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

circRNA detection from RNA-Seq Data 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 software. 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. circRNA Example installation instructions

1. Inside miARma folder, download data:

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

2. Uncompress it :

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

3. Check the parameters (optional step):

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

4. Execute the examples:

```
miARma>perl miARma Examples/basic_examples/circRNAs/1.Quality/1.Quality.ini
```

3.3. Other needed data for miARma execution

miARma uses BWA tool for read alignment. For the circRNA example included in miARma, human hg19 genome index for BWA reference genome in fasta file are needed.

3.3.1. BWA index installation:

1. Downloading from miARma folder:

```
miARma>curl -L -O http://miarmaseq.cbbio.es/download/Genome/Index_bwa_hg19.tar.bz2
```

2. Extracting:

```
miARma>tar -xjf Index_bwa_hg19.tar.bz2
```

3.3.2. 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
```

4. circRNA-SEQ ANALYSIS

miARma-Seq presents a highly flexible modular structure to perform the different stages of the circRNA 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.

In order to better explain mRNA-Seq analysis, data from GEO (GEO code: GSE52778) 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 mRNA profiles obtained via RNA-Seq for four primary human airway smooth muscle cell lines that were treated with dexamethasone, albuterol, dexamethasone+albuterol or were left untreated. To make easier the understanding of the pipeline, only samples treated with dexamethasone and control samples will be used. Examples installation is described in section 3.2.

A complete example of the pipeline can be executed using:

```
perl miARma Examples/basic_examples/circRNAs/miARma_circRNAs_pipeline.ini
```

4.1. General features

4.1.1. 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 using 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=circRNA
read_dir	Folder for input files where raw data from high thorughput sequencing in fastq format are located. Example: read_dir=Examples/basic_examples/circRNAs/reads/
label	Name to identify the analysis. This name will appear in the output files and plots. Example: label= Asthma
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/circRNAs/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: SRR1039508_1.fastq and SRR1039508_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

4.1.2. Examples of the general information in the configuration file

1) General information of circRNA analysis.- In this example, user defines general parameters to execute miARma from its own directory, the pipeline input files are fastq files from human located in the input directory (Examples/basic_examples/circRNAs/reads/ in the example) and the results will be saved in results directory (Examples/basic_examples/circRNAs/results/ in this case) including the name of the experiment (Asthma in this example). The example is composed of six paired-end stranded samples. The analysis will perform with 4 threads and the execution data will not showed in the screen.

```
[General]
type=circRNA
verbose=0
read_dir= Examples/basic_examples/circRNAs/reads/
threads=4
label= Asthma
miARmaPath=.
output_dir= Examples/basic_examples/circRNAs/results/
organism=human
seqtype=Paired
strand=no
```

4.2. 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.

4.2.1. 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. 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 accept the keyword "both" in

	case you want the analysis twice: before and after the pre-processing of the reads. Since circRNA analysis do not includes adapter trimming step, only prefix=pre is allowed. Example: prefix=pre

4.2.3. Examples of configuration file to run Quality analysis

1) Quality analysis of circRNA analysis. In this example, user will perform the quality analysis executing miARma from its own directory, the pipeline input files are fastq files from human samples located in the input directory (Examples/basic_examples/circRNAs/reads/in the example) and the results will be saved in results directory (Examples/basic_examples/circRNAs/results/ in this case) including the name of the experiment (Asthma in this example). The example consists of unstranded Paired-end samples. The analysis will perform with 4 threads and the execution data will not be showed in the screen.

```
[General]
type=circRNA
verbose=0
read_dir= Examples/basic_examples/circRNAs/reads/
threads=4
label= Asthma
miARmaPath=.
output_dir= Examples/basic_examples/circRNAs/results/
organism=human
seqtype=Paired
strand=no

[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/circRNAs/1.Quality/1.Quality.ini
```

To inspect results for a Fastq file named SRR1039508 , please check Examples/ basic_examples/circRNAs/results/Pre_fastqc_results/SRR1039508_1.fq_fastqc/fastqc_report.html

4.3. Aligner module

The aim of the aligner module is to align sequenced reads against a reference genome. For circRNA analysis, miARma-Seq includes BWA tool.

The reference genome to align reads is mandatory, so it can be provided as pre-built indexes or it must be created from a fasta file. To download bwa index or the human genome fasta file used in the examples, please see section 3.3.

4.3.1. Input/Output files

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

Output:

1. Aligned files in BAM format saved in the output directory provided by the user in the "bwa_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.
-[Fail to align]- Number of reads that fail to align.

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

4.3.2. 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 for circRNA-Seq analysis BWA is implemented in miARma-Seq. Others aligners are available for miRNA analysis (Bowtie1, Bowtie2, miRDeep) and mRNAs (topHat). Please see the specific documentation of these analyses to deep in their use. Example: aligner=BWA
Specific parameters for Pre-built BWA index:	
bwaindex	Path of the pre-built BWA index to perform the alignment of the reads. This index can be generated from a fasta file using miARma (See options below). Example: bwaindex=Genomes/Indexes/BWA/human/bwa_homo_sapiens19
Specific parameters to create a new index from fasta file:	
fasta	Path to the genome sequence fasta file to build the correspondent index. Example: fasta= Genomes/Indexes/BWA/human/homo_sapiens19.fa
indexname	Name to write in the generated index files. Example: hg19

4.3.3. Examples of configuration file to run Aligner module

1) Alignment with BWA using a pre-built index: In this example, user will perform the alignment of the input fastq files located at the input directory (Examples/basic_examples/circRNAs/reads/ in the example) against a pre-built BWA index located in index directory (Genomes/Indexes/BWA/human/ in this example). The results will be saved in results directory (Examples/basic_examples/circRNAs/results/ in this case) including the name of the experiment (Asthma in this example). The example is a paired-end experiment with no stranded samples. User will execute miARma from its own directory.

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

```
read_dir= Examples/basic_examples/circRNAs/reads/
threads=4
label= Asthma
miARmaPath=.
output_dir= Examples/basic_examples/circRNAs/results/
organism=human
seqtype=Paired
strand=no


[Aligner]
aligner=BWA
bwaindex=Genomes/Indexes/BWA/human/ bwa_homo_sapiens19
```

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/circRNAs/2.Aligner/2.1.BWA_prebuilt_index.ini
```

2) Alignment with BWA without index files: In this example, user will perform the alignment of the input fastq files located at (Examples/basic_examples/circRNAs/reads/ in the example) against a new generated index. This index will be named with the index name (bwa_homo_sapiens19 in this case) and created from a fasta file located in index directory (Genomes/Indexes/BWA/human/ in this example). The results will be saved in results directory (Examples/basic_examples/circRNAs/results/ in this case) including the name of the experiment (Asthma in this example). The example is a paired-end experiment with no stranded samples. User will execute miARma from its own directory.

```
[General]
type=circRNA
verbose=0
read_dir= Examples/basic_examples/circRNAs/reads/
threads=4
label= Asthma
miARmaPath=.
output_dir= Examples/basic_examples/circRNAs/results/
organism=human
seqtype=Paired
strand=no


[Aligner]
aligner=BWA
fasta= Genomes/Indexes/BWA/human/homo_sapiens19.fa
indexname= bwa_homo_sapiens19
```

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/circRNAs/2.Aligner/2.2.BWA_index_from_fasta.ini
```

4.4. 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 mRNA analysis, miARma-Seq has implemented featureCounts.

4.4.1. 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 mRNA or gene identifier 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:

	SRR1039508	SRR1039509	SRR1039512	SRR1039513	SRR1039516	SRR1039517
chr6:31964206|31996838	7	4	6	0	2	0
chr7:94052369|94053684	3	3	9	2	3	5
chr7:94056500|94056607	98	4	134	86	118	147
chrM:3599|3678	5	6	29	17	32	42
chrM:8922|8998	9	9	85	0	56	108
chrX:2700107|2700169	9	8	13	0	10	17

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 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).
-[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.

4.4.2. 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/circRNAs/data/Homo_sapiens_GRCh37.74_chr.gtf
fasta	Path of the genome sequence in fasta format. Example: fasta= Genomes/Indexes/BWA/human/homo_sapiens19.fa
method	Version of CIRI to identify and quantify circRNAs. CIRI1 or CIRI2 are available. Example: method=CIRI1 [Default]

4.4.3. Examples of configuration file to run ReadCount module

1) Quatification of circRNAs by CIRI: In this example, user will perform the read summarization corresponding to circRNAs taking as a reference the GTF from human genome (to download the gtf file used in this example see section 3.2.) and the genome sequence in fasta format. User will execute miARma from its own directory. The input files are aligned sam files from example 2.1. located in the input directory (Examples/basic_examples/circRNAs/results/ in the example) and the results will be saved in results directory (Examples/basic_examples/circRNAs/results/ in this case).

```
[General]
type=circRNA
verbose=0
read_dir= Examples/basic_examples/circRNAs/reads/
threads=4
label= Asthma
miARmaPath=.
output_dir= Examples/basic_examples/circRNAs/results/
organism=human
seqtype=Paired
strand=no

[ReadCount]
database=Examples/basic_examples/circRNAs/data/Homo_sapiens_GRCh37.74_chr.gtf
fasta= Genomes/Indexes/BWA/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/circRNAs/3.circRNAs_identification/3.1.ReadCount.ini
```

4.5. 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 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.

4.5.1. 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:

	SRR1039508	SRR1039509	SRR1039512	SRR1039513	SRR1039516	SRR1039517
chr6:31964206|31996838	7	4	6	0	2	0
chr7:94052369|94053684	3	3	9	2	3	5
chr7:94056500|94056607	98	4	134	86	118	147
chrM:3599|3678	5	6	29	17	32	42
chrM:8922|8998	9	9	85	0	56	108

chrX:2700107|2700169	9	8	13	0	10	17

Output:

1. <u>Tabulated results files</u> (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)_(name_of_readcount_output_file)_EdgeR_results_(label_of_the_comparison).xls

Example: For the comparison of Asthma experiment, the resultant file will be named as: Asthma_circRNAs.tab_EdgeR_results_Comp.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**
chr19:41117765|41117890	-8.789673158	12.8784057	0.155683015	1
chr4:100237372|100266238	9.801513485	13.84119864	0.157652248	1
chr16:55517996|55518076	8.648841063	12.74659716	0.174990097	1
chr6:31239776|31324734	-8.192615047	12.32858258	0.194091034	1
chr5:179146669|179146782	-7.943623307	12.10583752	0.196838419	1
chrM:4665|4748	-5.908332276	10.49878104	0.20352508	1
chrM:8928|9089	5.396090533	10.18302873	0.203766067	1

- **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)_(name_of_readcount_output_file)_Noiseq_results_(label_of_the_comparison).xls

Example: For the comparison of Asthma experiment, the resultant file will be: Asthma_circRNAs.tab_Noiseq_results_Comp.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.

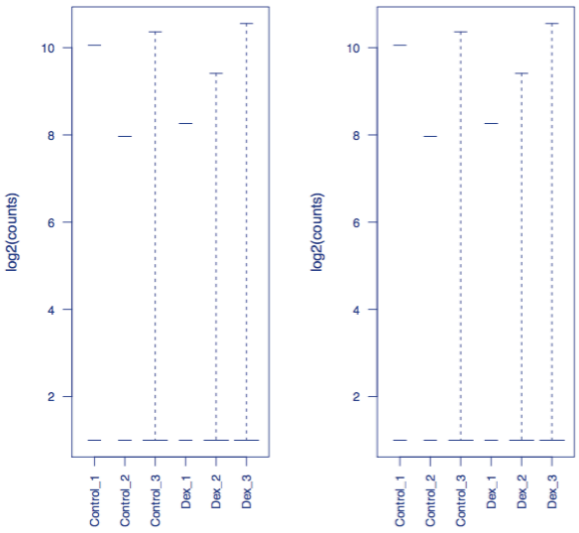
Example:

	Treated_mean	Untreated_mean	M	D	prob	ranking
chr17:48270355|48271987	34.02749422	37272.56668	-10.0971	37238.53	0.986008837	-37238.54056
chr15:48773852|48779397	34.02749422	28351.88129	-9.7025	28317.85	0.983308787	-28317.85546
chr4:100237372|100266238	34.02749422	24200.67126	-9.4741	24166.64	0.981590574	-24166.64562
chr4:100208007|100237459	23751.19096	44.16180887	9.0709	23707.02	0.981345115	23707.03089
chr10:5014393|5248360	34.02749422	22080.90443	-9.3418	22046.87	0.980363279	-22046.87892
chr10:5011014|5247797	34.02749422	21639.28635	-9.3127	21605.25	0.979626902	-21605.26086
chr22:45929641|45931217	21028.99143	44.16180887	8.8953	20984.82	0.979626902	20984.8315

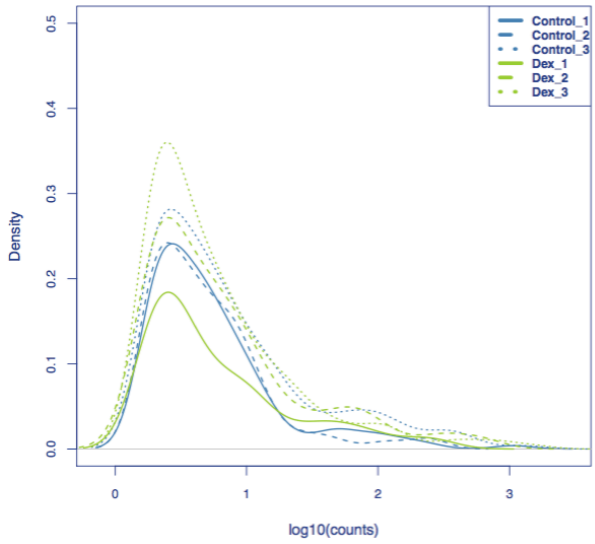
2. Exploratory plots of the analysis. miARma-Seq provides a exhaustive PDF report with different plots in order to make easier to the user the interpretation of the data. This report contains:

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:

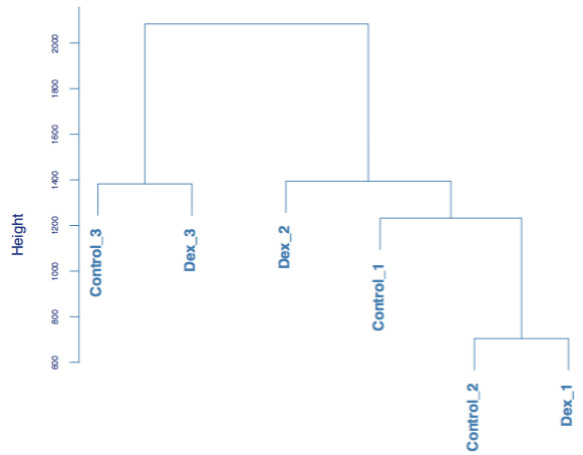


- Density plot of the distribution of the counts. The second and third page of the report contains 2 density plots with the distribution of the counts, before (second page) and after (third) the normalization process. The plot represents the density of the log10 of the counts for each sample. The different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

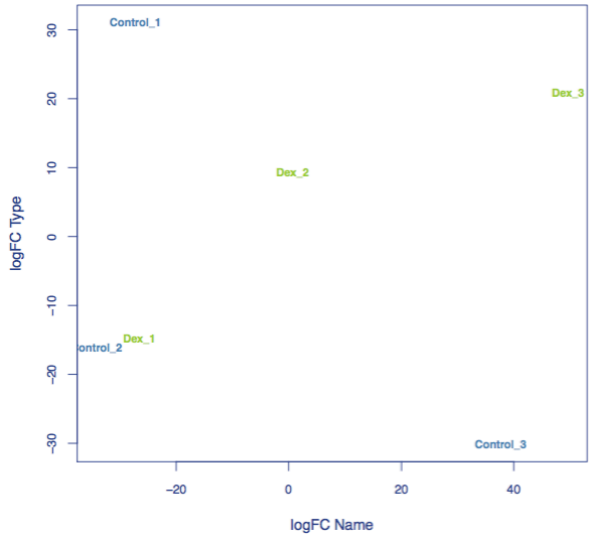


2.2. Analysis of the samples similarity- In order to examine the quality of the data obtained in the experiment, miARma-Seq has implemented different plots, which allows the inspection of the diversity between the samples. For a good quality experiment, the samples belonging to the same experimental 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.

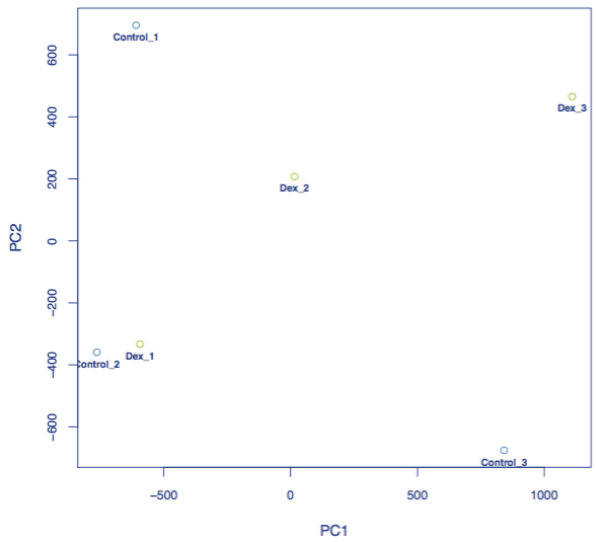
-Hierarchical clustering of the samples: The hierarchical clustering plots, before and after normalization process, classify the samples according to their similarity. The distance of the branch is proportional to the sample distance.



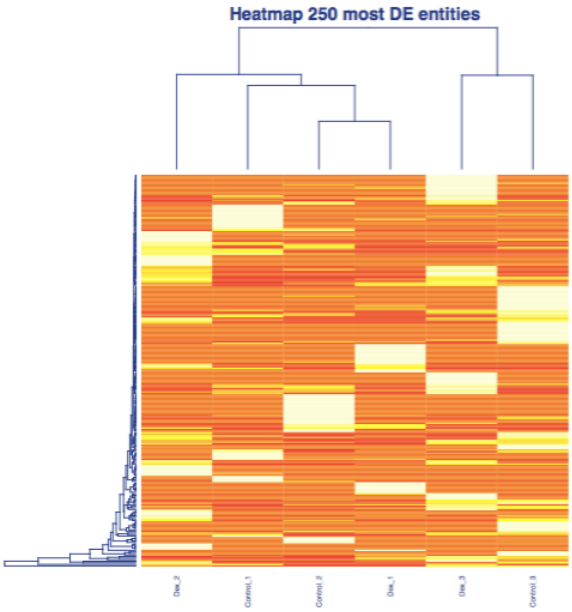
-Multidimensional plot (MDS): The MDS plot divides the samples in a two-dimensional plot according to their similarity. Each sample is represented with its name and coloured according to the experimental condition that belongs to.



-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.



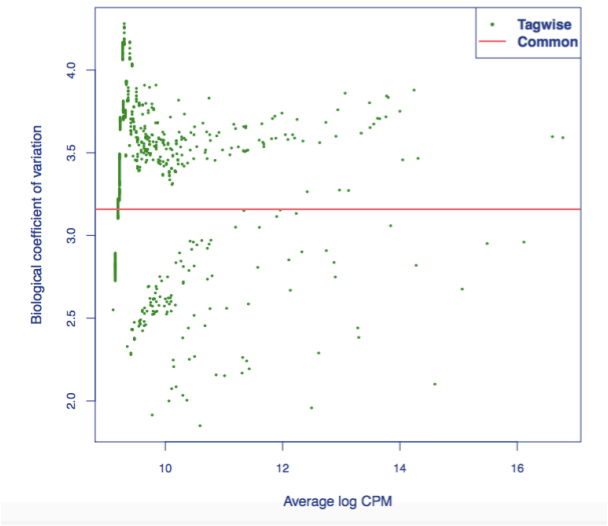
-Heatmap: The heatmap allows to the user evaluate the similarity between the samples according to the 250 most expressed miRNAs expression.



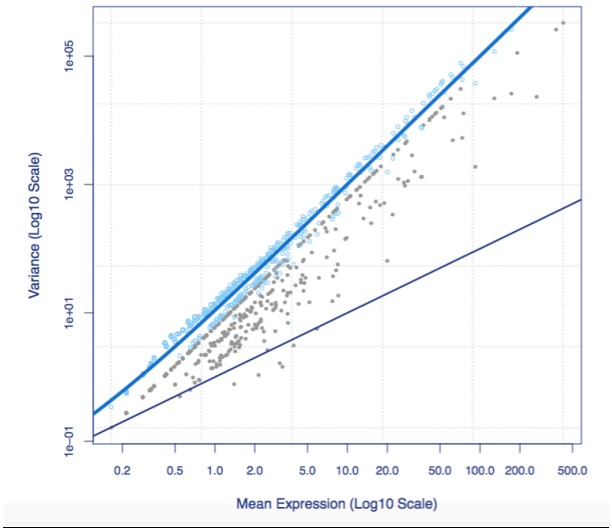
3. Results plots of the analysis.- miARma-Seq generates a PDF report with plots to explore the results with both tools, EdgeR and Noiseq.

3.1. Results plots with EdgeR:

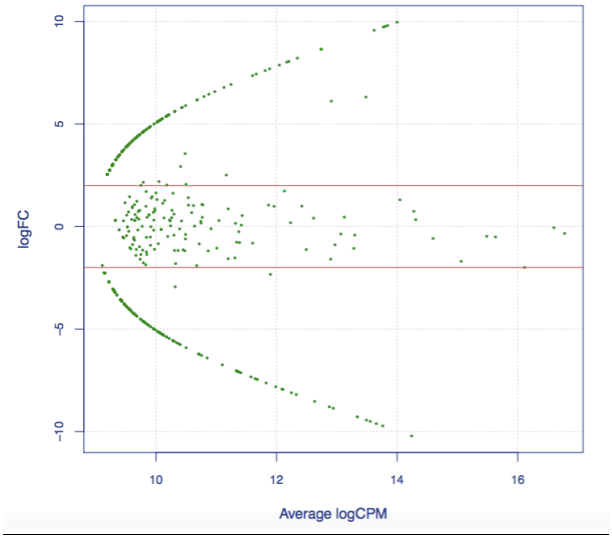
-Biological Variation Plot: The square root of dispersion is the coefficient of biological variation (BCV). This plot illustrates the relationship of biological coefficient of variation versus mean log CPM.



-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.

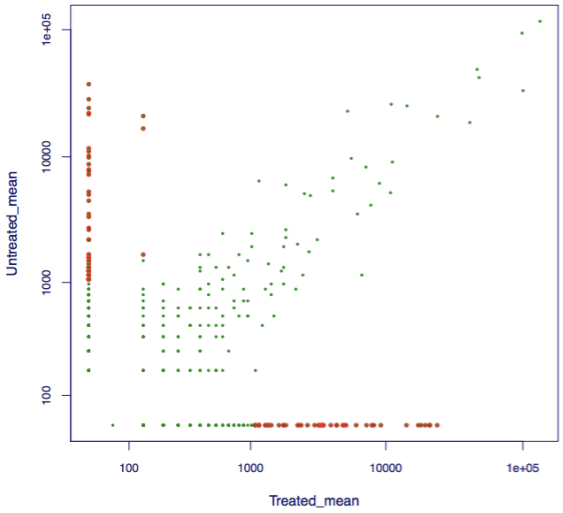


-Expression Plot: miARma-Seq generates one expression plot for each comparison. This plot shows all the logFCs against average count size, highlighting the DE genes.

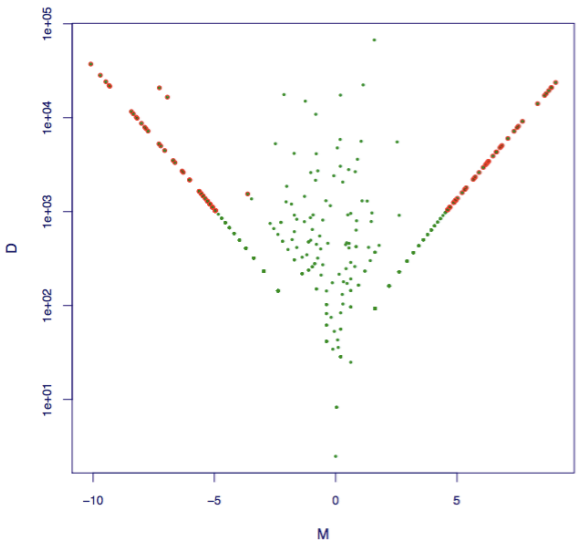


3.2. Results plots with Noiseq: For each comparison a PDF report with the results plot is generated

- 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.

4.5.2. 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 Type SRR1039508 Control_1 Untreated SRR1039512 Control_2 Untreated SRR1039516 Control_3 Untreated SRR1039509 Dex_1 Treated SRR1039513 Dex_2 Treated SRR1039517 Dex_3 Treated 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 "Type" corresponds to the experimental condition, which in this case is the treatment or not with dexamethasone. This target file can be downloaded as stated in section 3.2. Example: targetfile= Examples/basic_examples/circRNAs/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=Untreated-Treated In this example, there is 1 different contrast conditions: Untreated-Treated. This contrast file can be downloaded as stated in section 3.2. Example: contrastfile= Examples/basic_examples/circRNAs/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 offers different options to filter the reads. While EdgeR is implemented with a filter processing using a value ofcounts 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=no
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.

4.5.3. Examples of configuration file to run DEAnalysis module

1) Differential expression analysis of circRNAs by EdgeR: In this example, user will perform the differential expression analysis of the counts corresponding to circRNAs. User will execute miARma from its own directory, the input files are tabulated files with the counts from example 3.1. located in the input directory (Examples/basic_examples/circRNAs/results/ in the example) and the results will be saved in results directory (Examples/basic_examples/circRNAs/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 and normalization process will not be performed.

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

```
read_dir= Examples/basic_examples/circRNAs/reads/
threads=4
label= Asthma
miARmaPath=.
output_dir= Examples/basic_examples/circRNAs/results/
organism=human
seqtype=Paired
strand=no


[DEAnalysis]
desoft=EdgeR
targetfile=Examples/basic_examples/circRNAs/data/targets.txt
contrastfile=Examples/basic_examples/circRNAs/data/contrast.txt
filter=no
edger_normethod=none
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/circRNAs/4.DEAnalysis/4.1.DEAnalysis_EdgeR_circRNAs.ini
```

2) Differential expression analysis of circRNAs by Noiseq: In this example, user will perform the differential expression analysis of the counts corresponding to circRNAs. User will execute miARma from its own directory, the input files are tabulated files with the counts from example 3.1. located in the input directory (Examples/basic_examples/circRNAs/results/in the example) and the results will be saved in results directory (Examples/basic_examples/circRNAs/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. Filtering process will not be carried out and normalization process will be performed using rpkm method (by default).

```
[General]
type=circRNA
verbose=0
read_dir= Examples/basic_examples/circRNAs/reads/
threads=4
label= Asthma
miARmaPath=.
output_dir= Examples/basic_examples/circRNAs/results/
organism=human
seqtype=Paired
strand=no

[DEAnalysis]
desoft=Noiseq
targetfile=Examples/basic_examples/circRNAs/data/targets.txt
contrastfile=Examples/basic_examples/circRNAs/data/contrast.txt
filter=no
```

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/circRNAs/4.DEAnalysis/4.2.DEAnalysis_Noiseq_circRNAs.ini
```

3) Differential expression analysis of circRNAs by EdgeR and Noiseq: In this example, user will perform the differential expression analysis of the counts corresponding to circRNAs. User will execute miARma from its own directory, the input files are tabulated files with the counts from example 3.1. located in the input directory (Examples/basic_examples/circRNAs/results/ in the example) and the results will be saved in results directory (Examples/basic_examples/circRNAs/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 not be carried out and normalization process will be performed with the default option using rpkm for Noiseq and will not be performed for EdgeR analysis.

```
[General]
type=circRNA
verbose=0
read_dir= Examples/basic_examples/circRNAs/reads/
threads=4
label= Asthma
miARmaPath=.
output_dir= Examples/basic_examples/circRNAs/results/
organism=human
seqtype=Paired
strand=no

[DEAnalysis]
desoft=EdgeR-Noiseq
targetfile=Examples/basic_examples/circRNAs/data/targets.txt
contrastfile=Examples/basic_examples/circRNAs/data/contrast.txt
filter=no
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/circRNAs/4.DEAnalysis/4.3.DEAnalysis_EdgeR_Noiseq_circRNAs.ini
```