Package ‘ABSSeq’

January 14, 2017

Type Package

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

Version 1.10.0

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
groups ......................................................... 9
LevelstoNormFC .................................................. 10
maxRates ....................................................... 10
minimalDispersion ............................................... 11
minRates ....................................................... 12
normalFactors .................................................. 13
normMethod .................................................... 13
plotDifftoBase .................................................. 14
ReplaceOutliersByMAD ......................................... 15
Description
ABSDataSet object and constructors

Usage

ABSDataSet(counts, groups, normMethod = c("user", "total", "quartile", "geometric"), sizeFactor = 0, 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.
- `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 constructs 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.

Examples

```r
counts <- matrix(1:4, ncol=2)
groups <- factor(c("a","b"))
obj <- ABSDataSet(counts, groups)
```
Differential expression analysis based on the total counts difference.

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

Usage
ABSSeq(object, adjmethod = "BH", replaceOutliers = TRUE, 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.
- 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, pvalue from NB distribution model, adj.pvalue, adjusted p-value used p.adjust method.

Author(s)
Wentao Yang

References
Wentao Yang, Philip Rosenstiel & Hinrich Schulenburg: ABSSeq: a new RNA-Seq analysis method based on modelling absolute expression differences
**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")
```

**Arguments**

- `object`: an `ABSDataSet` object.
- `adjmethod`: the method for adjusting p-value, default is ‘BH’. For details, see `p.adjust.methods`.

**Details**

This function firstly calls p-value used `pnbinom` to call pvalue based on sum of counts difference between two groups, then adjusts the pvalues 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: shrunk 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`

**Examples**

```r
data(simuN5)
obj <- ABSDataSet(counts=simuN5$counts, groups=factor(simuN5$groups))
obj <- ABSSeq(obj)
res <- results(obj,c("Amean","Bmean","foldChange","pvalue","adj.pvalue"))
head(res)
```
callParameter

Calculate parameters for differential expression test based on absolute counts differences

Description

Calculate parameters for each gene (the moderating base mean and dispersions)

Usage

callParameter(object, replaceOutliers = TRUE, ...)

Arguments

- object: a ABSDataSet object.
- replaceOutliers: switch for outlier replacement, default is TRUE.
- ...: 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, base mean, mean of each group, variance, log2 of fold change, 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

data(simuN5)
obj <- ABSDataSet(counts=simuN5$counts, groups=factor(simuN5$groups))
obj <- normalFactors(obj)
obj <- callParameter(obj)
head(results(obj,c("foldChange","absD","baseMean")))
plotDiffToBase(obj)
callParameterwithoutReplicates

Calculate parameters for differential expression test base 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

buliding a pseudo group to esitimate parameter 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 con-
structs 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))
**counts**

**Accessors for the 'counts' slot of a ABSDataSet object.**

**Description**

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

**Usage**

```r
## 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**

```r
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**

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

**Description**

This function is borrowed from DESeq.

**Usage**

```r
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

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

Author(s)

Simon Anders

References

Simon Anders, Wolfgang Huber: Differential expression analysis for sequence count data. Genome

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.
groups

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))

---

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**

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

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

**Arguments**

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

**Details**

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

**See Also**

ABSDataSet, callParameter

**Examples**

```r
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**

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

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

Arguments

object an ABSDataSet object.
value a positive numeric object

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

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

## 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)
```
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.

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 (75

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)

normMethod  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
plotDiffToBase

Arguments

- `object` an ABSDataSet object.
- `value` a character object, should be one of 'user', 'total', 'quartile' and 'geometric'.

See `normalFactors`

Details

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

Examples

```r
data(simuN5)
obj <- ABSDataSet(counts=simuN5$counts, groups=factor(simuN5$groups))
normMethod(obj)
normMethod(obj) <- "geometric"
normMethod(obj)
```

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", ...)
```

Arguments

- `object` a ABSDataSet
- `foldname` indicates kind of fold-change in plotting, default is 'foldChange', see results
- `adj.pcut` cutoff for differential expressed genes, marked by different color, default is 0.05
- `cols` the colors to mark the non-DE and DE genes, default is black and red, respectively
- `pch` pch, default is 16
- `xlab` xlab, default is 'log2 of Expression level'
- `ylab` ylab, default is 'log2 fold-change'

Details

Plot absolute differences against expression levels and mark the gene with a color at a given cutoff of fold-change
ReplaceOutliersByMAD

Replacing outliers by moderated MAD

Description
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

Details
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`

Value
a ABSDataSet 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.

Examples
```
data(simuN5)
obj <- ABSDataSet(counts=simuN5$counts, groups=factor(simuN5$groups))
obj <- ABSSeq(obj)
plotDifftoBase(obj)

obj <- normalFactors(obj)
obj <- ReplaceOutliersByMAD(obj)
head(excounts(obj))
head(results(obj,c("trimmed")))
```
results  
Accessor functions for the result from an ABSDataSet

Description

Accessor functions for the result from an ABSDataSet by given names

Usage

```r
## S4 method for signature 'ABSDataSet'
results(object, cnames = c("Amean", "Bmean", "baseMean", "absD", "Variance", "rawFC", "lowFC", "foldChange", "pvalue", "adj.pvalue", "trimmed"))
```

Arguments

- `object`: an ABSDataSet
- `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 `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`

Value

a table according to canmes.

See Also

- `ABSSeq`

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))
```
sFactors

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

Description

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

Usage

```r
## 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

```r
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

```r
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 pvalue of 0.05.

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

References

Examples
data(simuN5)
Index

«Topic datasets

simuN5, 17
[[<-,SumInfo,character,missing-method (ABSDataSet), 2

ABSDataSet, 2, 3–6, 9–13, 15
ABSDataSet-class (ABSDataSet), 2
ABSSeq, 3, 16
callDEs, 3, 4, 5, 6
callParameter, 3, 4, 5, 10–12, 15
callParameterwithoutReplicates, 6
counts, 7
counts<-,ABSDataSet-method (counts), 7
counts<-,ABSDataSet,numeric-method (counts), 7
estimateSizeFactorsForMatrix, 7
excounts, 8, 15
excounts<-,ABSDataSet-method (excounts), 8
excounts<-,ABSDataSet,numeric-method (excounts), 8
groups, 9
groups<-,ABSDataSet-method (groups), 9
groups<-,ABSDataSet,numeric-method (groups), 9
LevelstoNormFC, 10
LevelstoNormFC<-,ABSDataSet-method (LevelstoNormFC), 10
LevelstoNormFC<-,ABSDataSet,numeric-method (LevelstoNormFC), 10
maxRates, 10
maxRates<-,ABSDataSet-method (maxRates), 10
maxRates<-,ABSDataSet,numeric-method (maxRates), 10
minimalDispersion, 11
minimalDispersion<-,ABSDataSet-method (minimalDispersion), 11
minimalDispersion<-,ABSDataSet,numeric-method (minimalDispersion), 11
minRates, 12
minRates<-,ABSDataSet-method (minRates), 12
minRates<-(minRates), 12
normMethod, 13
normMethod<-,ABSDataSet-method (normMethod), 13
normMethod<-,ABSDataSet,numeric-method (normMethod), 13
p.adjust, 4
p.adjust.methods, 3, 4
plotDifftoBase, 5, 14
pnbinom, 4, 16
ReplaceOutliersByMAD, 3, 5, 9, 15
results, 3–6, 15, 16
results<-,ABSDataSet-method (results), 16
sFactors, 7, 13, 17
sFactors<-,ABSDataSet-method (sFactors), 17
sFactors<-,ABSDataSet,numeric-method (sFactors), 17
simuN5, 17
SumInfo-class (ABSDataSet), 2