Package ‘sumSome’

May 9, 2026

Type Package

Title Permutation True Discovery Guarantee by Sum-Based Tests

Version 1.1.0

Date 2021-11-23

Author Anna Vesely

Maintainer Anna Vesely `<anna.vesely@phd.unipd.it>`

Description

It allows to quickly perform permutation-based closed testing by sum-based global tests, and construct lower confidence bounds for the TDP, simultaneously over all subsets of hypotheses. As a main feature, it produces simultaneous lower confidence bounds for the proportion of active voxels in different clusters for fMRI cluster analysis. Details may be found in Vesely, Finos, and Goeman (2020) <doi:10.48550/arXiv.2102.11759>.

License GPL (>= 2)

Imports Rcpp (>= 1.0.5), pARI, ARIbrain, RNifti

LinkingTo Rcpp

RoxygenNote 7.1.1

Encoding UTF-8

Language en-US

BugReports `https://github.com/annavesely/sumSome/issues`

URL `https://github.com/annavesely/sumSome`

NeedsCompilation yes

Repository CRAN

Date/Publication 2021-11-24 08:10:11 UTC

Contents

`sumSome-package` *True Discovery Guarantee by Sum-Based Tests*

Description

It provides permutation-based true discovery guarantees, using sum-based global statistics (sum of t-scores, p-value combinations, etc.). As a main feature, it produces simultaneous lower confidence bounds for the number of active voxels in different clusters for fMRI cluster analysis.

Author(s)

Anna Vesely.

Maintainer: Anna Vesely <anna.vesely@phd.unipd.it>

References

Goeman, J. J., and Solari, A. (2011). Multiple testing for exploratory research. Statistical Science 26 (4) 584-597.

Vesely, A., Finos, L., and Goeman, J. J. (2020). Permutation-based true discovery guarantee by sum tests. Pre-print arXiv:2102.11759.

See Also

True discovery guarantee: `sumStats`, `sumPvals`

Permutation statistics for brain imaging: `brainScores`, `brainPvals`

True discovery guarantee for cluster analysis: `clusterAnalysis`

`brainPvals` *Permutation p-Values for Brain Imaging*

Description

This function computes p-value combinations for different permutations of brain imaging data. A voxel's p-value is calculated by performing the one-sample `t` test for the null hypothesis that its mean contrast over the different subjects is zero.

Usage

```
brainPvals(copes, mask = NULL, alternative = "two.sided", alpha = 0.05, B = 200,
           seed = NULL, truncFrom = NULL, truncTo = 0.5,
           type = "vovk.wang", r = 0, rand = FALSE)
```

Arguments

`copes`	list of 3D numeric arrays (contrasts maps for each subject).
`mask`	3D logical array, where `TRUE` values correspond to voxels inside the brain, or character for a Nifti file name.
`alternative`	direction of the alternative hypothesis (`greater`, `lower`, `two.sided`).
`alpha`	significance level.
`B`	number of permutations, including the identity.
`seed`	seed.
`truncFrom`	truncation parameter: values greater than `truncFrom` are truncated. If `NULL`, it is set to `alpha`.
`truncTo`	truncation parameter: truncated values are set to `truncTo`. If `NULL`, p-values are not truncated.
`type`	p-value combination among `edgington`, `fisher`, `pearson`, `liptak`, `cauchy`, `vovk.wang` (see details).
`r`	parameter for Vovk and Wang's p-value combination.
`rand`	logical, `TRUE` to compute p-values by permutation distribution.

Details

A p-value `p` is transformed as following.

- Edgington: `-p`
- Fisher: `-log(p)`
- Pearson: `log(1-p)`
- Liptak: `-qnorm(p)`
- Cauchy: `tan(0.5 - p)/p`
- Vovk and Wang: `- sign(r)p^r`

An error message is returned if the transformation produces infinite values.

Truncation parameters should be such that `truncTo` is not smaller than `truncFrom`. As Pearson's and Liptak's transformations produce infinite values in 1, for such methods `truncTo` should be strictly smaller than 1.

The significance level `alpha` should be in the interval [1/`B`, 1).

Value

`brainPvals` returns an object of class `sumBrain`, containing

- `statistics`: numeric matrix of p-values, where columns correspond to voxels inside the brain, and rows to permutations. The first permutation is the identity
- `mask`: 3D logical array, where `TRUE` values correspond to voxels inside the brain
- `alpha`: significance level
- `truncFrom`: transformed first truncation parameter
- `truncTo`: transformed second truncation parameter

Author(s)

Anna Vesely.

References

Goeman, J. J. and Solari, A. (2011). Multiple testing for exploratory research. Statistical Science, 26(4):584-597.

Hemerik, J. and Goeman, J. J. (2018). False discovery proportion estimation by permutations: confidence for significance analysis of microarrays. JRSS B, 80(1):137-155.

Vesely, A., Finos, L., and Goeman, J. J. (2020). Permutation-based true discovery guarantee by sum tests. Pre-print arXiv:2102.11759.

See Also

Permutation statistics for brain imaging using t scores: `brainScores`

True discovery guarantee for cluster analysis: `clusterAnalysis`

Suprathreshold clusters: `findClusters`

Examples

```
# simulate 20 copes with dimensions 10x10x10
set.seed(42)
copes <- list()
for(i in seq(20)){copes[[i]] <- array(rnorm(10^3, mean = -10, sd = 30), dim=c(10,10,10))}

# cluster map where t scores are grater than 2.8, in absolute value
thr <- 2.8
cl <- findClusters(copes = copes, thr = thr)

# create object of class sumBrain (combination: Cauchy)
```

```
res <- brainPvals(copes = copes, alpha = 0.2, seed = 42, type = "cauchy")
res
summary(res)

# confidence bound for the number of true discoveries and the TDP within clusters
out <- clusterAnalysis(res, clusters = cl$clusters)
```

`brainScores` *Permutation t-Scores for Brain Imaging*

Description

This function computes t-scores for different permutations of brain imaging data. A voxel's score is calculated by performing the one-sample t test for the null hypothesis that its mean contrast over the different subjects is zero.

Usage

```
brainScores(copes, mask = NULL, alternative = "two.sided", alpha = 0.05, B = 200,
            seed = NULL, truncFrom = 3.2, truncTo = 0, squares = FALSE)
```

Arguments

`copes`	list of 3D numeric arrays (contrasts maps for each subject).
`mask`	3D logical array, where `TRUE` values correspond to voxels inside the brain, or character for a Nifti file name.
`alternative`	direction of the alternative hypothesis (`greater`, `lower`, `two.sided`).
`alpha`	significance level.
`B`	number of permutations, including the identity.
`seed`	seed.
`truncFrom`	truncation parameter: values less extreme than `truncFrom` are truncated. If `NULL`, statistics are not truncated.
`truncTo`	truncation parameter: truncated values are set to `truncTo`. If `NULL`, statistics are not truncated.
`squares`	logical, `TRUE` to use squared t-scores.

Details

Truncation parameters should be such that `truncTo` is not more extreme than `truncFrom`.

The significance level `alpha` should be in the interval [`1/B`, 1).

Value

`brainScores` returns an object of class `sumBrain`, containing

- `statistics`: numeric matrix of t-scores, where columns correspond to voxels inside the brain, and rows to permutations. The first permutation is the identity
- `mask`: 3D logical array, where `TRUE` values correspond to voxels inside the brain
- `alpha`: significance level
- `truncFrom`: transformed first truncation parameter
- `truncTo`: transformed second truncation parameter

Author(s)

Anna Vesely.

References

Goeman, J. J. and Solari, A. (2011). Multiple testing for exploratory research. Statistical Science, 26(4):584-597.

Hemerik, J. and Goeman, J. J. (2018). False discovery proportion estimation by permutations: confidence for significance analysis of microarrays. JRSS B, 80(1):137-155.

Vesely, A., Finos, L., and Goeman, J. J. (2020). Permutation-based true discovery guarantee by sum tests. Pre-print arXiv:2102.11759.

See Also

Permutation statistics for brain imaging using p-values: `brainPvals`

True discovery guarantee for cluster analysis: `clusterAnalysis`

Suprathreshold clusters: `findClusters`

Examples

```
# simulate 20 copes with dimensions 10x10x10
set.seed(42)
copes <- list()
for(i in seq(20)){copes[[i]] <- array(rnorm(10^3, mean = -10, sd = 30), dim=c(10,10,10))}

# cluster map where t scores are grater than 2.8, in absolute value
thr <- 2.8
cl <- findClusters(copes = copes, thr = thr)

# create object of class sumBrain
res <- brainScores(copes = copes, alpha = 0.2, seed = 42, truncFrom = thr)
res
summary(res)

# confidence bound for the number of true discoveries and the TDP within clusters
out <- clusterAnalysis(res, clusters = cl$clusters)
```

`clusterAnalysis` *True Discovery Guarantee for Cluster Analysis*

Description

This function determines a true discovery guarantee for fMRI cluster analysis. It computes confidence bounds for the number of true discoveries and the true discovery proportion within each cluster. The bounds are simultaneous over all sets, and remain valid under post-hoc selection.

Usage

```
clusterAnalysis(sumBrain, clusters, nMax = 50, silent = FALSE)
```

Arguments

`sumBrain`	an object of class sumBrain, as returned by the functions `brainScores` and `brainPvals`.
`clusters`	3D numeric array of cluster indices, or character for a Nifti file name. If NULL, the whole brain is considered.
`nMax`	maximum number of iterations per cluster.
`silent`	logical, `FALSE` to print the summary.

Value

`clusterAnalysis` returns a list containing `summary` (matrix) and `TDPmap` (3D numeric array of the true discovery proportions). The matrix `summary` contains, for each cluster,

- `size`: size
- `TD`: lower (1-`alpha`)-confidence bound for the number of true discoveries
- `maxTD`: maximum value of `TD` that could be found under convergence of the algorithm
- `TDP`: lower (1-`alpha`)-confidence bound for the true discovery proportion
- `maxTD`: maximum value of `TDP` that could be found under convergence of the algorithm
- `dim1`, `dim2`, `dim3`: coordinates of the center of mass.

Author(s)

Anna Vesely.

References

Goeman, J. J. and Solari, A. (2011). Multiple testing for exploratory research. Statistical Science, 26(4):584-597.

Hemerik, J. and Goeman, J. J. (2018). False discovery proportion estimation by permutations: confidence for significance analysis of microarrays. JRSS B, 80(1):137-155.

Vesely, A., Finos, L., and Goeman, J. J. (2020). Permutation-based true discovery guarantee by sum tests. Pre-print arXiv:2102.11759.

See Also

Permutation statistics for brain imaging: `brainScores`, `brainPvals`

Suprathreshold clusters: `findClusters`

Examples

```
# simulate 20 copes with dimensions 10x10x10
set.seed(42)
copes <- list()
for(i in seq(20)){copes[[i]] <- array(rnorm(10^3, mean = -10, sd = 30), dim=c(10,10,10))}

# cluster map where t scores are grater than 2.8, in absolute value
thr <- 2.8
cl <- findClusters(copes = copes, thr = thr)

# create object of class sumBrain
res <- brainScores(copes = copes, alpha = 0.2, seed = 42, truncFrom = thr)
res
summary(res)

# confidence bound for the number of true discoveries and the TDP within clusters
out <- clusterAnalysis(res, clusters = cl$clusters)
```

`discoveries` *Confidence Bound for the Number of True Discoveries*

Description

This function determines a lower confidence bound for the number of true discoveries within a set of interest. The bound remains valid under post-hoc selection.

Usage

```
discoveries(object)

## S3 method for class 'sumObj'
discoveries(object)
```

Arguments

`object`	an object of class `sumObj`, as returned by the functions `sumStats` and `sumPvals`.

Value

`discoveries` returns a lower (1-`alpha`)-confidence bound for the number of true discoveries in the set.

Author(s)

Anna Vesely.

References

Goeman, J. J. and Solari, A. (2011). Multiple testing for exploratory research. Statistical Science, 26(4):584-597.

Hemerik, J. and Goeman, J. J. (2018). False discovery proportion estimation by permutations: confidence for significance analysis of microarrays. JRSS B, 80(1):137-155.

Vesely, A., Finos, L., and Goeman, J. J. (2020). Permutation-based true discovery guarantee by sum tests. Pre-print arXiv:2102.11759.

See Also

Create a `sumObj` object: `sumStats`, `sumPvals`

Lower confidence bound for the TDP: `tdp`

Upper confidence bound for the FDP: `fdp`

Examples

```
# generate matrix of p-values for 5 variables and 10 permutations
G <- simData(prop = 0.6, m = 5, B = 10, alpha = 0.4, seed = 42)

# subset of interest (variables 1 and 2)
S <- c(1,2)

# create object of class sumObj
# combination: harmonic mean (Vovk and Wang with r = -1)
res <- sumPvals(G, S, alpha = 0.4, r = -1)
res
summary(res)

# lower confidence bound for the number of true discoveries in S
discoveries(res)

# lower confidence bound for the true discovery proportion in S
tdp(res)

# upper confidence bound for the false discovery proportion in S
fdp(res)
```

`fdp` *Confidence Bound for the FDP*

Description

This function determines an upper confidence bound for the false discovery proportion within a set of interest. The bound remains valid under post-hoc selection.

Usage

```
fdp(object)

## S3 method for class 'sumObj'
fdp(object)
```

Arguments

`object`	an object of class `sumObj`, as returned by the functions `sumStats` and `sumPvals`.

Value

`fdp` returns an upper (1-`alpha`)-confidence bound for the false discovery proportion in the set.

Author(s)

Anna Vesely.

References

Goeman, J. J. and Solari, A. (2011). Multiple testing for exploratory research. Statistical Science, 26(4):584-597.

Hemerik, J. and Goeman, J. J. (2018). False discovery proportion estimation by permutations: confidence for significance analysis of microarrays. JRSS B, 80(1):137-155.

Vesely, A., Finos, L., and Goeman, J. J. (2020). Permutation-based true discovery guarantee by sum tests. Pre-print arXiv:2102.11759.

See Also

Create a `sumObj` object: `sumStats`, `sumPvals`

Lower confidence bound for the number of true discoveries: `discoveries`

Lower confidence bound for the TDP: `tdp`

Examples

```
# generate matrix of p-values for 5 variables and 10 permutations
G <- simData(prop = 0.6, m = 5, B = 10, alpha = 0.4, seed = 42)

# subset of interest (variables 1 and 2)
S <- c(1,2)

# create object of class sumObj
# combination: harmonic mean (Vovk and Wang with r = -1)
res <- sumPvals(G, S, alpha = 0.4, r = -1)
res
summary(res)

# lower confidence bound for the number of true discoveries in S
discoveries(res)
```

```
# lower confidence bound for the true discovery proportion in S
tdp(res)

# upper confidence bound for the false discovery proportion in S
fdp(res)
```

`findClusters` *Suprathreshold Clusters for Brain Imaging*

Description

This function determines spatially connected clusters, where t-scores are more extreme than a given threshold.

Usage

```
findClusters(copes, mask = NULL, thr = 3.2, alternative = "two.sided", silent = FALSE)
```

Arguments

`copes`	list of 3D numeric arrays (contrasts maps for each subject).
`mask`	3D logical array, where `TRUE` values correspond to voxels inside the brain, or character for a Nifti file name.
`thr`	threshold.
`alternative`	direction of the alternative hypothesis (`greater`, `lower`, `two.sided`).
`silent`	logical, `FALSE` to print the number of clusters.

Value

`findClusters` returns a 3D numeric array, with integer values corresponding to clusters, and 0 to other voxels.

Author(s)

Anna Vesely.

See Also

Permutation statistics for brain imaging: `brainScores`, `brainPvals`

True discovery guarantee for cluster analysis: `clusterAnalysis`

Examples

```
# simulate 20 copes with dimensions 10x10x10
set.seed(42)
copes <- list()
for(i in seq(20)){copes[[i]] <- array(rnorm(10^3, mean = -10, sd = 30), dim=c(10,10,10))}

# cluster map where t scores are grater than 2.8, in absolute value
thr <- 2.8
cl <- findClusters(copes = copes, thr = thr)

# create object of class sumBrain
res <- brainScores(copes = copes, alpha = 0.2, seed = 42, truncFrom = thr)
res
summary(res)

# confidence bound for the number of true discoveries and the TDP within clusters
out <- clusterAnalysis(res, clusters = cl$clusters)
```

`simData` *Simulating Matrix of Statistics*

Description

This function simulates a matrix of permutation statistics, by performing a t test on normal data.

Usage

```
simData(prop, m, B = 200, rho = 0, n = 50, alpha = 0.05, pw = 0.8, p = TRUE, seed = NULL)
```

Arguments

`prop`	proportion of non-null hypotheses.
`m`	total number of variables.
`B`	number of permutations, including the identity.
`rho`	level of equicorrelation between pairs of variables.
`n`	number of observations.
`alpha`	significance level.
`pw`	power of the t test.
`p`	logical, `TRUE` to compute p-values, `FALSE` to compute t-scores.
`seed`	seed.

Details

The function applies the one-sample two-sided t test to a matrix of simulated data, for `B` data permutations. Data is obtained by simulating `n` independent observations from a multivariate normal distribution, where a proportion `prop` of the variables has non-null mean. This mean is such that the one-sample t test with significance level `alpha` has power equal to `pw`. Each pair of distinct variables has equicorrelation `rho`.

Value

`simData` returns a matrix where the `B` rows correspond to permutations (the first is the identity), and the `m` columns correspond to variables. The matrix contains p-values if `p` is `TRUE`, and t-scores otherwise. The first columns (a proportion `prop`) correspond to non-null hypotheses.

Author(s)

Anna Vesely.

See Also

True discovery guarantee: `sumStats`, `sumPvals`

Examples

```
# generate matrix of p-values for 5 variables and 10 permutations
G <- simData(prop = 0.6, m = 5, B = 10, alpha = 0.4, seed = 42)

# subset of interest (variables 1 and 2)
S <- c(1,2)

# create object of class sumObj
# combination: harmonic mean (Vovk and Wang with r = -1)
res <- sumPvals(G, S, alpha = 0.4, r = -1)
res
summary(res)

# lower confidence bound for the number of true discoveries in S
discoveries(res)

# lower confidence bound for the true discovery proportion in S
tdp(res)

# upper confidence bound for the false discovery proportion in S
fdp(res)
```

`sumPvals` *True Discovery Guarantee for p-Value Combinations*

Description

This function determines confidence bounds for the number of true discoveries, the true discovery proportion and the false discovery proportion within a set of interest, when using p-values as test statistics. The bounds are simultaneous over all sets, and remain valid under post-hoc selection.

Usage

```
sumPvals(G, S = NULL, alpha = 0.05, truncFrom = NULL, truncTo = 0.5,
         type = "vovk.wang", r = 0, nMax = 50)
```

Arguments

`G`	numeric matrix of p-values, where columns correspond to variables, and rows to data transformations (e.g. permutations). The first transformation is the identity.
`S`	vector of indices for the variables of interest (if not specified, all variables).
`alpha`	significance level.
`truncFrom`	truncation parameter: values greater than `truncFrom` are truncated. If `NULL`, it is set to `alpha`.
`truncTo`	truncation parameter: truncated values are set to `truncTo`. If `NULL`, p-values are not truncated.
`type`	p-value combination among `edgington`, `fisher`, `pearson`, `liptak`, `cauchy`, `vovk.wang` (see details).
`r`	parameter for Vovk and Wang's p-value combination.
`nMax`	maximum number of iterations.

Details

A p-value `p` is transformed as following.

- Edgington: `-p`
- Fisher: `-log(p)`
- Pearson: `log(1-p)`
- Liptak: `-qnorm(p)`
- Cauchy: `tan(0.5 - p)/p`
- Vovk and Wang: `- sign(r)p^r`

An error message is returned if the transformation produces infinite values.

Truncation parameters should be such that `truncTo` is not smaller than `truncFrom`. As Pearson's and Liptak's transformations produce infinite values in 1, for such methods `truncTo` should be strictly smaller than 1.

The significance level `alpha` should be in the interval [1/`B`, 1), where `B` is the number of data transformations (rows in `G`).

Value

`sumPvals` returns an object of class `sumObj`, containing

- `total`: total number of variables (columns in `G`)
- `size`: size of `S`
- `alpha`: significance level
- `TD`: lower (1-`alpha`)-confidence bound for the number of true discoveries in `S`
- `maxTD`: maximum value of `TD` that could be found under convergence of the algorithm
- `iterations`: number of iterations of the algorithm

Author(s)

Anna Vesely.

References

Goeman, J. J. and Solari, A. (2011). Multiple testing for exploratory research. Statistical Science, 26(4):584-597.

Hemerik, J. and Goeman, J. J. (2018). False discovery proportion estimation by permutations: confidence for significance analysis of microarrays. JRSS B, 80(1):137-155.

Vesely, A., Finos, L., and Goeman, J. J. (2020). Permutation-based true discovery guarantee by sum tests. Pre-print arXiv:2102.11759.

See Also

True discovery guarantee using generic statistics: `sumStats`

Access a `sumObj` object: `discoveries`, `tdp`, `fdp`

Examples

```
# generate matrix of p-values for 5 variables and 10 permutations
G <- simData(prop = 0.6, m = 5, B = 10, alpha = 0.4, seed = 42)

# subset of interest (variables 1 and 2)
S <- c(1,2)

# create object of class sumObj
# combination: harmonic mean (Vovk and Wang with r = -1)
res <- sumPvals(G, S, alpha = 0.4, r = -1)
res
summary(res)

# lower confidence bound for the number of true discoveries in S
discoveries(res)

# lower confidence bound for the true discovery proportion in S
tdp(res)

# upper confidence bound for the false discovery proportion in S
fdp(res)
```

`sumStats` *True Discovery Guarantee for Generic Statistics*

Description

This function determines confidence bounds for the number of true discoveries, the true discovery proportion and the false discovery proportion within a set of interest. The bounds are simultaneous over all sets, and remain valid under post-hoc selection.

Usage

```
sumStats(G, S = NULL, alternative = "greater", alpha = 0.05,
         truncFrom = NULL, truncTo = NULL, nMax = 50)
```

Arguments

`G`	numeric matrix of statistics, where columns correspond to variables, and rows to data transformations (e.g. permutations). The first transformation is the identity.
`S`	vector of indices for the variables of interest (if not specified, all variables).
`alternative`	direction of the alternative hypothesis (`greater`, `lower`, `two.sided`).
`alpha`	significance level.
`truncFrom`	truncation parameter: values less extreme than `truncFrom` are truncated. If `NULL`, statistics are not truncated.
`truncTo`	truncation parameter: truncated values are set to `truncTo`. If `NULL`, statistics are not truncated.
`nMax`	maximum number of iterations.

Details

Truncation parameters should be such that `truncTo` is not more extreme than `truncFrom`.

The significance level `alpha` should be in the interval `[1/B`, 1), where `B` is the number of data transformations (rows in `G`).

Value

`sumStats` returns an object of class `sumObj`, containing

- `total`: total number of variables (columns in `G`)
- `size`: size of `S`
- `alpha`: significance level
- `TD`: lower (1-`alpha`)-confidence bound for the number of true discoveries in `S`
- `maxTD`: maximum value of `TD` that could be found under convergence of the algorithm
- `iterations`: number of iterations of the algorithm

Author(s)

Anna Vesely.

References

Goeman, J. J. and Solari, A. (2011). Multiple testing for exploratory research. Statistical Science, 26(4):584-597.

Hemerik, J. and Goeman, J. J. (2018). False discovery proportion estimation by permutations: confidence for significance analysis of microarrays. JRSS B, 80(1):137-155.

Vesely, A., Finos, L., and Goeman, J. J. (2020). Permutation-based true discovery guarantee by sum tests. Pre-print arXiv:2102.11759.

See Also

True discovery guarantee using p-values: `sumPvals`

Access a `sumObj` object: `discoveries`, `tdp`, `fdp`

Examples

```
# generate matrix of t-scores for 5 variables and 10 permutations
G <- simData(prop = 0.6, m = 5, B = 10, alpha = 0.4, p = FALSE, seed = 42)

# subset of interest (variables 1 and 2)
S <- c(1,2)

# create object of class sumObj
res <- sumStats(G, S, alpha = 0.4, truncFrom = 0.7, truncTo = 0)
res
summary(res)

# lower confidence bound for the number of true discoveries in S
discoveries(res)

# lower confidence bound for the true discovery proportion in S
tdp(res)

# upper confidence bound for the false discovery proportion in S
fdp(res)
```

`tdp` *Confidence Bound for the TDP*

Description

This function determines a lower confidence bound for the true discovery proportion within a set of interest. The bound remains valid under post-hoc selection.

Usage

```
tdp(object)

## S3 method for class 'sumObj'
tdp(object)
```

Arguments

`object`	an object of class `sumObj`, as returned by the functions `sumStats` and `sumPvals`.

Value

`tdp` returns a lower (1-`alpha`)-confidence bound for the true discovery proportion in the set.

Author(s)

Anna Vesely.

References

Goeman, J. J. and Solari, A. (2011). Multiple testing for exploratory research. Statistical Science, 26(4):584-597.

Hemerik, J. and Goeman, J. J. (2018). False discovery proportion estimation by permutations: confidence for significance analysis of microarrays. JRSS B, 80(1):137-155.

Vesely, A., Finos, L., and Goeman, J. J. (2020). Permutation-based true discovery guarantee by sum tests. Pre-print arXiv:2102.11759.

See Also

Create a `sumObj` object: `sumStats`, `sumPvals`

Lower confidence bound for the number of true discoveries: `discoveries`

Upper confidence bound for the FDP: `fdp`

Examples

```
# generate matrix of p-values for 5 variables and 10 permutations
G <- simData(prop = 0.6, m = 5, B = 10, alpha = 0.4, seed = 42)

# subset of interest (variables 1 and 2)
S <- c(1,2)

# create object of class sumObj
# combination: harmonic mean (Vovk and Wang with r = -1)
res <- sumPvals(G, S, alpha = 0.4, r = -1)
res
summary(res)

# lower confidence bound for the number of true discoveries in S
discoveries(res)

# lower confidence bound for the true discovery proportion in S
tdp(res)

# upper confidence bound for the false discovery proportion in S
fdp(res)
```

Index