Package ‘npGSEA’

April 1, 2025

Type Package

Title Permutation approximation methods for gene set enrichment analysis (non-permutation GSEA)

Version 1.42.0

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

Collate 'getIncidence.R' 'miscFunctions.R' 'miscDataPrepFunctions.R' 'npGSEA.R' 'AllClasses.R' 'AllGenerics.R' 'npGSEAResultBeta-accessors.R' 'npGSEAResultChiSq-accessors.R' 'npGSEAResultNorm-accessors.R' 'npGSEAPlot-methods.R' 'show-methods.R' 'pValues-methods.R'

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

git_branch RELEASE_3_20

git_last_commit f2e2b8b

git_last_commit_date 2024-10-29

Repository Bioconductor 3.20

Date/Publication 2025-03-31

Contents

`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)
```

`betaHats-methods` ~~ *Methods for Function* `betaHats` ~~

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 contrubution 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`,`npGSEAResultNor` `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

```
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)
```

`betaStat-methods` ~~ *Methods for Function* `betaStat` ~~

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

```
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)
```

`betaValue-methods` ~~ *Methods for Function* `betaValue` ~~

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 `npGSEAResultBeta` object

`signature(object = "npGSEAResultBetaCollection")` Returns a list of the beta 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")
betaValue(res)
```

`chiSqStat-methods` ~~ *Methods for Function* `chiSqStat` ~~

Description

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

Usage

```
chiSqStat(object)
```

Arguments

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

Methods

`signature(object = "npGSEAResultChiSq")` Returns the chi-sq statistic for a `npGSEAResultChiSq` object

`signature(object = "npGSEAResultChiSqCollection")` Returns a list of the chi-sq statistics for a `npGSEAResultChiSqCollection` objects (1 for each set)

Author(s)

Jessica L. Larson

See Also

`npGSEAResultChiSq`-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= "chiSq")
chiSqStat(res)
```

`DF-methods` *~~ Methods for Function* `DF` *~~*

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

```
DF(object)
```

Arguments

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

Methods

`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

```
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 `npGSEAResultChiSqCollecti`

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

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)
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

```
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

Jessica L Larson and Art B Owen: Moment based gene set tests. BMC Bioinformatics 2015, 16:132. http://www.biomedcentral.com/1471-2105/16/132

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
```

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

Jessica L Larson and Art B Owen: Moment based gene set tests. BMC Bioinformatics 2015, 16:132. `http://www.biomedcentral.com/1471-2105/16/132`

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

```
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
- `varThatGw`: Object of class `"numeric"`, the variance of ThatGw
- `alpha`: Object of class `"numeric"`, the alpha value
- `beta`: Object of class `"numeric"`, the beta value
- `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

Jessica L Larson and Art B Owen: Moment based gene set tests. BMC Bioinformatics 2015, 16:132. http://www.biomedcentral.com/1471-2105/16/132

See Also

`npGSEAResultNorm`

Examples

```
showClass("npGSEAResultBeta")
```

`npGSEAResultBetaCollection-class`

Class `"npGSEAResultBetaCollection"`

Description

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

Jessica L Larson and Art B Owen: Moment based gene set tests. BMC Bioinformatics 2015, 16:132. `http://www.biomedcentral.com/1471-2105/16/132`

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

Jessica L Larson and Art B Owen: Moment based gene set tests. BMC Bioinformatics 2015, 16:132. `http://www.biomedcentral.com/1471-2105/16/132`

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

Jessica L Larson and Art B Owen: Moment based gene set tests. BMC Bioinformatics 2015, 16:132. `http://www.biomedcentral.com/1471-2105/16/132`

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

Jessica L Larson and Art B Owen: Moment based gene set tests. BMC Bioinformatics 2015, 16:132. `http://www.biomedcentral.com/1471-2105/16/132`

See Also

`npGSEAResultChiSq`

Examples

```
showClass("npGSEAResultNorm")
```

`npGSEAResultNormCollection-class` *Class* `"npGSEAResultNormCollection"`

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

Jessica L Larson and Art B Owen: Moment based gene set tests. BMC Bioinformatics 2015, 16:132. `http://www.biomedcentral.com/1471-2105/16/132`

See Also

`npGSEAResultChiSqCollection`

Examples

```
showClass("npGSEAResultNormCollection")
```

`pLeft-methods` ~~ *Methods for Function* `pLeft` ~~

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

```
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)
```

`pRight-methods` ~~ *Methods for Function* `pRight` ~~

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

```
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)
```

`pTwoSided-methods` ~~ *Methods for Function* `pTwoSided` ~~

Description

This function returns the two-sided p-value for the corresponding `GeneSet` or a list of p-values for a `GeneSetCollection`. This method is applicable for all three approximation methods.

Usage

`pTwoSided(object)`

Arguments

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

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)
rownames(xData) <- letters
geneSetABC15 <- GeneSet(geneIds=letters[1:15], setName="setABC15")
res <- npGSEA(x = xData, y = yFactor, set = geneSetABC15)
pTwoSided (res)
```

`pValues-methods` ~~ *Methods for Function* `pValues` ~~

Description

~~ Methods for function `pValues` ~~

Methods

- `signature(x = "npGSEAResultNorm")`,`signature(x = "npGSEAResultBeta")`, `signature(x = "npGSEAResultChiS`
  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)
```

`sigmaSq-methods` ~~ *Methods for Function* `sigmaSq` ~~

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 `npGSEAResultChiSqCollecti`

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

```
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)
```

`stat-methods` *~~ Methods for Function* `stat` *~~*

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

```
stat(object)
```

Arguments

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

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)
```

`summary-methods` ~~ *Methods for Function* `show` *in Package* **base** ~~

Description

~~ Methods for function `show` in package **base** ~~

Methods

`signature(x = "npGSEAResultNorm")`**,**`signature(x = "npGSEAResultBeta")`**,** `signature(x = "npGSEAResultChiS`
These methods display the corresponding statistics (linear or quadratic)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)
res
```

`xSet-methods` ~~ *Methods for Function* `xSet` ~~

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

```
xSet(object)
```

Arguments

`object`	An object of type `npGSEAResultNorm`, `npGSEAResultBeta`, `npGSEAResultChiSq`,`npGSEAResultNor` `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

```
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) )
```

`zStat-methods` *~~ Methods for Function* `zStat` *~~*

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

```
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

```
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