

# Package ‘isomiRs’

May 2, 2026

**Version** 1.41.0

**Date** 2025-12-23

**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, pheatmap, BiocStyle, testthat

**Depends** R (>= 4.4), SummarizedExperiment

**Imports** AnnotationDbi, BiocGenerics, Biobase, broom, cluster, cowplot, DEGreport, DESeq2, IRanges, dplyr, GenomicRanges, gplots, ggplot2, gtools, gridExtra, grid, grDevices, graphics, GGally, limma, methods, RColorBrewer, readr, reshape, rlang, stats, stringr, S4Vectors, tidyr, tibble

**LazyData** true

**License** MIT + file LICENSE

**VignetteBuilder** knitr

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.2

**Encoding** UTF-8

**git\_url** <https://git.bioconductor.org/packages/isomiRs>

**git\_branch** devel

**git\_last\_commit** 63a2ae2

**git\_last\_commit\_date** 2026-04-28

**Repository** Bioconductor 3.24

**Date/Publication** 2026-05-01

**Author** Lorena Pantano [aut, cre],  
Georgia Escaramis [aut] (CIBERESP - CIBER Epidemiologia y Salud Publica)

**Maintainer** Lorena Pantano <lorena.pantano@gmail.com>

## Contents

isomiRs-package . . . . .	2
counts . . . . .	3
dat286.long . . . . .	3
design . . . . .	4
ego . . . . .	4
gene_ex_rse . . . . .	5
isoAnnotate . . . . .	5
isoCounts . . . . .	6
isoDE . . . . .	7
IsomirDataSeq-class . . . . .	8
IsomirDataSeqFromFiles . . . . .	9
IsomirDataSeqFromMirtop . . . . .	10
IsomirDataSeqFromRawData . . . . .	11
isoNorm . . . . .	12
isoPlot . . . . .	13
isoPlotPosition . . . . .	14
isoSelect . . . . .	14
isoTop . . . . .	15
ma_ex . . . . .	16
mirData . . . . .	16
mirna_ex_rse . . . . .	17
mirTritation . . . . .	17
updateIsomirDataSeq . . . . .	18
<b>Index</b>	<b>19</b>

---

isomiRs-package	<i>isomiRs</i>
-----------------	----------------

---

## Description

Characterization of miRNAs and isomiRs, clustering and differential expression.

## Author(s)

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

Authors:

- Georgia Escaramis (CIBERESP - CIBER Epidemiologia y Salud Publica)

## See Also

Useful links:

- Report bugs at <https://github.com/lpantano/isomiRs/issues>

---

counts	<i>Accessors for the count matrix of a IsomirDataSeq object.</i>
--------	--

---

### 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	<i>Data frame containing mirna from Argyropoulos's paper</i>
-------------	--

---

### Description

Argyropoulos, Christos, et al. "Modeling bias and variation in the stochastic processes of small RNA sequencing." Nucleic Acids Research (2017).

### Usage

```
dat286.long
```

**Format**

mirna expression data in long format.

---

design	<i>Accessors for the 'design' slot of a IsomirDataSeq object.</i>
--------	---

---

**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 <a href="#">IsomirDataSeq</a> object.
value	A formula to pass to DESeq2.

**Value**

design for the experiment

**Examples**

```
data(mirData)
design(mirData) <- formula(~ 1)
```

---

ego	<i>enrichResult class</i>
-----	---------------------------

---

**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")`

---

gene_ex_rse	<i>Data frame containing gene expression data</i>
-------------	---

---

**Description**

Data frame containing gene expression data

**Usage**

```
gene_ex_rse
```

**Format**

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

---

isoAnnotate	<i>Annotate the rawData of the <a href="#">IsomirDataSeq</a> object</i>
-------------	---

---

**Description**

Get the sequence and the name information for each isomiR, and the importance value (isomir\_reads/mirna\_reads) for each sample.

**Usage**

```
isoAnnotate(ids)
```

**Arguments**

ids                    Object of class [IsomirDataSeq](#).

**Details**

edit\_mature\_position represents the position at the mature sequence + nucleotide at reference + nucleotide at isomiR.

**Value**

[data.frame](#) with the sequence, isomir name, and importance for each sample and isomiR.

**Examples**

```
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**

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

**Arguments**

<code>ids</code>	Object of class <a href="#">IsomirDataSeq</a> .
<code>ref</code>	Differentiate reference miRNA from rest.
<code>iso5</code>	Differentiate trimming at 5 miRNA from rest.
<code>iso3</code>	Differentiate trimming at 3 miRNA from rest.
<code>add</code>	Differentiate additions miRNA from rest.
<code>snv</code>	Differentiate nt substitution miRNA from rest.
<code>seed</code>	Differentiate changes in 2-7 nts from rest.
<code>all</code>	Differentiate all isomiRs.
<code>minc</code>	Int minimum number of isomiR sequences to be included.
<code>mins</code>	Int minimum number of samples with number of sequences bigger than minc counts.
<code>merge_by</code>	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](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

<code>ids</code>	Object of class <a href="#">IsomirDataSeq</a> .
<code>formula</code>	Formula used for DE analysis.
<code>...</code>	Options to pass to <code>isoCounts()</code> including <code>ref</code> , <code>iso5</code> , <code>iso3</code> , <code>add</code> , <code>subs</code> and <code>seed</code> parameters.

### 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 as 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 as 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 as 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 as `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**

```

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 <a href="#">DESeq2::DESeqDataSet</a>

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 <a href="#">IsomirDataSeqFromRawData</a> . including rowData.

### Details

This function parses the output of [http://seqcluster.readthedocs.org/mirna\\_annotation.html](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.

### 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)
```

---

IsomirDataSeqFromMirtop

*Import mirtop output into IsomirDataSeq*

---

### Description

The tabular output of [mirtop](#) is compatible with [IsomirDataSeq](#). This function allows to import the data and filter low confidence isomiRs for downstream analysis.

### Usage

```
IsomirDataSeqFromMirtop(mirtop, coldata, ...)
```

### Arguments

mirtop	data.frame with the output of mirtop export
coldata	data.frame with the metadata of the samples
...	It supports the same parameters as in <a href="#">IsomirDataSeqFromRawData</a> .

**Details**

The output is generated with `mirtop export --format isomir`.

**Value**

[IsomirDataSeq](#) class object.

**Examples**

```
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**

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

**Arguments**

<code>rawdata</code>	data.frame stored in metadata slot of <a href="#">IsomirDataSeq</a> object.
<code>coldata</code>	data frame containing groups for each sample
<code>design</code>	a formula to pass to <a href="#">DESeq2::DESeqDataSet</a>
<code>pct</code>	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.
<code>n_snv</code>	numeric used to remove isomiRs with more than this number of single nucleotide variants (indels are counted here).
<code>whitelist</code>	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.
<code>...</code>	arguments provided to <code>SummarizedExperiment</code> . including <code>rowData</code> .

**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)
```

---

 isoNorm

*Normalize count matrix*


---

**Description**

This function normalizes raw count matrix using [DESeq2::rlog\(\)](#) function from [DESeq2::DESeq2-](#)[package](#).

**Usage**

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

**Arguments**

ids	Object of class <a href="#">IsomirDataSeq</a> .
formula	Formula that will be used for normalization.
maxSamples	Maximum number of samples to use with <a href="#">DESeq2::rlog()</a> , if not <a href="#">limma::voom()</a> 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

<code>ids</code>	Object of class <a href="#">IsomirDataSeq</a> .
<code>type</code>	String (iso5, iso3, add, snv, all) to indicate what isomiRs to use for the plot. See details for explanation.
<code>column</code>	String indicating the column in <code>colData</code> to color samples.
<code>use</code>	Character vector to only use these isomiRs for the plot. The id used is the rownames that comes from using <code>isoCounts</code> with all the arguments on TRUE.
<code>nts</code>	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.

## Value

[ggplot2::ggplot\(\)](#) Object showing different isomiRs changes at different positions.

## Examples

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

---

isoPlotPosition	<i>Plot nucleotides changes at a given position</i>
-----------------	---

---

### Description

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

### Usage

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

### Arguments

ids	Object of class <code>IsomirDataSeq</code> .
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

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

---

isoSelect	<i>Method to select specific miRNAs from an IsomirDataSeq object.</i>
-----------	---

---

### Description

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

### Usage

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

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

**Arguments**

object	A <a href="#">IsomirDataSeq</a> 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

**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, condition = NULL)
```

**Arguments**

ids	Object of class <a href="#">IsomirDataSeq</a> .
top	Number of isomiRs/miRNAs used.
condition	Give condition to color PCA samples

**Value**

PCA of the top expressed miRNAs

**Examples**

```
data(mirData)
isoTop(mirData)
```

---

ma_ex	<i>Data frame containing gene-mirna relationship</i>
-------	--

---

**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	<i>Example of IsomirDataSeq with human brain miRNA counts data</i>
---------	--

---

**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](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**

Pantano L, Friedlander MR, Escaramis G, Lizano E et al. Specific small-RNA signatures in the amygdala at premotor and motor stages of Parkinson's disease revealed by deep sequencing analysis. *Bioinformatics* 2016 Mar 1;32(5):673-81. PMID: 26530722

---

mirna_ex_rse	<i>Data frame containing mirna expression data</i>
--------------	--

---

**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	<i>Data frame containing mirna from Argyropoulos's paper</i>
--------------	--

---

**Description**

Argyropoulos, Christos, et al. "Modeling bias and variation in the stochastic processes of small RNA sequencing." *Nucleic Acids Research* (2017).

**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, 3
  - ego, 4
  - gene\_ex\_rse, 5
  - ma\_ex, 16
  - mirna\_ex\_rse, 17
  - mirTritation, 17
  
- base::matrix, 3
  
- counts, 3
- counts(), 8
- counts, IsomirDataSeq-method (counts), 3
- counts.IsomirDataSeq (counts), 3
- counts<-, IsomirDataSeq,matrix-method (counts), 3
  
- dat286.long, 3
- data.frame, 5, 8
- DESeq2::DESeq2-package, 7, 12
- DESeq2::DESeqDataSet, 7, 9, 11
- DESeq2::results(), 7
- DESeq2::rlog(), 12
- design, 4
- design, IsomirDataSeq-method (design), 4
- design<-, IsomirDataSeq, formula-method (design), 4
  
- ego, 4
  
- gene\_ex\_rse, 5
- ggplot2::ggplot(), 13, 14
  
- isoAnnotate, 5
- isoCounts, 6
- isoCounts(), 7, 8
- isoDE, 7
- isoDE(), 7, 8
- IsomirDataSeq, 4–16, 18
- IsomirDataSeq (IsomirDataSeq-class), 8
- IsomirDataSeq-class, 8
- IsomirDataSeqFromFiles, 8, 9, 16
- IsomirDataSeqFromMirtop, 10
- IsomirDataSeqFromRawData, 10, 11
- isomiRs (isomiRs-package), 2
  
- isomiRs-package, 2
- isoNorm, 12
- isoNorm(), 8
- isoPlot, 13
- isoPlot(), 8
- isoPlotPosition, 14
- isoSelect, 14
- isoSelect(), 8
- isoSelect, IsomirDataSeq-method (isoSelect), 14
- isoSelect.IsomirDataSeq (isoSelect), 14
- isoTop, 15
  
- limma::voom(), 12
  
- ma\_ex, 16
- mirData, 16
- mirna\_ex\_rse, 17
- mirTritation, 17
  
- S4Vectors::DataFrame, 15
  
- updateIsomirDataSeq, 18