Package ‘npGSEA’

March 14, 2024

Type Package
Title Permutation approximation methods for gene set enrichment analysis (non-permutation GSEA)
Version 1.38.0
Date 2015-5-6
Author Jessica Larson and Art Owen
Maintainer Jessica Larson <larson.jess@gmail.com>
Imports Biobase, methods, BiocGenerics, graphics, stats
Suggests ALL, genefilter, limma, hgu95av2.db, ReportingTools, BiocStyle
Depends GSEABase (>= 1.24.0)

Description Current gene set enrichment methods rely upon permutations for inference. These approaches are computationally expensive and have minimum achievable p-values based on the number of permutations, not on the actual observed statistics. We have derived three parametric approximations to the permutation distributions of two gene set enrichment test statistics. We are able to reduce the computational burden and granularity issues of permutation testing with our method, which is implemented in this package. npGSEA calculates gene set enrichment statistics and p-values without the computational cost of permutations. It is applicable in settings where one or many gene sets are of interest. There are also built-in plotting functions to help users visualize results.

License Artistic-2.0
biocViews GeneSetEnrichment, Microarray, StatisticalMethod, Pathways
’npgSEA.R’ ‘AllClasses.R’ ‘AllGenerics.R’
’npgGSEAResultBeta-accessors.R’ ‘npgGSEAResultChiSq-accessors.R’
’npgGSEAResultNorm-accessors.R’ ‘npgGSEAPlot-methods.R’
’show-methods.R’ ‘pValues-methods.R’
git_url https://git.bioconductor.org/packages/npGSEA
R topics documented:

alphaValue-methods .......................................................... 2
betaHats-methods .................................................................. 3
betaStat-methods ................................................................. 4
betaValue-methods ............................................................... 5
chiSqStat-methods ............................................................... 6
DF-methods ........................................................................... 7
geneSetName-methods ......................................................... 8
getIncidence .......................................................................... 9
npGSEA ................................................................. 10
npGSEAPlot-methods ............................................................ 11
npGSEAResultBeta-class ....................................................... 12
npGSEAResultBetaCollection-class ....................................... 13
npGSEAResultChiSq-class ....................................................... 14
npGSEAResultChiSqCollection-class ..................................... 15
npGSEAResultNorm-class ....................................................... 16
npGSEAResultNormCollection-class ...................................... 17
pLeft-methods ................................................................. 17
pRight-methods ............................................................... 18
pTwoSided-methods ............................................................ 19
pValues-methods .............................................................. 20
sigmaSq-methods .............................................................. 21
stat-methods ................................................................. 22
summary-methods ............................................................ 23
xSet-methods ................................................................. 24
zStat-methods ............................................................... 25

Index 27

alphaValue-methods ~~ Methods for Function alphaValue ~~

Description

This function returns the corresponding alpha value for the reference beta distribution for the npGSEA analysis in the gene set in the given experiment. This method is applicable for only the beta approximation method.
Usage
alphaValue(object)

Arguments
object An object of type npGSEAResultBeta or npGSEAResultBetaCollection

Methods
signature(object = "npGSEAResultBeta") Returns the value for alpha for a npGSEAResultBeta object
signature(object = "npGSEAResultBetaCollection") Returns a list of the alpha values for a npGSEAResultBetaCollection objects (1 for each set)

Author(s)
Jessica L. Larson

See Also
npGSEAResultBeta-class

Examples
set.seed(15)
yFactor <- as.factor( c(rep("treated", 5), rep("control", 5)) )
xData <- matrix(data=rnorm(length(letters)*10) ,nrow=length(letters), ncol=10)
rownames(xData) <- letters
geneSetABC15 <- GeneSet(geneIds=letters[1:15], setName="setABC15")
res <- npGSEA(x = xData, y = yFactor, set = geneSetABC15, approx= "beta")
alphaValue(res)

Description
This function returns the betaHats for all genes in the corresponding GeneSet in the given experiment, or a list of such vectors for each set in a GeneSetCollection. This corresponds to each gene’s contribution to the test statistic. This method is applicable for all three approximation methods.

Usage
betaHats(object)
Arguments

**object**  
An object of type `npGSEAResultNorm`, `npGSEAResultBeta`, `npGSEAResultChiSq`, `npGSEAResultNormCollection`, `npGSEAResultBetaCollection`, or `npGSEAResultChiSqCollection`.

Methods

- `signature(object = "npGSEAResultNorm")` Returns the betaHats used for analysis to create a `npGSEAResultNorm` object.
- `signature(object = "npGSEAResultBeta")` Returns the betaHats used for analysis to create a `npGSEAResultBeta` object.
- `signature(object = "npGSEAResultChiSq")` Returns the betaHats used for analysis to create a `npGSEAResultChiSq` object.
- `signature(object = "npGSEAResultNormCollection")` Returns the betaHats used for analysis to create the `npGSEAResultNormCollection` objects (1 for each set).
- `signature(object = "npGSEAResultBetaCollection")` Returns a list of the betaHats used for analysis to create the `npGSEAResultBetaCollection` objects (1 for each set).
- `signature(object = "npGSEAResultChiSqCollection")` Returns a list of the betaHats used for analysis to create the `npGSEAResultChiSqCollection` objects (1 for each set).

See Also

- `npGSEAResultNorm-class`

Examples

```r
set.seed(15)
yFactor <- as.factor(c(rep("treated", 5), rep("control", 5)))
xData <- matrix(data=rnorm(length(letters)*10), nrow=length(letters), ncol=10)
rownames(xData) <- letters
geneSetABC15 <- GeneSet(geneIds=letters[1:15], setName="setABC15")
res <- npGSEA(x = xData, y = yFactor, set = geneSetABC15)
betaHats(res)
```

Description

This function returns the corresponding beta statistic which is compared to the reference beta distribution for the npGSEA analysis in the gene set in the given experiment.

Usage

```r
betaStat(object)
```
Arguments

object An object of type npGSEAResultBeta or npGSEAResultBetaCollection

Methods

signature(object = "npGSEAResultBeta") Returns the beta-statistic for a npGSEAResultBeta object
signature(object = "npGSEAResultBetaCollection") Returns a list of the beta-statistics for a npGSEAResultBetaCollection objects (1 for each set)

Author(s)

Jessica L. Larson

See Also

npGSEAResultBeta-class

Examples

set.seed(15)
yFactor <- as.factor( c(rep("treated", 5), rep("control", 5)) )
xData <- matrix(data=rnorm(length(letters)*10) ,nrow=length(letters), ncol=10)
rownames(xData) <- letters
geneSetABC15 <- GeneSet(geneIds=letters[1:15], setName="setABC15")
res <- npGSEA(x = xData, y = yFactor, set = geneSetABC15, approx = "beta")
betaStat(res)

Description

This function returns the corresponding beta value for the reference beta distribution for the npGSEA analysis in the gene set in the given experiment. This method is applicable for only the beta approximation method.

Usage

betaValue(object)

Arguments

object An object of type npGSEAResultBeta or npGSEAResultBetaCollection
Methods

signature(object = "npGSEAResultBeta") Returns the value for beta for a \texttt{npGSEAResultBeta} object
signature(object = "npGSEAResultBetaCollection") Returns a list of the beta values for a \texttt{npGSEAResultBetaCollection} objects (1 for each set)

Author(s)

Jessica L. Larson

See Also

\texttt{npGSEAResultBeta-class}

Examples

set.seed(15)
yFactor <- as.factor( c(rep("treated", 5), rep("control", 5)) )
xData <- matrix(data=rnorm(length(letters)*10) ,nrow=length(letters), ncol=10)
rownames(xData) <- letters
geneSetABC15 <- GeneSet(geneIds=letters[1:15], setName="setABC15")
res <- npGSEA(x = xData, y = yFactor, set = geneSetABC15, approx = "beta")
betaValue(res)

\texttt{chiSqStat-methods} \hspace{1cm} \texttt{\textasciitilde Methods for Function chiSqStat \textasciitilde}

Description

This function returns the chi-sq statistic (which is compared to a reference Chi-sq distribution) for the chi-sq approximation of \texttt{npGSEA} for a corresponding \texttt{GeneSet} or a list of these statistics for a \texttt{GeneSetCollection}. This method is applicable for only the chi-sq approximation method.

Usage

\texttt{chiSqStat(object)}

Arguments

\texttt{object} An object of type \texttt{npGSEAResultChiSq} or \texttt{npGSEAResultChiSqCollection}

Methods

signature(object = "npGSEAResultChiSq") Returns the chi-sq statistic for a \texttt{npGSEAResultChiSq} object
signature(object = "npGSEAResultChiSqCollection") Returns a list of the chi-sq statistics for a \texttt{npGSEAResultChiSqCollection} objects (1 for each set)
DF-methods

Author(s)
Jessica L. Larson

See Also
npGSEAResultChiSq-class

Examples

```r
set.seed(15)
yFactor <- as.factor(c(rep("treated", 5), rep("control", 5)))
xData <- matrix(data=rnorm(length(letters)*10), nrow=length(letters), ncol=10)
rownames(xData) <- letters
geneSetABC15 <- GeneSet(geneIds=letters[1:15], setName="setABC15")
res <- npGSEA(x = xData, y = yFactor, set = geneSetABC15, approx = "chiSq")
chiSqStat(res)
```

Description
This function returns the degrees of freedom for the chi-sq approximation of a corresponding GeneSet or a list of degrees of freedom for a GeneSetCollection. This method is applicable for only the chi-sq approximation method.

Usage

```r
DF(object)
```

Arguments

```r
object An object of type npGSEAResultChiSq or npGSEAResultChiSqCollection
```

Methods

```r
signature(object = "npGSEAResultChiSq") Returns the degrees of freedom for a npGSEAResultChiSq object
signature(object = "npGSEAResultChiSqCollection") Returns a list of the degrees of freedom for a npGSEAResultChiSqCollection objects (1 for each set)
```

Author(s)
Jessica L. Larson

See Also

npGSEAResultChiSq-class
Examples

```r
set.seed(15)
yFactor <- as.factor( c(rep("treated", 5), rep("control", 5)) )
xData <- matrix(data=rnorm(length(letters)*10) ,nrow=length(letters), ncol=10)
rownames(xData) <- letters
geneSetABC15 <- GeneSet(geneIds=letters[1:15], setName="setABC15")
res <- npGSEA(x = xData, y = yFactor, set = geneSetABC15, approx= "chiSq")
DF(res)
```

---

geneSetName-methods  ~~ Methods for Function geneSetName ~~

Description

This function returns the name of the corresponding GeneSet or a list of names for a GeneSetCollection. This method is applicable for all three approximation methods.

Usage

geneSetName(object)

Arguments

- **object**
  - An object of type npGSEAResultNorm, npGSEAResultBeta, npGSEAResultChiSq, npGSEAResultNormCollection, npGSEAResultBetaCollection, or npGSEAResultChiSqCollection

Methods

- `signature(object = "npGSEAResultNorm")` Returns a the name of the gene set from a `npGSEAResultNorm` object
- `signature(object = "npGSEAResultBeta")` Returns a the name of the gene set from a `npGSEAResultBeta` object
- `signature(object = "npGSEAResultChiSq")` Returns a the name of the gene set from a `npGSEAResultChiSq` object
- `signature(object = "npGSEAResultNormCollection")` Returns a list of the names of the gene sets from a `npGSEAResultNormCollection` objects (1 for each set)
- `signature(object = "npGSEAResultBetaCollection")` Returns a list of the names of the gene sets from a `npGSEAResultBetaCollection` objects (1 for each set)
- `signature(object = "npGSEAResultChiSqCollection")` Returns a list of the names of the gene sets from a `npGSEAResultChiSqCollection` objects (1 for each set)

Author(s)

Jessica L. Larson
getIncidence

**See Also**

npGSEAResultNorm-class

**Examples**

```r
set.seed(15)
yFactor <- as.factor(c(rep("treated", 5), rep("control", 5)))
xData <- matrix(data=rnorm(length(letters)*10), nrow=length(letters), ncol=10)
rownames(xData) <- letters
geneSetABC15 <- GeneSet(geneIds=letters[1:15], setName="setABC15")
res <- npGSEA(x = xData, y = yFactor, set = geneSetABC15)
geneSetName(res)
```

---

**getIncidence**

Determines the incidence of a gene set in a list of genes.

**Description**

Calculates the incidence of a gene set in an experiment

**Usage**

```r
getIncidence(universeIDs, set)
```

**Arguments**

- **universeIDs**: A vector containing the list of possible gene ids in the universe (or experiment).
- **set**: A GeneSet object containing a set of genes of interest

**Details**

getIncidence returns an incidence vector of the location of the genes within a gene set in a list of genes in an experiment and vise-versa.

**Value**

A list of inSet and inExp. inSet is a vector with the same length as universeIDs. Each value of inSet is 1 if the gene is in the set and 0 otherwise. inExp is a vector with the same length as geneIds(set), the number of genes in the set. Each value of inExp is 1 if the gene is in universeIDs and 0 otherwise.

**Author(s)**

Jessica L. Larson and Art Owen

**References**

Examples

geneSetABC15 <- GeneSet(geneIds=letters[1:15], setName="setABC15")
getIncidence(letters, geneSetABC15)

npGSEA
Calculates an approximation of the permutation GSEA statistics and p-values

Description

This function calculates the permutation gene set enrichment analysis test statistic and p-value without actually running the permutation. We account for the covariance among the genes within the set and approximate the corresponding permutation distribution. For more details on the method see Larson and Owen (2014).

Usage

npGSEA(x, y, set, covars = NULL, approx = c("norm", "beta", "chiSq"), w = NULL, epsilonBetaAdj=TRUE, scaleXY=TRUE, uniVarX=TRUE)

Arguments

x A matrix of expression data or an object of type ExpressionSet. The columns of x represent samples in a given experiment. The rows are genes. The names of each row (or featureNames of the eSet) must be of the same type (e.g., entrez ids) as the ids of the gene set.

y A vector containing the treatment for each sample. The length of y must be more than 4 for the "chisq" approximation. Each treatment group must have at least two observations for all approximation methods. There can only be two treatment groups.

covars A vector or matrix containing covariate(s) of interest, optional

set A GeneSet object containing a set of genes of interest or a GeneSetCollection object containing a collection of GeneSets

approx A string of either "norm" (default), "beta" or "chiSq". If "norm", the normal approximation to the non-permutation GSEA is calculated and returned. If "beta", the beta approximation is reported. If "chiSq", the Chi-squared approximation to the permutation GSEA is calculated.

w A vector or list containing the weights of each gene in the set or sets, optional. If w is a list, the number of elements in the list must correspond to the number of gene sets in the collection.

epsilonBetaAdj A boolean indicating whether or to not to use an epsilon adjusted p-value for the Beta approximation. When TRUE, this prevents observed p-values of 0. The default is TRUE.

scaleXY A boolean indicating whether or to not to scale x and y. The default is TRUE.

uniVarX A boolean indicating whether or to not to scale x to have unit variance. The default is TRUE.
Value

An object with the corresponding GSEA results. If approx="norm" an npGSEAResultNorm object is returned. If approx="beta" a npGSEAResultBeta object is returned. If approx="chiSq" a npGSEAResultChiSq object is returned. If set is a GeneSetCollection (i.e., multiple sets of interest), then the corresponding npGSEAResultNormCollection, npGSEAResultBetaCollection, or npGSEAResultChiSqCollection is returned.

Author(s)

Jessica L. Larson and Art Owen

References


Examples

set.seed(15)
yFactor <- as.factor( c(rep("treated", 5), rep("control", 5)) )
xData <- matrix(data=rnorm(length(letters)*10) ,nrow=length(letters), ncol=10)
rownames(xData) <- letters
geneSetABC15 <- GeneSet(geneIds=letters[1:15], setName="setABC15")
res <- npGSEA(x = xData, y = yFactor, set = geneSetABC15)

npGSEAPlot-methods

Methods for Function npGSEAPlot

Description

This function plots the reference distribution and the corresponding scaled statistic (Z, Beta, or Chi-sq) from the npGSEA analysis for a given GeneSet. This method is applicable for all three approximation methods.

Usage

npGSEAPlot(object)

Arguments

object An object of type npGSEAResultNorm, npGSEAResultBeta, or npGSEAResultChiSq
Methods

signature(object = "npGSEAResultNorm") Plots the Z-statistic for a npGSEAResultNorm object and the standard normal distribution

signature(object = "npGSEAResultBeta") Plots the beta statistic for a npGSEAResultBeta object and the corresponding reference beta distribution (with alpha and beta calculated from npGSEA).

signature(object = "npGSEAResultChiSq") Plots the beta statistic for a npGSEAResultChiSq object and the corresponding reference chi-squared distribution (with degrees of freedom calculated from npGSEA).

Author(s)

Jessica L. Larson

See Also

npGSEAResultNorm-class

Examples

```r
set.seed(15)
yFactor <- as.factor(c(rep("treated", 5), rep("control", 5)))
xData <- matrix(data=rnorm(length(letters)*10), nrow=length(letters), ncol=10)
rownames(xData) <- letters
geneSetABC15 <- GeneSet(geneIds=letters[1:15], setName="setABC15")
res <- npGSEA(x = xData, y = yFactor, set = geneSetABC15)
##npGSEAPlot (res)
```

---

npGSEAResultBeta-class

Class "npGSEAResultBeta"

Description

Objects of this class store results from running npGSEA with the beta approximation.

Objects from the Class

Objects can be created by calls of npGSEA.

Slots

geneSetName: Object of class "character", the name of the geneSet
betaStat: Object of class "numeric", the test statistic, scaled to the corresponding beta distribution
ThatGw: Object of class "numeric", the test statistic for the set
Objects of this class store results from running npGSEA with the beta approximation with a Gene-SetCollection. npGSEAResultBetaCollection objects contain a list of npGSEAResultBeta objects (one result for each GeneSet).

Objects from the Class
Objects can be created by calls of npGSEA.

Author(s)
Jessica L. Larson
References

See Also
npGSEAResultNormCollection

Examples

showClass("npGSEAResultBetaCollection")

npGSEAResultChiSq-class

Class "npGSEAResultChiSq"

Description
Objects of this class store results from running npGSEA with the Chi-square approximation.

Objects from the Class
Objects can be created by calls of npGSEA.

Slots
geneSetName: Object of class "character", the name of the geneSet
chiSqStat: Object of class "numeric", the test statistic, scaled to the corresponding chi-sq distribution
ChatGw: Object of class "numeric", the test statistic for the set
sigmaSq: Object of class "numeric", the variance
DF: Object of class "numeric", the degrees of freedom
pTwoSided: Object of class "numeric", the p-value for the two-sided hypothesis
xSet: Object of class "matrix", the centered and scaled x data for this set
betaHats: Object of class "vector", the betaHats for each gene in this set

Author(s)
Jessica L. Larson

References
See Also

npGSEAResultNorm

Examples

showClass("npGSEAResultChiSq")

---

npGSEAResultChiSqCollection-class

Class "npGSEAResultChiSqCollection"

Description

Objects of this class store results from running npGSEA with the Chi-square approximation with a GeneSetCollection. npGSEAResultChiSqCollection objects contain a list of npGSEAResultChiSq objects (one result for each GeneSet).

Objects from the Class

Objects can be created by calls of npGSEA.

Author(s)

Jessica L. Larson

References


See Also

npGSEAResultNormCollection

Examples

showClass("npGSEAResultChiSqCollection")
npGSEAResultNorm-class

Class "npGSEAResultNorm"

Description

Objects of this class store results from running npGSEA with the Gaussian approximation.

Objects from the Class

Objects can be created by calls of npGSEA.

Slots

geneSetName: Object of class "character", the name of the geneSet
zStat: Object of class "numeric", the test statistic, scaled to a standard normal
ThatGw: Object of class "numeric", the test statistic for the set
varThatGw: Object of class "numeric", the variance of ThatGw
pLeft: Object of class "numeric", the p-value for the left-side hypothesis
pRight: Object of class "numeric", the p-value for the right-side hypothesis
pTwoSided: Object of class "numeric", the p-value for the two-sided hypothesis
xSet: Object of class "matrix", the centered and scaled x data for this set
betaHats: Object of class "vector", the betaHats for each gene in this set

Author(s)

Jessica L. Larson

References


See Also

npGSEAResultChiSq

Examples

showClass("npGSEAResultNorm")
Description

Objects of this class store results from running npGSEA with the Gaussian approximation with a GeneSetCollection. npGSEAResultNormCollection objects contain a list of npGSEAResultNorm objects (one result for each GeneSet).

Objects from the Class

Objects can be created by calls of npGSEA.

Author(s)

Jessica L. Larson

References


See Also

npGSEAResultChiSqCollection

Examples

showClass("npGSEAResultNormCollection")

Description

This function returns the left-sided p-value for the corresponding GeneSet or a list of p-values for a GeneSetCollection. This method is only applicable for the normal and beta approximation methods.

Usage

pLeft(object)
Arguments

object An object of type npGSEAResultNorm, npGSEAResultBeta, npGSEAResultNormCollection, or npGSEAResultBetaCollection

Methods

signature(object = "npGSEAResultNorm") Returns a left-sided p-value for a npGSEAResultNorm object

signature(object = "npGSEAResultBeta") Returns a left-sided p-value for a npGSEAResultBeta object

signature(object = "npGSEAResultNormCollection") Returns a list of left-sided p-values for a npGSEAResultNormCollection objects (1 for each set)

signature(object = "npGSEAResultBetaCollection") Returns a list of left-sided p-values for a npGSEAResultBetaCollection objects (1 for each set)

Author(s)

Jessica L. Larson

See Also

npGSEAResultNorm-class, pRight

Examples

```r
set.seed(15)
yFactor <- as.factor( c(rep("treated", 5), rep("control", 5)) )
xData <- matrix(data=rnorm(length(letters)*10) ,nrow=length(letters), ncol=10)
rownames(xData) <- letters
geneSetABC15 <- GeneSet(geneIds=letters[1:15], setName="setABC15")
res <- npGSEA(x = xData, y = yFactor, set = geneSetABC15)
pLeft (res)
```

Description

This function returns the right-sided p-value for the corresponding GeneSet or a list of p-values for a GeneSetCollection. This method is only applicable for the normal and beta approximation methods.

Usage

pRight(object)
Arguments

object
An object of type npGSEAResultNorm, npGSEAResultBeta, npGSEAResultNormCollection, or npGSEAResultBetaCollection

Methods

signature(object = "npGSEAResultNorm") Returns a right-sided p-value for a npGSEAResultNorm object
signature(object = "npGSEAResultBeta") Returns a right-sided p-value for a npGSEAResultBeta object
signature(object = "npGSEAResultNormCollection") Returns a list of right-sided p-values for a npGSEAResultNormCollection objects (1 for each set)
signature(object = "npGSEAResultBetaCollection") Returns a list of right-sided p-values for a npGSEAResultBetaCollection objects (1 for each set)

Author(s)

Jessica L. Larson

See Also

npGSEAResultNorm-class, pLeft

Examples

```r
set.seed(15)
yFactor <- as.factor(c(rep("treated", 5), rep("control", 5)))
xData <- matrix(data=rnorm(length(letters)*10), nrow=length(letters), ncol=10)
rownames(xData) <- letters
geneSetABC15 <- GeneSet(geneIds=letters[1:15], setName="setABC15")
res <- npGSEA(x = xData, y = yFactor, set = geneSetABC15)
pRight(res)
```
Arguments

object  

An object of type npGSEAResultNorm, npGSEAResultBeta, npGSEAResultChiSq, npGSEAResultNormCollection, npGSEAResultBetaCollection, or npGSEAResultChiSqCollection

Methods

signature(object = "npGSEAResultNorm") Returns a two-sided p-value for a npGSEAResultNorm object
signature(object = "npGSEAResultBeta") Returns a two-sided p-value for a npGSEAResultBeta object
signature(object = "npGSEAResultChiSq") Returns a two-sided p-value for a npGSEAResultChiSq object
signature(object = "npGSEAResultNormCollection") Returns a list of left-sided p-values for a npGSEAResultNormCollection objects (1 for each set)
signature(object = "npGSEAResultBetaCollection") Returns a list of left-sided p-values for a npGSEAResultBetaCollection objects (1 for each set)
signature(object = "npGSEAResultChiSqCollection") Returns a list of two-sided p-values for a npGSEAResultChiSqCollection objects (1 for each set)

Author(s)

Jessica L. Larson

See Also

npGSEAResultNorm-class, pRight

Examples

set.seed(15)
yFactor <- as.factor( c(rep("treated", 5), rep("control", 5)) )
xData <- matrix(data=rnorm(length(letters)*10) ,nrow=length(letters), ncol=10)ownames(xData) <- letters
geneSetABC15 <- GeneSet(geneIds=letters[1:15], setName="setABC15")
res <- npGSEA(x = xData, y = yFactor, set = geneSetABC15)
pTwoSided (res)
Methods

signature(x = "npGSEAResultNorm"), signature(x = "npGSEAResultBeta"), signature(x = "npGSEAResultChiSq")

These methods display the corresponding p-values for the npGSEA analysis in the gene set in the given experiment.

Author(s)

Jessica L. Larson

Examples

set.seed(15)
yFactor <- as.factor(c(rep("treated", 5), rep("control", 5)))
xData <- matrix(data=rnorm(length(letters)*10), nrow=length(letters), ncol=10)
rownames(xData) <- letters
geneSetABC15 <- GeneSet(geneIds=letters[1:15], setName="setABC15")
res <- npGSEA(x = xData, y = yFactor, set = geneSetABC15)
pValues(res)

Description

This function returns the corresponding variance of the statistic (linear or quadratic) from the npGSEA analysis for a given GeneSet, or a list of these variances for a given GeneSetCollection. This method is applicable for all three approximation methods.

Usage

sigmaSq(object)

Arguments

object An object of type npGSEAResultNorm, npGSEAResultBeta, npGSEAResultChiSq, npGSEAResultNormCollection, npGSEAResultBetaCollection, or npGSEAResultChiSqCollection

Methods

signature(object = "npGSEAResultNorm") Returns the variance of the linear statistic for a npGSEAResultNorm object
signature(object = "npGSEAResultBeta") Returns the variance of the linear statistic for a npGSEAResultBeta object
signature(object = "npGSEAResultChiSq") Returns the variance of the quadratic statistic for a npGSEAResultChiSq object
signature(object = "npGSEAResultNormCollection") Returns a list of the variances of the linear statistics for a npGSEAResultNormCollection objects (1 for each set)
signature(object = "npGSEAResultBetaCollection") Returns a list of the variances of the linear statistics for a npGSEAResultBetaCollection objects (1 for each set)

signature(object = "npGSEAResultChiSqCollection") Returns a list of the variances of the linear statistics for a npGSEAResultChiSqCollection objects (1 for each set)

Author(s)
Jessica L. Larson

See Also
npGSEAResultNorm-class

Examples

```r
set.seed(15)
yFactor <- as.factor(c(rep("treated", 5), rep("control", 5)))
xData <- matrix(data=rnorm(length(letters)*10) ,nrow=length(letters), ncol=10)
rownames(xData) <- letters
geneSetABC15 <- GeneSet(geneIds=letters[1:15], setName="setABC15")
res <- npGSEA(x = xData, y = yFactor, set = geneSetABC15)
sigmaSq(res)
```

Description

This function returns the corresponding statistic (linear or quadratic) from the npGSEA analysis for a given GeneSet, or a list of these statistics for a given GeneSetCollection. This method is applicable for all three approximation methods.

Usage

```r
stat(object)
```

Arguments

- **object** An object of type `npGSEAResultNorm`, `npGSEAResultBeta`, `npGSEAResultChiSq`, `npGSEAResultNormCollection`, `npGSEAResultBetaCollection`, or `npGSEAResultChiSqCollection`

Methods

- signature(object = "npGSEAResultNorm") Returns the linear statistic for a npGSEAResultNorm object
- signature(object = "npGSEAResultBeta") Returns the linear statistic for a npGSEAResultBeta object
signature(object = "npGSEAResultChiSq")  Returns the quadratic statistic for a npGSEAResultChiSq object
signature(object = "npGSEAResultNormCollection")  Returns a list of the linear statistics for a npGSEAResultNormCollection objects (1 for each set)
signature(object = "npGSEAResultBetaCollection")  Returns a list of the linear statistics for a npGSEAResultBetaCollection objects (1 for each set)
signature(object = "npGSEAResultChiSqCollection")  Returns a list of the quadratic statistics for a npGSEAResultChiSqCollection objects (1 for each set)

Author(s)
Jessica L. Larson

See Also
npGSEAResultNorm-class

Examples

set.seed(15)
yFactor <- as.factor( c(rep("treated", 5), rep("control", 5)) )
xData <- matrix(data=rnorm(length(letters)*10) ,nrow=length(letters), ncol=10)
rownames(xData) <- letters
geneSetABC15 <- GeneSet(geneIds=letters[1:15], setName="setABC15")
res <- npGSEA(x = xData, y = yFactor, set = geneSetABC15)
stat(res)
Examples

```r
set.seed(15)
yFactor <- as.factor( c(rep("treated", 5), rep("control", 5)) )
xData <- matrix(data=rnorm(length(letters)*10) ,nrow=length(letters), ncol=10)
rownames(xData) <- letters
geneSetABC15 <- GeneSet(geneIds=letters[1:15], setName="setABC15")
res <- npGSEA(x = xData, y = yFactor, set = geneSetABC15)
res
```

---

**xSet-methods**

### Description

This function returns the scaled and centered expression data for all genes in the corresponding 
GeneSet in the given experiment, or a list of such matrices for each set in a GeneSetCollection. 
This method is applicable for all three approximation methods.

### Usage

```r
xSet(object)
```

### Arguments

- `object` An object of type npGSEAResultNorm, npGSEAResultBeta, npGSEAResultChiSq, npGSEAResultNormCollection, npGSEAResultBetaCollection, or npGSEAResultChiSqCollection

### Methods

- `signature(object = "npGSEAResultNorm")` Returns the centered and scaled X matrix used for analysis to create a npGSEAResultNorm object
- `signature(object = "npGSEAResultBeta")` Returns the centered and scaled X matrix used for analysis to create a npGSEAResultBeta object
- `signature(object = "npGSEAResultChiSq")` Returns the centered and scaled X matrix used for analysis to create a npGSEAResultChiSq object
- `signature(object = "npGSEAResultNormCollection")` Returns a list of centered and scaled X matrices used for analysis to create the npGSEAResultNormCollection objects (1 for each set)
- `signature(object = "npGSEAResultBetaCollection")` Returns a list of centered and scaled X matrices used for analysis to create the npGSEAResultBetaCollection objects (1 for each set)
- `signature(object = "npGSEAResultChiSqCollection")` Returns a list of centered and scaled X matrices used for analysis to create the npGSEAResultChiSqCollection objects (1 for each set)
See Also

npGSEAResultNorm-class

Examples

```r
set.seed(15)
yFactor <- as.factor(c(rep("treated", 5), rep("control", 5)))
xData <- matrix(data=rnorm(length(letters)*10), nrow=length(letters), ncol=10)
rownames(xData) <- letters
geneSetABC15 <- GeneSet(geneIds=letters[1:15], setName="setABC15")
res <- npGSEA(x = xData, y = yFactor, set = geneSetABC15)
head(xSet(res))
```

Description

This function returns the Z-statistic (which is compared to a reference standard normal distribution) for the normal approximation of npGSEA for a corresponding GeneSet or a list of these statistics for a GeneSetCollection. This method is applicable for only the normal approximation method.

Usage

```r
zStat(object)
```

Arguments

- **object**: An object of type `npGSEAResultNorm` or `npGSEAResultNormCollection`

Methods

- `signature(object = "npGSEAResultNorm")` Returns the Z-statistic for a `npGSEAResultNorm` object
- `signature(object = "npGSEAResultNormCollection")` Returns a list of the Z-statistics for a `npGSEAResultNormCollection` objects (1 for each set)

Author(s)

Jessica L. Larson

See Also

- `npGSEAResultNorm-class`
Examples

```r
set.seed(15)
yFactor <- as.factor(c(rep("treated", 5), rep("control", 5)))
xData <- matrix(data=rnorm(length(letters)*10), nrow=length(letters), ncol=10)
rownames(xData) <- letters
geneSetABC15 <- GeneSet(geneIds=letters[1:15], setName="setABC15")
res <- npGSEA(x = xData, y = yFactor, set = geneSetABC15, approx = "norm")
zStat(res)
```
Index

* classes
  npGSEAResultBeta-class, 12
  npGSEAResultBetaCollection-class, 13
  npGSEAResultChiSq-class, 14
  npGSEAResultChiSqCollection-class, 15
  npGSEAResultNorm-class, 16
  npGSEAResultNormCollection-class, 17

* methods
  alphaValue-methods, 2
  betaHats-methods, 3
  betaStat-methods, 4
  betaValue-methods, 5
  chiSqStat-methods, 6
  DF-methods, 7
  geneSetName-methods, 8
  npGSEAPlot-methods, 11
  pLeft-methods, 17
  pRight-methods, 18
  pTwoSided-methods, 19
  pValues-methods, 20
  sigmaSq-methods, 21
  stat-methods, 22
  summary-methods, 23
  xSet-methods, 24
  zStat-methods, 25

  alphaValue (alphaValue-methods, 2
  alphaValue, npGSEAResultBeta-method (alphaValue-methods, 2
  alphaValue, npGSEAResultBetaCollection-method (alphaValue-methods, 2
  alphaValue-methods, 2

  betaHats (betaHats-methods, 3
  betaHats, npGSEAResultBeta-method (betaHats-methods, 3
  betaHats, npGSEAResultBetaCollection-method (betaHats-methods, 3
  betaHats, npGSEAResultChiSq-method (betaHats-methods, 3
  betaHats, npGSEAResultChiSqCollection-method (betaHats-methods, 3
  betaHats, npGSEAResultNorm-method (betaHats-methods, 3
  betaHats, npGSEAResultNormCollection-method (betaHats-methods, 3
  betaHats-methods, 3
  betaStat (betaStat-methods, 4
  betaStat, npGSEAResultBeta-method (betaStat-methods, 4
  betaStat, npGSEAResultBetaCollection-method (betaStat-methods, 4
  betaStat-methods, 4
  betaValue (betaValue-methods, 5
  betaValue, npGSEAResultBeta-method (betaValue-methods, 5
  betaValue, npGSEAResultBetaCollection-method (betaValue-methods, 5
  betaValue-methods, 5

  chiSqStat (chiSqStat-methods, 6
  chiSqStat, npGSEAResultChiSq-method (chiSqStat-methods, 6
  chiSqStat, npGSEAResultChiSqCollection-method (chiSqStat-methods, 6
  chiSqStat-methods, 6

  DF (DF-methods, 7
  DF, npGSEAResultChiSq-method (DF-methods, 7
  DF, npGSEAResultChiSqCollection-method (DF-methods, 7
  DF-methods, 7

  geneSetName (geneSetName-methods, 8
<table>
<thead>
<tr>
<th>Function</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>geneSetName, npGSEAResultBeta-method</td>
<td>8</td>
</tr>
<tr>
<td>geneSetName, npGSEAResultBetaCollection-method</td>
<td>8</td>
</tr>
<tr>
<td>geneSetName, npGSEAResultChiSq-method</td>
<td>8</td>
</tr>
<tr>
<td>geneSetName, npGSEAResultChiSqCollection-method</td>
<td>8</td>
</tr>
<tr>
<td>geneSetName, npGSEAResultNorm-method</td>
<td>8</td>
</tr>
<tr>
<td>geneSetName, npGSEAResultNormCollection-method</td>
<td>8</td>
</tr>
<tr>
<td>getIncidence</td>
<td>9</td>
</tr>
<tr>
<td>npGSEA, 10</td>
<td></td>
</tr>
<tr>
<td>npGSEAPlot (npGSEAPlot-methods), 11</td>
<td></td>
</tr>
<tr>
<td>npGSEAPlot, npGSEAResultBeta-method</td>
<td>11</td>
</tr>
<tr>
<td>npGSEAPlot, npGSEAResultChiSq-method</td>
<td>11</td>
</tr>
<tr>
<td>npGSEAPlot, npGSEAResultNorm-method</td>
<td>11</td>
</tr>
<tr>
<td>npGSEAPlot-methods, 11</td>
<td></td>
</tr>
<tr>
<td>npGSEAResultBeta, 3, 3, 6</td>
<td></td>
</tr>
<tr>
<td>npGSEAResultBeta-class</td>
<td>12</td>
</tr>
<tr>
<td>npGSEAResultBetaCollection-class, 13</td>
<td></td>
</tr>
<tr>
<td>npGSEAResultChiSq, 7, 16</td>
<td></td>
</tr>
<tr>
<td>npGSEAResultChiSq-class, 14</td>
<td></td>
</tr>
<tr>
<td>npGSEAResultChiSqCollection, 17</td>
<td></td>
</tr>
<tr>
<td>npGSEAResultChiSqCollection-class, 15</td>
<td></td>
</tr>
<tr>
<td>npGSEAResultNorm, 4, 9, 12, 13, 15, 18–20</td>
<td></td>
</tr>
<tr>
<td>npGSEAResultNorm-class, 16</td>
<td></td>
</tr>
<tr>
<td>npGSEAResultNormCollection, 14, 15</td>
<td></td>
</tr>
<tr>
<td>npGSEAResultNormCollection-class, 17</td>
<td></td>
</tr>
<tr>
<td>pLeft (pLeft-methods), 17</td>
<td></td>
</tr>
<tr>
<td>pLeft, npGSEAResultBeta-method</td>
<td>17</td>
</tr>
<tr>
<td>pLeft, npGSEAResultBetaCollection-method</td>
<td>17</td>
</tr>
<tr>
<td>pLeft, npGSEAResultNorm-method</td>
<td>17</td>
</tr>
<tr>
<td>pLeft, npGSEAResultNormCollection-method</td>
<td>17</td>
</tr>
<tr>
<td>pLeft-methods, 17</td>
<td></td>
</tr>
<tr>
<td>pRight (pRight-methods), 18</td>
<td></td>
</tr>
<tr>
<td>pRight, npGSEAResultBeta-method</td>
<td>18</td>
</tr>
<tr>
<td>pRight, npGSEAResultBetaCollection-method</td>
<td>18</td>
</tr>
<tr>
<td>pRight, npGSEAResultNorm-method</td>
<td>18</td>
</tr>
<tr>
<td>pRight, npGSEAResultNormCollection-method</td>
<td>18</td>
</tr>
<tr>
<td>pRight-methods, 18</td>
<td></td>
</tr>
<tr>
<td>pTwoSided (pTwoSided-methods), 19</td>
<td></td>
</tr>
<tr>
<td>pTwoSided, npGSEAResultBeta-method</td>
<td>19</td>
</tr>
<tr>
<td>pTwoSided, npGSEAResultBetaCollection-method</td>
<td>19</td>
</tr>
<tr>
<td>pTwoSided, npGSEAResultChiSq-method</td>
<td>19</td>
</tr>
<tr>
<td>pTwoSided, npGSEAResultChiSqCollection-method</td>
<td>19</td>
</tr>
<tr>
<td>pTwoSided, npGSEAResultNorm-method</td>
<td>19</td>
</tr>
<tr>
<td>pTwoSided, npGSEAResultNormCollection-method</td>
<td>19</td>
</tr>
<tr>
<td>pTwoSided-methods, 19</td>
<td></td>
</tr>
<tr>
<td>pValues (pValues-methods), 20</td>
<td></td>
</tr>
<tr>
<td>pValues, npGSEAResultBeta-method</td>
<td>20</td>
</tr>
<tr>
<td>pValues, npGSEAResultBetaCollection-method</td>
<td>20</td>
</tr>
<tr>
<td>pValues, npGSEAResultChiSq-method</td>
<td>20</td>
</tr>
<tr>
<td>pValues, npGSEAResultChiSqCollection-method</td>
<td>20</td>
</tr>
<tr>
<td>pValues, npGSEAResultNorm-method</td>
<td>20</td>
</tr>
<tr>
<td>pValues, npGSEAResultNormCollection-method</td>
<td>20</td>
</tr>
<tr>
<td>pValues-methods, 20</td>
<td></td>
</tr>
<tr>
<td>show (summary-methods), 23</td>
<td></td>
</tr>
<tr>
<td>show, npGSEAResultBeta-method</td>
<td>23</td>
</tr>
<tr>
<td>show, npGSEAResultBetaCollection-method</td>
<td>23</td>
</tr>
<tr>
<td>show, npGSEAResultChiSq-method</td>
<td>23</td>
</tr>
<tr>
<td>show, npGSEAResultChiSqCollection-method</td>
<td>23</td>
</tr>
<tr>
<td>show, npGSEAResultNorm-method</td>
<td>23</td>
</tr>
<tr>
<td>show, npGSEAResultNormCollection-method</td>
<td>23</td>
</tr>
<tr>
<td>show-methods (summary-methods), 23</td>
<td></td>
</tr>
<tr>
<td>sigmaSq (sigmaSq-methods), 21</td>
<td></td>
</tr>
<tr>
<td>sigmaSq, npGSEAResultBeta-method</td>
<td>21</td>
</tr>
<tr>
<td>sigmaSq, npGSEAResultBetaCollection-method</td>
<td>21</td>
</tr>
<tr>
<td>sigmaSq, npGSEAResultChiSq-method</td>
<td>21</td>
</tr>
<tr>
<td>sigmaSq, npGSEAResultChiSqCollection-method</td>
<td>21</td>
</tr>
<tr>
<td>sigmaSq-methods, 21</td>
<td></td>
</tr>
</tbody>
</table>
sigmaSq, npGSEAResultNorm-method
(sigmaSq-methods), 21
sigmaSq, npGSEAResultNormCollection-method
(sigmaSq-methods), 21
sigmaSq-methods, 21
stat (stat-methods), 22
stat, npGSEAResultBeta-method
(stat-methods), 22
stat, npGSEAResultBetaCollection-method
(stat-methods), 22
stat, npGSEAResultChiSq-method
(stat-methods), 22
stat, npGSEAResultChiSqCollection-method
(stat-methods), 22
stat, npGSEAResultNorm-method
(stat-methods), 22
stat, npGSEAResultNormCollection-method
(stat-methods), 22
stat-methods, 22
summary-methods, 23

xSet (xSet-methods), 24
xSet, npGSEAResultBeta-method
(xSet-methods), 24
xSet, npGSEAResultBetaCollection-method
(xSet-methods), 24
xSet, npGSEAResultChiSq-method
(xSet-methods), 24
xSet, npGSEAResultChiSqCollection-method
(xSet-methods), 24
xSet, npGSEAResultNorm-method
(xSet-methods), 24
xSet, npGSEAResultNormCollection-method
(xSet-methods), 24
xSet-methods, 24

zStat (zStat-methods), 25
zStat, npGSEAResultNorm-method
(zStat-methods), 25
zStat, npGSEAResultNormCollection-method
(zStat-methods), 25
zStat-methods, 25