Package ‘isomiRs’

January 5, 2024

Version 1.30.0
Date 2022-06-17
Type Package
Title Analyze isomiRs and miRNAs from small RNA-seq
Description Characterization of miRNAs and isomiRs, clustering and
differential expression.
biocViews miRNA, RNASeq, DifferentialExpression, Clustering,
      ImmunoOncology
BugReports https://github.com/lpantano/isomiRs/issues
Suggests knitr, rmarkdown, org.Mm.eg.db, targetscan.Hs.eg.db,
      pheatmap, BiocStyle, testthat
Depends R (>= 4.0), SummarizedExperiment
Imports AnnotationDbi, assertive.sets, BiocGenerics, Biobase, broom,
      cluster, cowplot, DEGreport, DESeq2, IRanges, dplyr,
      GenomicRanges, gplots, ggplot2, gtools, gridExtra, grid,
      grDevices, graphics, GGally, limma, methods, RColorBrewer,
      readr, reshape, rlang, stringr, S4Vectors, tidyr, tibble
LazyData true
Maintainer Lorena Pantano <lorena.pantano@gmail.com>
License MIT + file LICENSE
VignetteBuilder knitr
Roxygen list(markdown = TRUE)
RoxygenNote 7.1.2
Encoding UTF-8
git_url https://git.bioconductor.org/packages/isomiRs
git_branch RELEASE_3_18
git_last_commit ac97130
git_last_commit_date 2023-10-24
Repository Bioconductor 3.18
**Description**

Characterization of miRNAs and isomiRs, clustering and differential expression.
counts

Author(s)

Maintainer: Lorena Pantano <lorena.pantano@gmail.com>
Authors:

• Georgia Escaramis (CIBERESP - CIBER Epidemiología y Salud Publica)

See Also

Useful links:

• Report bugs at https://github.com/lpantano/isomiRs/issues

counts

Accessors for the count matrix of a IsomirDataSeq object.

Description

The counts slot holds the count data as a matrix of non-negative integer count values, one row for each isomiR, and one column for each sample. The normalized matrix can be obtained by using the parameter norm=TRUE.

Usage

counts.IsomirDataSeq(object, norm = FALSE)

## S4 method for signature 'IsomirDataSeq'
counts(object, norm = FALSE)

## S4 replacement method for signature 'IsomirDataSeq,matrix'
counts(object) <- value

Arguments

object A IsomirDataSeq object.
norm Boolean, return log2-normalized counts.
value An integer matrix.

Value

base::matrix with raw or normalized count data.

Author(s)

Lorena Pantano

Examples

data(mirData)
head(counts(mirData))
dat286.long

Data frame containing mirna from Argyropoulos's paper

Description

Usage
dat286.long

Format
mirna expression data in long format.

design

Accessors for the 'design' slot of a IsomirDataSeq object.

Description
The design holds the R formula which expresses how the counts depend on the variables in colData. See IsomirDataSeq for details.

Usage
## S4 method for signature 'IsomirDataSeq'
design(object)

## S4 replacement method for signature 'IsomirDataSeq,formula'
design(object) <- value

Arguments
object A IsomirDataSeq object.
value A formula to pass to DESeq2.

Value
design for the experiment

Examples
data(mirData)
design(mirData) <- formula(~ 1)
ego

**enrichResult class**

**Description**

enrichResult class

**Usage**

ego

**Format**

enrichResult class with the output of: `ego <- enrichGO(row.names(assay(gene_ex_rse, "norm")), org.Mm.eg.db, "ENSEMBL", ont = "BP")`

---

**findTargets**

Find miRNAs target using mRNA/miRNA expression

**Description**

This function creates a matrix with rows (genes) and columns (mirnas) with values indicating if miRNA-gene pair is target according putative targets and negative correlation of the expression of both molecules.

**Usage**

`findTargets(mirna_rse, gene_rse, target, summarize = "group", min_cor = -0.6)`

**Arguments**

- **mirna_rse**: SummarizedExperiment with miRNA information. See details.
- **gene_rse**: SummarizedExperiment with gene information. See details.
- **target**: Data.frame with two columns: gene and miRNA.
- **summarize**: Character column name in colData(rse) to use to group samples and compare between miRNA/gene expression.
- **min_cor**: Numeric cutoff for correlation value that will be use to consider a miRNA-gene pair as valid.

**Value**

miRNA-gene matrix
Examples

```r
data(isoExample)
mirna_ma <- data.frame(gene = names(gene_ex_rse)[1:20],
                      mir = names(mirna_ex_rse))
corMat <- findTargets(mirna_ex_rse, gene_ex_rse, mirna_ma)
```

---

**gene_ex_rse**

_Data frame containing gene expression data_

**Description**

Data frame containing gene expression data

**Usage**

```r
gene_ex_rse
```

**Format**

gene expression data with 18 samples: example of a time series data

---

**isoAnnotate**

_Annotate the rawData of the IsomirDataSeq object_

**Description**

Get the sequence and the name information for each isomiR, and the importance value (isomir_reads/mirna_reads) for each sample.

**Usage**

```r
isoAnnotate(ids)
```

**Arguments**

```r
ids Object of class IsomirDataSeq.
```

**Details**

```r
edit_mature_position represents the position at the mature sequence + nucleotide at reference + nucleotide at isomiR.
```

**Value**

```r
data.frame with the sequence, isomir name, and importance for each sample and isomiR.
```
isoCounts

Examples

```r
data(mirData)
head(isoAnnotate(mirData))
```

isoCounts  
Create count matrix with different summarizing options

Description

This function collapses isomiRs into different groups. It is a similar concept than how to work with gene isoforms. With this function, different changes can be put together into a single miRNA variant. For instance all sequences with variants at 3’ end can be considered as different elements in the table or analysis having the following naming hsa-miR-124a-5p.iso.t3:AAA.

Usage

```r
isoCounts(
  ids,
  ref = FALSE,
  iso5 = FALSE,
  iso3 = FALSE,
  add = FALSE,
  snv = FALSE,
  seed = FALSE,
  all = FALSE,
  minc = 1,
  mins = 1,
  merge_by = NULL
)
```

Arguments

- `ids` Object of class IsomirDataSeq.
- `ref` Differentiate reference miRNA from rest.
- `iso5` Differentiate trimming at 5 miRNA from rest.
- `iso3` Differentiate trimming at 3 miRNA from rest.
- `add` Differentiate additions miRNA from rest.
- `snv` Differentiate nt substitution miRNA from rest.
- `seed` Differentiate changes in 2-7 nts from rest.
- `all` Differentiate all isomiRs.
- `minc` Int minimum number of isomiR sequences to be included.
- `mins` Int minimum number of samples with number of sequences bigger than minc counts.
- `merge_by` Column in coldata to merge samples into a single column in counts. Useful to combine technical replicates.
Details

You can merge all isomiRs into miRNAs by calling the function only with the first parameter `isoCounts(ids)`. You can get a table with isomiRs altogether and the reference miRBase sequences by calling the function with `ref=TRUE`. You can get a table with 5' trimming isomiRs, miRBase reference and the rest by calling with `isoCounts(ids, ref=TRUE, iso5=TRUE)`. If you set up all parameters to TRUE, you will get a table for each different sequence mapping to a miRNA (i.e. all isomiRs).

Examples for the naming used for the isomiRs are at http://seqcluster.readthedocs.org/mirna_annotation.html#mirna-annotation.

Value

`IsomirDataSeq` object with new count table. The count matrix can be access with `counts(ids)`.

Examples

data(mirData)
ids <- isoCounts(mirData, ref=TRUE)
head(counts(ids))
# taking into account isomiRs and reference sequence.
ids <- isoCounts(mirData, ref=TRUE, minc=10, mins=6)
head(counts(ids))

isoDE

Differential expression analysis with DESeq2

Description

This function does differential expression analysis with `DESeq2::DESeq2-package` using the specific formula. It will return a `DESeq2::DESeqDataSet` object.

Usage

`isoDE(ids, formula = NULL, ...)`

Arguments

- `ids`: Object of class `IsomirDataSeq`.
- `formula`: Formula used for DE analysis.
- `...`: Options to pass to `isoCounts()` including ref, iso5, iso3, add, subs and seed parameters.
IsomirDataSeq-class

Details

First, this function collapses all isomiRs in different types. Read more at isoCounts() to know the different options available to collapse isomiRs.

After that, DESeq2::DESeq2-package is used to do differential expression analysis. It uses the count matrix and design experiment stored at (counts(ids) and colData(ids)) IsomirDataSeq object to construct a DESeq2::DESeqDataSet object.

Value

DESeq2::DESeqDataSet object. To get the differential expression isomiRs, use DESeq2::results() from DESeq2 package. This allows to ask for different contrast without calling again isoDE(). Read results manual to know how to access all the information.

Examples

data(mirData)
ids <- isoCounts(mirData, minc=10, mins=6)
dds <- isoDE(mirData, formula=~condition)

IsomirDataSeq-class Class that contains all isomiRs annotation for all samples

Description

The IsomirDataSeq is a subclass of SummarizedExperiment, used to store the raw data, intermediate calculations and results of an miRNA/isomiR analysis. This class stores all raw isomiRs data for each sample, processed information, summary for each isomiR type, raw counts, normalized counts, and table with experimental information for each sample.

Details

IsomirDataSeqFromFiles creates this object using seqbuster output files.

Methods for this objects are counts() to get count matrix and isoSelect() for miRNA/isomiR selection. Functions available for this object are isoCounts() for count matrix creation, isoNorm() for normalization, isoDE() for differential expression. isoPlot() helps with basic expression plot.

metadata contains one list:

• rawData is a data.frame with the information of each sequence found in the data and the counts for each sample.

The naming of isomiRs follows these rules:

• miRNA name
• type:ref if the sequence is the same than the miRNA reference. iso if the sequence has variations.
• **iso_5p** tag: indicates variations at 5 position. The naming contains two words: direction - nucleotides, where direction can be UPPER CASE NT (changes upstream of the 5 reference position) or LOWER CASE NT (changes downstream of the 5 reference position). 0 indicates no variation, meaning the 5 position is the same than the reference. After direction, it follows the nucleotide/s that are added (for upstream changes) or deleted (for downstream changes).

• **iso_3p** tag: indicates variations at 3 position. The naming contains two words: direction - nucleotides, where direction can be LOWER CASE NT (upstream of the 3 reference position) or UPPER CASE NT (downstream of the 3 reference position). 0 indicates no variation, meaning the 3 position is the same than the reference. After direction, it follows the nucleotide/s that are added (for downstream changes) or deleted (for upstream changes).

• **iso_add** tag: indicates nucleotides additions at 3 position. The naming contains two words: direction - nucleotides, where direction is UPPER CASE NT (upstream of the 5 reference position). 0 indicates no variation, meaning the 3 position has no additions. After direction, it follows the nucleotide/s that are added.

• **iso_snv** tag: indicates nucleotides substitutions along the sequences. The naming contains three words: position-nucleotide@isomiR-nucleotide@reference.

• **iso_snv_seed** tag: same than iso_snv tag, but only if the change happens between nucleotide 2 and 8.

In general nucleotides in UPPER case mean insertions respect to the reference sequence, and nucleotides in LOWER case mean deletions respect to the reference sequence.

**Examples**

```r
path <- system.file("extra", package="isomiRs")
fn_list <- list.files(path, pattern="mirna", full.names = TRUE)
de <- data.frame(row.names=c("f1", "f2"),
                         condition = c("newborn", "newborn"))
ids <- IsomirDataSeqFromFiles(fn_list, coldata=de)
head(counts(ids))
```

---

**IsomirDataSeqFromFiles**

*Loads miRNA annotation from seqbuster tool or pre-processed data.*

---

**Description**

This function parses output of seqbuster tool to allow isomiRs/miRNAs analysis of samples in different groups such as characterization, differential expression and clustering. It creates an IsomirDataSeq object.
Usage

IsomirDataSeqFromFiles(
  files,  
  coldata,  
  rate = 0.2,  
  canonicalAdd = TRUE,  
  uniqueMism = TRUE,  
  uniqueHits = FALSE,  
  design = ~1L,  
  minHits = 1L,  
  header = TRUE,  
  skip = 0,  
  quiet = TRUE,  
  ...
)

Arguments

files files with the output of seqbuster tool
coldata data frame containing groups for each sample
rate minimum counts fraction to consider a mismatch a real mutation
canonicalAdd boolean only keep A/T non-template addition. All non-template nucleotides at the 3’ end will be removed if they contain C/G nts.
uniqueMism boolean only keep mutations that have a unique hit to one miRNA molecule. For instance, if the sequence map to two different miRNAs, then it would be removed.
uniqueHits boolean whether filtering ambiguous sequences or not.
design a formula to pass to DESeq2::DESeqDataSet
minHits Minimum number of reads in the sample to consider it in the final matrix.
header boolean to indicate files contain headers
skip skip first line when reading files
quiet boolean indicating to print messages while reading files. Default FALSE.
... arguments provided to SummarizedExperiment and IsomirDataSeqFromRawData. including rowData.

Details

This function parses the output of http://seqcluster.readthedocs.org/mirna_annotation.html for each sample to create a count matrix for isomiRs, miRNAs or isomiRs grouped in types (i.e all sequences with variations at 5’ but ignoring any other type). It creates IsomirDataSeq object (see link to example usage of this class) to allow visualization, queries, differential expression analysis and clustering. To create the IsomirDataSeq, it parses the isomiRs files, and generates an initial matrix having all isomiRs detected among samples. As well, it creates a summary for each isomiR type (trimming, addition and substitution) to visualize general isomiRs distribution.
IsomirDataSeqFromMirtop

Value

IsomirDataSeq class object.

Examples

path <- system.file("extra", package="isomiRs")
fn_list <- list.files(path, pattern="mirna", full.names = TRUE)
de <- data.frame(row.names=c("f1", "f2"),
  condition = c("newborn", "newborn"))
ids <- IsomirDataSeqFromFiles(fn_list, coldata=de)

head(counts(ids))
IsomirDataSeqFromRawData(metadata(ids)[["rawData"]], de)
**Examples**

```r
library(readr)
path <- system.file("extra", "mirtop", package="isomiRs")
fn <- list.files(path, full.names = TRUE)
de <- data.frame(row.names=c("sample1", "sample2"),
  condition = c("cc", "cc"))
# mirtop export --format isomir ....
IsomirDataSeqFromMirtop(read_tsv(fn), de)
```

**IsomirDataSeqFromRawData**

*Loads miRNA annotation from seqbuster tool or pre-processed data.*

**Description**

Process raw data like tables to speed up filtering steps.

**Usage**

```r
IsomirDataSeqFromRawData(
  rawdata,
  coldata,
  design = ~1L,
  pct = 0.1,
  n_snv = 1,
  whitelist = NULL,
  ...
)
```

**Arguments**

- `rawdata` data.frame stored in metadata slot of `IsomirDataSeq` object.
- `coldata` data frame containing groups for each sample.
- `design` a formula to pass to `DESeq2::DESeqDataSet`.
- `pct` numeric used to remove isomiRs with an importance lower than this value. Importance is calculated by dividing the isomiR count by the total counts of the miRNA to which it maps.
- `n_snv` numeric used to remove isomiRs with more than this number of single nucleotide variants (indels are counted here).
- `whitelist` character vector with sequences to keep even if the filtering step would have removed them. They have to match the `seq` column in the table.
- `...` arguments provided to `SummarizedExperiment` including rowData.

**Value**

`IsomirDataSeq` class object.
isoNetwork

Clustering miRNAs-genes pairs in similar pattern expression

Description

Clustering miRNAs-genes pairs

Usage

isoNetwork(
  mirna_rse,
  gene_rse,
  summarize = NULL,
  target = NULL,
  org = NULL,
  enrich = NULL,
  genename = "ENSEMBL",
  min_cor = -0.6,
  min_fc = 0.5
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mirna_rse</td>
<td>SummarizedExperiment with miRNA information. See details.</td>
</tr>
<tr>
<td>gene_rse</td>
<td>SummarizedExperiment with gene information. See details.</td>
</tr>
<tr>
<td>summarize</td>
<td>Character column name in colData(rse) to use to group samples and compare</td>
</tr>
<tr>
<td></td>
<td>between miRNA/gene expression.</td>
</tr>
<tr>
<td>target</td>
<td>Matrix with miRNAs (columns) and genes (rows) target prediction (1 if it is</td>
</tr>
<tr>
<td></td>
<td>a target, 0 if not).</td>
</tr>
<tr>
<td>org</td>
<td>AnnotationDb object. For example:(org.Mm.eg.db)</td>
</tr>
<tr>
<td>enrich</td>
<td>The output of clusterProfiler of similar functions.</td>
</tr>
<tr>
<td>genename</td>
<td>Character keytype of the gene names in gene_rse object.</td>
</tr>
<tr>
<td>min_cor</td>
<td>Numeric cutoff to consider a miRNA to regulate a target.</td>
</tr>
<tr>
<td>min_fc</td>
<td>Numeric cutoff to consider as the minimum log2FoldChange between groups to</td>
</tr>
<tr>
<td></td>
<td>be considered in the analysis.</td>
</tr>
</tbody>
</table>
**Details**

This function will correlate miRNA and gene expression data using a specific metadata variable to group samples and detect pattern of expression that will be annotated with GO terms. `mirna_rse` and `gene_rse` can be created using the following code:

```r
mi_rse = SummarizedExperiment(assays=SimpleList(norm=mirna_matrix), colData, metadata=list(sign=mirna_keep))
```

where, `mirna_matrix` is the normalized counts expression, `colData` is the metadata information and `mirna_keep` the list of miRNAs to be used by this function.

**Value**

list with network information

**Examples**

```r
# library(org.Mm.eg.db)
# library(clusterProfiler)
data(isoExample)
# ego <- enrichGO(row.names(assay(gene_ex_rse, "norm")),
# org.Mm.eg.db, "ENSEMBL", ont = "BP")
data <- isoNetwork(mirna_ex_rse, gene_ex_rse,
                   summarize = "group", target = ma_ex,
                   enrich = ego)
isoPlotNet(data, minGenes = 5)
```

---

### isoNorm

**Normalize count matrix**

**Description**

This function normalizes raw count matrix using `DESeq2::rlog()` function from `DESeq2::DESeq2-package`.

**Usage**

```r
isoNorm(ids, formula = NULL, maxSamples = 50)
```

**Arguments**

- `ids` Object of class `IsomirDataSeq`.
- `formula` Formula that will be used for normalization.
- `maxSamples` Maximum number of samples to use with `DESeq2::rlog()`, if not `limma::voom()` is used.

**Value**

`IsomirDataSeq` object with the normalized count matrix in a slot. The normalized matrix can be access with `counts(ids, norm=TRUE)`.
Examples

data(mirData)
ids <- isoCounts(mirData, minc=10, mins=6)
ids <- isoNorm(mirData, formula=~condition)
head(counts(ids, norm=TRUE))

isoPlot

Plot the amount of isomiRs in different samples

Description

This function plot different isomiRs proportion for each sample. It can show trimming events at both side, additions and nucleotides changes.

Usage

isoPlot(ids, type = "iso5", column = NULL, use = NULL, nts = FALSE)

Arguments

ids Object of class IsomirDataSeq.
type String (iso5, iso3, add, snv, all) to indicate what isomiRs to use for the plot. See details for explanation.
column String indicating the column in colData to color samples.
use Character vector to only use these isomiRs for the plot. The id used is the rownames that comes from using isoCounts with all the arguments on TRUE.
nts Boolean to indicate whether plot positions of nucleotides changes when showing single nucleotides variants.

Details

There are four different values for type parameter. To plot trimming at 5’ or 3’ end, use type="iso5" or type="iso3". Get a summary of all using type="all". In this case, it will plot 3 positions at both side of the reference position described at miRBase site. Each position refers to the % of sequences that start/end before or after the miRBase reference. The color indicates the sample group. The size of the point is proportional to the abundance considering the total as all the sequences in the sample. The position at y is the % of different sequences considering the total as all sequences with changes for the specific isomiR showed.

Same logic applies to type="add" and type="subs". However, when type="add", the plot will refer to addition events from the 3’ end of the reference position. Note that this additions don’t match to the precursor sequence, they are non-template additions. In this case, only 3 positions after the 3’ end will appear in the plot. When type="subs", it will appear one position for each nucleotide in the reference miRNA. Points will indicate isomiRs with nucleotide changes at the given position. When type="all" a color coordinate map will show the abundance of each isomiR type in a single plot. Note the position is relatively to the sequence not the miRNA.
**isoPlotNet**

**Value**

`ggplot2::ggplot()` Object showing different isomiRs changes at different positions.

**Examples**

```r
data(mirData)
isoPlot(mirData)
```

---

**isoPlotNet**

*Functional miRNA / gene expression profile plot*

**Description**

Plot analysis from `isoNetwork()`. See that function for an example of the figure.

**Usage**

```r
isoPlotNet(obj, minGenes = 2)
```

**Arguments**

- `obj` Output from `isoNetwork()`.
- `minGenes` Minimum number of genes per term to be kept.

**Value**

Network ggplot.

---

**isoPlotPosition**

*Plot nucleotides changes at a given position*

**Description**

This function plot different isomiRs proportion for each sample at a given position focused on the nucleotide change that happens there.

**Usage**

```r
isoPlotPosition(ids, position = 1L, column = NULL)
```

**Arguments**

- `ids` Object of class `IsomirDataSeq`.
- `position` Integer indicating the position to show.
- `column` String indicating the column in `colData` to color samples.
Details

It shows the nucleotides changes at the given position for each sample in each group. The color indicates the sample group. The size of the point is proportional to the number of total counts of isomiRs with changes. The position at y is the % of different isomiRs supporting the change. Note the position is relatively to the sequence not the miRNA.

Value

`ggplot2::ggplot()` Object showing nucleotide changes at a given position.

Examples

```r
data(mirData)
isoPlotPosition(mirData)
```

---

**isoSelect**

Method to select specific miRNAs from an IsomirDataSeq object.

Description

This method allows to select a miRNA and all its isomiRs from the count matrix.

Usage

```r
isoSelect.IsomirDataSeq(object, mirna, minc = 10)
```

```r
## S4 method for signature 'IsomirDataSeq'
isoSelect(object, mirna, minc = 10)
```

Arguments

- **object** A IsomirDataSeq object.
- **mirna** String referring to the miRNA to show.
- **minc** Minimum number of isomiR reads needed to be included in the table.

Value

`S4Vectors::DataFrame` with count information. The row.names show the isomiR names, and each of the columns shows the counts for this isomiR in that sample. Mainly, it will return the count matrix only for isomiRs belonging to the miRNA family given by the `mirna` parameter. IsomiRs need to have counts bigger than `minc` parameter at least in one sample to be included in the output. Annotation of isomiRs follows these rules:

- miRNA name
- mismatches
- additions
- 5 trimming events
- 3 trimming events
isoTop

Author(s)

Lorena Pantano

Examples

data(mirData)
# To select isomiRs from let-7a-5p miRNA
# and with 10000 reads or more.
isoSelect(mirData, mirna="hsa-let-7a-5p", minc=10000)

isoTop

Heatmap of the top expressed isomiRs

Description

This function creates a heatmap with the top N isomiRs/miRNAs. It uses the matrix under `counts(ids)` to get the top expressed isomiRs/miRNAs using the average expression value and plot a heatmap with the raw counts for each sample.

Usage

isoTop(ids, top = 20)

Arguments

top
Object of class IsomirDataSeq.

Arguments

top
Number of isomiRs/miRNAs used.

Value

PCA of the top expressed miRNAs

Examples

data(mirData)
isoTop(mirData)
ma_ex

Data frame containing gene-mirna relationship

Description
Data frame containing gene-mirna relationship

Usage
ma_ex

Format
A data frame with rows same as gene_ex_rse and columns same as mirna_ex_rse.

mirData

Example of IsomirDataSeq with human brain miRNA counts data

Description
This data set is the object return by IsomirDataSeqFromFiles. It contains miRNA count data from 14 samples: 7 control individuals (pc) and 7 patients with Parkinson’s disease in early stage (Pantano et al, 2016). Use colData to see the experiment design.

Usage
data("mirData")

Format
a IsomirDataSeq class.

Author(s)
Lorena Pantano, 2018-04-27

Source
Data is available from GEO dataset under accession number GSE97285
Every sample was analyzed with seqbuster tool, see http://seqcluster.readthedocs.org/mirna_annotation.html for more details. You can get same files running the small RNA-seq pipeline from https://github.com/bcbio/bcbio-nextgen.
bcbio_nextgen was used for the full analysis.
See raw-data.R to know how to recreate the object. This script is inside "extra" folder of the package.
References


mirna2targetscan

Find targets in targetscan database

Description

From a list of miRNA names, find their targets in targetscan.Hs.eg.db annotation package.

Usage

mirna2targetscan(mirna, species = "hsa", org = NULL, keytype = NULL)

Arguments

mirna Character vector with miRNA names as in miRBase 21.
species hsa or mmu supported right now.
org AnnotationDb object. For example:(org.Mm.eg.db)
keytype Character mentioning the gene id to use. For example, ENSEMBL.

Value

data.frame with 4 columns:

- miRFamily
- Seedmatch
- PCT
- entrezGene

Examples

library(targetscan.Hs.eg.db)
mirna2targetscan(c("hsa-miR-34c-5p"))
mirna_ex_rse  Data frame containing mirna expression data

Description
Data frame containing mirna expression data

Usage
mirna_ex_rse

Format
mirna expression data with 18 samples: example of a time series data

mirTritation  Data frame containing mirna from Argyropoulos’s paper

Description

Usage
mirTritation

Format
mirna expression data in long format. Train and test data to use with isoCorrect

updateIsomirDataSeq  Update IsomirDataSeq object from version < 1.7

Description
In version 1.9 IsomirDataSeq object changed their internal structure to save space and speed up loading and downstream functions.

Usage
updateIsomirDataSeq(object)
Arguments

- object: IsomirDataSeq.

Details

This function will update to the current structure.
Index

* datasets
  dat286.long, 4
  ego, 5
  gene_ex_rse, 6
  ma_ex, 20
  mirna_ex_rse, 22
  mirTritation, 22

base::matrix, 3

counts, 3
counts(), 9
counts,IsomirDataSeq-method (counts), 3
counts,IsomirDataSeq (counts), 3
counts<-,IsomirDataSeq, matrix-method (counts), 3

dat286.long, 4
data.frame, 6, 9, 21
DESeq2::DESeq2-package, 8, 9, 15
DESeq2::DESeqDataSet, 8, 9, 11, 13
DESeq2::results(), 9
DESeq2::rlog(), 15
design, 4
design,IsomirDataSeq-method (design), 4
design<-,IsomirDataSeq, formula-method (design), 4

ego, 5

findTargets, 5

gene_ex_rse, 6
ggplot2::ggplot(), 17, 18

isoAnnotate, 6
isoCounts, 7
isoCounts(), 8, 9
isoDE, 8
isoDE(), 9
IsomirDataSeq, 4, 6–13, 15–20, 22, 23

IsomirDataSeq (IsomirDataSeq-class), 9
IsomirDataSeq-class, 9
IsomirDataSeqFromFiles, 9, 10, 20
IsomirDataSeqFromMirtop, 12
IsomirDataSeqFromRawData, 11, 12, 13
isomiRs (isomiRs-package), 2
isomiRs-package, 2
isoNetwork, 14
isoNetwork(), 17
isoNorm, 15
isoNorm(), 9
isoPlot, 16
isoPlot(), 9
isoPlotNet, 17
isoPlotPosition, 17
isoSelect, 18
isoSelect(), 9
isoSelect,IsomirDataSeq-method (isoSelect), 18
isoSelect.IsomirDataSeq(isoSelect), 18
isoTop, 19
limma::voom(), 15

ma_ex, 20
mirData, 20
mirna2targets, 21
mirna_ex_rse, 22
mirTritation, 22

S4Vectors::DataFrame, 18

updateIsomirDataSeq, 22