters included in this section are:

Mandatory parameters:	
type	Type of analysis to perform with miARma-Seq. Allowed values for this parameter are: miRNA, mRNA or circRNAs. Example: type=miRNA
read_dir	Folder for input files where raw data from high thorughput sequencing in fastq format are located. Example: read_dir=Examples/basic_examples/miRNAs/reads/
label	Name to identify the analysis. This name will appear in the output files and plots. Example: label=Hypoxia
miARmaPath	Folder where miARma-Seq has been installed. Example: miARmaPath=/opt/miARma/
output_dir	Folder to store the results.

	Example: output_dir=Examples/basic_examples/miRNAs/Known_miRNAs/results/
organism	Organism analysed in the experiment. Example: organism=human

Optional parameters:

verbose	Parameter to show the execution data on the screen. Value of 0 for no verbose, otherwise to print "almost" everything. Example: verbose=0
threads	Number of process to run at the same time. The maximum value of this parameter is defined for user's computer. Example: threads=4
stats_file	File where stats data will be saved. Example: stats_file=stats.log
logfile	File to print the information about the execution process. Example: logfile=run_log.log
seqtype	Sequencing procedure of RNA-Seq experiment. Allowed values: Paired/Paired-End or Single/Single-End (by default). Please note that paired-end analysis samples must be named with the final end of "_1" and "_2" before file extension to correctly identify paired samples. Example: SRR873382_1.fastq and SRR873382_2.fastq. Example: seqtype=Paired
strand	Parameter to specify whether the data is from a strand-specific assay. The allowed values are: yes (by default), no or reverse. Example: strand=no

4.1.1.2. Examples of the general information in the configuration file

1) General information of miRNA analysis.- In this example, user is defining the general parameters of the analysis executing miARma from its own directory, the pipeline input files are fastq files from human located in the input directory (Examples/basic_examples/miRNAs/reads/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/Known_miRNAs/results/ in this case) including the name of the experiment (Hypoxia in this example). The analysis will perform with 4 threads and the execution data will not be showed in the screen.

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/ Known_miRNAs/results/
organism=human
strand=yes
```

4.1.2. Quality module

The aim of the Quality module is to provide a simple way to check the quality of our sequenced samples and avoid the inclusion of outliers. This analysis will be performed with FASTQC software and it can be performed before the data processing and after read trimming, in the case of miRNA analysis.

4.1.2.1. Input/Output files

Input: Raw data from high throughput sequencing in fastq format (compressed files are allowed).

Output:

1. HTML report with different plots and statistics of the quality of the data. These files will be saved inside a folder called Pre_fastqc_results under the path specified in output_dir. For each fastq file, an independent quality analysis process will be performed and stored in a folder with the same name of the fastq file. In order to examine the results, a html file called fastqc_report.html is included. Please visit FastQC help page to better understand the FastQC report.

2. Summary results report (xls format) with the main statistics of the analysis. This report will be generated in the output directory provided by the user. In this report, "Quality" section with the path of the quality results can be founded, together with a summary table below with the columns:

-[Filename]- Name of the sample
-[Number of reads] –Number of reads contained in the fastq files.
-[%GC Content]- Proportion of GC content of the reads
-[Read Length]- Length of the reads.
-[Encoding]- Type of encoding of the fastq files.

An example of the summary report can be consulted in the following link.

4.1.2.2. Configuration file

To execute this analysis the heading **[Quality]** must be included in the configuration file. The parameters included in this analysis are:

Mandatory parameters	
prefix	Parameter to define when miARma will perform the quality analysis. Use "pre" to perform a quality analysis for unprocessed reads and "post" for processed reads (after adapter trimming step). miARma also accepts the keyword "both" in case you want the analysis twice: before and after the pre-processing of the reads. Example: prefix=both

4.1.2.3. Examples of configuration file to run Quality analysis

1) Quality analysis of miRNA analysis. In this example, user will perform the quality analysis executing miARma from its own directory, the pipeline input files are fastq files located in the input

directory (Examples/basic_examples/miRNAs/reads/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/Known_miRNAs/results/ in this case).

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/Known_miRNAs/results/
organism=human
strand=yes

[Quality]
prefix=Pre
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/Known_miRNAs/1.Quality/1.Quality.ini
```

To inspect results for a Fastq file named SRR873382, please check Examples/ basic_examples/ miRNAs/DeNovo_miRNAs/results/Pre_fastqc_results/SRR873382_fastqc/fastqc_report.html

4.1.3. Adapter module

The aim of the adapter package is the pre-processing of the reads from a fastq file to remove the adapter sequence and/or the low-quality bases. Adapter module offers numerous options for the trimming of the reads, which can be used as a single option or combined:

- Adapter sequence removal- miARma-Seq includes two alternative software for this purpose: Cutadapt and Reaper. Cutadapt is one of the most used tools for adapter trimming, mainly due to its convenience, efficacy and the several parameters to customize the analysis. For these reasons, Cutadapt is the recommended tool for pre-processing of the reads. Alternatively, miARma-Seq includes Reaper software, which allow adapter trimming with additional features such as filtering for low complexity bases or the possibility to include additional sequences to remove for specific regions of the reads.

- Adapter sequence prediction- If adapter sequence used in the experiment setup is unknown, miARma-Seq will execute Minion software to predict it. Predicted sequences will be checked with Blat software to avoid selection of over-represented biological sequences and the most probable adapter sequence will be selected and provided to Cutadapt or reaper for the trimming process.

- End nucleotides removal- Removal of a specific number of nucleotides from a specific end of the read is usual for example when quality analysis showed low quality nucleotides in a specific end of the reads. For this purpose, miARma-Seq includes an in-house tool to remove a specific number of nucleotides from 3' or 5' end, called Adaptrimming.

4.1.3.1. Input/Output files

Input: Raw data from high throughput sequencing in fastq format (compressed files are allowed).

Output:

1. Fastq files with the trimmed reads in the output directory provided by the user in the "cutadapt_results", "reaper_results" or "adaptrimming_results" folder. Since different trimming tools can be executed simultaneously, a results folder will be generated for each analysis.

2. Summary results report (xls format) with the main statistics of the analysis. This report will be generated in the output directory provided by the user. In this report, "Adapter Removal" section with the path of the adapter results can be founded. The information obtained is different according to the trimming tool selected.

For read trimming using Cutadapt tool, a summary table will include the columns:

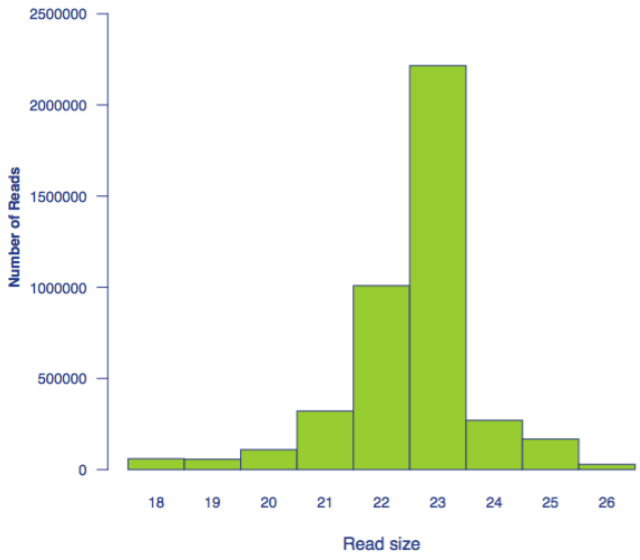
-[Filename]- Name of the sample
-[Processed Reads]- Initial number of reads contained in the fastq file
-[Processed Bases]- Initial number of bases contained in the fastq file
-[Trimmed reads]- Number of reads after trimming process
-[Trimmed bases]- Number of bases after trimming process
-[Quality-Discarded]- Number of reads discarded due to low quality
-[Too short reads]- Number of reads discarded due to a length less than the minimum established
-[Too long reads]- Number of reads discarded due to a length greater than the maximum established

For read trimming using Reaper tool, the summary table will include the columns:

-[Filename]- Name of the sample
-[Processed Reads]- Initial number of reads contained in the fastq file
-[Accepted Reads]- Number of reads after trimming process
-[Discarded Reads]- Number of reads discarded in the trimming process
-[Size-Discarded]- Number of reads discarded due to their length
-[Quality-Discarded]- Number of reads discarded due to low quality
-[Low Complexity-Discarded]- Number of reads discarded due to their low complexity.

An example of the summary report can be consulted in the following link.

3. Plot with the read size distribution of each sample. This plot shows the number of reads of each size after the pre-processing of the reads. This plot will look like this:



4.1.3.2. Configuration file

To execute this analysis the heading **[Adapter]** must be included in the configuration file. The parameters included in this analysis are:

Mandatory parameters	
adaptersoft	Specific software to perform the pre-processing of the reads. As state above the tools Cutadapt and Reaper are implemented in miARma-Seq, as well as, an in house tool, called adaptrimming, to trim the end of the reads. These tools can be selected alone or in combination. Thus allowed values for this parameter are: cutadapt, reaper, adapttrimming, cutadapt-reaper, cutadapt-adapttrimming, reaper-adapttrimming or cutadapt-reaper-adapttrimming. Note that each specific tool might require specific additional parameters for its correct execution. See examples below to deep in the analysis possibilities. Example: adaptersoft= cutadapt-reaper
Optional parameters	
adapter	Adapter sequence to be removed in the analysis with Cuadapt or Reaper (if this sequence is not provided Minion software will be used to predict one). Example: adapter=ATCTCGTATGCCGTCTTCTGCTTGAA
min	Minimum length of the sequence read to keep during the trimming with Cutadapt and Reaper software. This value is predefined as 15 by default to miRNA analysis. Example: min=18
Specific parameters for minion	
adaptpredictionnumber	Number of adapter predictions to show by minion in the adapter prediction analysis. By default Minion shows 2 sequences, which will be automatically checked with Blat to discard biological sequences. This value is recommended for the analysis but if adapter sequence is not founded with

	it, we recommend increasing the number of the predicted sequences. Example: adaptpredictionnumber=4
minionadaptersequence	Known adapter sequence to compare with the sequence predicted by Minion. This parameter is useful for check the reliability of the known adapter sequences. Example: minionadaptersequence=ATCTCGTATGCCGTCTTCTGCTTGAA

Specific parameters for Cutadapt:

max	Maximum length of the sequence read to keep during the trimming with Cutadapt software. This value is predefined as 35 by default to miRNA analysis. Example: max=26
min_quality	Minimum quality value of the sequence read to use with Cutadapt software. This value is predefined as 0 by default to miRNA analysis. Example: min_quality=25
cutadaptparameters	Other parameters to perform the Cutadapt analysis using the Cutadapt recommended syntaxis in Cutadapt's user guide. Example: cutadaptparameters= -O 4
adapter_file	Complete path of the file to specify a different adapter for each fastq file (recommended for multiplexed files). This is a tabulated file that contains two columns: the name of the fastq file and the correspondent adapter sequence to remove it. The name of the fastq file in this file must be exactly identical to fastq file of the input directory; otherwise adapter will not be removed from this sample. For example, for the hypoxia example input the correspondent adapter_file will contain the next information: Filename Adapter SRR873382.fastq.bz2 ATCTCGTATGCCGTCTTCTGCTTGA SRR873383.fq.bz2 ATCTCGTATGCCGTCTTCTGCTTG SRR873384.fastq.bz2 ATCTCGTATGCCGTCTTCTGCTT SRR873385.fq.bz2 ATCTCGTATGCCGTCTTCTGCT SRR873386.fastq.bz2 TCTCGTATGCCGTCTTCTGCTTGAA SRR873387.fastq.bz2 CTCGTATGCCGTCTTCTGCTTGAA SRR873388.fastq.bz2 TCGTATGCCGTCTTCTGCTTGAA SRR873389.fastq.bz2 CGTATGCCGTCTTCTGCTTGAA Example: adapter_file=Examples/basic_examples/miRNAs/data/Adapter_file.txt

Specific parameters for Reaper

geom	Geometry used in the analysis with Reaper referring to the position of the barcode in the read. The geometry can be: No barcode (no-bc) (default value), 3'end (3p-bc) or 5'end (5p-bc). Note that other geometries different than the default value (no-bc) require the metafile parameter. Example: geom= 3p-bc
metafile	Metadata file with the information of the adapter sequences to use in the analysis with Reaper. This parameter is mandatory for 3p-bc and 5p-bc geometries. See reaper instructions to generate this file. Example: metafile= /data/reaper_metafile.txt
tabu	Tabu sequence to remove the read which contain this sequence with

	Reaper Software (usually this sequence is 5' primer sequence) Example: tabu=GTTCAGAGTTCTACAGTCCGACGATC
reaperparameters	Other parameters to perform the Reaper analysis using the Reaper recommended syntaxis in Reaper user guide. Example: reaperparameters=-3p-prefix 12/2/0/0 -dust-suffix-late 20

Specific parameters for Adapttrimmig

trimmingnumber	Number of nucleotides to remove from the sequence and the quality data of the reads with AdaptTrimming function Example: trimmingnumber=12
readposition	End of the read to remove the nucleotides with AdaptTrimming function. Nucleotides can be removed from 3'end (3) or 5'end (5). Example: readposition=5

4.1.3.3. Examples of configuration file to run Adapter module

1) Adapter removal of a known adapter sequence by Cutadapt: In this case, user provides the adapter sequence (ATCTCGTATGCCGTCTTCTGCTTGAA) and wants to remove it from each read using Cutadapt software. User will execute miARma from its own directory, the pipeline input files are fastq files located in the input directory (Examples/basic_examples/miRNAs/reads/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/results/ in this case). Cutadapt analysis will be performed keeping the reads with a minimum read length of 18 and a maximum of 26 nucleotides and with a minimum quality value of 25.

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/Known_miRNAs/results/
organism=human
strand=yes

[Adapter]
adapter=ATCTCGTATGCCGTCTTCTGCTTGAA
adaptersoft=Cutadapt
min=18
max=26
min_quality=25
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/Known_miRNAs/2.Adapter/2.1.Cutadapt_adapter_removal.ini
```

2) Removing nucleotides from an end of the read with Adapttrimming: In this example, user will perform the trimming of 12 nucleotides of the 5'end of the reads with Adaptrimming tool. User will execute miARma from its own directory, the pipeline input files are fastq files located in the input

directory (Examples/basic_examples/miRNAs/reads/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/results/in this case).

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/Known_miRNAs/results/
organism=human
strand=yes


[Adapter]
adaptersoft=Adapttrimming
trimmingnumber=12
readposition=5
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/Known_miRNAs/2.Adapter/2.2.AdapterTrimming_adapter_removal.ini
```

3) Adapter removal of a known adapter sequence with Reaper: In this case, user knows the adapter sequence (ATCTCGTATGCCGTCTTCTGCTTGAA) and want to remove it from the reads using Reaper. User will execute miARma from its own directory, the pipeline input files are fastq files located in the input directory (Examples/basic_examples/miRNAs/reads/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/results/ in this case). Reaper analysis will be performed with dust threshold for read suffix in 20, with a minimum read length of 18, and a an alignment that matches the start of the 3' adapter with the end of the read and with a length of 12, an edit distance of 2, and without gap size and offset.

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/results/
organism=human
strand=yes


[Adapter]
adapter=ATCTCGTATGCCGTCTTCTGCTTGAA
adaptersoft=Reaper
reaperparameters=-3p-prefix 12/2/0/0 -dust-suffix-late 20
min=18
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/Known_miRNAs/2.Adapter/2.3.Reaper_adapter_removal.ini
```

4) Adapter removal of an unknown adapter sequence with Cutadapt: In this example, user will perform the removal of an unknown adapter sequence with Cutadapt. User will execute miARma from its own directory, the pipeline input files are fastq files located in the input directory (Examples/basic_examples/miRNAs/reads/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/results/ in this case). In this case, adapter sequence is unknown and miARma-Seq will use Minion software to predict it. Minion will show 4 results to be tested in Blat as a possible adapter sequence. Cutadapt analysis will be performed keeping the reads with a minimum read length of 18 and a maximum of 26 nucleotides and with a minimum quality value of 25.

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/results/
organism=human
strand=yes


[Adapter]
adaptersoft= Cutadapt
min=18
max=26
min_quality=25
adaptpredictionnumber=4
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/2.Adapter/2.4.Cutadapt_minion_adapter_removal.ini
```

5) Adapter removal of a known adapter with Cutadapt and Reaper: In this example, user will perform the removal of a known adapter sequence with both, Cutadapt and Reaper software. User will execute miARma from its own directory, the pipeline input files are fastq files located in the input directory (Examples/basic_examples/miRNAs/reads/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/results/ in this case). In this case, user knows the adapter sequence (ATCTCGTATGCCGTCTTCTGCTTGAA) and want to remove it from the reads using both softwares. Cutadapt analysis will be performed keeping the reads with a minimum read length of 18 and a maximum of 26 nucleotides and with a minimum quality value of 25. In the case of Reaper,

the analysis will be performed with dust threshold for read suffix in 20, with a minimum read length of 18, and a an alignment that matches the start of the 3' adapter with the end of the read and with a length of 12, an edit distance of 2, and without gap size and offset.

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/results/
organism=human
strand=yes


[Adapter]
adapter=ATCTCGTATGCCGTCTTCTGCTTGAA
adaptersoft=Reaper-Cutadapt
reaperparameters=-3p-prefix 12/2/0/0 -dust-suffix-late 20
min=18
max=26
min_quality=25
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/2.Adapter/2.5.Cutadapt_reaper_adapter_removal.ini
```

6) Adapter removal of different adapter sequences with Cutadapt: In this example, user will perform the removal of different adapter sequences for each sample with Cutadapt. User will execute miARma from its own directory, the pipeline input files are fastq files located in the input directory (Examples/basic_examples/miRNAs/reads/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/results/ in this case). In this case, user knows the different adapter sequences for each sample and these are specified in the adapter_file. Cutadapt analysis will be performed keeping the reads with a minimum read length of 18 and a maximum of 26 nucleotides and with a minimum quality value of 25.

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/results/
organism=human
strand=yes


[Adapter]
adapter_file=Examples/basic_examples/miRNAs/data/Adapter_file.txt
adaptersoft=Cutadapt
```

```
min=18
max=26
min_quality=25
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/2.Adapter/2.6.Cutadapt_multiplexed_adapter_removal.ini
```

4.1.4. Aligner module

The aim of the aligner module is to align sequenced reads against the reference genome. For miRNA analysis, miARma-Seq has implemented Bowtie1 and Bowtie2, which can be used as a single option or combined. The reference genome to align the reads against is mandatory. The reference genome can be provided in two ways:

i) As a pre-built Bowtie indexes, which can be downloaded for most of organism from Bowtie1 webpage and Bowtie2 webpage. Note that, Bowtie1 and Bowtie 2 use different index formats. Bowtie 1 index must have .ebwt extension and is only allowed for Bowtie1 analysis, while Bowtie 2 index must have .bt2 extension and is only allowed for Bowtie2 analysis. To download bowtie indexes of human genome 19 used in the examples see section 3.3.2. and 3.3.3.

ii) As a genome sequence in fasta format. This option is allowed for Bowtie1 and Bowtie2 analysis, since miARma-Seq will generate the corresponding genome index for the selected analysis. This option allows to the user the analysis of any organism with a sequenced genome. Nevertheless, the generation of a new index is a long process, which can take several hours. For this reason, we strongly recommend generate the specific index at the first time and use it for future analysis. To download the fasta of human genome 19 used in the examples see section 3.3.1.

4.1.4.1. Input/Output files

Input: Raw data or pre-processed data from high throughput sequencing in fastq format.

Output:

1. Aligned files in SAM/BAM format saved in the output directory provided by the user within the "bowtie1_results" or "bowtie2_results" folder.

2. Summary results report (xls format) with the main statistics of the analysis. This report will be generated in the output directory provided by the user. In this report, "Alignment" section with the path of the aligner results can be founded, together with a summary table below with the columns:

-[Filename]- Name of the sample.

-[Processed Reads]- Initial number of reads contained in the fastq file (trimmed file).

-[Aligned reads]- Number of aligned reads against the reference genome provided.

-[Multimapping reads]- Number of reads with multiple alignment.

-[Overall alignment]- Proportion of aligned read/total number of reads.
-[Fail to align]- Number of reads that fail to align.

An example of the summary report can be consulted in the following link.

4.1.4.2. Configuration file

To execute this analysis the heading **[Aligner]** must be included in the configuration file. The parameters included in this analysis are:

Mandatory parameters	
aligner	Specific software to perform the alignment against the corresponding index. As state above the tools Bowtie 1 and Bowtie 2 are implemented in miARma-Seq. These tools can be selected alone or in combination. Thus allowed values for this parameter are: Bowtie1, Bowtie2, Bowtie1-Bowtie2 and Bowtie2-Bowtie1. Note that, each specific aligner requires a specific genome index. See examples below to deep in the analysis possibilities. Example: aligner=Bowtie1-Bowtie2
Specific parameters for Pre-built Bowtie indexes:	
bowtie1index	Path of the pre-built Bowtie 1 index to the alignment of the reads. This index can be downloaded from Bowtie 1 web page as stated above or can be built from genome sequence in fasta format. In any case, the index is composed by various files with the same index name, followed by a number and the .ebwt extension (i.e. bw1_homo_sapiens19.1.ebwt, bw1_homo_sapiens19.2.ebwt, bw1_homo_sapiens19.3.ebwt, etc). See the example below to define bowtie1index parameter supposing that index would be placed in Genomes/Indexes/bowtie1/human/. Example: bowtie1index=Genomes/Indexes/bowtie1/human/bw1_homo_sapiens19
bowtie2index	Path of the pre-built Bowtie 2 index to the alignment of the reads. This index can be downloaded from Bowtie 2 web page as stated above or can be built from genome sequence in fasta format. In any case, the index is composed by various files with the same index name, followed by a number and the .bt2 extension (i.e. bw2_homo_sapiens19.1.bt2, bw2_homo_sapiens19.2.bt2, bw2_homo_sapiens19.3.bt2, etc). See the example below to define bowtie2index parameter supposing that index would be placed in Genomes/Indexes/bowtie2/human/. Example: bowtie2index=Genomes/Indexes/bowtie2/human/bw2_homo_sapiens19
Specific parameters to create a new index from fasta file:	
fasta	Path of the genome sequence in fasta format to build the correspondent index. Example: Genomes/Indexes/ homo_sapiens19.fa
indexname	Name to write in the generated index files. Example: hg19

Optional parameters	
bowtiemiss	Maximum number mismatches in seed alignment in bowtie analysis. Allowed values are: 0-3 for Bowtie1 analysis (2 by default) and 0-1 for Bowtie 2 analysis (0 by default). Example: bowtiemiss=1
bowtielength	Length of the seed substrings to align during multiseed alignment. Smaller values make alignment slower but more senstive. Allowed values are comprised between 5-32. Example: bowtielength=19
bowtie1parameters	Other parameters to perform the Bowtie 1 analysis using the Bowtie 1 recommended syntaxis in Bowtie 1 user guide. Example: bowtie1parameters=--best --nomaqround -e 70 -k 1
bowtie2parameters	Other parameters to perform the Bowtie 2 analysis using the Bowtie 2 recommended syntaxis in Bowtie 2 user guide. Example: bowtie2parameters= -N 1 -D 10

4.1.4.3. Examples of configuration file to run Aligner module

1) Alignment with Bowtie1 against a pre-built index: In this example, user will perform the alignment of processed reads against a Bowtie 1 index located in the directory (Examples/basic_examples/miRNAs/Genomes/Indexes/bowtie1/human/ bw1_homo_sapiens19 in this example). Note that miARma-Seq automatically searches for processed reads, thus, in this example is assumed that the [Adapter] module has been previously executed. Otherwise, miARma-Seq will align the reads located at the input directory (Examples/basic_examples/miRNAs/reads in the example). User will execute miARma from its own directory. The alignment will be performed a allowing 0 mismatches in seed alignment, with 19 of length of seed substrings, eliminating strand bias and preventing the rounding of the quality values.

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/Known_miRNAs/miRNAs/results/
organism=human
strand=yes


[Aligner]
aligner=Bowtie1
bowtie1index= Genomes/Indexes/bowtie1/human/bw1_homo_sapiens19
bowtiemiss=0
bowtieleng=19
bowtie1parameters=--best --nomaqround
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/Known_miRNAs/3.Aligner/3.1.Bowtie1_prebuilt_index.ini
```

2) Alignment by Bowtie2 without index files: In this example, user will perform the alignment of processed reads against a new index generated by miARma-Seq. It will use the fasta-sequenced file located in the genome's directory. Note that miARma-Seq automatically searches for processed reads, thus, in this example it is assumed that the [Adapter] modules has been previously executed and results are located in their corresponding directory (Examples/basic_examples/miRNAs/results/ in this case). Otherwise, miARma-Seq will take the reads located at the input directory (Examples/basic_examples/miRNAs/reads in the example). User will execute miARma from its own directory. The alignment will be performed a allowing 0 mismatches in seed alignment with 19 of length of seed substrings.

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/Known_miRNAs/results/
organism=human
strand=yes


[Aligner]
aligner=Bowtie2
fasta=Genomes/Indexes/bowtie2/human/homo_sapiens19.fa
indexname=bw2_homo_sapiens19
bowtiemiss=0
bowtieleng=19
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/Known_miRNAs/3.Aligner/3.2.Bowtie2_index_from_fasta.ini
```

3) Alignment with Bowtie1 and Bowtie2 against a pre-built index: In this example, user will perform the alignment of processed reads using both tools, Bowtie 1 and Bowtie 2, against their corresponding indexes located in index directory (Examples/basic_examples/miRNAs/Genomes/Indexes/bowtie1/human/ and Examples/basic_examples/miRNAs/Genomes/Indexes/bowtie2/human/ in this example). Note that miARma-Seq automatically searches for processed reads, thus, in this example it is assumed that the [Adapter] module has been previously executed and results are located in their corresponding directory. Otherwise, miARma-Seq will take the reads located at the input directory (Examples/basic_examples/miRNAs/reads in the example). User will execute miARma from its own directory. The alignment will be performed a allowing 0 mismatches in seed alignment, with 19 of length of seed substrings in both, Bowtie 1 and Bowtie 2 analysis, and eliminating strand bias and preventing the rounding of the quality values in Bowtie 1 analysis.

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
```

```
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/Known_miRNAs/results/
organism=human
strand=yes


[Aligner]
aligner=Bowtie1-Bowtie2
bowtie1index=Genomes/Indexes/bowtie1/human/bw1_homo_sapiens19
bowtie2index=Genomes/Indexes/bowtie2/human/bw2_homo_sapiens19
bowtiemiss=0
bowtieleng=19
bowtie1parameters=--best --nomaqround
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/Known_miRNAs/3.Aligner/3.3.Bowtie1_Bowtie2_prebuilt_index.ini
```

4.1.5. ReadCount module

The aim of the ReadCount module is the summarization of mapped reads into genomic features such as genes, exons, promoter, gene bodies, genomic bins and chromosomal locations. For miRNA analysis, miARma-Seq has implemented featureCounts.

4.1.5.1. Input/Output files

Input: Aligned files in SAM/BAM format.

Output:

1. Tabulated text file with the entities and the correspondent counts in the output directory provided by the user within "Readcount_results" folder. In this file, each row corresponds to an miRNA and each column to the number of reads of that selected feature in each sample. The names of the columns are the name of each sample. Example:

	SRR873382	SRR873383	SRR873384	SRR873385	SRR873386	SRR873387	SRR873388	SRR873389
hsa-let-7a-2-3p	0	0	0	0	0	0	0	0
hsa-let-7a-3p	180	163	101	92	121	82	124	86
hsa-let-7a-5p	270720	306188	376418	204502	218299	158334	209985	224984
hsa-let-7b-3p	20	40	8	8	7	10	13	8
hsa-let-7b-5p	90881	121568	46482	76478	78113	88608	76335	87498
hsa-let-7c-3p	4	12	1	3	5	2	3	2
hsa-let-7c-5p	13669	17626	10830	11561	12842	11041	10913	11904

2. Summary results report (xls format) with the main statistics of the analysis. This report will be generated in the output directory provided by the user. In this report, "ReadCount" section with the path of the readcount results can be found, together with a summary table below with the columns:

-[Filename]- Name of the sample.
-[Processed Reads]- Initial number of reads contained in the fastq file (trimmed file).
-[Assigned reads]- Number of assigned reads using the database in gtf format provided.
-[Strand]- Type of experiment.
-[Number of identified entities]- Number of identified entities

An example of the summary report can be consulted in the following link.

4.1.5.2. Configuration file

To execute this analysis the heading **[Readcount]** must be included in the configuration file. The parameters included in this analysis are:

Mandatory parameters	
database	File in GTF format used to calculate the number of reads. Example: database=Examples/basic_examples/miRNAs/data/miRBase_Annotation_20_for_hsa_mature_miRNA.gtf
Optional parameters	
featuretype	Feature type (3rd column in GTF file) to be used, all features of other type are ignored (default:exon) for featureCounts analysis Example: featuretype=miRNA
seqid	GTF attribute to be used as feature ID. This parameter has a default value of gene_id for miRNA analysis. Example: seqid=transcript_id
quality	Quality value threshold to avoid counting low quality reads. Example: quality=10
parameters	Other featureCounts parameters to perform the analysis using the featureCounts recommended syntaxis. Example: parameters= -d 50 -D 600

4.1.5.2. Examples of configuration file to run ReadCount module

1) Quantification of mRNAs by Readcount: In this example, user will perform the read summarization corresponding to miRNAs taking as a reference the GTF from miRBase v20 (to download the gtf file used in this example see section 3.2.1.). User will execute miARma from its own directory, the input files are aligned sam files from 3rd example from 4.1.4.3. located in the input directory (Examples/basic_examples/miRNAs/results/ in the example) section (example file 3.3. from miARma example files) and the results will be saved in results directory

(Examples/basic_examples/miRNAs/results/ in this case). The quantification will be performed from a strand specific assay and with a minimum quality value of 10.

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/Known_miRNAs/results/
organism=human
strand=yes

[ReadCount]
database=Examples/basic_examples/miRNAs/data/miRBase_Annotation_20_for_hsa_mature_miRNA.gtf
seqid=transcript_id
quality=10
featuretype=miRNA
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/Known_miRNAs/4.ReadCount/4.1.ReadCount.ini
```

4.1.6. Differential Expression module

The aim of this module is to perform the differential expression analysis between different experimental conditions. For this purpose, miARma-Seq is implemented with NOISeq software and EdgeR software. Both are valuable tools to identify differentially expressed (DE) elements, which covers different requirements. edgeR is a widely employed tool for differential expression analysis that allows not only the identification of DE elements between two experimental conditions but more complicated comparisons in the same analysis process. On other hand, Noiseq allows the simulation of technical replicates to increase the reliability of the results, when no replicates are available for the analysis.

4.1.6.1. Input/Output files

Input: Tabulated file with the counts of the reads. In this file, each row corresponds to a feature and each column to the number of reads of that feature. The names of the columns are the name of each sample. Example:

	SRR873382	SRR873383	SRR873384	SRR873385	SRR873386	SRR873387	SRR873388	SRR873389
hsa-let-7a-2-3p	0	0	0	0	0	0	0	0
hsa-let-7a-3p	180	163	101	92	121	82	124	86
hsa-let-7a-5p	270720	306188	376418	204502	218299	158334	209985	224984
hsa-let-7b-3p	20	40	8	8	7	10	13	8

hsa-let-7b-5p	90881	121568	46482	76478	78113	88608	76335	87498
hsa-let-7c-3p	4	12	1	3	5	2	3	2
hsa-let-7c-5p	13669	17626	10830	11561	12842	11041	10913	11904

Output:

1. Tabulated results files (excel compatible) with the entities differentially expressed (DE) and the statistical values of the analysis for any of the comparison between the different experimental conditions. According to the selected tool for the analysis, the format of the results differs. Specific format will be detailed below.

- **EdgeR results**- EdgeR results will be located in the "EdgeR_results" directory in the output_dir directory defined by the user. The results with the DE entities of each condition will be saved in different files. The name of the results files will be constructed as follows:

(Label_defined_by_user)_(Adapter_tool)_(Aligner_tool)_EdgeR_results_(label_of_the_comparison).xls

Example: For the comparison at 16 hours of hypoxia experiment performed with Cutadapt and Bowtie1 tools, the resultant file will be named as : Hypoxia_cut_bw1_EdgeR_results_Comp_16.xls

EdgeR result file contains 5 columns:

- [Entity]- Name of the DE entity, which according to the experiment could be the name of miRNAs, mRNAs or circRNAs.
- [logFC]- Log2-fold- change value.
- [logCPM]- Log2 counts-per-million.
- [Pvalue]- Probability value.
- [FDR]- False discovery rate obtained by Benjamini and Hochberg's algorithm

Example:

	logFC	**logCPM**	**PValue**	**FDR**
hsa-miR-503-3p	-1.890652364	4.098298363	1.60E-07	9.39E-05
hsa-miR-210-3p	1.702083488	8.770057331	1.37E-06	0.000400593
hsa-miR-4521	-3.308571177	1.407874884	3.48E-06	0.000678891
hsa-miR-210-5p	2.666024708	2.435923273	1.24E-05	0.001809943
hsa-miR-222-5p	-1.835901985	2.791957916	3.81E-05	0.004455154

- **Noiseq results**- Noiseq results will be located in the "Noiseq_results" directory in the output_dir directory defined by the user. Noiseq generates a results file with the statistical values of every expressed entity for each condition. The name of this file will be constructed as follows:

(Label_defined_by_user)_(Adapter_tool)_(Aligner_tool)_Noiseq_results_(label_of_the_comparison).xls

Example: For the comparison at 16 hours of hypoxia experiment performed with Cutadapt and Bowtie1 tools, the resultant file will be: Hypoxia_cut_bw1_Noiseq_results_Comp_16.xls

Both files contain 7 columns:

- [Entity]- Name of the DE entity, which according to the experiment could be the name of miRNAs, mRNAs or circRNAs.
- [Condition1_mean]- Expression values for condition 1.
- [Condition2_mean]- Expression values for condition 2.
- [M] - log2-ratio of the two conditions.
- [D] - value of the difference between conditions.
- [prob] - probability of differential expression.
- [ranking] – summary statistic of "M" and "D" values.

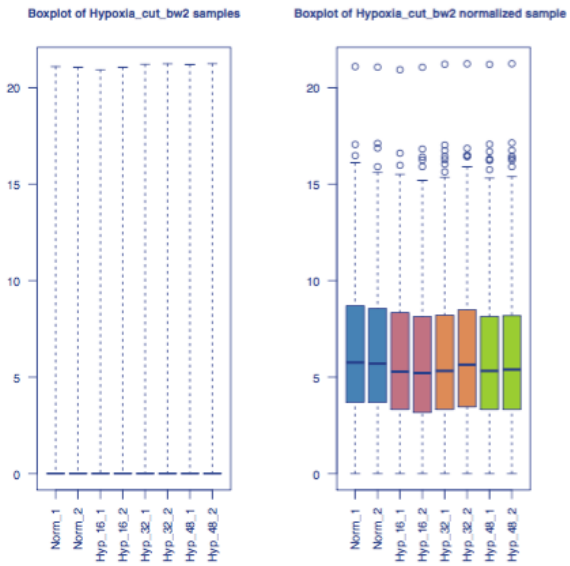
Example:

	T48h_mean	**T0h_mean**	**M**	**D**	**prob**	**ranking**
hsa-miR-19b-3p	96.66	392.96	-2.02	296.30	0.9358	-296.31
hsa-miR-296-3p	27.06	159.27	-2.55	132.20	0.9280	-132.22
hsa-miR-7-5p	321.28	970.53	-1.59	649.24	0.9168	-649.24
hsa-miR-22-3p	90.79	319.31	-1.81	228.51	0.9090	-228.51
hsa-miR-30d-5p	20290.89	8331.66	1.28	11959.22	0.9046	11959.2

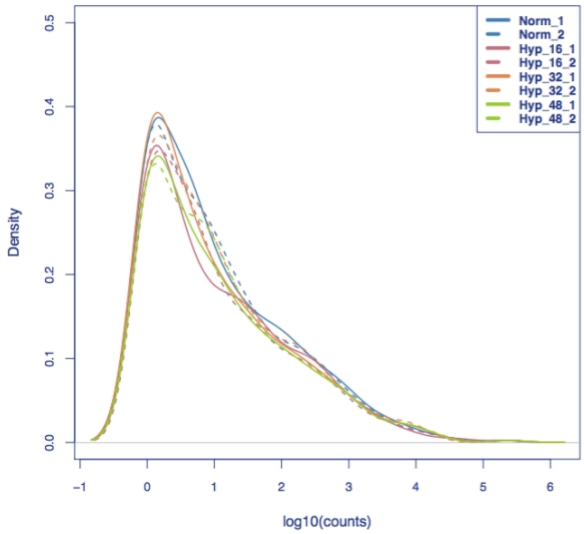
2. Exploratory plots of the analysis. miARma-Seq provides a exhaustive PDF report with different plots in order to make easier to the user the interpretation of the data. This report contains:

2.1. Analysis of the distribution of the reads in the samples. The detailed inspection of the distribution of the reads in the different samples allows to the user identify samples with abnormal distribution of the reads. These samples are recommended to be removed from the analysis since may introduce noise or affect to the final results. In order to inspect the distribution of the reads in the different samples, miARma-Seq generates two kinds of plots:

- Boxplot of the distribution of the counts. The first page of the report contains 2 boxplots with the distribution of the counts, before (left) and after (right) the normalization process. The log2(number of counts) is represented for each sample. Boxplot of non-normalized data usually will have a lower limit near to – infinite due to the miRNAs with no counts. The different replicates will be represented with the same colour. The expected boxplot will look like this:

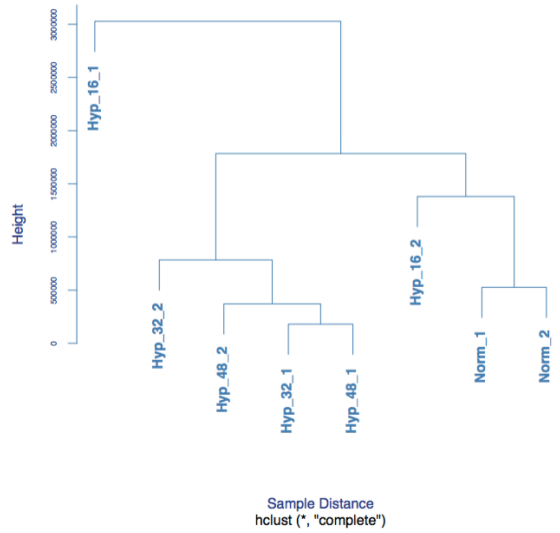


- Density plot of the distribution of the counts. The second and third page of the report contains 2 density plots with the distribution of the counts, before (second page) and after (third) the normalization process. The plot represents the density of the log10 of the counts for each sample. The different replicates will be represented with the same colour. These plots will look like this:

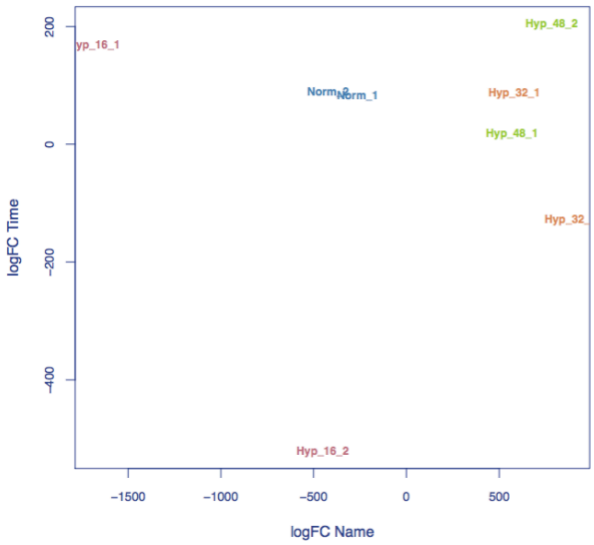


2.2. Analysis of the samples similarity- In order to examine the quality of the data obtained in the experiment, miARma-Seq has implemented different plots, which allows the inspection of the diversity between the samples. For a good quality experiment, the samples belonging to the same experimental conditions should present more similarity between them than with the samples of others experimental conditions. Thus, with these analyses user can identify samples with low quality to remove from the analysis.

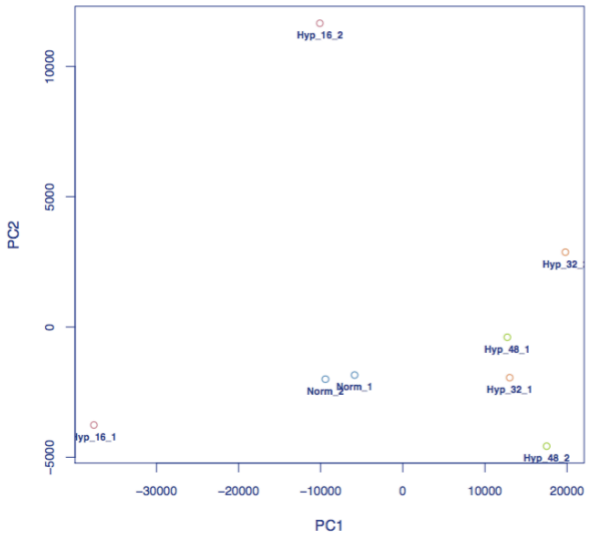
-Hierarchical clustering of the samples: The hierarchical clustering plots, before and after normalization process, classify the samples according to their similarity. The distance of the branch is proportional to the sample distance.



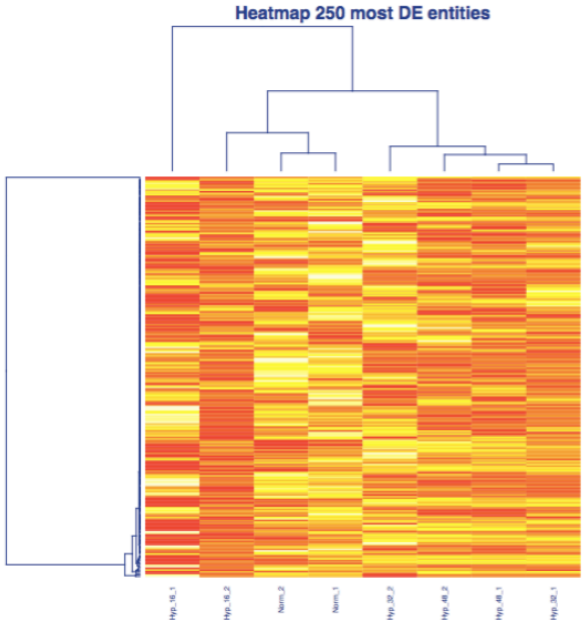
-Multidimensional plot (MDS): The MDS plot divides the samples in a two-dimensional plot according to their similarity. Each sample is represented with its name and coloured according to the experimental condition that belongs to.



-Principal Component Analysis (PCA) plot: The PCA plot divides the samples in a two-dimensional plot according to their similarity. Each sample is represented with its name and coloured according to the experimental condition that belongs to.



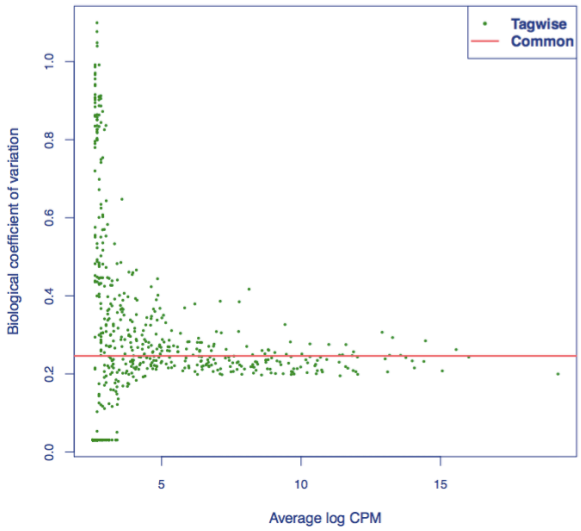
-Heatmap: The heatmap allows to the user evaluate the similarity between the samples according to the 250 most expressed miRNAs expression.



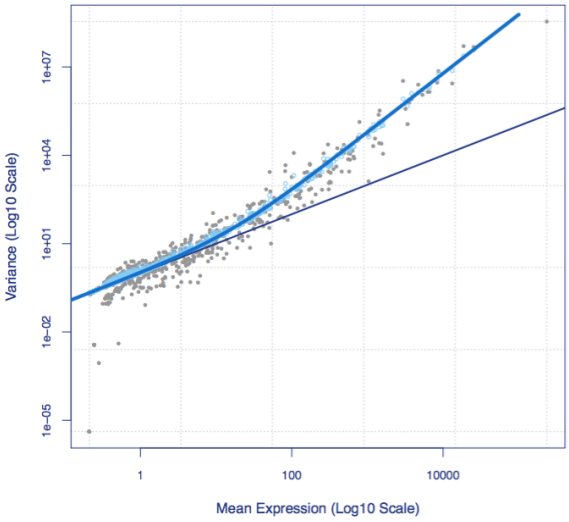
3. Results plots of the analysis.- miARma-Seq generates a PDF report with plots to explore the results with both tools, EdgeR and Noiseq.

3.1. Results plots with EdgeR:

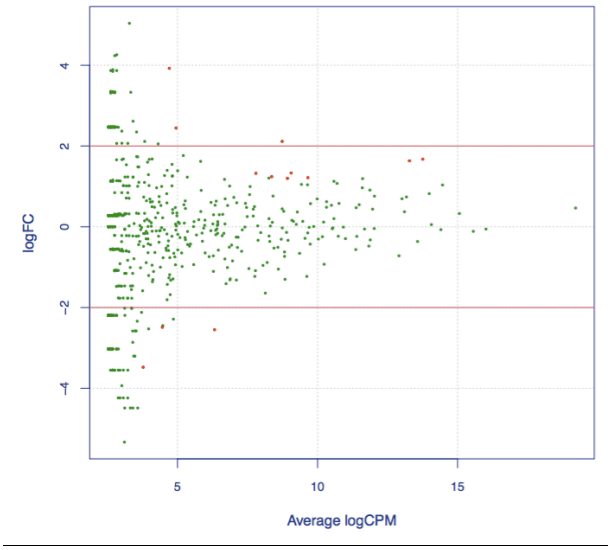
-Biological Variation Plot: The square root of dispersion is the coefficient of biological variation (BCV). This plot illustrates the relationship of biological coefficient of variation versus mean log CPM.



-Mean Variance Plot: This plot can be used to explore the mean-variance relationship; each dot represents the estimated mean and variance for each gene, with binned variances as well as the trended common dispersion overlaid.

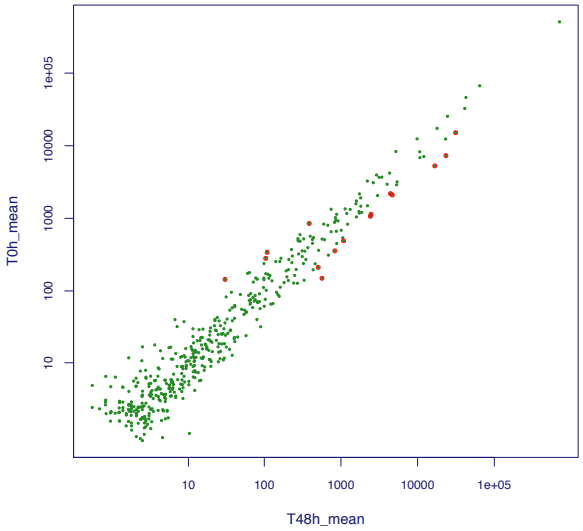


-Expression Plot: miARma-Seq generates one expression plot for each comparison. This plot shows all the logFCs against average count size, highlighting the DE genes.

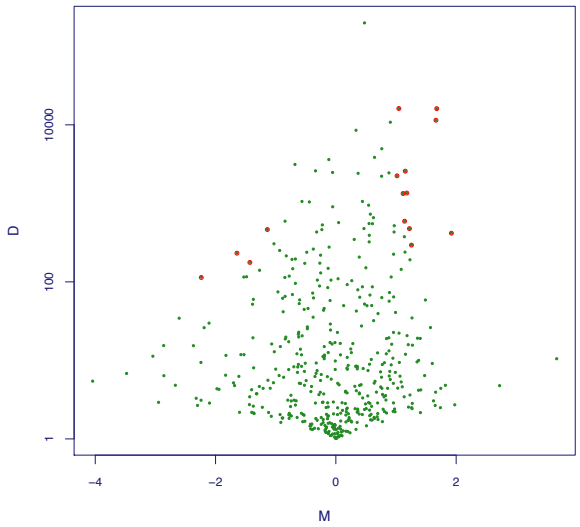


3.2. Results plots with Noiseq: For each comparison a PDF report with the results plot is generated

- Expression Plot: Summary plot of the expression values for both conditions (green), where differentially expressed genes are highlighted (red)



- MD Plot: Summary plot for (M,D) values (green) and the differentially expressed genes (red).



4. Summary results report (xls format) with the main statistics of the analysis. This report will be generated in the output directory provided by the user. In this report, "Differential Expression Analysis" section with the path of the Differential Expression Analysis results can be founded for each tool EdgeR and Noiseq. Each tool shows different information.

For EdgeR analysis the summary table shows the columns:

-[Comparison]- Name of the comparison defined in the contrastfile.
-[File]- Name of the file with the DE elements.
-[Number of DE elements (Pval <=0.05)]- Number of DE elements with a p-value <=0.05.
-[Number of DE elements (FDR <=0.05)]- Number of DE elements with a FDR <=0.05.

For Noiseq analysis the summary table shows the columns:

-[Comparison]- Name of the comparison defined in the contrastfile.
-[File]- Name of the file with the DE elements.
-[Number of DE elements (Prob >=0.8)Number of DE elements with a p-value <=0.05.

An example of the summary report can be consulted in the following link.

4.1.6.2. Configuration file

To execute this analysis the heading **[DEAnalysis]** must be included in the configuration file. The parameters included in this analysis are:

Mandatory parameters	
desoft	Specific software to perform the differential expression analysis. As state above the tools EdgeR and Noiseq are implemented in miARma-Seq. These tools can be selected alone or in combination. Thus allowed values for this parameter are: edger, noiseq or edger-noiseq. Note that, each specific tool requires specific

	parameters. See examples below to deep in the analysis possibilities. Example: desoft=EdgeR-Noiseq
targetfile	Complete path of the target file. This is a tabulated file that contains the experimental condition of each sample. The first column of this file must coincide with the column names of the input files. Note that, only those samples present in the target file will be used for the analysis. The second column must contain the names of the samples to be used in the plots, and the next columns the condition of each factor. For example, for the input previously showed the correspondent target file will contain the next information: Filename Name Time SRR873382 Norm_1 T0h SRR873383 Norm_2 T0h SRR873384 Hyp_16_1 T16h SRR873385 Hyp_16_2 T16h SRR873386 Hyp_32_1 T32h SRR873387 Hyp_32_2 T32h SRR873388 Hyp_48_1 T48h SRR873389 Hyp_48_2 T48h In this example, the first column “Filename” contain the name of the samples obtained from SRA, the second column “Name” contain the names to use in the exploratory plots and the third column “Time” corresponds to the experimental condition, which in this case is the number of hours in hypoxic conditions. This target file can be downloaded as stated in section 3.2.1. Example: targetfile=Examples/basic_examples/miRNAs/data/targets.txt
contrastfile	Path of the contrast file o perform the DE analysis with EdgeR. This file has one column with the contrasts user want to evaluate. The syntax of the contrast should be: name_of_contrast=contrast to evaluate. Any type of contrast can be done but condition name must be one of the conditions present in targets file. In addition, contrast must differ of 0 (ie: cond=WT-WT is not allowed). There is no limit in the number of contrasts. For example, for the input previously showed the correspondent contrast file will contain the next information: "Name" "Comp_16=T16h-T0h" "Comp_32=T32h-T0h" "Comp_48=T48h-T0h" In this example, there are 3 different contrasts conditions: T16h-T0h, T32h-T0h and T48h-T0h, which will be evaluated as separated analysis, generating a xls file with the correspondent results for each condition. This contrast file can be downloaded as stated in section 3.2.1. Example: contrastfile=Examples/basic_examples/miRNAs/data/contrast.txt
filter	This value refers to filter processing in the reads. Filter process is usually recommended to remove the noise and less informative reads, such as low expressed elements with very low read counts. This low read counts might not reveal a real biological information, being due to sequencing errors or inaccuracy during the procedure of read alignment to the reference genome, such as cross mapping artefacts. For this reason, a minimum read count value could be used

	to filter out reads detected below the cutoff. EdgeR and Noiseq offer different options to filter the reads. While EdgeR is implemented with a filter processing using a value of counts per million as a cutoff, Noiseq offers 3 different methods of filtering. See in the specific parameters below for more information. Thus, allowed values for this parameter are: yes or no. Example: filter=yes

Optional parameters

cpmvalue	Cutoff for the counts per million value to be used in filter processing with methods 1 and 3 with Noiseq software (see below for more information about these methods) and in filter processing with EdgeR (1 cpm by default). Example: cpmvalue=2

Specific parameters for EdgeR:

edger_normethod	Normalization method to perform the DE analysis with EdgeR. Normalization allows the comparison between the samples by means of the elimination of the effects of artefacts, noise or outlier values. If data comes from different experiments or datasets normalization process is highly recommended to minimize the effect of the different experimental conditions. EdgeR allows the normalization with 3 methods: "TMM" (default), "RLE", "upperquartile" or "none" (no normalization). Example: edger_normethod=TMM
repthreshold	Number of replicates that have to contains at least a defined number of reads per million to perform the filtering process with EdgeR software (2 replicates by default) Example: repthreshold=3
replicates	Value to indicate if replicates samples are present in the analysis to perform the DE analysis with EdgeR. It is highly recommended to perform the analysis with replicates, but if there are not available a biological coefficient variation value (see below for more information about this parameter) can be used to perform the differential expression analysis. The allowed values for this parameter are: "yes" (by default) or "no". Example: replicates=no
bcvvalue	Value for the common biological coefficient variation (bcv) (square- root-dispersion) in experiments without replicates to perform the DE analysis with EdgeR. Standard values from well-controlled experiments are 0.4 for human data (by default), 0.1 for data on genetically identical model organisms or 0.01 for technical replicates. Example: bcvvalue=0.3

Specific parameters for Noiseq

qvalue	Probability of differential expression to perform the DE analysis with Noiseq. The elements with a probability greater than the defined q-value will be highlighted in the results plots. Please remember that, when using NOISeq, the probability of differential expression is not equivalent to 1 − pvalue. Noiseq team recommends for q to use values around 0.8. If no replicates are available, then it is preferable to use a higher threshold such as q = 0.9. See Noiseq user's manual

	for more information. By default qvalue is 0.8 Example: qvalue=0.9
filtermethod	Method that will be used to filtering process with Noiseq software. See filter parameter above for general recommendations. Noiseq allows filtering with 3 methods: CPM method (1) (by default), Wilcoxon method (2) and Proportion test (3). See Noiseq user's manual for more information. Thus allowed values are: 1, 2 or 3, to refer the previously stated filtering methods. Example: filtermethod=2
cutoffvalue	Cutoff for the coefficient of variation per condition to be used in filter processing with CPM method (1) in Noiseq analysis. This cutoff is expressed in percentage (100 by default). See Noiseq user's manual for more information. Example: cutoffvalue=80
noiseq_normethod	Normalization method to perform the DE analysis with Noiseq. Normalization allows the comparison between the samples by means of the elimination of the effects of artefacts, noise or outlier values. If data comes from different experiments or datasets normalization process is highly recommended to minimize the effect of the different experimental conditions Noiseq allows the following normalization methods: "rpkm" (default), "uqua" (upper quartile), "tmm" (trimmed mean of M) or "n" (no normalization). See Noiseq user's manual for more information. Example: noiseq_normethod=tmm
replicatevalue	Type of replicates to be used to perform the DE analysis with Noiseq. Allowed values are: Technical, biological or no. Inclusion of technical or biological replicates is strongly recommended. Technical replicates involve taking one sample from the same source tube, and analysing it across multiple conditions, while biological replicates are different samples measured across multiple conditions. See Noiseq user's manual for more information. By default, technical replicates option is chosen. Example: replicatevalue=biological
kvalue	Counts equal to 0 are replaced by k value to perform the DE analysis with Noiseq. See Noiseq user's manual for more information. By default, kvalue = 0.5. Example: kvalue = 1
lcvalue	Additional length correction in the normalization process. This correction is done by dividing expression by length^lc to perform the DE analysis with Noiseq. See Noiseq user's manual for more information. By default, lcvalue = 0 for no length correction is applied. Example: lcvalue = 0.5.
pnrvalue	Percentage of the total reads used to simulate each sample when no replicates are available to perform the DE analysis with Noiseq. See Noiseq user's manual for more information. By default, pnrvalue = 0.2. Example: pnrvalue = 0.5.
nssvalue	Number of samples to simulate for each condition (nss>= 2) to perform the DE analysis with Noiseq. See Noiseq user's manual for more information. By default, nssvalue = 5. Example: nssvalue = 3.
vvalue	Variability in the simulated sample total reads to perform the DE analysis with

	Noiseq. Sample total reads is computed as a random value from a uniform distribution in the interval [(pnr-v)*sum(counts), (pnr+v)*sum(counts)]. See Noiseq user's manual for more information. By default, vvalue = 0.02. Example: vvalue = 0.05.

4.1.6.3. Examples of configuration file to run DEAnalysis module

1) Differential expression analysis of miRNAs by EdgeR: In this example, user will perform the differential expression analysis of the counts corresponding to miRNAs. User will execute miARma from its own directory, the input files are tabulated files with the counts from example 4.1. located in the input directory (Examples/basic_examples/miRNAs/results/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/results/ in this case). The differential expression analysis will be performed by EdgeR, using the experimental conditions defined in the target file and the comparisons defined in the contrast file. Filtering process will be carry out discarding those counts less than 2 counts per million in at least 1 replicate. Normalization process will be performed using TMM method.

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/Known_miRNAs/results/
organism=human
strand=yes


[DEAnalysis]

desoft=EdgeR
targetfile=Examples/basic_examples/miRNAs/data/targets.txt
contrastfile=Examples/basic_examples/miRNAs/data/contrast.txt
filter=yes
cpmvalue=2
repthreshold=1
edger_normethod=TMM
replicates=yes
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/Known_miRNAs/5.DEAnalysis/5.1.DEAnalysis_EdgeR_miRNAs.ini
```

2) Differential expression analysis of miRNAs with Noiseq: In this example, user will perform the differential expression analysis of the counts corresponding to miRNAs. User will execute miARma from its own directory, the input files are tabulated files with the counts from example 4.1. located in the input directory (Examples/basic_examples/miRNAs/results/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/results/ in this case). The differential

expression analysis will be performed with Noiseq tool, using the experimental conditions defined in the target file and the comparisons defined in the contrast file. This analysis will be performed using biological replicates. Filtering process will be carried out by CPM method using a cutoff or 2 counts per million and a coefficient of variation per condition of 80%. Normalization process will be performed using tmm method. In addition, counts equal to 0 will be replaced by 1 in the normalization process and a length correction will also performed be performed using 0.5 value. In this analysis the q value cutoff has been established in 0.9 to select the differentially expressed elements with Noiseq.

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/Known_miRNAs/results/
organism=human
strand=yes

[DEAnalysis]
desoft=Noiseq
targetfile=Examples/basic_examples/miRNAs/data/targets.txt
contrastfile=Examples/basic_examples/miRNAs/data/contrast.txt
qvalue=0.9
filter=yes
filtermethod=1
cpmvalue=2
cutoffvalue=80
noiseq_normethod=tmm
replicates=yes
replicatevalue=biological
kvalue = 1
lcvalue = 0.5
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/Known_miRNAs/5.DEAnalysis/5.2.DEAnalysis_Noiseq_miRNAs.ini
```

3) Differential expression analysis of miRNAs by EdgeR and Noiseq: In this example, user will perform the differential expression analysis of the counts corresponding to miRNAs. User will execute miARma from its own directory, the input files are tabulated files with the counts from example 4.1. located in the input directory (Examples/basic_examples/miRNAs/results/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/results/ in this case). The differential expression analysis will be performed with both, EdgeR and Noiseq tools, using the experimental conditions defined in the target file and the comparisons defined in the contrast file. Filtering process will be carry out with default filter methods (those counts less than 1 counts per million will be discarded), and normalization process will be performed as default option using tmm

method for analysis with EdgeR and rpkm for Noiseq. In this analysis the default cutoff of q value (0.8) will be used to select the differentially expressed elements with Noiseq.

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/Known_miRNAs/results/
organism=human
strand=yes

[DEAnalysis]
desoft=EdgeR-Noiseq
targetfile=Examples/basic_examples/miRNAs/data/targets.txt
contrastfile=Examples/basic_examples/miRNAs/data/contrast.txt
filter=yes
replicates=yes
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/Known_miRNAs/5.DEAnalysis/5.3.DEAnalysis_EdgeR_Noiseq_miRNAs.ini
```

4) Differential expression analysis of miRNAs by EdgeR and Noiseq without replicates: In this example, user will perform the differential expression analysis of the counts corresponding to miRNAs. User will execute miARma from its own directory, the input files are tabulated files with the counts from example 4.1. located in the input directory (Examples/basic_examples/miRNAs/results/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/results/ in this case). The differential expression analysis will be performed by both, EdgeR and Noiseq tools, using the experimental conditions defined in the target file and the comparisons defined in the contrast file. Filtering process will be carried out with default filter parameters (those counts less than 1 counts per million will be discarded), and normalization process will be performed as default option using tmm method for analysis with EdgeR and rpkm for Noiseq. In this case, the analysis contains no replicates, so as recommended by edgeR vignette, a manually selected valued should be introduced by the user. For analysis using EdgeR, the common biological coefficient variation (bcv) has been established in 0.3. For Noiseq analysis the simulation of the replicates is more complex being characterized for the percentage of the total reads used to simulated each sample (pnrvalue), the number of samples to simulate for each condition (nssvalue) and the variability in the simulated sample total reads (vvalue). In this analysis, a pnrvalue of 0.5, a nssvalue of 3 and a vvalue of 0.05 has been established. This analysis the default cutoff of q value (0.8) will be used to select the differentially expressed elements with Noiseq.

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/Known_miRNAs/results/
organism=human
strand=yes

[DEAnalysis]
desoft=EdgeR-Noiseq
targetfile=Examples/basic_examples/miRNAs/data/targets_no_replicates.txt
contrastfile=Examples/basic_examples/miRNAs/data/contrast.txt
filter=yes
replicates=no
bcvvalue=0.3
replicatevalue=no
pnrvalue = 0.5
nssvalue = 3
vvalue = 0.05
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/Known_miRNAs/5.DEAnalysis/5.4.DEAnalysis_EdgeR_Noiseq_miRNAs_no_replicates.ini
```

4.1.7. Target prediction module

The aim of this module is to perform the target prediction analysis. To achieve this goal, miARma-Seq has implemented miRGate tool. miRGate is a new database containing novel computational predicted miRNA-mRNA pairs that are calculated using well-established algorithms such as miRanda, Pita, TargetScan, RNAhybrid or MicroTar. In addition, miRGate includes experimental validated miRNA-mRNAs pairs providing to miARma-Seq a high reliability tool to miRNA-mRNA target prediction. Thus, miRGate provides the target genes of the input DE miRNA data, or the DE mRNAs targeted by DE miRNAs from the negative correlations between DE miRNAs and DE mRNAs.

4.1.7.1. Input/Output files

Input: Tabulated file (xls format) with the DE expressed miRNAs from the Noiseq or EdgeR analysis with the DEAnalysis module. To obtain more information about the format of these files please consult the output format of the Section 4.1.6. Differential Expression module.

Output:

1. Tabulated file (excel compatible) with the predicted targets and the statistical values of the prediction. The standard output file will contain 12 columns:

- [miRNA]- Name of the miRNA.
- [miRNA FC]- Fold Change of the miRNA obtained in the DE analysis.

- [miRNA FDR]- FDR value of the miRNA obtained in the DE analysis.
- [Ensembl Gene]- Ensembl code of the targeted gene.
- [Gene Symbol]- Symbol of the targeted gene.
- [Ensembl Transcript]- Ensembl code of the targeted transcript.
- [Gene FC]- Fold Change of the mRNA obtained in the DE analysis.
- [Gene FDR]- FDR value of the miRNA obtained in the DE analysis.
- [Method]- Method of prediction.
- [Target Site]- Type of target site.
- [Score]- Z-score of the prediction.
- [Energy]- Energy of the prediction.

2. Summary results report (xls format) with the main statistics of the analysis. This report will be generated in the output directory provided by the user. In this report, "miRNA-mRNA Target Predictions by miRGate" section with the path of the target prediction results can be founded, together with 2 summary tables:

miRNAs with more associations:

-[File]- Name of the DE Analysis results file.
-[miRNA]- Name of the miRNA with more associations (5 for each file).
-[Number of associations]- Number of associations.

Genes more regulated:
-[File]- Name of the DE Analysis results file.
-[GeneName]- Name of the gene regulated for more miRNAs (5 for each file).
-[Number of associations]- Number of associations.

An example of the summary report can be consulted in the following link.

4.1.7.2. Configuration file:

To execute this analysis the heading **[TargetPrediction]** must be included in the configuration file. The parameters included in this analysis are:

Optional parameters	
noiseq_cutoff	Cutoff value to select statistically significant results performed with Noiseq tool. The selected entities will be included in the target prediction analysis. By default, this value has been established in 0.8. Example: noiseq_cutoff=0.9.
edger_cutoff	Cutoff value to select statistically significant results performed with EdgeR tool. The selected entities will be included in the target prediction analysis. By default, this value has been established in 0.05. Example: edger_cutoff=0.01

fc_threshold	Value to filter low DE expressed miRNAs. As logFC is expressed in log2 a fc_threshold=1 means a real change in expression of 2 folds. Example: fc_threshold=3
genes_folder	Path of the folder with DE mRNAs to obtain DE mRNAs-miRNAs pairs with miRGate. Example: Examples/basic_examples/mRNAs/results/

4.1.7.3. Examples of configuration file to run Target Prediction module

1) Target prediction analysis of DE miRNAs: In this example, user will perform the target prediction analysis of the DE miRNAs. User will execute miARma from its own directory, the input files are tabulated files with the results of the DE analysis from example 5.3. located in the input directory (Examples/basic_examples/miRNAs/results/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/results/ in this case). The target prediction will be performed only in those miRNAs with a probability greater than 0.8 in Noiseq analysis and a p-value less than 0.05 in EdgeR analysis.

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/results/
organism=human
strand=yes

[TargetPrediction]
noiseq_cutoff=0.8
edger_cutoff=0.05
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/6.TargetPrediction/6.1.miRGate.ini
```

2) Target prediction analysis of DE pairs miRNAs-mRNAs: In this example, user will perform the target prediction analysis of the DE miRNAs using as targets the DE genes contained in the genes_folder. User will execute miARma from its own directory, the input files are tabulated files with the results of the DE analysis from example 5.3. located in the input directory (Examples/basic_examples/miRNAs/results/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/results/ in this case). The target prediction will be performed only in those miRNAs with a probability greater than 0.8 in Noiseq analysis and a p-value less than 0.05 in EdgeR analysis.

```
[General]
type=miRNA
verbose=0
```

```
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/results/
organism=human
strand=yes

[TargetPrediction]
noiseq_cutoff=0.8
edger_cutoff=0.05
genes_folder=Examples/basic_examples/mRNAs/results/
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/6.TargetPrediction/6.2.miRGate_miRNAs_and_genes.ini
```

4.2. De novo prediction and known miRNAs analysis

This analysis allows the identification of differentially expressed known miRNAs from high throughput sequencing data and predicts novel miRNAs. A complete example of the pipeline can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/DeNovo_miRNAs/miARma_miRNAs_DeNovo.ini
```

4.2.1 General features

4.2.1.1. Configuration file

In order to execute miARma-Seq, a configuration file in INI format is mandatory with information about your experiment setup. General information must be provided by the heading **[General]** at the beginning of the configuration file. Although general section is required for any analysis with miARma-Seq a configuration file only with this section will not perform any analysis. See below a detailed explanation in order to configure the different steps of the analysis. This information is mainly oriented to the path of input files and output directories.

The parameters included in this section are:

Mandatory parameters:	
type	Type of analysis to perform with miARma-Seq. Allowed values for this parameter are: miRNA, mRNA or circRNAs. Example: type=miRNA
read_dir	Folder for input files where raw data from high thorughput sequencing in fastq format are located. Example: read_dir=Examples/basic_examples/miRNAs/reads/

label	Name to identify the analysis. This name will appear in the output files and plots. Example: label=Hypoxia
miARmaPath	Folder where miARma-Seq has been installed. Example: miARmaPath=/opt/miARma/
output_dir	Folder to store the results. Example:output_dir=Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/
organism	Organism analysed in the experiment. Example: organism=human

Optional parameters:

verbose	Parameter to show the execution data on the screen. Value of 0 for no verbose, otherwise to print "almost" everything. Example: verbose=0
threads	Number of process to run at the same time. The maximum value of this parameter is defined for user's computer. Example: threads=4
stats_file	File where stats data will be saved. Example: stats_file=stats.log
logfile	File to print the information about the execution process. Example: logfile=run_log.log
seqtype	Sequencing procedure of RNA-Seq experiment. Allowed values: Paired/Paired-End or Single/Single-End (by default). Please note that paired-end analysis samples must be named with the final end of "_1" and "_2" before file extension to correctly identify paired samples. Example: SRR873382_1.fastq and SRR873382_2.fastq. Example: seqtype=Paired
strand	Parameter to specify whether the data is from a strand-specific assay. The allowed values are: yes (by default), no or reverse. Example: strand=no

4.2.1.2. Examples of the general information in the configuration file

1) General information of miRNA analysis.- In this example, user is defining the general parameters of the analysis executing miARma from its own directory, the pipeline input files are fastq files from human located in the input directory (Examples/basic_examples/miRNAs/reads/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/ DeNovo_miRNAs /results/ in this case) including the name of the experiment (Hypoxia in this example). The analysis will perform with 4 threads and the execution data will not be showed in the screen.

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
```

```
output_dir=Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/
organism=human
strand=yes
```

4.2.2. Quality module

The aim of the Quality module is to provide a simple way to check the quality of our sequenced samples and avoid the inclusion of outliers. This analysis will be performed with FASTQC software and it can be performed before the data processing and after read trimming, in the case of miRNA analysis.

4.2.2.1. Input/Output files

Input: Raw data from high throughput sequencing in fastq format (compressed files are allowed).

Output:

1. HTML report with different plots and statistics of the quality of the data. These files will be saved inside a folder called Pre_fastqc_results under the path specified in output_dir. For each fastq file, an independent quality analysis process will be performed and stored in a folder with the same name of the fastq file. In order to examine the results, a html file called fastqc_report.html is included. Please visit FastQC help page to better understand the FastQC report.

2. Summary results report (xls format) with the main statistics of the analysis. This report will be generated in the output directory provided by the user. In this report, "Quality" section with the path of the quality results can be founded, together with a summary table below with the columns:

-[Filename]- Name of the sample.
-[Number of reads] –Number of reads contained in the fastq files.
-[%GC Content]- Proportion of GC content of the reads.
-[Read Length]- Length of the reads.
-[Encoding]- Type of encoding of the fastq files.

An example of the summary report can be consulted in the following link.

4.2.2.2. Configuration file:

To execute this analysis the heading **[Quality]** must be included in the configuration file. The parameters included in this analysis are:

Mandatory parameters	
prefix	Parameter to define when miARma will perform the quality analysis. Use "pre" to perform a quality analysis for unprocessed reads and "post" for processed reads (after adapter section). miARma also accept the keyword "both" in case you want the analysis twice: before and after the pre-processing of the reads. Example: prefix=both

4.2.2.3. Examples of configuration file to run Quality analysis

1) Quality analysis of miRNA analysis.- In this example, user will perform the quality analysis executing miARma from its own directory, the pipeline input files are fastq files located in the input directory (Examples/basic_examples/miRNAs/reads/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/ in this case).

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/
organism=human
strand=yes

[Quality]
prefix=Pre
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/DeNovo_miRNAs/1.Quality/1.Quality.ini
```

To inspect results for a Fastq file named SRR873382, please check Examples/basic_examples/ miRNAs/Known_miRNAs/results/Pre_fastqc_results/SRR873382_fastqc/fastqc_report.html

4.2.3. DeNovo module

In order to perform de novo prediction of miRNAs, miARma-Seq has implemented miRDeep2, which allow novel miRNAs prediction based on different aspects such as their ability to form a hairpin structure of a pre-miRNA or to follow the pattern of Dicer processing. This tool performs the pre-processing of the reads (adapter removal), the alignment against a genome reference using Bowtie1, the prediction of new miRNAs and the quantification of novel and known miRNAs according to aligned reads.

4.2.3.1. Input/Output files

Input: Raw data from high throughput sequencing in fastq format (compressed files are allowed).

Output:

1. Different results files (.arf, .fa, .tab and .xls) in the provided output directory within the "mirdeep_results" folder. To obtain more information about these files please visit miRDeep webpage. To simplify the results obtained, a tabulated results files (excel compatible) including the information of novel miRNAs as well as detected mature miRBase miRNAs in each sample will be generated. The output directory will be named as "DeNovo_ReadCount". Of note, known miRNAs will be identified

with the name of the mature miRNA while new miRNAs detected will be identified with the genomic position. In this file, each row corresponds to a miRNA and each column to the number of reads of that feature in each sample. The names of the columns are the name of each sample. Example:

	SRR873382	SRR873383	SRR873384	SRR873385	SRR873386	SRR873387	SRR873388	SRR873389
chr1:175937535 ..175937596:-	130	115	86	91	0	188	0	133
chr5:180633867 ..180633931:+	1212	0	0	0	0	0	0	0
hsa-let-7a-5p	273585	300555	382141	205273	219394	159367	211469	227299
hsa-let-7b-5p	93847	120503	47565	78418	80563	90990	78404	90161
hsa-let-7c-5p	14676	18346	11641	12028	13683	11622	11659	12721

2. Summary results report (xls format) with the main statistics of the analysis. This report will be generated in the output directory provided by the user. In this report, "miRDeep" section with the path of the quality results can be founded, together with a summary table below with the columns:

-[Filename]- Name of the sample.
-[Number of novel miRNAs] –Number of novel miRNAs identified.
-[Number of known miRNAs]- Number of known miRNAs identified.

An example of the summary report can be consulted in the following link.

4.2.3.2. Configuration file

To execute this analysis the heading **[DeNovo]** must be included in the configuration file. The parameters included in this analysis are:

Mandatory parameters	
bowtie1index	Indexed genome to align your reads in format .ebwt Example: bowtie1index=Genomes/Indexes/bowtie1/human/bw1_homo_sapiens19
adapter	Adapter sequence to trimm at read 3' Example: adapter=ATCTCGTATGCCGTCTTCTGCTTGAA
mature_miRNA_file	Fasta file with all mature sequence from your organism Example: mature_miRNA_file=Examples/basic_examples/miRNAs/data/hsa_mature_miRBase20.fasta
precursor_miRNA_file	Fasta file with all known pre-miRNA sequence Example: precursor_miRNA_file=Examples/miRNAs/data/precursors_miRBase20.fasta
genome	Fasta file for the complete genome of our organism Example: genome= Genomes/Indexes/bowtie1/human/homo_sapiens19.fa
Optional parameters:	
adapter_file	Complete path of the file to specify a different adapter for each fastq file

(recommended por multiplexed files). This is a tabulated file that contains two columns: the name of the fastq file and the correspondent adapter sequence to remove it. The name of the fastq file in this file must be exactly identical to fastq file of the input directory; otherwise adapter will not be removed from this sample. For example, for the hypoxia example input the correspondent adapter_file will contain the next information:

Filename	Adapter
SRR873382.fastq.bz2	ATCTCGTATGCCGTCTTCTGCTTGA
SRR873383.fq.bz2	ATCTCGTATGCCGTCTTCTGCTTG
SRR873384.fastq.bz2	ATCTCGTATGCCGTCTTCTGCTT
SRR873385.fq.bz2	ATCTCGTATGCCGTCTTCTGCT
SRR873386.fastq.bz2	TCTCGTATGCCGTCTTCTGCTTGAA
SRR873387.fastq.bz2	CTCGTATGCCGTCTTCTGCTTGAA
SRR873388.fastq.bz2	TCGTATGCCGTCTTCTGCTTGAA
SRR873389.fastq.bz2	CGTATGCCGTCTTCTGCTTGAA

Example: adapter_file=Examples/basic_examples/miRNAs/data/Adapter_file.txt

4.2.3.3. Examples of configuration file to run DeNovo analysis

1) DeNovo miRNA prediction.- In this example, user will perform the DeNovo miRNA prediction executing miARma from its own directory. The input files are fastq files located in the input directory (Examples/basic_examples/miRNAs/reads/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/ in this case). First, a specific adapter sequence will be removed from the reads and then, the reads will be aligned against the bowtie1 index and the non-aligned reads will be used to predict new miRNAs using the human genome in fasta format. The files including mature and precursor miRNAs are necessary for the counting of the miRNAs.

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/
organism=human
strand=yes

[DeNovo]
bowtie1index= Genomes/Indexes/bowtie1/human/bw1_homo_sapiens19
adapter=ATCTCGTATGCCGTCTTCTGCTTGAA
mature_miRNA_file=Examples/basic_examples/miRNAs/data/hsa_mature_miRBase20.fasta
precursor_miRNA_file=Examples/miRNAs/data/precursors_miRBase20.fasta
genome=Genomes/Indexes/bowtie1/human/homo_sapiens19.fa
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/DeNovo_miRNAs/2.DeNovo/2.1.DeNovo.ini
```

2) DeNovo miRNA prediction removing different adapter sequences for each file.- In this example, user will perform the DeNovo miRNA prediction executing miARma from its own directory. The input files are fastq files located in the input directory (Examples/basic_examples/miRNAs/reads/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/ in this case). First, a specific adapter sequence for each sample specified in the adapter_file will be removed from the reads and then, the reads will be aligned against the bowtie1 index and the non-aligned reads will be used to predict new miRNAs using the human genome in fasta format. The files including mature and precursor miRNAs are necessary for the counting of the miRNAs.

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/
organism=human
strand=yes

[DeNovo]
bowtie1index= Genomes/Indexes/bowtie1/human/bw1_homo_sapiens19
adapter_file=Examples/basic_examples/miRNAs/data/Adapter_file.txt
mature_miRNA_file=Examples/basic_examples/miRNAs/data/hsa_mature_miRBase20.fasta
precursor_miRNA_file=Examples/miRNAs/data/precursors_miRBase20.fasta
genome=Genomes/Indexes/bowtie1/human/homo_sapiens19.fa
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/DeNovo_miRNAs/2.DeNovo/2.2. DeNovo_different_barcodes.ini
```

4.2.4. Differential Expression module

The aim of this module is to perform the differential expression analysis between different experimental conditions. For this purpose, miARma-Seq implements NOISeq and EdgeR software. Both are valuable tools to identify differentially expressed (DE) elements, which covers different requirements. edgeR is a widely employed tool for differential expression analysis that allows not only the identification of DE elements between two experimental conditions but more complicated comparison in the same analysis process. On other hand, Noiseq allows the simulation of technical replicates to increase the reliability of the results, when no replicates are available for the analysis.

4.2.4.1. Input/Output files

Input: Tabulated file with the counts of the reads. In this file, each row corresponds to a feature and each column to the number of reads of that feature. The names of the columns are the name of each sample. Example:

	SRR873382	SRR873383	SRR873384	SRR873385	SRR873386	SRR873387	SRR873388	SRR873389
hsa-let-7a-2-3p	0	0	0	0	0	0	0	0
hsa-let-7a-3p	180	163	101	92	121	82	124	86
hsa-let-7a-5p	270720	306188	376418	204502	218299	158334	209985	224984
hsa-let-7b-3p	20	40	8	8	7	10	13	8
hsa-let-7b-5p	90881	121568	46482	76478	78113	88608	76335	87498
hsa-let-7c-3p	4	12	1	3	5	2	3	2
hsa-let-7c-5p	13669	17626	10830	11561	12842	11041	10913	11904

Output:

1. Tabulated results files (excel compatible) with the entities differentially expressed (DE) and the statistical values of the analysis for any of the comparison between the different experimental conditions. According to the tool selected for the analysis, the format of the results differs. Specific format will be detailed explained below.

- **EdgeR results**- EdgeR results will be located in the "EdgeR_results" directory in the output_dir directory defined by the user. The results with the differentially expressed entities of each condition will be saved in different files. The name of the results files will be constructed as follows:

(Label_defined_by_user)_(Adapter_tool)_(Aligner_tool)_EdgeR_results_(label_of_the_comparison).xls

Example: For the comparison at 16 hours of hypoxia experiment performed with Cutadapt and Bowtie1 tools, the resultant file will be: Hypoxia_cut_bw1_EdgeR_results_Comp_16.xls

EdgeR result file contains 5 columns:

- [Entity]- Name of the DE entity, which according to the experiment could be the name of miRNAs, mRNAs or circRNAs.
- [logFC]- Log2-fold- change value.
- [logCPM]- Log2 counts-per-million.
- [Pvalue]- Probability value.
- [FDR]- False discovery rate obtained by Benjamini and Hochberg's algorithm.

Example:

	logFC	logCPM	PValue	FDR
hsa-miR-503-3p	-1.890652364	4.098298363	1.60E-07	9.39E-05

hsa-miR-210-3p	1.702083488	8.770057331	1.37E-06	0.000400593
hsa-miR-4521	-3.308571177	1.407874884	3.48E-06	0.000678891
hsa-miR-210-5p	2.666024708	2.435923273	1.24E-05	0.001809943
hsa-miR-222-5p	-1.835901985	2.791957916	3.81E-05	0.004455154

- **Noiseq results**- Noiseq results will be located in the "Noiseq_results" directory in the output_dir directory defined by the user. Noiseq generates a results file with the statistical values of every expressed entity for each condition. The name of this file will be constructed as follows:

(Label_defined_by_user)_(Adapter_tool)_(Aligner_tool)_Noiseq_results_(label_of_the_comparison).xls

Example: For the comparison at 16 hours of hypoxia experiment performed with Cutadapt and Bowtie1 tools, the resultant file will be: Hypoxia_cut_bw1_Noiseq_results_Comp_16.xls

Both files contain 7 columns:

- [Entity]- Name of the DE entity, which according to the experiment could be the name of miRNAs, mRNAs or circRNAs.
- [Condition1_mean]- Expression values for condition 1.
- [Condition2_mean]- Expression values for condition 2.
- [M] - log2-ratio of the two conditions.
- [D] - value of the difference between conditions.
- [prob] - probability of differential expression.
- [ranking] – summary statistic of "M" and "D" values.

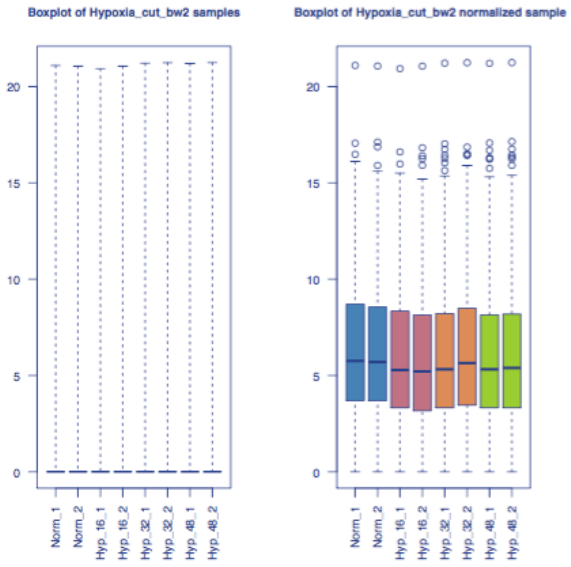
Example:

	T48h_mean	**T0h_mean**	**M**	**D**	**prob**	**ranking**
hsa-miR-19b-3p	96.66	392.96	-2.02	296.30	0.9358	-296.31
hsa-miR-296-3p	27.06	159.27	-2.55	132.20	0.9280	-132.22
hsa-miR-7-5p	321.28	970.53	-1.59	649.24	0.9168	-649.24
hsa-miR-22-3p	90.79	319.31	-1.81	228.51	0.9090	-228.51
hsa-miR-30d-5p	20290.89	8331.66	1.28	11959.22	0.9046	11959.2

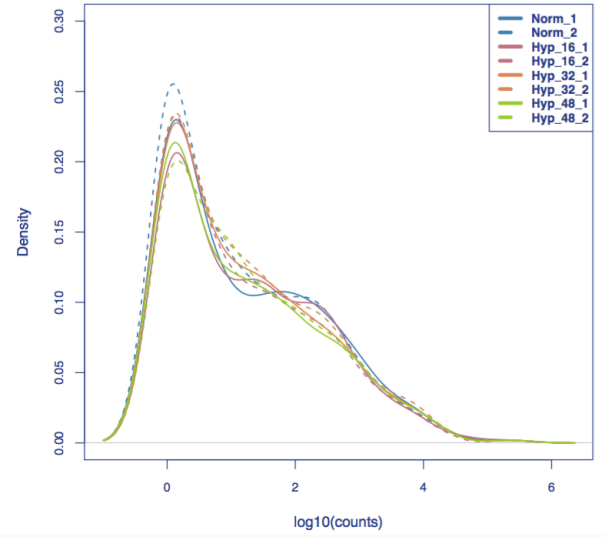
2. Exploratory plots of the analysis. miARma-Seq provides a exhaustive PDF report with different plots in order to make easier to the user the interpretation of the data. This report contains:

2.1. Analysis of the distribution of the reads in the samples. The detailed inspection of the distribution of the reads in the different samples allows to the user identify samples with abnormal distribution of the reads. These samples are recommended to be removed from the analysis since may introduce noise or affect to the final results. In order to inspect the distribution of the reads in the different samples, miARma-Seq generates two kinds of plots:

- Boxplot of the distribution of the counts. The first page of the report contains 2 boxplots with the distribution of the counts, before (left) and after (right) the normalization process. The log2(number of counts) is represented for each sample. Boxplot of non-normalized data usually will have a lower limit near to – infinite due to the miRNAs with no counts. The different replicates will be represented with the same colour. The expected boxplot will look like this:



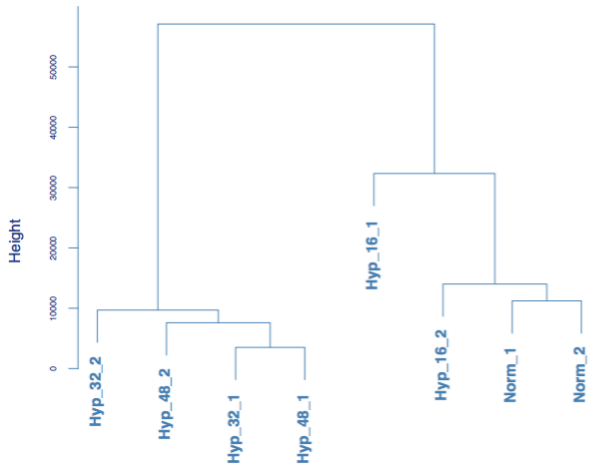
- Density plot of the distribution of the counts. The second and third page of the report contains 2 density plots with the distribution of the counts, before (second page) and after (third) the normalization process. The plot represents the density of the log10 of the counts for each sample. The different replicates will be represented with the same colour. These plots will look like this:



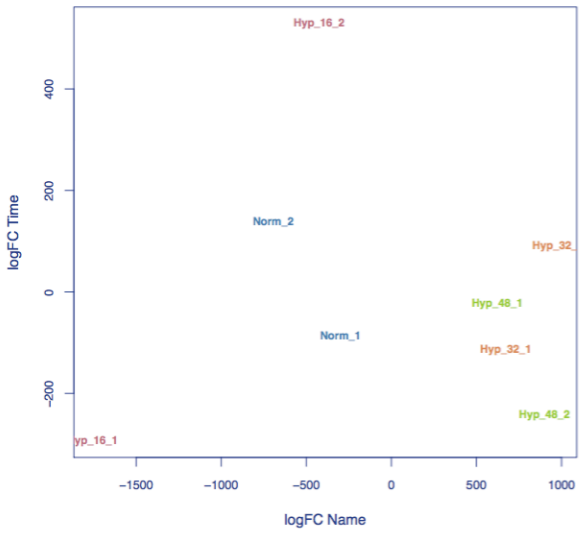
2.2. Analysis of the samples similarity- In order to examine the quality of the data obtained in the experiment, miARma-Seq has implemented different plots, which allows the inspection of the

diversity between the samples. For a good quality experiment, the samples belonging to the same experimental conditions should present more similarity between them than with the samples of others experimental conditions. Thus, with these analyses user can identify samples with low quality to remove from the analysis.

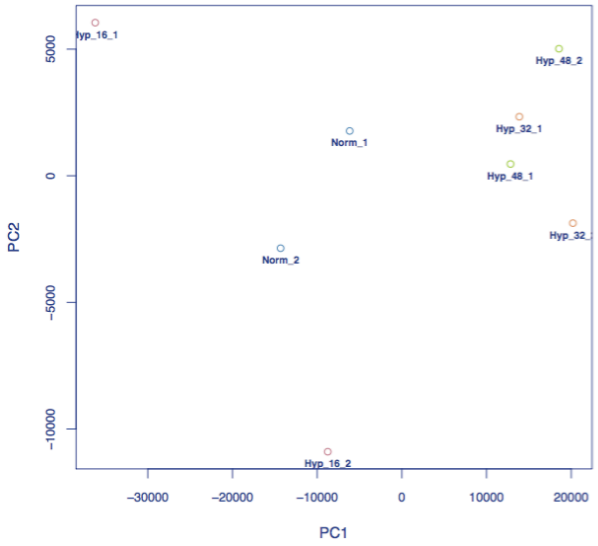
-Hierarchical clustering of the samples: The hierarchical clustering plots, before and after normalization process, classify the samples according to their similarity. The distance of the branch is proportional to the sample distance.



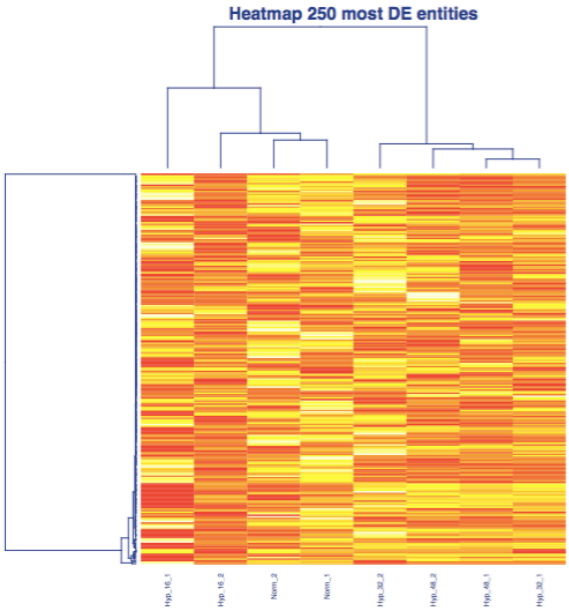
-Multidimensional plot (MDS): The MDS plot divides the samples in a two-dimensional plot according to their similarity. Each sample is represented with its name and coloured according to the experimental condition that belongs to.



-Principal Component Analysis (PCA) plot: The PCA plot divides the samples in a two-dimensional plot according to their similarity. Each sample is represented with its name and coloured according to the experimental condition that belongs to.



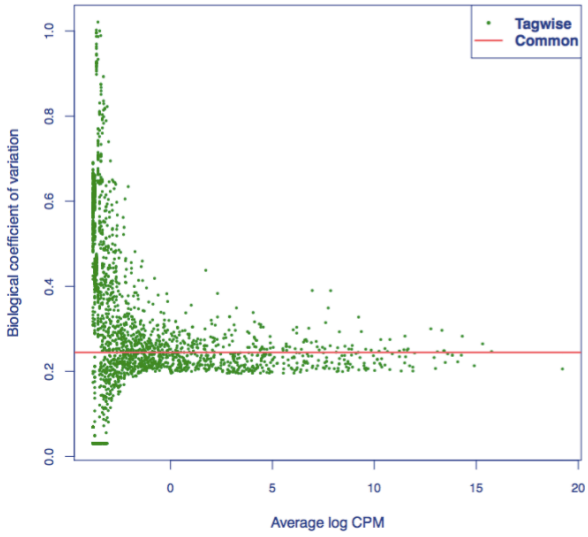
-Heatmap: The heatmap allows to the user evaluate the similarity between the samples according to the 250 most expressed miRNAs expression.



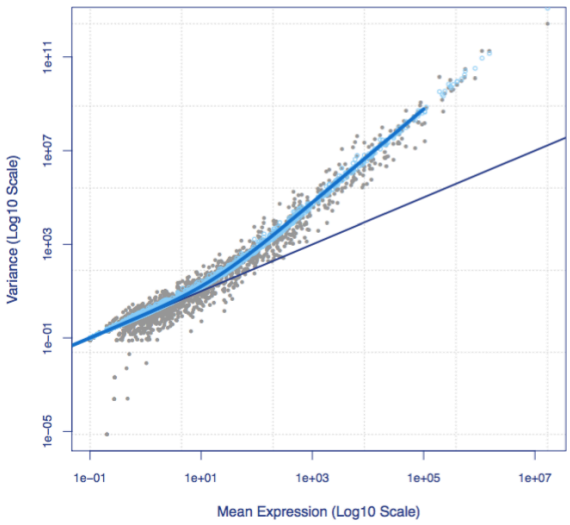
3. Results plots of the analysis.- miARma-Seq generates a PDF report with plots to explore the results with both tools, EdgeR and Noiseq.

3.1. Results plots with EdgeR:

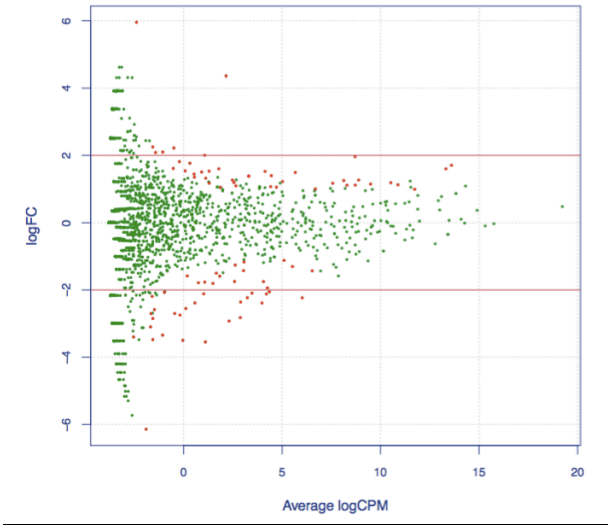
-Biological Variation Plot: The square root of dispersion is the coefficient of biological variation (BCV). This plot illustrates the relationship of biological coefficient of variation versus mean log CPM.



-Mean Variance Plot: This plot can be used to explore the mean-variance relationship; each dot represents the estimated mean and variance for each gene, with binned variances as well as the trended common dispersion overlaid.

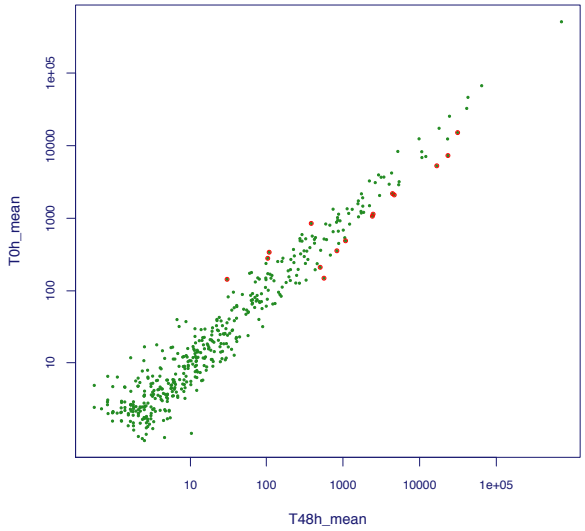


-Expression Plot: miARma-Seq generates one expression plot for each comparison. This plot shows all the logFCs against average count size, highlighting the DE genes.

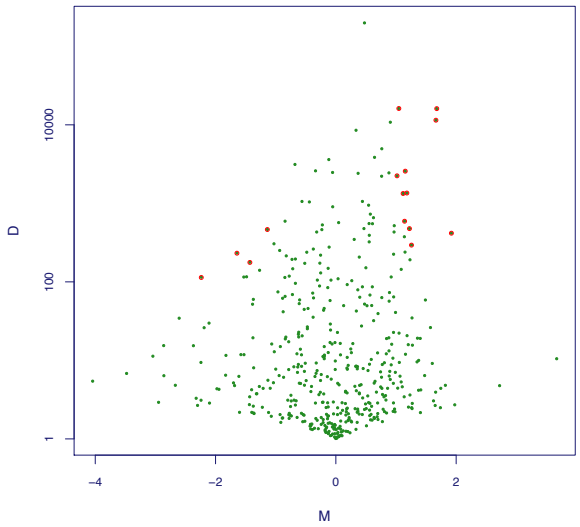


3.2. Results plots with Noiseq: For each comparison a PDF report with the results plot is generated

- Expression Plot: Summary plot of the expression values for both conditions (green), where differentially expressed genes are highlighted (red).



- MD Plot: Summary plot for (M,D) values (green) and the differentially expressed genes (red).



4. Summary results report (xls format) with the main statistics of the analysis. This report will be generated in the output directory provided by the user. In this report, "Differential Expression Analysis" section with the path of the Differential Expression Analysis results can be founded for each tool EdgeR and Noiseq. Each tool shows different information.

For EdgeR analysis the summary table shows the columns:

-[Comparison]- Name of the comparison defined in the contrastfile.
-[File]- Name of the file with the DE elements.
-[Number of DE elements (Pval <=0.05)]- Number of DE elements with a p-value <=0.05.
-[Number of DE elements (FDR <=0.05)]- Number of DE elements with a FDR <=0.05.

For Noiseq analysis the summary table shows the columns:

-[Comparison]- Name of the comparison defined in the contrastfile.
-[File]- Name of the file with the DE elements.
-[Number of DE elements (Prob >=0.8)]-Number of DE elements with a probability >=0.8.

An example of the summary report can be consulted in the following link.

4.2.4.2. Configuration file

To execute this analysis the heading **[DEAnalysis]** must be included in the configuration file. The parameters included in this analysis are:

Mandatory parameters	
desoft	Specific software to perform the differential expression analysis. As state above the tools EdgeR and Noiseq are implemented in miARma-Seq. These tools can be selected alone or in combination. Thus allowed values for this parameter are: edger, noiseq or edger-noiseq. Note that, each specific tool requires specific

	parameters. See examples below to deep in the analysis possibilities. Example: desoft=EdgeR-Noiseq
targetfile	Complete path of the target file. This is a tabulated file that contains the experimental condition of each sample. The first column of this file must coincide with the column names of the input files. Note that, only those samples present in the target file will be used for the analysis. The second column must contain the names of the samples to be used to the plots, and the next columns the condition of each factor. For example, for the input previously showed the correspondent target file will contain the next information: Filename Name Time SRR873382 Norm_1 T0h SRR873383 Norm_2 T0h SRR873384 Hyp_16_1 T16h SRR873385 Hyp_16_2 T16h SRR873386 Hyp_32_1 T32h SRR873387 Hyp_32_2 T32h SRR873388 Hyp_48_1 T48h SRR873389 Hyp_48_2 T48h In this example, the first column “Filename” contain the name of the samples obtained from SRA, the second column “Name” contain the names to use in the exploratory plots and the third column “Time” corresponds to the experimental condition, which in this case is the number of hours in hypoxic conditions. This target file can be downloaded as stated in section 3.2. Example: targetfile=Examples/basic_examples/miRNAs/data/targets.txt
contrastfile	Path of the contrast file o perform the DE analysis with EdgeR. This file has one column with the contrasts user want to evaluate. The syntax of the contrast should be: name_of_contrast=contrast to evaluate. Any type of contrast can be done but condition name must be one of the conditions present in targets file. In addition, contrast must differ of 0 (ie: cond=WT-WT is not allowed). There is no limit in the number of contrasts. For example, for the input previously showed the correspondent contrast file will contain the next information: "Name" "Comp_16=T16h-T0h" "Comp_32=T32h-T0h" "Comp_48=T48h-T0h" In this example, there are 3 different contrasts conditions: T16h-T0h, T32h-T0h and T48h-T0h, which will be evaluated as separated analysis, generating a xls file with the correspondent results for each condition. This contrast file can be downloaded as stated in section 3.2. Example: contrastfile=Examples/basic_examples/miRNAs/data/contrast.txt
filter	This value refers to filter processing in the reads. Filter process is usually recommended to remove the noise and less informative reads, such as low expressed elements with very low read counts. This low read counts might not reveal a real biological information, being due to sequencing errors or inaccuracy during the procedure of read alignment to the reference genome, such as cross mapping artefacts. For this reason, a minimum read count value could be used

	to filter out reads detected below the cutoff. EdgeR and Noiseq offer different options to filter the reads. While EdgeR is implemented with a filter processing using a value of counts per million as a cutoff, Noiseq offers 3 different methods of filtering. See in the specific parameters below for more information. Thus, allowed values for this parameter are: yes or no. Example: filter=yes

Optional parameters

cpmvalue	Cutoff for the counts per million value to be used in filter processing with methods 1 and 3 with Noiseq software (see below for more information about these methods) and in filter processing with EdgeR (1 cpm by default). Example: cpmvalue=2

Specific parameters for EdgeR:

edger_normethod	Normalization method to perform the DE analysis with EdgeR. Normalization allows the comparison between the samples by means of the elimination of the effects of artefacts, noise or outlier values. If data comes from different experiments or datasets normalization process is highly recommended to minimize the effect of the different experimental conditions. EdgeR allows the normalization with 3 methods: "TMM" (default), "RLE", "upperquartile" or "none" (no normalization). Example: edger_normethod=TMM
repthreshold	Number of replicates that have to contains at least a defined number of reads per million to perform the filtering process with EdgeR software (2 replicates by default) Example: repthreshold=3
replicates	Value to indicate if replicates samples are present in the analysis to perform the DE analysis with EdgeR. It is highly recommended to perform the analysis with replicates, but if there are not available a biological coefficient variation (bcv) value (see below for more information about this parameter) can be used to perform the differential expression analysis. The allowed values for this parameter are: "yes" (by default) or "no". Example: replicates=no
bcvvalue	Value for the common biological coefficient variation (bcv) (square- root-dispersion) in experiments without replicates to perform the DE analysis with EdgeR. Standard values from well-controlled experiments are 0.4 for human data (by default), 0.1 for data on genetically identical model organisms or 0.01 for technical replicates. Example: bcvvalue=0.3

Specific parameters for Noiseq

qvalue	Probability of differential expression to perform the DE analysis with Noiseq. The elements with a probability greater than the defined q-value will be highlighted in the results plots. Please remember that, when using NOISeq, the probability of differential expression is not equivalent to 1 − pvalue. Noiseq team recommends for q to use values around 0.8. If no replicates are available, then it is preferable to use a higher threshold such as q = 0.9. See Noiseq user's manual for more information. By default qvalue is 0.8

	Example: qvalue=0.9
filtermethod	Method that will be used to filtering process with Noiseq software. See filter parameter above for general recommendations. Noiseq allows filtering with 3 methods: CPM method (1) (by default), Wilcoxon method (2) and Proportion test (3). See Noiseq user's manual for more information. Thus allowed values are: 1, 2 or 3, to refer the previously stated filtering methods. Example: filtermethod=2
cutoffvalue	Cutoff for the coefficient of variation per condition to be used in filter processing with CPM method (1) in Noiseq analysis. This cutoff is expressed in percentage (100 by default). See Noiseq user's manual for more information. Example: cutoffvalue=80
noiseq_normethod	Normalization method to perform the DE analysis with Noiseq. Normalization allows the comparison between the samples by means of the elimination of the effects of artefacts, noise or outlier values. If data comes from different experiments or datasets normalization process is highly recommended to minimize the effect of the different experimental conditions Noiseq allows the following normalization methods: "rpkm" (default), "uqua" (upper quartile), "tmm" (trimmed mean of M) or "n" (no normalization). See Noiseq user's manual for more information. Example: noiseq_normethod=tmm
replicatevalue	Type of replicates to be used to perform the DE analysis with Noiseq. Allowed values are: Technical, biological or no. Inclusion of technical or biological replicates is highly recommended. Technical replicates involve taking one sample from the same source tube, and analysing it across multiple conditions, while biological replicates are different samples measured across multiple conditions. See Noiseq user's manual for more information. By default, technical replicates option is chosen. Example: replicatevalue=biological
kvalue	Counts equal to 0 are replaced by k value to perform the DE analysis with Noiseq. See Noiseq user's manual for more information. By default, kvalue = 0.5. Example: kvalue = 1
lcvalue	Additional length correction in the normalization process. This correction is done by dividing expression by length^lc to perform the DE analysis with Noiseq. See Noiseq user's manual for more information. By default, lcvalue = 0 for no length correction is applied. Example: lcvalue = 0.5.
pnrvalue	Percentage of the total reads used to simulate each sample when no replicates are available to perform the DE analysis with Noiseq. See Noiseq user's manual for more information. By default, pnrvalue = 0.2. Example: pnrvalue = 0.5.
nssvalue	Number of samples to simulate for each condition (nss>= 2) to perform the DE analysis with Noiseq. See Noiseq user's manual for more information. By default, nssvalue = 5. Example: nssvalue = 3.
vvalue	Variability in the simulated sample total reads to perform the DE analysis with Noiseq. Sample total reads is computed as a random value from a uniform

	distribution in the interval [(pnr-v)*sum(counts), (pnr+v)*sum(counts)]. See Noiseq user's manual for more information. By default, vvalue = 0.02. Example: vvalue = 0.05.

4.2.4.3. Examples of configuration file to run DEAnalysis module

1) Differential expression analysis of miRNAs by EdgeR: In this example, user will perform the differential expression analysis of the quantified miRNAs. User will execute miARma from its own directory, the input files are tab delimited files with the counts from example 2.1. located in the input directory (Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/ in this case). The differential expression analysis will be performed with EdgeR, using the experimental conditions defined in the target file and the comparisons defined in the contrast file. Filtering process will be carry out discarding those counts less than 2 counts per million in at least 1 replicate. Normalization process will be performed using TMM method.

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/
organism=human
strand=yes


[DEAnalysis]

desoft=EdgeR
targetfile=Examples/basic_examples/miRNAs/data/targets.txt
contrastfile=Examples/basic_examples/miRNAs/data/contrast.txt
filter=yes
cpmvalue=2
repthreshold=1
edger_normethod=TMM
replicates=yes
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/DeNovo_miRNAs/3.DEAnalyis/3.1.DEAnalysis_EdgeR_miRNAs.ini
```

2) Differential expression analysis of miRNAs by Noiseq: In this example, user will perform the differential expression analysis of the counts corresponding to miRNAs. User will execute miARma from its own directory, the input files are tabulated files with the counts from example 2.1. located in the input directory (Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/ in this case). The differential expression

analysis will be performed by Noiseq tool, using the experimental conditions defined in the target file and the comparisons defined in the contrast file. Filtering process will be carry out with CPM method using as cutoff 2 counts per million and a coefficient of variation per condition of 80%. Normalization process will be performed using tmm method. In addition, counts equal to 0 are replaced by 1 in the normalization process and a length correction will also performed be performed using 0.5 value. In this analysis the q value cufoff has been established in 0.9 to select the differentially expressed elements with Noiseq.

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/
organism=human
strand=yes

[DEAnalysis]
desoft=Noiseq
targetfile=Examples/basic_examples/miRNAs/data/targets.txt
contrastfile=Examples/basic_examples/miRNAs/data/contrast.txt
qvalue=0.9
filter=yes
filtermethod=1
cpmvalue=2
cutoffvalue=80
noiseq_normethod=tmm
replicates=yes
replicatevalue=biological
kvalue=1
lcvalue=0.5
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/DeNovo_miRNAs/3.DEAnalyis/3.2.DEAnalysis_Noiseq_miRNAs.ini
```

3) Differential expression analysis of miRNAs by EdgeR and Noiseq: In this example, user will perform the differential expression analysis of the counts corresponding to miRNAs. User will execute miARma from its own directory, the input files are tabulated files with the counts from example 2.1. located in the input directory (Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/ in this case). The differential expression analysis will be performed with both, EdgeR and Noiseq tools, using the experimental conditions defined in the target file and the comparisons defined in the contrast file. Filtering process will be carry out with default filter methods (those counts less than 1 counts per million will be discarded), and normalization process will be performed as default option using tmm method for analysis with EdgeR

and rpkm for Noiseq. In this analysis the default cutoff of q value (0.8) will be used to select the differentially expressed elements with Noiseq.

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/
organism=human
strand=yes

[DEAnalysis]
desoft=EdgeR-Noiseq
targetfile=Examples/basic_examples/miRNAs/data/targets.txt
contrastfile=Examples/basic_examples/miRNAs/data/contrast.txt
filter=yes
replicates=yes
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/DeNovo_miRNAs/3.DEAnalyis/3.3.DEAnalysis_EdgeR_Noiseq_miRNAs.ini
```

4) Differential expression analysis of miRNAs by EdgeR and Noiseq without replicates: In this example, user will perform the differential expression analysis of the counts corresponding to miRNAs. User will execute miARma from its own directory, the input files are tabulated files with the counts from example 2.1. located in the input directory (Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/ in this case). The differential expression analysis will be performed with both, EdgeR and Noiseq tools, using the experimental conditions defined in the target file and the comparisons defined in the contrast file. Filtering process will be carry out with default filter methods (those counts less than 1 counts per million will be discarded), and normalization process will be performed as default option using tmm method for analysis with EdgeR and rpkm for Noiseq. In this case, the analysis contains no replicates, so as recommended by edgeR vignette, a manually selected valued should be introduced by the user. For analysis with EdgeR the common biological coefficient variation (bcv) used to simlulate the dispersion of the data has been established in 0.3. For Noiseq analysis the simulation of the replicates is more complex being characterized for the percentage of the total reads used to simulated each sample (pnrvalue), the number of samples to simulate for each condition (nssvalue) and the variability in the simulated sample total reads (vvalue). In this analysis, a pnrvalue of 0.5, a nssvalue of 3 and a vvalue of 0.05 has been established. This analysis the default cutoff of q value (0.8) will be used to select the differentially expressed elements with Noiseq.

```
[General]
```

```
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/
organism=human
strand=yes

[DEAnalysis]
desoft=EdgeR-Noiseq
targetfile=Examples/basic_examples/miRNAs/data/targets_no_replicates.txt
contrastfile=Examples/basic_examples/miRNAs/data/contrast.txt
filter=yes
replicates=no
bcvvalue=0.3
replicatevalue=no
pnrvalue = 0.5
nssvalue = 3
vvalue = 0.05
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/DeNovo_miRNAs/3.DEAnalyis/3.4.DEAnalysis_EdgeR_Noiseq_miRNAs_no_replicates.ini
```

4.2.5. Target prediction module

The aim of this module is to perform the target prediction analysis. To achieve this goal, miARma-Seq has implemented miRGate tool. miRGate is a new database containing novel computational predicted miRNA-mRNA pairs that are calculated using well-established algorithms such as miRanda, Pita, TargetScan, RNAhybrid or MicroTar. In addition, miRGate includes experimental validated miRNA-mRNAs pairs providing to miARma-Seq a high reliability tool to miRNA-mRNA target prediction. Thus, miRGate provides the target genes of the input DE miRNA data, the target miRNAs of the input DE mRNA data or even the DE mRNAs targeted by DE miRNAs from the negative correlations between DE miRNAs and DE mRNAs.

4.2.5.1. Input/Output files

Input: Tabulated file (xls format) with the DE expressed miRNAs or mRNAs from the Noiseq or EdgeR analysis with the DEAnalysis module. To obtain more information about the format of these files please consult the output format of the Section 4.2.4. Differential Expression module.

Output:

1. Tabulated file (excel compatible) with the predicted targets and the statistical values of the prediction. The standard output file will contain 12 columns:

- [miRNA]- Name of the miRNA.
- [miRNA FC]- Fold Change of the miRNA obtained in the DE analysis.

- [miRNA FDR]- FDR value of the miRNA obtained in the DE analysis.
- [Ensembl Gene]- Ensembl code of the targeted Gene.
- [Gene Symbol]- Symbol of the targeted gene.
- [Ensembl Transcript]- Ensembl code of the targeted transcript.
- [Gene FC]- Fold Change of the mRNA obtained in the DE analysis.
- [Gene FDR]- FDR value of the miRNA obtained in the DE analysis.
- [Method]- Method of prediction.
- [Target Site]- Type of target site.
- [Score]- Z-score of the prediction.
- [Energy]- Energy of the prediction.

2. Summary results report (xls format) with the main statistics of the analysis. This report will be generated in the output directory provided by the user. In this report, "miRNA-mRNA Target Predictions by miRGate" section with the path of the target prediction results can be founded, together with 2 summary tables:

miRNAs with more associations:

-[File]- Name of the DE Analysis results file.
-[miRNA]- Name of the miRNA with more associations (5 for each file).
-[Number of associations]- Number of associations.

Genes more regulated:
-[File]- Name of the DE Analysis results file.
-[GeneName]- Name of the gene regulated for more miRNAs (5 for each file).
-[Number of associations]- Number of associations.

An example of the summary report can be consulted in the following link.

4.2.5.2. Configuration file:

To execute this analysis the heading **[TargetPrediction]** must be included in the configuration file. The parameters included in this analysis are:

Optional parameters	
noiseq_cutoff	Cutoff value to select statistically significant results performed with Noiseq tool. The selected entities will be included in the target prediction analysis. By default, this value has been established in 0.8. Example: noiseq_cutoff=0.9.
edger_cutoff	Cutoff value to select statistically significant results performed with EdgeR tool. The selected entities will be included in the target prediction analysis. By default, this value has been established in 0.05. Example: edger_cutoff=0.01

fc_threshold	Value to filter low DE expressed miRNAs. As logFC is expressed in log2 a fc_threshold=1 means a real change in expression of 2 folds. Example: fc_threshold=3
genes_folder	Path of the folder with DE mRNAs to obtain DE mRNAs-miRNAs pairs with miRGate. Example: Examples/basic_examples/mRNAs/results/

4.2.5.3. Examples of configuration file to run Target Prediction module

1) Target prediction analysis of DE miRNAs: In this example, user will perform the target prediction analysis of the DE miRNAs. User will execute miARma from its own directory, the input files are tabulated files with the results of the DE analysis from example 3.1. located in the input directory (Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/ in the example) and the results will be saved in results directory (Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/ in this case). The target prediction will be performed only in those miRNAs with a probability greater than 0.8 in Noiseq analysis and a p-value less than 0.05 in EdgeR analysis.

```
[General]
type=miRNA
verbose=0
read_dir=Examples/basic_examples/miRNAs/reads/
threads=4
label=Hypoxia
miARmaPath=.
output_dir=Examples/basic_examples/miRNAs/DeNovo_miRNAs/results/
organism=human
strand=yes

[TargetPrediction]
noiseq_cutoff=0.8
edger_cutoff=0.05
fc_threshold=1
```

This configuration file can be founded in the examples folder of the miARma downloaded directory and can be executed using:

```
perl miARma Examples/basic_examples/miRNAs/DeNovo_miRNAs/4.TargetPrediction/4.1.TargetPrediction.ini
```