Package ‘crmn’

October 12, 2022

Version 0.0.21

Date 2020-02-09

Author Henning Redestig

Maintainer Henning Redestig <henning.red@gmail.com>

Title CCMN and Other Normalization Methods for Metabolomics Data

Depends R (>= 2.10), pcaMethods (>= 1.56.0), Biobase, methods

Description Implements the Cross-contribution Compensating Multiple standard Normalization (CCMN) method described in Redestig et al. (2009) Analytical Chemistry <doi:10.1021/ac901143w> and other normalization algorithms.

URL https://github.com/hredestig/crmn

License GPL (>= 3)

Collate 'classes.R' 'crmn-package.R' 'misc.R' 'norm.R' 'generic.R'

Repository CRAN

Date/Publication 2020-02-10 21:50:10 UTC

RoxygenNote 7.0.2

NeedsCompilation no

R topics documented:

`analytes` *Accessor for the analytes*

Description

Subset an data set to only contain the analytes.

Usage

```
analytes(object, standards=NULL, ...)
```

Arguments

`object`	an `ExpressionSet`, `matrix` or `data.frame`
`standards`	a logical vector indicating which rows are internal analytes
`...`	not used

Value

subsetted dataset

Author(s)

Henning Redestig

Examples

```
data(mix)
analytes(mix)
analytes(exprs(mix), fData(mix)$tag == 'IS')
```

`analytes_eset` *Accessor for the analytes*

Description

Subset an expression set to remove the internal standards

Usage

```
analytes_eset(object, where = "tag", what = "IS", ...)
```

Arguments

`object`	an `ExpressionSet`
`where`	Column index or name of fData which equals `what` for the ISs (and something else for the analytes)
`what`	What the column `where` does not equal for analytes. Can be vector values too.
`...`	not used

Value

`ExpressionSet`

Author(s)

Henning Redestig

Examples

```
data(mix)
analytes(mix)
fData(mix)$test <- fData(mix)$tag
analytes(mix, where="test")
```

`analytes_other` *Accessor for the analytes*

Description

Subset an expression set to remove the internal standards

Usage

```
analytes_other(object, standards, ...)
```

Arguments

`object`	an `ExpressionSet`
`standards`	a logical vector indicating which rows are internal standards
`...`	not used

Value

`ExpressionSet`

Author(s)

Henning Redestig

Examples

```
data(mix)
analytes(exprs(mix), fData(mix)$tag == 'IS')
```

crmn *CRMN*

Description

Normalize metabolomics data using CCMN and other methods

Details

Package:	crmn
Type:	Package
Developed since:	2009-05-14
Depends:	Biobase, pcaMethods (>= 1.20.2), pls, methods
License:	GPL (>=3)
LazyLoad:	yes

A package implementing the 'Cross-contribution compensating multiple standard normalization' described in Redestig et al. (2009) Analytical Chemistry, https://doi.org/10.1021/ac901143w. Can be used to normalize metabolomics data. Do `openVignette("crmn")` to see the manual.

Author(s)

Henning Redestig

`dropunusedlevels` *Drop unused levels*

Description

Drop unused factor levels in a data frame.

Usage

```
dropunusedlevels(x)
```

Arguments

`x`	the data frame

Author(s)

Henning Redestig

Examples

```
iris[1:10,]$Species
dropunusedlevels(iris[1:10,])$Species
```

`makeX-methods` *Make X*

Description

Construct a design matrix

Usage

```
makeX(object, factors, ...)

## S4 method for signature 'ANY,matrix'
makeX(object, factors, ...)

## S4 method for signature 'ExpressionSet,character'
makeX(object, factors, ...)
```

Arguments

`object`	an `ExpressionSet`
`factors`	column names from the pheno data of `object` or a design matrix
`...`	not used

Details

Make a design matrix from the pheno data slot of an expression set, taking care that factors and numerical are handled properly. No interactions are included and formula is the most simple possible, i.e. `y~-1+term1+term2+...`. Can also be given anything as object in which case `factor` must be a design matrix. It that case the same design matrix is returned.

Value

a design matrix

Author(s)

Henning Redestig

Examples

```
data(mix)
makeX(mix, "runorder")
runorder <- mix$runorder
makeX(mix, model.matrix(~-1+runorder))
```

`method-methods` *Accessor for the method*

Description

Get the method

Usage

```
method(object, ...)

method(object, ...)
```

Arguments

`object`	an `nFit` object
`...`	not used

Value

the method (content differs between normlization methods)

Author(s)

Henning Redestig

mexprs-methods *Matrix safe accessor of expression slot*

Description

Get the expression data from an `ExpressionSet` or just return the given matrix

Usage

```
mexprs(object)

mexprs(object)

## S4 method for signature 'ExpressionSet'
mexprs(object)
```

Arguments

`object`	an `ExpressionSet` or `matrix`

Value

the expression data

Author(s)

Henning Redestig

Examples

```
data(mix)
head(mexprs(mix))
head(mexprs(exprs(mix)))
```

mexprs-rep-methods *Accessor*

Description

Matrix safe setter of expression slot

Usage

```
mexprs(object) <- value

## S4 replacement method for signature 'ExpressionSet,matrix'
mexprs(object) <- value

mexprs(object) <- value
```

Arguments

`object`	an `ExpressionSet` or `matrix`
`value`	the value to assign

Details

Set the expression data in an `ExpressionSet` or just return the given matrix

Value

the expression data

Author(s)

Henning Redestig

Examples

```
data(mix)
test <- mix
mexprs(test) <- exprs(mix) * 0
head(mexprs(test))
test <- exprs(mix)
mexprs(test) <- test * 0
head(mexprs(test))
```

`mix` *Dilution mixture dataset.*

Description

Mixture dilution series

Usage

```
data(mix)
```

Details

Multi-component dilution series. GC-TOF/MS measurements by Miyako Kusano. Input concentrations are known and given in the original publication.

Author(s)

Henning Redestig

Examples

```
data(mix)
fData(mix)
exprs(mix)
pData(mix)
```

model-methods *Accessor for the model*

Description

Get the model

Usage

```
model(object, ...)

model(object, ...)
```

Arguments

`object`	an `nFit` object
`...`	not used

Value

the model (content differs between normlization models)

Author(s)

Henning Redestig

nFit *Normalization model*

Description

Common class representation for normalization models.

Author(s)

Henning Redestig

normalize *Normalize a metabolomics dataset*

Description

Normalization methods for metabolomics data

Usage

```
normalize(object, method, segments = NULL, ...)
```

Arguments

`object`	an `ExpressionSet`
`method`	the desired method
`segments`	normalization in a cross-validation setup, only to use for validation/QC purposes.
`...`	passed on to `normFit` and `normPred`

Details

Wrapper function for `normFit` and `normPred`

Value

the normalized dataset

Author(s)

Henning Redestig

See Also

`normFit`, `normPred`

Examples

```
data(mix)
normalize(mix, "crmn", factor="type", ncomp=3)
#other methods
normalize(mix, "one")
normalize(mix, "avg")
normalize(mix, "nomis")
normalize(mix, "t1")
normalize(mix, "ri")
normalize(mix, "median")
normalize(mix, "totL2")
## can also do normalization with matrices
Y <- exprs(mix)
G <- with(pData(mix), model.matrix(~-1+type))
isIS <- with(fData(mix), tag == "IS")
normalize(Y, "crmn", factor=G, ncomp=3, standards=isIS)
```

normFit *Fit a normalization model*

Description

Fit the parameters for normalization of a metabolomics data set.

Usage

```
normFit(
  object,
  method,
  one = "Succinate_d4",
  factors = NULL,
  lg = TRUE,
  fitfunc = lm,
  formula = TRUE,
  ...
)
```

Arguments

object	an ExpressionSet or a matrix (with samples as columns) in which case the standards must be passed on via ...
method	chosen normalization method
one	single internal standard to use for normalization
factors	column names in the pheno data slot describing the biological factors. Or a design matrix directly.
lg	logical indicating that the data should be log transformed

fitfunc the function that creates the model fit for normalization, must use the same interfaces as lm.

formula if fitfunc has formula interface or not

... passed on to standardsFit, standards, analytes

Details

Normalization is first done by fitting a model and then applying that model either to new data or the same data using normPred. Five different methods are implemented.

t1 divide by row-means of the L_2 scaled internal standards

one divide by value of a single, user defined, internal standard

totL2 divide by the square of sums of the full dataset

nomis See Sysi-Aho et al.

crmn See Redestig et al.

Value

a normalization model

Author(s)

Henning Redestig

References

Sysi-Aho, M.; Katajamaa, M.; Yetukuri, L. & Oresic, M. Normalization method for metabolomics data using optimal selection of multiple internal standards. BMC Bioinformatics, 2007, 8, 93

Redestig, H.; Fukushima, A.; Stenlund, H.; Moritz, T.; Arita, M.; Saito, K. & Kusano, M. Compensation for systematic cross-contribution improves normalization of mass spectrometry based metabolomics data Anal Chem, 2009, 81, 7974-7980

See Also

normPred, standards, model.matrix

Examples

```
data(mix)
nfit <- normFit(mix, "crmn", factors="type", ncomp=3)
slplot(sFit(nfit)$fit$pc, scol=as.integer(mix$runorder))
## same thing
Y <- exprs(mix)
G <- model.matrix(~-1+mix$type)
isIS <- fData(mix)$tag == 'IS'
nfit <- normFit(Y, "crmn", factors=G, ncomp=3, standards=isIS)
slplot(sFit(nfit)$fit$pc, scol=as.integer(mix$runorder))
```

`normPred` *Predict for normalization*

Description

Predict the normalized data using a previously fitted normalization model.

Usage

```
normPred(normObj, newdata, factors = NULL, lg = TRUE, predfunc = predict, ...)
```

Arguments

`normObj` the result from `normFit`

`newdata` an `ExpressionSet` or a `matrix` (in which case the `standards` must be passed on via `...`), possibly the same as used to fit the normalization model in order to get the fitted data.

`factors` column names in the pheno data slot describing the biological factors. Or a design matrix.

`lg` logical indicating that the data should be log transformed

`predfunc` the function to use to get predicted values from the fitted object (only for crmn)

`...` passed on to `standardsPred`, `standardsFit`, odestandards, `analytes`

Details

Apply fitted normalization parameters to new data to get normalized data. Current can not only handle matrices as input for methods 'RI' and 'one'.

Value

the normalized data

Author(s)

Henning Redestig

See Also

`normFit`

Examples

```
data(mix)
nfit <- normFit(mix, "crmn", factor="type", ncomp=3)
normedData <- normPred(nfit, mix, "type")
slplot(pca(t(log2(exprs(normedData)))), scol=as.integer(mix$type))
## same thing
Y <- exprs(mix)
G <- with(pData(mix), model.matrix(~-1+type))
isIS <- fData(mix)$tag == 'IS'
nfit <- normFit(Y, "crmn", factors=G, ncomp=3, standards=isIS)
normedData <- normPred(nfit, Y, G, standards=isIS)
slplot(pca(t(log2(normedData))), scol=as.integer(mix$type))
```

pcaMuffle *Muffle the pca function*

Description

PCA and Q2 issues warnings about biasedness and poorly estimated PCs. The first is non-informative and the poorly estimated PCs will show up as poor overfitting which leads to a choice of fewer PCs i.e. not a problem. This function is mean to muffle those warnings. Only used for version of pcaMethods before 1.26.0.

Usage

```
pcaMuffle(w)
```

Arguments

`w`	a warning

Value

nothing

Author(s)

Henning Redestig

plot.nFit *Plot a statistics for CRMN normalization model*

Description

Simple plot function for a CRMN normalization model.

Usage

```
## S3 method for class 'nFit'
plot(x, y = NULL, ...)
```

Arguments

x	an nFit object
y	not used
...	passed on to the scatter plot calls

Details

Shows Tz and the optimization (if computed) of the PCA model. The number of components used for normalization should not exceed the maximum indicated by Q2. The structure shown in the Tz plot indicate the analytical variance which is exactly independent of the experimental design. The corresponding loading plot shows how this structure is capture by the used ISs.

Value

nothing

Author(s)

Henning Redestig

See Also

slplot

Examples

```
data(mix)
nfit <- normFit(mix, "crmn", factors="type", ncomp=2)
plot(nfit)
```

`sFit-method` *Accessor for the standards model*

Description

Get the sFit

Usage

```
sFit(object, ...)

sFit(object, ...)
```

Arguments

`object`	an `nFit` object
`...`	not used

Value

the sFit is only defined for CRMN

Author(s)

Henning Redestig

`show` *Show method for nFit*

Description

Show some basic information for an nFit model

Usage

```
## S4 method for signature 'nFit'
show(object)
```

Arguments

`object`	the `nFit` object

Value

prints some basic information

Author(s)

Henning Redestig

Examples

```
data(mix)
normFit(mix, "avg")
```

`show_nfit` *Show nfit*

Description

Show method for nFit

Usage

```
show_nfit(object)
```

Arguments

`object`	the `nFit` object

Value

prints some basic information

Author(s)

Henning Redestig

`standards` *Accessor for the Internal Standards*

Description

Subset an data set to only contain the labeled internal standards.

Usage

```
standards(object, standards=NULL, ...)
```

Arguments

`object`	an `ExpressionSet`, `matrix` or `data.frame`
`standards`	a logical vector indicating which rows are internal standards
`...`	not used

Value

subsetted dataset

Author(s)

Henning Redestig

Examples

```
data(mix)
standards(mix)
standards(exprs(mix), fDataMix)$tag == 'IS')
```

`standardsFit` *Standards model*

Description

Fit a model which describes the variation of the labeled internal standards from the biological factors.

Usage

```
standardsFit(object, factors, ncomp = NULL, lg = TRUE, fitfunc = lm, ...)
```

Arguments

`object` an `ExpressionSet` or a `matrix`. Note that if you pass `amatrix` have to specify the identity of the standards by passing the appropriate argument to `standards`.

`factors` the biological factors described in the pheno data slot if `object` is an `ExpressionSet` or a design matrix if `object` is a `matrix`.

`ncomp` number of PCA components to use. Determined by cross-validation if left `NULL`

`lg` logical indicating that the data should be log transformed

`fitfunc` the function that creates the model fit for normalization, must use the same interfaces as `lm`.

`...` passed on to `Q2`, `pca` (if pcaMethods > 1.26.0), `standards` and `analytes`

Details

There is often unwanted variation in among the labeled internal standards which is related to the experimental factors due to overlapping peaks etc. This function fits a model that describes that overlapping variation using a scaled and centered PCA / multiple linear regression model. Scaling is done outside the PCA model.

Value

a list containing the PCA/MLR model, the recommended number of components for that model, the standard deviations and mean values and Q2/R2 for the fit.

Author(s)

Henning Redestig

See Also

makeX, standardsPred

Examples

```
data(mix)
sfit <- standardsFit(mix, "type", ncomp=3)
slplot(sfit$fit$pc)
## same thing
Y <- exprs(mix)
G <- model.matrix(~-1+mix$type)
isIS <- fDataset(mix)$tag == 'IS'
sfit <- standardsFit(Y, G, standards=isIS, ncomp=3)
```

---

standardsPred *Predict effect for new data (or get fitted data)*

Description

Predicted values for the standards

Usage

```
standardsPred(model, newdata, factors, lg = TRUE, ...)
```

Arguments

model	result from standardsFit
newdata	an ExpressionSet or matrix with new data (or the data used to fit the model to get the fitted data)
factors	the biological factors described in the pheno data slot if object is an ExpressionSet or a design matrix if object is a matrix.
lg	logical indicating that the data should be log transformed
...	passed on to standards and analytes

Details

There is often unwanted variation in among the labeled internal standards which is related to the experimental factors due to overlapping peaks etc. This predicts this effect given a model of the overlapping variance. The prediction is given by $\hat{X}_{IS} = X_{IS} - X_{IS}B$

Value

the corrected data

Author(s)

Henning Redestig

See Also

makeX, standardsFit

Examples

```
data(mix)
fullFit <- standardsFit(mix, "type", ncomp=3)
sfit <- standardsFit(mix[,-1], "type", ncomp=3)
pred <- standardsPred(sfit, mix[,1], "type")
cor(scores(sfit$fit$pc)[1,], scores(fullFit$fit$pc)[1,])
## could just as well have been done as
Y <- exprs(mix)
G <- model.matrix(~-1+mix$type)
isIS <- fData(mix)$tag == 'IS'
fullFit <- standardsFit(Y, G, ncomp=3, standards=isIS)
sfit    <- standardsFit(Y[,-1], G[-1,], ncomp=3,
                        standards=isIS)
pred <- standardsPred(sfit, Y[,1,drop=FALSE], G[1,,drop=FALSE], standards=isIS)
cor(scores(sfit$fit$pc)[1,], scores(fullFit$fit$pc)[1,])
```

---

standards_eset *Accessor for the Internal Standards*

Description

Subset an data set to only contain the labeled internal standards.

Usage

```
standards_eset(object, where = "tag", what = "IS", ...)
```

Arguments

`object`	an `ExpressionSet`
`where`	Column index or name in fData which equals `what` for the ISs
`what`	What the column `where` equals for ISs
`...`	not used

Value

subsetted dataset

Author(s)

Henning Redestig

Examples

```
data(mix)
standards(mix)
fData(mix)$test <- fData(mix)$tag
standards(mix, where="test")
```

`standards_other` *Accessor for the Internal Standards*

Description

Subset an data set to only contain the labeled internal standards.

Usage

```
standards_other(object, standards, ...)
```

Arguments

`object`	an `matrix` or `data.frame`
`standards`	a logical vector indicating which rows are internal standards
`...`	not used

Value

subsetted dataset

Author(s)

Henning Redestig

Examples

```
data(mix)
standards(exprs(mix), fData(mix)$tag == 'IS')
```

weightnorm *Normalize by sample weight*

Description

Normalize samples by their weight (as in grams fresh weight)

Usage

```
weightnorm(object, weight = "weight", lg = FALSE)
```

Arguments

`object`	an `ExpressionSet`
`weight`	a string naming the pheno data column with the weight or a numeric vector with one weight value per sample.
`lg`	is the assay data already on the log-scale or not. If lg, the weight value is also log-transformed and subtraction is used instead of division.

Details

Normalize each sample by dividing by the loaded sample weight. The weight argument is takes from the pheno data (or given as numerical vector with one value per sample). Missing values are not tolerated.

Value

the normalized expression set

Author(s)

Henning Redestig

Examples

```
data(mix)
w <- runif(ncol(mix),1, 1.3)
weightnorm(mix, w)
```

Index