|  |
| --- |
| *Supplementary Information*  **Circr: Identify and Prioritize circRNA-miRNA Associations through Integration of Bioinformatic Predictions and Experimental Data**  Martina Dori1, Jimmy Caroli2, Silvio Bicciato1  1 Department of Life Sciences, University of Modena and Reggio Emilia, Modena, Italy  2 Department of Drug Design and Pharmacology, University of Copenhagen, Copenhagen, Denmark |

|  |  |
| --- | --- |
| **WORKFLOW** **DESCRIPTION**…………………………………………………………………………………… | 2 |
| Step 1: input preparation……………………………………………………………………………………... | 2 |
| Step 2: prediction of miRNA-circRNA interactions…………………………….…………………………... | 2 |
| Step 3: predict interactions with third party software…....……………………………………………….... | 2 |
| Step 4: comparing predicted circulars to database..……………………………………….……………… | 3 |
| Step 5: filtering validated interactions…………………………………………………………..……………... | 3 |
| **INSTALLATION AND USAGE**……………………………..…………………………………………………… | 4 |
| Installation...……………………………………………………………………………………..……………... | 3 |
| Dependencies……………………………………………………………………………………..……………... | 3 |
| Parameters………………………………………………………………………………………..……………... | 5 |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| **REFERENCES**…………………………………………………………………………………………………… | 11 |

**WORKFLOW DESCRIPTION**

Here we present Circr, a computational tool for the prediction of circRNA-miRNA associations. It combines publicly available algorithms for *de novo* prediction of miRNA binding sites on target sequences with experimentally validated miRNA-AGO and miRNA-RNA sites in different organisms. Circr can be used with either the provided support files or with custom ones, allowing the analysis of novel and not previously annotated circRNAs in virtually any species of interest. Circr is designed to perform its analysis workflow in three sequential steps: I) the input preparation step, which calculates the exonic coordinates starting from the circRNA input file, ii) the prediction step, where three different softwares are used to perform interaction prediction (namely miRanda, targetscan and RNAhybrid) and iii) the result comparison step, in which output derived from the prediction software are retrieved, merged and compared against a database of validated interactions and AGO peaks. Finally, the resulting annotated table is provided in a comma-separated file that allows the user an easy filtering and selection of interesting circRNA-miRNA sites for validation and functional investigations.

**Step 1: input preparation**

In the first step, Circr generates a dataframe of fasta sequences necessary for the following analysis step starting from the genomic coordinates of circRNAs provided as input ( -i or --input parameter, required).

For circRNA overlapping genes, Circr assumes them to include only exons, therefore each transcript is split into its intron/exon coordinates and only the coordinates of the latter are retained for fasta sequence reconstruction. Conversely, antisense circRNA and intergenic ones are considered as a single exon. By providing the organism and genome build required for the analysis via the command line variables (-s or --organism and -v or –genome-version), the software will automatically fetch the other required input files from its own database for the calculation purposes. Users can also provide custom genome, reference genes and rRNA files via the dedicated input command line commands (--genome, --gtf and –rRNA respectively).

Since transcripts that overlap genes can undergo alternative splicing, implying that they might not include all exon within the circRNA start/end coordinates or retain introns, it is also possible to provide the coordinates of all the features (exon/intron) known to be included in each circRNA. In this case, it is necessary to submit the -c (or –coordinates) flag, which instructs Circr to not perform the initial intron/exon splitting. This flag can also be used to avoid exon/intron splitting without prior knowledge of the feature composition, thus considering each circRNA as a single exon.

**Step 2: prediction of miRNA-circRNA interactions**

*Miranda ref (version 3.3, (Enright AJ, John B, Gaul U, Tuschl T, Sander C, Marks DS. MicroRNA targets in drosophila. Genome Biol. 2003;5(1):R1.))*

*TargetScan ref (version 7.0, (Agarwal V, Bell GW, Nam J, Bartel DP. Predicting effective microRNA target sites in mammalian mRNAs. eLife, 4:e05005, (2015)))*

*RNAhybrid ref (Rehmsmeier, M., Steffen, P., Höchsmann, M., Giegerich, R. 2004 Fast and effective prediction of microRNA/target duplexes RNA 10 1507 –1517)*

In the second step, the circular database generated in the previous step is processed and prepared for the prediction of interaction using three different software integrated in Circr: Miranda [ref], Targetscan [ref] and RNAhybrid [ref]. Circr is designed to perform analysis using a multi-threaded approach, limiting computation time and maximizing computational power. The number of spawned threads can be adjusted using the parameter --threads when launching Circr. Default parameters are set for the analyses performed via the third party software. Results generated are then collected in a single internal dataframe, and generated files and data are garbage collected and removed from the system to prevent space constraints.

**Step 3: comparison of predicted interaction against validated databases**

In the last step, predicted interactions obtained via the analysis using Miranda, Targetscan and RNAhybrid are compared against databases of validated interactions and AGO peaks, which are provided with the data packages of Circr. A table is then generated, reporting the information of each interaction calculated, along with the number of software able to predict that specific interaction, and presence/absence of intersection with the validation and AGO peaks databases.

**INSTALLATION AND USAGE**

**Installation**

Circr is freely available at [chissà] and is compatible with Linux, Mac OS and the MS Windows subsystem for Linux. While Targetscan is provided as a standalone software along the complete download package of Circr, Mirand and RNAhybrid are required to be already present in the working system. However, a yaml file is provided to produce a fully functional environment using Anaconda 3.0 where analysis with Circr can be performed. Moreover, Circr builds on some of the most common Python modules, which are listed in the **Dependencies** section of this Supplementary Materials. Circr is written in Python 3.6 and includes Targetscan (version 7.0).

**Dependencies**

MiDori is written in python3 and builds on several different modules, which are mandatory for the correct installation and usage of the software. Here we will list all the required modules for MiDori:*pandas, pybedtools, collections, multiprocessing, functools, itertools, operator.* All these modules can be installed via pip in python, by calling the appropriate module using the command install. For example, the command:  
  
pip install pandas  
  
will extract the module pandas from the python repositories and install it in your environment.   
Moreover, MiDori compares the prediction efficiency of three different software, namely Miranda, RNAhybrid and Targetscan. While the latter is provided as a standalone perl script, included in the downloadable package, the first two software must be downloaded and installed by the user from their respective official source (<http://cbio.mskcc.org/miRNA2003/miranda.html>; [https://bibiserv.cebitec.uni-bielefeld.de/rnahybrid](https://bibiserv.cebitec.uni-bielefeld.de/rnahybrid?id=rnahybrid_view_download)/).

**Parameters**

Circr is invoked in Python on an input bed file, representing either a list of circulars or associated coordinates, using the following example command, eventually specifying a set of additional parameters ([options]):

python3 Circr.py -i [INPUT] -o [OUTPUT][-h][options]

Most parameters are pre-set to optimized values, but users can easily modify default values to meet specific characteristics in their experimental data.

-i, --input [INPUT]

This parameter defines the input file to be analyzed with MiDori. The input file must contain either a list of circular RNAs or their exon coordinates. When the latter is provided, the -c (--coord) parameter should be provided too in the command line command.

-c, --coord [COORDINATES]

This parameter states if the provided input contains a list of exon coordinates rather than a simple list of circular RNAs coordinates.

-s, --organism [ORGANISM]

This parameter specifies the organisms used for the analysis. Organisms available in the database are mouse, human, worm, and fruitfly. The default organism used is human. Any other organism provided will result in an error message and exit.

-v, --genome-version [GENOME VERSION]

This parameter specifies the genome version used for the analysis. Versions available in the database are the following: hg19 and hg38 for human, mm9 and mm10 for mouse, ce10 and ce11 for worm, dm3 and dm5 for fruit fly. Any other genome version provided or mismatched input will result in an error message and exit.

--gtf [GTF]

This parameter specifies an alternative path for a user defined GTF file. This will override the default GTF file that would be fetched from the provided database. By default, defining the genome and the genome version would point the software to the specified files necessary for the correct functioning of the software.

--genome [GENOME]

This parameter specifies an alternative path for a user defined genome file. This will override the default genome file that would be fetched from the provided database. By default, defining the genome and the genome version would point the software to the specified files necessary for the correct functioning of the software.

--rRNA [rRNA]

This parameter specifies an alternative path for a user defined ribosomal RNA file. This will override the default ribosomal RNA file that would be fetched from the provided database. By default, defining the genome and the genome version would point the software to the specified files necessary for the correct functioning of the software.  
  
--miRNA [miRNA]

This parameter specifies an alternative path for a user defined miRNA file. This will override the default miRNA file that would be fetched from the provided database. By default, defining the genome and the genome version would point the software to the specified files necessary for the correct functioning of the software.  
  
--AGO [AGO PEAKS]

This parameter specifies an alternative path for a user defined AGO peaks file. This will override the default AGO peaks file that would be fetched from the provided database. By default, defining the genome and the genome version would point the software to the specified files necessary for the correct functioning of the software.  
  
--validated-interactions [VALIDATED INTERACTIONS]

This parameter specifies an alternative path for a user defined file containing validated interactions. This will override the default file containing validated interactions that would be fetched from the provided database. By default, defining the genome and the genome version would point the software to the specified files necessary for the correct functioning of the software.  
  
--threads [THREADS]

This parameter specifies the number of threads that will be spawned during the analysis. Default is 8 threads for each analysis. The number of threads should be defined according to the availability of the machine operating.   
  
-o, --output [OUTPUT]

This parameter specifies the output file name generated, which will be in the current working directory. The default name is “Midori\_Analysis.csv”. We suggest to specify an output name that will be helpful for your purposes when performing multiple analyses.

**REFERENCES**