Package ‘ABSSeq’

April 25, 2017

Type Package

Title ABSSeq: a new RNA-Seq analysis method based on modelling absolute expression differences

Version 1.22.0

Author Wentao Yang

Maintainer Wentao Yang <wyang@zoologie.uni-kiel.de>

Description Inferring differential expression genes by absolute counts difference between two groups, utilizing Negative binomial distribution and moderating fold-change according to heterogeneity of dispersion across expression level.

License GPL (>= 3)

biocViews DifferentialExpression

Imports locfit, limma

Depends R (>= 2.10), methods

Suggests

NeedsCompilation no

R topics documented:

ABSDataSet ................................................................. 2
ABSSeq ................................................................. 3
callDEs ................................................................. 4
callParameter .......................................................... 5
callParameterwithoutReplicates ................................... 6
counts ................................................................. 7
estimateSizeFactorsForMatrix ....................................... 7
excounts ............................................................... 8
genAFold ............................................................... 9
groups ................................................................. 10
LevelstoNormFC ......................................................... 10
maxRates .............................................................. 11
minimalDispersion .................................................... 12
minRates .............................................................. 13
normalFactors ......................................................... 13
normMethod .......................................................... 14
paired ................................................................. 15
Description

ABSDataSet object and constructors

Usage

ABSDataSet(counts, groups, normMethod = c("user", "qtotal", "total", "quartile", "geometric"), sizeFactor = 0, paired = FALSE, minDispersion = NULL, minRates = 0.1, maxRates = 0.3, LevelstoNormFC = 100)

Arguments

counts: a matrix or table with at least two columns and one row.
groups: a factor with two groups, whose length should be equal with sample size
normMethod: method for estimating the size factors, should be one of 'user', 'total', 'quartile' and 'geometric'. See normalFactors for description.
sizeFactor: size factors for 'user' method, self-defined size factors by user.
paired: switch for differential expression detection in paired samples.
minDispersion: a positive double for user-defined penalty of dispersion estimation
minRates: low bounder rate of baseline estimation for counts difference, default is 0.1
maxRates: up bounder rate of baseline estimation for counts difference, default is 0.3. Setting minRates equal with maxRates will result in a testing on user-define rate,
LevelstoNormFC: maximal level of average standard deviation in fold-change normalization according to expression level, default is 100.

Details

The function contracts an ABSDataSet object with counts table and groups. It also checks the structure of counts and groups. The ABSDataSet is a class, used to store the input values, intermediate calculations and results of an analysis of differential expression. It also contains information for the running time of an analysis.

Value

An ABSDataSet object.
ABSSeq

Examples

```
counts <- matrix(1:4,ncol=2)
groups <- factor(c("a","b"))
obj <- ABSDataSet(counts, groups)
obj <- ABSDataSet(counts, groups, paired=TRUE)
```

Description

This function performs a default analysis by calling, in order, the functions: normalFactors, callParameter, callDEs.

Usage

```
ABSSeq(object, adjmethod = "BH", replaceOutliers = TRUE, useaFold = FALSE,
        quiet = FALSE, ...)
```

Arguments

- **object**: an ABSDataSet object, contains the reads count matrix, groups and normalization method.
- **adjmethod**: default is 'BH', method for p-value adjusted, see p.adjust.methods for details
- **replaceOutliers**: default is TRUE, switch for outlier replacement.
- **useaFold**: default is FALSE, switch for DE detection through fold-change, see callDEs for details
- **quiet**: default is FALSE, whether to print messages at each step
- **...**: parameters passed to ReplaceOutliersByMAD from callParameter

Details

The differential expression analysis models the total counts difference by a Negative binomial distribution

\[
NB(\mu, r)
\]

Value

An ABSDataSet object with additional elements, which can be retrieved by results: Amean and Bmean, mean of log2 normalized reads count for group A and B, foldChange, shrinked (expression level and gene-specific) log2 of fold-change, B - A, rawFC, raw log2 of fold-change, B - A (without shrinkage), lowFC, expression level corrected log2 fold-change, pvalue, adj.pvalue from NB distribution model, adj.pvalue, adjusted p-value used p.adjust method.

Author(s)

Wentao Yang
callDEs

Testing the differential expression by counts difference

Description

Using NB distribution to calculate p-value for each gene as well as adjust p-value

Usage

```r
callDEs(object, adjmethod = "BH", useaFold = FALSE)
```

Arguments

- **object**: an `ABSDataSet` object.
- **adjmethod**: the method for adjusting p-value, default is ‘BH’. For details, see `p.adjust.methods`.
- **useaFold**: switch for DE detection through fold-change, which will use a normal distribution (N(0,sd)) to test the significance of log2 fold-change. The sd is estimated through a quantile function of gamma distribution at `callParameter`.

Details

This function firstly calls p-value used `pbinom` to call p-value based on sum of counts difference between two groups or used `pnorm` to call p-value via log2 fold-change, then adjusts the p-values via `p.adjust` method. In addition, it also shrink the log2 fold-change towards a common dispersion after pvalue calling.

Value

- an `ABSDataSet` object with additional elements: shrinked log2 fold-change, pvalue and adjusted p-value, denoted by foldChange pvalue and adj-pvalue, respectively. Use the `results` method to get access it.

Note

- this function should run after `callParameter`
**callParameter**

Calculate parameters for differential expression test based on absolute counts differences

**Description**

Calculate parameters for each gene (the moderating basemean, dispersions, moderated fold-change and general sd)

**Usage**

```r
callParameter(object, replaceOutliers = TRUE, qforGeneralSD = 0.75, ...)
```

**Arguments**

- `object` a `ABSDataSet` object.
- `replaceOutliers` switch for outlier replacement, default is TRUE.
- `qforGeneralSD` quantile for estimating general SD of fold-change, default is 0.75.
- `...` parameters past to `ReplaceOutliersByMAD`

**Details**

shifted and calculate a set of parameters from normalized counts table before `callDEs`

**Value**

A `ABSDataSet` object with absolute differences, basemean, mean of each group, variance, log2 of foldchange, named as 'absD', 'baseMean', 'Amean', 'Bmean', 'Variance' and 'foldChange', respectively. Use the `results` to get access it and `plotDifftoBase` to plot it.

**Note**

This function should run after `normalFactors` or providing size factors.

**Examples**

```r
data(simuN5)
obj <- ABSDataSet(counts=simuN5$counts, groups=factor(simuN5$groups))
obj <- normalFactors(obj)
obj <- callParameter(obj)
obj <- callDEs(obj)
head(results(obj))
plotDifftoBase(obj)
```
callParameterwithoutReplicates

Calculate parameters for differential expression test based on absolute counts differences without replicates

Description

Calculate parameters for each gene (the moderating basemean and dispersions), without replicates

Usage

callParameterwithoutReplicates(object)

Arguments

object a ABSDataSet object.

Details

Building a pseudo group to estimate parameters by mean difference. Shifted and calculate a set of parameters from normalized counts table before callDEs

Value

A ABSDataSet object with absolute differences, basemean, mean of each group, variance, log2 of foldchange, named as 'absD', 'baseMean', 'Amean', 'Bmean', 'Variance' and 'foldChange', respectively. Use the results to get access it

Note

This function should run after normalFactors or providing size factors. This function firstly constructs an expression level depended fold-change cutoffs and then separate the data into two groups. The group with fold-change less than cutoffs is used to training the dispersion. However, the cutoff might be too small when applied on data set without or with less DEs. To avoid it, we set a prior value (0.5) to it.

Examples

data(simuN5)
obj <- ABSDataSet(counts=(simuN5$counts)[,c(1,2)], groups=factor(c(1,2)))
obj <- normalFactors(obj)
obj <- callParameterwithoutReplicates(obj)
obj <- callDEs(obj)
head(results(obj))
Accessors for the 'counts' slot of a ABSDataSet object.

Description
Accessors for the 'counts' slot of a ABSDataSet object, return a matrix.

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

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

Arguments
object a ABSDataSet object.
norm logical indicating whether or not to normalize the counts before returning
value an numeric matrix

Details
The counts slot holds the count data as a matrix of non-negative integer count values, rows and columns for genes and samples, respectively.

See Also
sFactors, normalFactors

Examples
data(simuN5)
obj <- ABSDataSet(counts=simuN5$counts, groups=factor(simuN5$groups))
head(counts(obj))
counts(obj) <- matrix(1:50, nrow=5, ncol=10)
head(counts(obj))

estimateSizeFactorsForMatrix

Description
Low-level function to estimate size factors with robust regression.

Usage
estimateSizeFactorsForMatrix(counts, locfunc = median)
Arguments

- `counts`: a matrix or data frame of counts, i.e., non-negative integer values
- `locfunc`: a function to compute a location for a sample. By default, the median is used.

Details

Given a matrix or data frame of count data, this function estimates the size factors as follows: Each column is divided by the geometric means of the rows. The median (or, if requested, another location estimator) of these ratios (skipping the genes with a geometric mean of zero) is used as the size factor for this column. Typically, you will not call this function directly.

Value

da vector with the estimates size factors, one element per column

Author(s)

Simon Anders

References


Examples

data(simuN5)
dat <- simuN5
estimateSizeFactorsForMatrix(dat$counts)

excounts

Accessors for the 'excounts' slot of a ABSDataSet object.

Description

Accessors for the 'excounts' slot of a ABSDataSet object, return a matrix

Usage

## S4 replacement method for signature 'ABSDataSet,matrix'
excounts(object) <- value

Arguments

- `object`: a ABSDataSet object.
- `value`: an numeric matrix

Details

The excounts slot holds the nomarlized (trimmed or not) count data as a matrix of non-negative integer count values, rows and columns for genes and samples, respectively.
genAFold

See Also
ABSDataSet, ReplaceOutliersByMAD

Examples

data(simuN5)
obj <- ABSDataSet(counts=simuN5$counts, groups=factor(simuN5$groups))
obj <- normalFactors(obj)
obj <- ReplaceOutliersByMAD(obj)
head(excounts(obj))

---

genAFold

*Calculate parameters for differential expression test based on absolute counts differences*

Description

Calculate aFold for each gene and general sd

Usage

genAFold(nncounts, cond, preval = 0.05, qforGeneralSD = 0.75)

Arguments

- **nncounts** matrix for read count.
- **cond** factor for condition.
- **preval** pre-defined uncertainty level as control to penalize absolute count difference, default is 0.05.
- **qforGeneralSD** quantile for estimating general SD of fold-change, default is 0.75.

Details

shifted and calculate a set of parameters from normalized counts table before callDEs

Value

A list with log2 foldchange and general SD for calculating pvalue

Note

This function should run after normalFactors.

Examples

data(simuN5)
obj <- ABSDataSet(counts=simuN5$counts, groups=factor(simuN5$groups))
mtx <- counts(obj,TRUE)
aFold <- genAFold(mtx,factor(simuN5$groups))
hist(aFold[[1]])
groups

Accessors for the ‘groups’ slot of a ABSDataSet object.

Description

Accessor functions for the ‘groups’ information in a ABSDataSet object.

Usage

## S4 method for signature 'ABSDataSet'
groups(object)

## S4 replacement method for signature 'ABSDataSet,factor'
groups(object) <- value

Arguments

object

an ABSDataSet object.

value

a factor object, includes two groups, equal with the number of samples

Details

The ‘groups’ is a factor object, contains the experiment design for differential expression analysis. Its length should be equal with the sample size.

Examples

data(simuN5)
obj <- ABSDataSet(counts=simuN5$counts, groups=factor(simuN5$groups))
groups(obj)
groups(obj) <- factor(rep(c("A","B"),c(5,5)))
groups(obj)

LevelstoNormFC

Accessors for the ‘LevelstoNormFC’ slot of a ABSDataSet object.

Description

Accessor functions for the ‘LevelstoNormFC’ slot of a ABSDataSet object.

Usage

## S4 method for signature 'ABSDataSet'
LevelstoNormFC(object)

## S4 replacement method for signature 'ABSDataSet,numeric'
LevelstoNormFC(object) <- value
Arguments

<table>
<thead>
<tr>
<th>object</th>
<th>an ABSDataSet object.</th>
</tr>
</thead>
<tbody>
<tr>
<td>value</td>
<td>a positive numeric object</td>
</tr>
</tbody>
</table>

Details

The 'LevelstoNormFC' is maximal level of average standard deviation in fold-change normalization according to expression level.

See Also

ABSDataSet, callParameter

Examples

data(simuN5)
obj <- ABSDataSet(counts=simuN5$counts, groups=factor(simuN5$groups))
LevelstoNormFC(obj)
LevelstoNormFC(obj) <- 200
LevelstoNormFC(obj)

maxRates

Accessors for the 'maxRates' slot of a ABSDataSet object.

Description

Accessor functions for the 'maxRates' slot of a ABSDataSet object.

Usage

### S4 method for signature 'ABSDataSet'
maxRates(object)

### S4 replacement method for signature 'ABSDataSet,numeric'
maxRates(object) <- value

Arguments

<table>
<thead>
<tr>
<th>object</th>
<th>an ABSDataSet object.</th>
</tr>
</thead>
<tbody>
<tr>
<td>value</td>
<td>a positive numeric object</td>
</tr>
</tbody>
</table>

Details

The 'maxRates' is the upper bound of rate for baseline of counts difference estimation.

See Also

callParameter, ABSDataSet
Examples

data(simuN5)
obj <- ABSDataSet(counts=simuN5$counts, groups=factor(simuN5$groups))
maxRates(obj)
maxRates(obj) <- 0.4
maxRates(obj)

minimalDispersion

Accessors for the ’minDispersion’ slot of a ABSDataSet object.

Description

Accessor functions for the ’minDispersion’ slot of a ABSDataSet object.

Usage

## S4 method for signature 'ABSDataSet'
minimalDispersion(object)

## S4 replacement method for signature 'ABSDataSet,numeric'
minimalDispersion(object) <- value

Arguments

object      an ABSDataSet object.
value       a positive numeric object

Details

The ’minimalDispersion’ is the penalty of dispersion estimation. User can set the penalty of dispersion by this function

See Also

callParameter,ABSDataSet

Examples

data(simuN5)
obj <- ABSDataSet(counts=simuN5$counts, groups=factor(simuN5$groups))
minimalDispersion(obj)
minimalDispersion(obj) <- 0.2
minimalDispersion(obj)
**minRates**

*Accessors for the `minRates` slot of a ABSDataSet object.*

**Description**

Accessor functions for the `minRates` slot of a ABSDataSet object.

**Usage**

```r
## S4 method for signature 'ABSDataSet'
minRates(object)

## S4 replacement method for signature 'ABSDataSet,numeric'
minRates(object) <- value
```

**Arguments**

- `object` an ABSDataSet object.
- `value` a positive numeric object

**Details**

The `minRates` is the lower bound of rate for baseline of counts difference estimation.

**See Also**

callParameter.ABSDataset

**Examples**

```r
data(simuN5)
obj <- ABSDataSet(counts=simuN5$counts, groups=factor(simuN5$groups))
minRates(obj)
minRates(obj) <- 0.3
minRates(obj)
```

---

**normalFactors**

*Estimating size factors from the reads count table*

**Description**

Function for estimating size factors

**Usage**

`normalFactors(object)`

**Arguments**

- `object` a ABSSeq object with element of `counts` and `normMethod`, see the constructor functions `ABSDataset`.

---
normMethod

Details

Given a matrix of count data, this function estimates the size factors by selected method. It also provides four different methods for normalizing according to user-defined size factors, total reads, up quantile (75th Value)

a ABSDataset object with the estimates size factors, one element per column. Use the sFactors to show it.

Examples

data(simuN5)
obj <- ABSDataset(counts=simuN5$counts, groups=factor(simuN5$groups))
obj <- normalFactors(obj)
sFactors(obj)

attr(## Accessors for the 'normMethod' slot of a ABSDataset object.

Description

Accessor functions for the 'normMethod' information in a ABSDataset object.

Usage

## S4 method for signature 'ABSDataset'

normMethod(object)

## S4 replacement method for signature 'ABSDataset,character'

normMethod(object) <- value

Arguments

object an ABSDataset object.
value a character object, should be one of 'user', 'qtoatl', 'total', 'quartile' and 'geometric'. See normalFactors

Details

The 'normMethod' is the method for calculating the size factors. Currently, Four methods: 'user', 'qtoatl', 'total', 'quartile' and 'DESeq' are available.

Examples

data(simuN5)
obj <- ABSDataset(counts=simuN5$counts, groups=factor(simuN5$groups))
normMethod(obj)
normMethod(obj) <- "geometric"
normMethod(obj)
Accessors for the ’paired’ slot of a ABSDataSet object.

Description

Accessors for the ’paired’ slot of a ABSDataSet object, return a logical value.

Usage

```r
## S4 method for signature 'ABSDataSet'
paired(object)
```

```r
## S4 replacement method for signature 'ABSDataSet,logical'
paired(object) <- value
```

Arguments

- `object`: an ABSDataSet object.
- `value`: value a boolean object, should be either TRUE or FALSE.

Details

The ’paired’ is the switch for differential expression detection among paired samples, with a boolean value: TRUE or FALSE (default). When ”paired” is TRUE, the replicates in each group should be equal.

Examples

```r
data(simuN5)
obj <- ABSDataSet(counts=simuN5$counts, groups=factor(simuN5$groups))
paired(obj)
paired(obj) <- TRUE
```

plotDifftoBase

Plot absolute log2 fold-change against base mean of expression

Description

Plot absolute differences against expression levels

Usage

```r
plotDifftoBase(object, foldname = "foldChange", adj.pcut = 0.05,
cols = c("black", "red"), pch = 16, xlab = "log2 of Expression level",
ylab = "log2 fold-change", ...)
```
ReplaceOutliersByMAD

ReplaceOutliersByMAD

Replacing outliers by moderated MAD

Function for replacing the outliers by MAD

Usage

ReplaceOutliersByMAD(object, replaceOutlier = TRUE, cutoff = 2,
baseMean = 100, limitMad = 0.707, spriors = 2, Caseon = TRUE)

Arguments

object a ABSSeq object with element of 'counts' and 'normMethod', see the constructor functions ABSDataset.
replaceOutlier switch for replacing, default is TRUE.
cutoff cutoff of moderating MAD for outliers, default is 2
baseMean parameter for limiting the trimming at low expression level by baseMean/(sample size), default is 100.
limitMad the minimal prior for moderating MAD, default is set to 0.707, which is usually the highest standard deviation at expression level of 1
spriors prior weight size for prior MAD, default is 2
Caseon switch for dealing with outlier trimming at sample size of 2
Given a matrix of count data, this function replacing the outliers by MAD. Noticeably, this function also provides part of parameters for DEs calling. It is called by callParameter.

a ABSDDataSet object with normalized counts after trimming (replaceOutlier=TRUE) or not (replaceOutlier=FALSE). Use the excounts to show it. Use results with name ‘trimmed’ to view the trimming status.

```
data(simuN5)
obj <- ABSDDataSet(counts=simuN5$counts, groups=factor(simuN5$groups))
obj <- normalFactors(obj)
obj <- ReplaceOutliersByMAD(obj)
head(excounts(obj))
head(results(obj,c("trimmed")))
```

This function returns the result of ABSSeq as a table or a vector depended on the given names, see ABSSeq.

Arguments

- **object**: a ABSDDataSet
- **cnames**: a vector of names for output, which are among: 'Amean','Bmean', log2 of mean counts for group A and B, 'baseMean', estimated mean for absolute counts difference (absD), used for mu in \texttt{pnbinom} 'absD', absolute counts difference in total 'Variance', pooled Variance for two groups 'rawFC','lowFC', 'foldChange', log2 fold-change of original (Bmean-Amean), corrected by expression level and corrected by both expression level and gene-specific dispersion 'pvalue','adj.pvalue', pvalue and adjusted pvalue 'trimmed', number of trimmed outliers.

Details

This function returns the result of ABSSeq as a table or a vector depended on the given names, see ABSSeq.
sFactors

Value

A table according to cannes.

See Also

ABSSeq

Examples

data(simuN5)
obj <- ABSDataSet(counts=simuN5$counts, groups=factor(simuN5$groups))
obj <- normalFactors(obj)
obj <- callParameter(obj)
obj <- callDEs(obj)
head(results(obj))

sFactors

Accessors for the 'sizeFactor' slot of a ABSDataSet object.

Description

Accessor functions for the 'sizeFactor' slot of a ABSDataSet object.

Usage

## S4 method for signature 'ABSDataSet'
sFactors(object)

## S4 replacement method for signature 'ABSDataSet,numeric'
sFactors(object) <- value

Arguments

object an ABSDataSet object.
value a numeric object, one for each sample

Details

The sizeFactors vector assigns to each sample a value, used to normalize the counts in each sample according to selected normMethod.

See Also

normalFactors

Examples

data(simuN5)
obj <- ABSDataSet(counts=simuN5$counts, groups=factor(simuN5$groups))
obj <- normalFactors(obj)
sFactors(obj)
sFactors(obj) <- runif(10,1,2)
sFactors(obj)
**simuN5**

*Simulated study with random outliers*

**Description**

Simulated study with random outliers, include five samples for two groups. It contains counts table, groups and defined differential expression genes.

**Usage**

data(simuN5)

**Format**

The format is: List of 3

- $ counts: integer, reads count matrix
- $ groups: two groups
- $ DEs : differential expression genes

**Details**

Multiple each gene with a value from 5-10 by chance at p-value of 0.05.

**Source**

http://bcf.isb-sib.ch/data/compcodeR/

**References**


**Examples**

data(simuN5)
Index

«Topic datasets
  simuN5, 19
  [[-, SumInfo, character, missing-method (ABSDataSet), 2

  ABSDataSet, 2, 3–6, 9, 11–13, 16
  ABSDataSet-class (ABSDataSet), 2

  ABSSeq, 3, 17, 18

  callDEs, 3, 4, 5, 6, 9
  callParameter, 3, 4, 5, 11–13, 17
  callParameterwithoutReplicates, 6
  counts, 7
  counts<-,ABSDataSet-method (counts), 7
  counts<-,ABSDataSet,matrix-method (counts), 7

  estimateSizeFactorsForMatrix, 7
  excounts, 8, 17
  excounts<-,ABSDataSet-method (excounts), 8
  excounts<-,ABSDataSet,matrix-method (excounts), 8

  genAFold, 9
  groups, 10
  groups<-,ABSDataSet-method (groups), 10
  groups<-,ABSDataSet,factor-method (groups), 10

  LevelstoNormFC, 10
  LevelstoNormFC<-,ABSDataSet-method (LevelstoNormFC), 10
  LevelstoNormFC<-,ABSDataSet,numeric-method (LevelstoNormFC), 10

  maxRates, 11
  maxRates<-,ABSDataSet-method (maxRates), 11
  maxRates<-,ABSDataSet,numeric-method (maxRates), 11

  minimalDispersion, 12
  minimalDispersion<-,ABSDataSet-method (minimalDispersion), 12
  minimalDispersion<-,ABSDataSet,numeric-method (minimalDispersion), 12

  minRates, 13
  minRates<-,ABSDataSet-method (minRates), 13
  minRates<-,ABSDataSet,numeric-method (minRates), 13

  normalFactors, 2, 3, 5–7, 9, 13, 14, 18
  normMethod, 14
  normMethod<-,ABSDataSet-method (normMethod), 14
  normMethod<-,ABSDataSet,character-method (normMethod), 14

  p.adjust, 4
  p.adjust.methods, 3, 4
  paired, 15
  paired<-,ABSDataSet-method (paired), 15
  paired<-,ABSDataSet,logical-method (paired), 15
  plotDifftoBase, 5, 15
  pnbinom, 4, 17
  pnorm, 4

  ReplaceOutliersByMAD, 3, 5, 9, 16
  results, 3–6, 17, 17
  results<-,ABSDataSet-method (results), 17
  sFactors, 7, 14, 18
  sFactors<-,ABSDataSet-method (sFactors), 18
  sFactors<-,sFactors,numeric-method (sFactors), 18

  simuN5, 19
  SumInfo-class (ABSDataSet), 2