Package 'ExiMiR'

February 1, 2026

Type Package

Title R functions for the normalization of Exiqon miRNA array data

Version 2.53.0

Date 2012-11-06

Author Sylvain Gubian <DL.RSupport@pmi.com>, Alain Sewer
<DL.RSupport@pmi.com>, PMP SA

Maintainer Sylvain Gubian <DL.RSupport@pmi.com>

Depends R (>= 2.10), Biobase (>= 2.5.5), affy (>= 1.26.1), limma

Imports affyio(>= 1.13.3), Biobase(>= 2.5.5), preprocessCore(>=
1.10.0)

Suggests mirna10cdf

Description This package contains functions for reading raw data in
ImaGene TXT format obtained from Exiqon miRCURY LNA arrays,
annotating them with appropriate GAL files, and normalizing
them using a spike-in probe-based method. Other platforms and
data formats are also supported.

License GPL-2

Collate make.gal.env.R read.exi.R createAB.R NormiR.R NormiR.methods.R

LazyLoad yes

biocViews Microarray, OneChannel, TwoChannel, Preprocessing,
GeneExpression, Transcription

git_url https://git.bioconductor.org/packages/ExiMiR

git_branch devel

git_last_commit 6d3b680

git_last_commit_date 2025-10-29

Repository Bioconductor 3.23

Date/Publication 2026-02-01

Contents

ExiMiR-package   *R functions for the normalization of Exiqon miRNA array data*

Description

This package contains functions for reading raw data in ImaGene TXT format obtained from Exiqon miRCURY LNA arrays, annotating them with appropriate GAL files, and normalizing them using a spike-in probe-based method. Other platforms and data formats are also supported by ExiMiR.

Details

Package:	ExiMiR
Type:	Package
Version:	1.99.0
Date:	2012-04-24
License:	GPL-2
LazyLoad:	yes

Author(s)

Sylvain Gubian, Alain Sewer, PMP SA
Maintainer: DL.RSupport@pmi.com

---

bg.correct.miR *ExiMiR low-level function for background correction*

Description

This function performs background correction on an `AffyBatch` object.

The methods supported by `bg.correct.miR` are provided by the `affy` or `limma` packages, depending on whether the input `AffyBatch` object has been created with `ReadAffy` or `ReadExi/createAB`, respectively.

Usage

```
bg.correct.miR(abatch,
        bgcorrect.method='auto',
        bgcorrect.param=list(),
        verbose=FALSE)
```

Arguments

abatch   An `AffyBatch` object.

bgcorrect.method

Character vector. It contains the name of the background correction method. Running `NormiR.bgcorrect.methods(abatch)` indicates which methods can be used, depending on the raw data contained in the `abatch` object. The `auto` option corresponds to the default choice of applying `rma` for single-channel arrays and `normexp` for dual-channel arrays.

bgcorrect.param

A R list containing the parameters required by the selected background correction method, as specified in the documentation of the original functions `bg.correct` of the `affy` package or `backgroundCorrect` of the `limma` package. As an illustration the parameters of the `normexp` method of the `limma` package are given below.

normexp.method Character vector. The variant of the `normexp` method, matching exactly the argument `normexp.method` of the `backgroundCorrect` function.

offset Numeric value to add to intensities. It matches exactly the argument `offset` of the `backgroundCorrect` function.

verbose   Logical. The default value is `TRUE`. The details of the function execution are displayed on the console.

Details

See accompanying vignette.

Value

An `AffyBatch` object containing the background-corrected raw expression data.

Author(s)

Sylvain.Gubian, Alain Sewer, PMP SA

See Also

`NormiR.bgcorrect.methods`, `NormiR`.

Examples

```
data(galenv)
data(GSE20122)
NormiR.bgcorrect.methods(GSE20122)
GSE20122.bgcorrected <- bg.correct.miR(GSE20122,
                                       bgcorrect.method='normexp',
                                       bgcorrect.param=list(offset=50))
```

---

createAB	*ExiMiR function for creating an AffyBatch object from other object types (RGList, EListRaw, MAList or list)*

Description

This function creates an `AffyBatch` object from a `limma` object (`RGList`, `EListRaw`, `MAList`) or from any appropriate `list` object.

Usage

```
createAB(object,
         verbose=TRUE,
         ref.channel="R",
         genes.block=NULL,
         genes.row=NULL,
         genes.col=NULL,
         genes.id=NULL,
         genes.name=NULL,
         galname=NULL,
         env.overwrite=TRUE,
         ...)
```

Arguments

`object`	An appropriate `EListRaw`, `RGList`, `MAList` or `list` object.
`verbose`	Logical. The default value is `TRUE`. The details of the function execution are displayed on the console.
`genes.block`	Optional character vector in case the platform is neither ImaGene, Exiqon nor Agilent. The name of the column in the `object$genes` data frame that contains the block or field values.

genes.row	Optional character vector in case the platform is neither ImaGene, Exiqon nor Agilent. The name of the column in the `object$genes` data frame that contains the row values.
genes.col	Optional character vector in case the platform is neither ImaGene, Exiqon nor Agilent. The name of the column in the `object$genes` data frame that contains the column values.
genes.id	Optional character vector in case the platform is neither ImaGene, Exiqon nor Agilent. The name of the column in the `object$genes` data frame that contains the gene IDs.
genes.name	Optional character vector in case the platform is neither ImaGene, Exiqon nor Agilent. The name of the column in the `object$genes` data frame that contains gene names.
ref.channel	Character vector. The value of the reference channel for two-color arrays ('R' or 'G')
galname	Character vector. The default value is `NULL`. In this case the GAL annotation environment used by `createAB` for generating the resulting `AffyBatch` object is lost. Assigning `galname` a non-empty value allows to control this GAL environment, which is useful in two specific situations. First, it gives a handle to this GAL annotation environment for later use. Second, if an adequate GAL annotation environment already exists in the memory (e.g. after having been generated by `createAB` or by `make.gal.env`), `galname` allows to force `createAB` to use it for generating the resulting `AffyBatch` object.
env.overwrite	Logical. The default value is `TRUE`. If a GAL annotation environment with the same name already exists in the memory, it will be overwritten. This may be useful when the piece of code containing `createAB` is run several times.
...	Any additional argument that can be given to the `AffyBatch` constructor, as specified in the documentation of the `AffyBatch` object provided in the `affy` package.

Details

See accompanying vignette.

Value

An `AffyBatch` object containing the raw expression data.

Author(s)

Sylvain Gubian, Alain Sewer, PMP SA

See Also

`ReadExi`, `make.gal.env`.

---

galenv   *R annotation environment for GEO series GSE20122*

Description

The galenv environment is a hash table for the annotation of the Exiqon miRCURY LNA arrays used in the GEO series GSE20122 (Exiqon miRCURY LNA array v.11).

Details

See accompanying vignette.

See Also

make.gal.env.

GSE19183   *Affybatch object for the raw data from GEO series GSE19183*

Description

The Affybatch object GSE19183 contains the raw expression data contained in the CEL files of the GEO series GSE19183, obtained from the Affymetrix miRNA-1_0 platform. The annotation is included in the Affybatch object.

Details

See accompanying vignette.

GSE20122   *Affybatch object for the raw data from GEO series GSE20122*

Description

The Affybatch object GSE20122 contains the raw expression data contained in the ImaGene TXT files of the GEO series GSE20122, obtained from the Exiqon miRCURY LNA platform v.11.

Details

See accompanying vignette.

---

make.gal.env *GAL Environment Maker*

Description

Reads an Exiqon GAL file and creates an annotation environment used as a hash table for the probeset mapping location.

Usage

```
make.gal.env(galname=NULL,
             filename=NULL,
             gal.path=getwd(),
             verbose=FALSE)
```

Arguments

galname   Character vector. Name to be used for the annotation environment.

filename   Character vector. Name of the GAL file.

gal.path   Character vector. Path to the GAL file.

verbose   Logical. The default value is TRUE. The details of the function execution are displayed on the console.

Details

This function is designed similarly to make.cdf.env from the makecdfenv package. If no filename is provided as argument, the function tries to read the first GAL file in the input path. The returned environment is a hash table. For every probeset name we have a matrix with 2 columns. The first column contains the PM locations and the second column the MM locations. For PM only chips the MM column will have NAs.

Value

None.

Author(s)

Sylvain Gubian, Alain Sewer, PMP SA

Examples

```
# The folder 'Exiqon' contains a GAL file
## Not run: make.gal.env(galname='galenv', gal.path='Exiqon')
```

---

norm.miR	*ExiMiR low-level function for miRNA raw data normalization.*

Description

This function performs low-level normalization on an `AffyBatch` object and returns the result in a new `AffyBatch` object.

By default, it applies the spike-in probe-based normalization method. In case the spike-in probe-based method cannot be applied, a median normalization is executed instead. Several options allow however to force the execution of the spike-in probe-based normalization and to fine-tune the resulting correction functions.

Usage

```
norm.miR(abatch,
         normalize.method="spikein",
         normalize.param=list(),
         verbose=TRUE,
    ...)
```

Arguments

abatch   An `AffyBatch` object.

normalize.method

Character vector. It contains the name of normalization method. By default, the `spikein` method is used. Running `NormiR.normalize.methods(abatch)` indicates which other methods can be chosen, depending on the raw data contained in the `abatch` object.

normalize.param

A R list of the arguments that are used to control the `spikein` normalization. Running `NormiR.spikein.args()` provides a complete list of all the tunable parameters supported by `norm.miR` and explained below.

figures.output Character vector. By default, `display` is used. Figures are shown to the screen. Using `file` generates the figures in PDF format in the working directory.

min.corr Numeric. Default value is 0.5. Minimal allowed value for the average of the off-diagonal elements of the Pearson correlation matrix of the spike-in probeset intensities across the arrays.

loess.span Numeric. Default value is -1, which corresponds to a loess smoothing neighbourhood spanning a fraction 3/(number of spike-in probesets) of the total number of points. Other positive values are allowed, see the `span` argument of the R `loess` function

extrap.points Numeric. Default value is 2. The number of spike-in probesets used in the high-intensity extrapolation of the normalization correction function.

> **extrap.method** Character vector. Default value is mean. The method used for the high-intensity extrapolation of the normalization correction function.
>
> **force.zero** Logical. Default value is FALSE. If TRUE, it forces the normalization correction functions to have zero values at the lower end of the probe intensity range.
>
> **cover.ext** Numeric. Default value is 1/2. Minimal allowed relative coverage of the spike-in probesets intensities. It is computed as the ratio between the intensity range covered by the spike-in probes and the one covered by all probes on the array.
>
> **cover.int** Numeric. Default value is 1/3. Maximal allowed relative intensity interval between two consecutive spike-in probesets. It is computed as the largest intensity difference between two consecutive spike-in probesets divided by the overall probe intensity range.
>
> **verbose** Logical. Default is TRUE; some details are provided on the console.
>
> **max.log2span** Numeric. Default value is 1. Gives the maximal (log2) intensity interval allowed for the probes belonging to one spike-in probeset.
>
> **probeset.list** Vector of probes names that will be used as the "spike-in probes". By default, norm.miR uses the probes annotated as "spike-in" by Exiqon or Affymetrix.

verbose   Logical. The default value is TRUE. The details of the function execution are displayed on the console.

...   Any additional argument. Used for backward compatibility.

Details

See accompanying vignette.

Value

An AffyBatch object containing the normalized (but not summarized) expression data.

Author(s)

Sylvain.Gubian, Alain.Sewer, PMP SA

See Also

NormiR.normalize.methods, NormiR.spikein.args, NormiR.

Examples

```
data(galenv)
data(GSE20122)
GSE20122.normalized <- norm.miR(GSE20122,
                                normalize.param=list(figures.show=FALSE))
# Apply the affy method hist on the generated AffyBatch object GSE20122.normalized
layout(matrix(c(1,2), 1, 2, byrow = TRUE))
hist(GSE20122)
hist(GSE20122.normalized)
```

```
layout(1)
```

NormiR   *ExiMiR high-level function for miRNA raw data normalization.*

Description

This function applies a standard raw data normalization pipeline (i.e. background correction, normalization, PM correction if needed, and summarization) on the input `AffyBatch` object and returns the result in an `ExpressionSet` object.

The methods supported by `NormiR` for the background correction are provided by the `affy` or `limma` packages, depending on whether the input `AffyBatch` object has been created with `ReadAffy` or `ReadExi/createAB`, respectively.

By default, it applies the spike-in probe-based method for the second step of normalization. In case the spike-in probe-based method cannot be applied, a median normalization is executed instead. Several options allow however to force the execution of the spike-in probe-based normalization and to fine-tune the resulting correction functions.

The next step of PM correction is enabled only when numerical values are available for the MM probes of the input `AffyBatch` object. In this case the methods proposed by `NormiR` are provided by the `affy` package.

The methods supported by `NormiR` for the last step of summarization are also provided by the `affy` package. They do not depend on how the input `AffyBatch` object has been created.

Usage

```
NormiR(abatch,
        # background correction
        bg.correct=TRUE,
        bgcorrect.method='auto',
        bgcorrect.param=list(),
        # normalize
        normalize=TRUE,
        normalize.method='spikein',
        normalize.param=list(),
        # pm correction (enabled only when MM-values are available)
        pmcorrect.method='pmonly',
        pmcorrect.param=list(),
        # expression values
        summary.method='medianpolish',
        summary.param=list(),
        summary.subset=NULL,
        # misc.
        verbose=FALSE,
        ...)
```

Arguments

abatch   An `AffyBatch` object.

bg.correct   Logical. Default is `TRUE`: the background correction step will be performed.

bgcorrect.method

                Character vector. It contains the name of the background correction method. Running `NormiR.bgcorrect.methods(abatch)` indicates which methods can be used, depending on the raw data contained in the `abatch` object. The `auto` option corresponds to the default choice of applying `rma` for single-channel arrays and `normexp` for dual-channel arrays.

bgcorrect.param

                A R list containing the parameters required by the selected background correction method, as specified in the documentation of the original functions `bg.correct` of the `affy` package or `backgroundCorrect` of the `limma` package. As an illustration the parameters of the `normexp` method of the `limma` package are given below.

                normexp.method Character vector. The variant of the `normexp` method, matching exactly the argument `normexp.method` of the `backgroundCorrect` function.

                offset Numeric value to add to intensities. It matches exactly the argument `offset` of the `backgroundCorrect` function.

normalize   Logical. Default is `TRUE`: the normalization step will be performed.

normalize.method

                Character vector. It contains the name of normalization method. By default, the `spikein` method is used. Running `NormiR.normalize.methods(abatch)` indicates which other methods can be chosen, depending on the raw data contained in the `abatch` object.

normalize.param

                A R list of the arguments that are used to control the `spikein` normalization. Running `NormiR.spikein.args()` provides a complete list of all the tunable parameters supported by `NormiR` and explained below.

                figures.output Character vector. By default, `display` is used. Figures are shown to the screen. Using `file` generates the figures in PDF format in the working directory.

                min.corr Numeric. Default value is 0.5. Minimal allowed value for the average of the off-diagonal elements of the Pearson correlation matrix of the spike-in probeset intensities across the arrays.

                loess.span Numeric. Default value is -1, which corresponds to a loess smoothing neighbourhood spanning a fraction 3/(number of spike-in probesets) of the total number of points. Other positive values are allowed, see the `span` argument of the R `loess` function

                extrap.points Numeric. Default value is 2. The number of spike-in probesets used in the high-intensity extrapolation of the normalization correction function.

                extrap.method Character vector. Default value is `mean`. The method used for the high-intensity extrapolation of the normalization correction function.

           force.zero Logical. Default value is `FALSE`. If `TRUE`, it forces the normaliza-
                tion correction functions to have zero values at the lower end of the probe
                intensity range.

           cover.ext Numeric. Default value is 1/2. Minimal allowed relative coverage of
                the spike-in probesets intensities. It is computed as the ratio between the
                intensity range covered by the spike-in probes and the one covered by all
                probes on the array.

           cover.int Numeric. Default value is 1/3. Maximal allowed relative intensity
                interval between two consecutive spike-in probesets. It is computed as the
                largest intensity difference between two consecutive spike-in probesets di-
                vided by the overall probe intensity range.

           verbose Logical. Default is `TRUE`; some details are provided on the console.

           max.log2span Numeric. Default value is 1. Gives the maximal (log2) intensity
                interval allowed for the probes belonging to one spike-in probeset.

           probeset.list Vector of probes names that will be used as the "spike-in probes".
                By default, `NormiR` uses the probes annotated as "spike-in" by Exiqon or
                Affymetrix.

`pmcorrect.method`

        Character vector. It contains the name of the PM correction method, which is
        enabled only when numerical values are available for the MM probes of the input
        `AffyBatch` object. Running `NormiR.pmcorrect.methods(abatch)` indicates
        which other methods can be chosen instead of the default one `pmonly`.

`pmcorrect.param`

        A R list of optional parameters for the selected `pmcorrect.method`, as specified
        in the documentation of the original function `pmcorrect` function of the `affy`
        package.

`summary.method` Character vector. It contains the name of the summarization method. Running
        `NormiR.summary.methods()` indicates which other methods can be chosen in-
        stead of the default one `medianpolish`.

`summary.param` A R list of optional parameters for the selected `summary.method`, as specified
        in the documentation of the original `AffyBatch` method `computeExprSet` con-
        tained in the `affy` package.

`summary.subset` A R list of probe set identifiers. When set to its default `NULL` value, the summa-
        rized expression values are computed for all probe sets available on the array.

`verbose`   Logical. The default value is `TRUE`. The details of the function execution are
        displayed on the console.

`...`   Any additional argument. Used for backward compatibility.

Details

See accompanying vignette.

Value

An `ExpressionSet` object containing the normalized expression data.

Author(s)

Sylvain Gubian, Alain Sewer, PMP SA

See Also

bg.correct.miR, NormiR.bgcorrect.methods, norm.miR, NormiR.normalize.methods, NormiR.spikein.args, NormiR.pmcorrect.methods, summarize.miR, NormiR.summary.methods.

Examples

```
data(galenv)
data(GSE20122)
eset.spike <- NormiR(GSE20122,
                     bg.correct=FALSE,
                     normalize.method='spikein',
                     summary.method='medianpolish')
```

NormiR.methods	*ExiMiR functions for enumerating the names of low-level normalization methods or arguments*

Description

These functions enumerate the names of methods or arguments of the low-level functions for miRNA raw data normalization (i.e. background correction, (spike-in probe-based) normalization, PM corrrection, summarization). They take into account how the input AffyBatch object was created as well as the underlying array type.

Usage

```
NormiR.bgcorrect.methods(object)
NormiR.normalize.methods(object)
NormiR.pmcorrect.methods(object)
NormiR.summary.methods()
NormiR.spikein.args()
```

Arguments

object   An AffyBatch object.

Details

See accompanying vignette.

Value

List of strings containing the names of the methods or arguments available for the input AffyBatch object.

Author(s)

Sylvain.Gubian, Alain.Sewer, PMP SA

See Also

`NormiR`, `bg.correct.miR`, `norm.miR`, `summarize.miR`.

ReadExi   *ExiMiR function for reading Exiqon raw data into an AffyBatch object.*

Description

This function reads Exiqon raw data in ImaGene file format and creates an `AffyBatch` object.

Usage

```
ReadExi(txtfile.path=getwd(),
        galname=NULL,
        description=NULL,
 notes='',
 rm.background=FALSE,
 verbose=TRUE)
```

Arguments

txtfile.path	Character vector. It contains the path to the folder containing the `samplesinfo.txt` file and the Exiqon raw data files in ImaGene `txt` format.
galname	Character vector. The default value is `NULL`. In this case the GAL annotation environment used by `ReadExi` for generating the resulting `AffyBatch` object is lost. Assigning `galname` a non-empty value allows to control this GAL environment, which is useful in two specific situations. First, it gives a handle to this GAL annotation environment for later use. Second, if an adequate GAL annotation environment already exists in the memory (e.g. after having been generated by `ReadExi` or by `make.gal.env`), `galname` allows to force `ReadExi` to use it for generating the resulting `AffyBatch` object.
description	Object of class MIAME, as specified in the documentation of the `AffyBatch` object provided in the `affy` package.
notes	Character vector, as specified in the documentation of the `AffyBatch` object provided in the `affy` package.
rm.background	Logical. This option is kept for compatibility reasons but it is not used anymore. See the `NormiR` options for background correction.
verbose	Logical. The default value is `TRUE`. The details of the function execution are displayed on the console.

Details

The Exiqon miRNA raw expression data are normally in ImageGene txt file format and accompanied by a samplesinfo.txt description file. It enumerates the names of the sample files for each channel. Therefore the txtfile.path argument of ReadExi must be a folder that contains the ImageGene and the samplesinfo.txt files. If this is not the case, ReadExi stops.

The galname argument of ReadExi must be the name of a GAL annotation environment created with the make.gal.env or the ReadExi functions. If galname is provided a NULL value, which is the default situation, a minimal GAL annotation environment is created based on the annotation contained in the ImageGene txt files.

Value

An AffyBatch object containing the raw expression data.

Warning

The image method of the AffyBatch object might not work properly when the galname argument of ReadExi has not been assigned.

Author(s)

Sylvain Gubian, Alain Sewer, PMP SA

See Also

make.gal.env, createAB.

Examples

```
# The folder 'Exiqon' contains the file 'samplesinfo.txt' and  the corresponding raw data files in ImaGene format
## Not run: ebatch <- ReadExi(txtfile.path='Exiqon')
# If the GAL environment has already created by the function make.gal.env
## Not run: ebatch <- ReadExi(galenv='galenv', txtfile.path='Exiqon')
```

---

summarize.miR *ExiMiR low-level function for summarization.*

Description

This function performs summarization on an AffyBatch object using a GAL or CDF annotation environment and generates an ExpressionSet object containing the results.

Usage

```
summarize.miR(abatch,
        pmcorrect.method='pmonly',
        pmcorrect.param=list(),
        summary.method='medianpolish',
        summary.param=list(),
        summary.subset=NULL)
```

Arguments

abatch   An `AffyBatch` object.

pmcorrect.method

                Character vector. It contains the name of the PM correction method, which is enabled only when numerical values are available for the MM probes of the input `AffyBatch` object. Running `NormiR.pmcorrect.methods(abatch)` indicates which other methods can be chosen instead of the default one `pmonly`.

pmcorrect.param

                A R list of optional parameters for the selected `pmcorrect.method`, as specified in the documentation of the original function `pmcorrect` function of the `affy` package.

summary.method   Character vector. It contains the name of the summarization method. Running `NormiR.summary.methods()` indicates which other methods can be chosen instead of the default one `medianpolish`.

summary.param   A R list of optional parameters for the selected `summary.method`, as specified in the documentation of the original `AffyBatch` method `computeExprSet` contained in the `affy` package.

summary.subset   A R list of probe set identifiers. When left to its default `NULL` value, the summarized expression values are computed for all probe sets available on the array.

Value

An `ExpressionSet` containing the summarized expression data.

Author(s)

Sylvain.Gubian, Alain Sewer, PMP SA

See Also

`NormiR.pmcorrect.methods`, `NormiR.summary.methods`, `NormiR`.

Examples

```
data(galenv)
data(GSE20122)
eset <- summarize.miR(GSE20122,
                      summary.method="medianpolish")
```

Index