Package 'OVESEG'

November 12, 2026

Type Package

Title OVESEG-test to detect tissue/cell-specific markers

Version 1.28.0

Author Lulu Chen <luluchen@vt.edu>

Maintainer Lulu Chen <luluchen@vt.edu>

biocViews Software, MultipleComparison, CellBiology, GeneExpression

Description An R package for multiple-group comparison to detect tissue/cell-specific marker genes among subtypes. It provides functions to compute OVESEG-test statistics, derive component weights in the mixture null distribution model and estimate p-values from weightedly aggregated permutations. Obtained posterior probabilities of component null hypotheses can also portrait all kinds of upregulation patterns among subtypes.

License GPL-2

Encoding UTF-8

Depends R (>= 3.6)

Suggests knitr, rmarkdown, BiocStyle, testthat, ggplot2, gridExtra, grid, reshape2, scales

VignetteBuilder knitr

Imports stats, utils, methods, BiocParallel, SummarizedExperiment, limma, fdrtool, Rcpp

BugReports https://github.com/Lululuella/OVESEG

RoxygenNote 6.1.1

LinkingTo Rcpp

SystemRequirements C++11

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

git_branch RELEASE_3_23

git_last_commit 977248c

git_last_commit_date 2026-04-28

Repository Bioconductor 3.23

Date/Publication 2026-11-11

Contents

OVESEG-package . . . 2
countBT . . . 3
nullDistri . . . 3
OVESEGtest . . . 4
OVESEGtstat . . . 6
OVEtstatPermTopM . . . 7
pairwise_tstat_unscaled . . . 8
patternDistri . . . 9
permfunc . . . 9
postProbNull . . . 10
pvalueWeightedEst . . . 11
RocheBT . . . 12
row_min . . . 13
row_which_max . . . 13
shuffle_topm . . . 14

Index **15**

`OVESEG-package` *OVESEG: A package for marker gene test.*

Description

Function `OVESEGtest` performs OVESEG-test for expression profiles from multiple groups to detect subtype-specific marker genes. While it may take a long time to execute permutations for p-value estimation, users can apply `OVESEGtstat` to obtain OVESEG-test statistics to rank genes and apply `postProbNull` to obtain each gene's posterior probabilities of component null hypotheses. `nullDistri` estimates probabilities of any one group being upregulated under null hypotheses. `patternDistri` estimates probabilities of all kinds of upregulation patterns among groups.

References

Chen, L., Herrington, D., Clarke, R., Yu, G., and Wang, Y. (2019). "Data-Driven Robust Detection of Tissue/Cell-Specific Markers." bioRxiv. https://doi.org/10.1101/517961.

countBT *RNAseq count data downsampled from GSE60424*

Description

Three cell subtypes (B cells, CD4+ T cells, CD8+ T cells) were purified from 20 fresh blood samples. RNA was extracted from each of these cell subsets and processed into RNA sequencing libraries (Illumina TruSeq). Sequencing libraries were analyzed on an Illumina HiScan, with a target read depth of ~20M reads. Reads were demultiplexed, mapped to human gene models (ENSEMBL), and tabulated using HTSeq. We downsample the original data to 10000 genes. Subtype labels for purified populations are also included. (Data generation script can be found in ./data_raw folder.)

Usage

```
data(countBT)
```

Format

A list with one count mixture (count) and a categorical vector giving subtypes (group).

References

Linsley et al. PLoS One 2014;9(10):e109760. PMID: 25314013

nullDistri *Probability of one group being upregulated under null*

Description

This function estimates probabilities of any one group being upregulated than other groups under null hypotheses.

Usage

```
nullDistri(ppnull)
```

Arguments

ppnull	a list returned by postProbNull or OVESEGtest.

Details

The probability of one group being upregulated under null hypotheses is calculated by accumulating and normalizing genewise posterior probability of null hypotheses. The group with higher probability tends to get more False Positive MGs.

Value

a numeric vector indicating probabilities of each group being upregulated than others under null hypotheses.

Examples

```
data(RocheBT)
ppnull <- postProbNull(RocheBT$y, RocheBT$group, alpha='moderated')
pk <- nullDistri(ppnull)
```

`OVESEGtest` *OVESEG-test*

Description

This function performs OVESEG-test to assess significance of molecular markers.

Usage

```
OVESEGtest(y, group, weights = NULL, alpha = "moderated",
  NumPerm = 999, seed = 111, detail.return = TRUE,
  BPPARAM = bpparam())
```

Arguments

`y`	a numeric matrix containing log-expression or logCPM (log2-counts per million) values. Data frame or SummarizedExperiment object will be internally coerced into a matrix. Rows correspond to probes and columns to samples. Missing values are not permitted.
`group`	categorical vector or factor giving group membership of columns of y. At least two categories need to be presented.
`weights`	optional numeric matrix containing prior weights for each spot.
`alpha`	parameter specifying within-group variance estimator to be used. 'moderated': empirical Bayes moderated variance estimator as used in `eBayes`. Numeric value: a constant value added to pooled variance estimator ($\alpha + \sigma$). NULL: no estimator; all variances are set to be 1.
`NumPerm`	an integer specifying the number of permutation resamplings (default 999).
`seed`	an integer seed for the random number generator.
`detail.return`	a logical indicating whether more details about posterior probability estimation will be returned.
`BPPARAM`	a BiocParallelParam object indicating whether parallelization should be used for permutation resamplings. The default is bpparam().

Details

OVESEG-test is a statistically-principled method that can detect tissue/cell-specific marker genes among many subtypes. OVESEG-test statistics are designed to mathematically match the definition of molecular markers, and a novel permutation scheme are employed to estimate the corresponding distribution under null hypotheses where the expression patterns of non-markers can be highly complex.

Value

a list containing the following components:

`pv.overall`	a vector of p-values calculated by all permutations regardless of upregulated subtypes.
`pv.oneside`	a vector of subtype-specific p-values calculated specifically for the upregulated subtype of each probe.
`pv.oneside.max`	subtype-specific p-values when observed test statistic equal to zero.
`pv.multiside`	pv.oneside*K (K-time comparison correction) and truncated at 1.
`W`	a matrix of posterior probabilities for each component null hypothesis given an observed probe. Rows correspond to probes and columns to one hypothesis.
`label`	a vector of group labels.
`groupOrder`	a matrix with each row being group indexes ordered based on decreasing expression levels. Group indexes are positions in `label`.
`F.p.value`	a matrix with each column giving p-values corresponding to F-statistics on certain groups.
`lfdr`	a matrix with each column being local false discovery rates estimated based on one column of weighted F.p.value matrix.
`fit`	a `MArrayLM` fitted model object produced by `lmFit`.

`F.p.value`, `lfdr` and `fit` are returned only when `detail.return` is TRUE.

Examples

```
data(RocheBT)
rtest <- OVESEGtest(RocheBT$y, RocheBT$group, NumPerm=99,
                    BPPARAM=BiocParallel::SerialParam())
## Not run:
#parallel computing
rtest <- OVESEGtest(RocheBT$y, RocheBT$group, NumPerm=99,
                    BPPARAM=BiocParallel::SnowParam())

## End(Not run)
```

OVESEGtstat *OVESEG-test statistics*

Description

This function computes OVESEG-test statistics.

Usage

```
OVESEGtstat(y, group, weights = NULL, alpha = "moderated",
  order.return = FALSE, lmfit.return = FALSE)
```

Arguments

y	a numeric matrix containing log-expression or logCPM (log2-counts per million) values. Data frame or SummarizedExperiment object will be internally coerced into a matrix. Rows correspond to probes and columns to samples. Missing values are not permitted.
group	categorical vector or factor giving group membership of columns of y. At least two categories need to be presented.
weights	optional numeric matrix containing prior weights for each spot.
alpha	parameter specifying within-group variance estimator to be used. 'moderated': empirical Bayes moderated variance estimator as used in eBayes. Numeric value: a constant value added to pooled variance estimator ($\alpha + \sigma$). NULL: no estimator; all variances are set to be 1.
order.return	a logical indicating whether the order of groups will be returned. If FALSE, only the highest expressed group index is return for each probe.
lmfit.return	a logical indicating whether a MArrayLM fitted model object produced by lmFit should be returned.

Details

OVESEG-test statistics are designed to mathematically match the definition of molecular markers:

$$\max_{k=1,\ldots,K}\left\{min_{l\neq k}\left\{\frac{\mu_k(j)-\mu_l(j)}{\sigma(j)\sqrt{\frac{1}{N_k}+\frac{1}{N_l}}}\right\}\right\}$$

where j is probe index, K is the number of groups, and μ_k is the mean expression of group k.

Value

a list containing the following components:

tstat	a vector of OVESEG-test statistics for probes.
label	a vector of group labels.

groupOrder	If order.return is TRUE, a matrix with each row being group indexes ordered based on decreasing expression levels. If order.return is FALSE, a vector with each element being a probe's highest expressed group index. Group indexes are positions in label.
fit	a MArrayLM fitted model object produced by lmFit. Returned only when lmfit.return is TRUE.

Examples

```
data(RocheBT)
rtstat <- OVESEGtstat(RocheBT$y, RocheBT$group, alpha='moderated')
rtstat <- OVESEGtstat(RocheBT$y, RocheBT$group, alpha=0.1)
```

OVEtstatPermTopM *OVESEG-test statistics after permuting top M groups*

Description

This function permutes group labels among highest expressed M groups and then computes new OVESEG-test statistics.

Usage

```
OVEtstatPermTopM(y, group, groupOrder, M, weights = NULL,
  alpha = "moderated", NumPerm = 999, seed = 111,
  BPPARAM = bpparam())
```

Arguments

y	a numeric matrix containing log-expression or logCPM (log2-counts per million) values. Data frame or SummarizedExperiment object will be internally coerced into a matrix. Rows correspond to probes and columns to samples. Missing values are not permitted.
group	categorical vector or factor giving group membership of columns of y. At least two categories need to be presented.
groupOrder	an integer matrix with each row giving group indexes ordered based on decreasing expression levels .
M	an integer indicating the number of groups being permutated. The range is $[2, K]$, where K is the total number of groups.
weights	optional numeric matrix containing prior weights for each spot.
alpha	parameter specifying within-group variance estimator to be used. 'moderated': empirical Bayes moderated variance estimator as used in eBayes. Numeric value: a constant value added to pooled variance estimator ($\alpha + \sigma$). NULL: no estimator; all variances are set to be 1.
NumPerm	an integer specifying the number of permutation resamplings (default 999).
seed	an integer seed for the random number generator.
BPPARAM	a BiocParallelParam object indicating whether parallelization should be used for permutation resamplings. The default is bpparam().

Details

Top M expressed groups will be involved in permutation. There are C_K^M probe patterns in which probes are highly expressed in certain M groups among the total K groups. Probes of the same pattern share the same shuffled labels.

To improve the time efficiency, some functions within permutation loops are implemented using c++.

Value

a list containing the following components:

`tstat.perm`	a numeric matrix with each column giving OVESEG-test statistics over the expressions after one permutation.
`topidx.perm`	a integer matrix with each colulmn giving the highest expressed group index over the expressions after one permutation.

Examples

```
data(RocheBT)
ppnull <- postProbNull(RocheBT$y, RocheBT$group, detail.return = TRUE)
rperm <- OVEtstatPermTopM(RocheBT$y, RocheBT$group, ppnull$groupOrder, M=2,
                          NumPerm=99, BPPARAM=BiocParallel::SerialParam())
## Not run:
#parallel computing
rperm <- OVEtstatPermTopM(RocheBT$y, RocheBT$group, ppnull$groupOrder, M=2,
                          NumPerm=99, BPPARAM=BiocParallel::SnowParam())

## End(Not run)
```

`pairwise_tstat_unscaled` *pairwise t-statistics (unscaled)*

Description

pairwise t-statistics (unscaled)

Usage

```
pairwise_tstat_unscaled(ymean, stdevUnscaled)
```

Arguments

`ymean`	a numeric matrix containing group means.
`stdevUnscaled`	a numeric matrix containing unscaled standard deviations of the group means.

Value

unscaled pairwise t-statistics

patternDistri *Probabilities of all upregulation patterns*

Description

This function estimates probabilities of all kinds of upregulation patterns among subtypes.

Usage

```
patternDistri(ppnull)
```

Arguments

ppnull	a list returned by postProbNull or OVESEGtest.

Details

The probability of each upregulation pattern is calculated by accumulating and normalizing genewise posterior probability of null hypotheses and of alternative hypotheses.

Value

a data frame object containing all possible upregulation patterns and corresponding probabilities.

Examples

```
data(RocheBT)
ppnull <- postProbNull(RocheBT$y, RocheBT$group, alpha='moderated')
pd<- patternDistri(ppnull)
```

permfunc *Internal function for one permutation task*

Description

Internal function for one permutation task

Usage

```
permfunc(p, y, group, weights, alpha, combM, geneSubset, seeds)
```

Arguments

p	an integer indicating permutation index
y	an expressions matrix
group	a integer vector indicating group labels
weights	optional numeric matrix containing prior weights
alpha	parameter specifying within-group variance estimator to be used
combM	a integer matrix with each row giving one choice of M groups
geneSubset	a integer vector indicating the probe pattern of combM
seed	an integer seed for the random number generator

Value

test statistics and upregulated group indexes after one permutaion

---

postProbNull *Posterior probabilities of each component null hypothesis*

Description

This function computes posterior probabilities of each component null hypothesis given observed probes. Such probe-wise probabilities will be used as weights for aggregating permutations.

Usage

```
postProbNull(y, group, weights = NULL, alpha = "moderated",
  detail.return = TRUE)
```

Arguments

y	a numeric matrix containing log-expression or logCPM (log2-counts per million) values. Data frame or SummarizedExperiment object will be internally coerced into a matrix. Rows correspond to probes and columns to samples. Missing values are not permitted.
group	categorical vector or factor giving group membership of columns of y. At least two categories need to be presented.
weights	optional numeric matrix containing prior weights for each spot.
alpha	parameter specifying within-group variance estimator to be used. 'moderated': empirical Bayes moderated variance estimator as used in eBayes. Numeric value: a constant value added to pooled variance estimator ($\alpha + \sigma$). NULL: no estimator; all variances are set to be 1.
detail.return	a logical indicating whether more details (e.g. lfdr) will be returned.

Details

Posterior probabilities of each component null hypothesis given observed probes are estimated from ANOVA test on certain groups and local fdr. There are totally $(K-1)$ null hypotheses, where K is the number of groups.

Value

a list containing the following components:

`W`	a matrix of posterior probabilities for each component null hypothesis given an observed probe. Rows correspond to probes and columns to one hypothesis.
`label`	a vector of group labels.
`groupOrder`	a matrix with each row being group indexes ordered based on decreasing expression levels. Group indexes are positions in `label`.
`F.p.value`	a matrix with each column giving p-values corresponding to F-statistics on certain groups.
`lfdr`	a matrix with each column being local false discovery rates estimated based on one column of weighted F.p.value matrix.
`fit`	a `MArrayLM` fitted model object produced by `lmFit`.

`F.p.value`, `lfdr` and `fit` are returned only when `detail.return` is TRUE.

Examples

```
data(RocheBT)
ppnull <- postProbNull(RocheBT$y, RocheBT$group, alpha='moderated')
ppnull <- postProbNull(RocheBT$y, RocheBT$group, alpha=0.1)
```

`pvalueWeightedEst` *p-value by weighted permutation scheme*

Description

This function estimates p-values by aggregating weighted permutations.

Usage

```
pvalueWeightedEst(tt, ttperm, w)
```

Arguments

`tt`	a vector of test statistics.
`ttperm`	a matrix of test statistics from permutations. Rows correspond to probes and columns to one permutation.
`w`	a matrix containing weights for each spot in `ttperm`. Provided by `postProbNull`.

Details

P-values are estimated by weightedly accumulating test statistics from permutations that are larger than observations

Value

p-values.

Examples

```
#generate some example data
t.obs <- rnorm(100)
t.perm <- matrix(rnorm(100*1000),nrow=100)
w <- matrix(runif(100*1000),nrow=100)

pv <- pvalueWeightedEst(t.obs, t.perm, w)
```

RocheBT *mRNA expression data downsampled from GSE28490 (Roche)*

Description

Three cell subtypes (B cells, CD4+ T cells, CD8+ T cells) were isolated from 5 pools of 5 healthy donors each. RNA obtained from these 15 purified populations were subsequently used for mRNA expression profiling by HG-U133Plus2.0 microarrays. We downsample the original data to 5000 probes/probesets. Subtype labels for purified populations are also included. (Data generation script can be found in ./data_raw folder.)

Usage

```
data(RocheBT)
```

Format

A list with one expression matrix (y) and a categorical vector giving subtypes (group).

References

Allantaz et al. PLoS One 2012;7(1):e29979. PMID: 22276136

row_min *min value for each row*

Description

min value for each row

Usage

```
row_min(Y)
```

Arguments

Y	a numeric matrix

Value

a numeric vector indicating min value in each row

row_which_max *which.max for each row*

Description

which.max for each row

Usage

```
row_which_max(Y)
```

Arguments

Y	a numeric matrix

Value

a integer vector indicating the location of max value in each row

`shuffle_topm` *Shuffle the top M groups*

Description

Shuffle the top M groups

Usage

```
shuffle_topm(y, group, weights, combM, geneSubset, seed)
```

Arguments

`y`	a numeric matrix to be shuffled.
`group`	a integer vector indicating group indexes.
`weights`	optional numeric matrix containing prior weights.
`combM`	a integer matrix with each row giving one choice of M groups.
`geneSubset`	a integer vector indicating the probe pattern of combM.
`seed`	an integer seed for the random number generator.

Value

shuffled expression matrix and weight matrix in top M groups.

Index

∗ **internal**
    pairwise_tstat_unscaled, 8
    permfunc, 9
    row_min, 13
    row_which_max, 13
    shuffle_topm, 14

countBT, 3

eBayes, *4*, *6, 7*, *10*

lmFit, *6*

nullDistri, *2*, 3

OVESEG *(*OVESEG-package*)*, 2
OVESEG-package, 2
OVESEGtest, *2, 3*, 4, *9*
OVESEGtstat, *2*, 6
OVEtstatPermTopM, 7

pairwise_tstat_unscaled, 8
patternDistri, *2*, 9
permfunc, 9
postProbNull, *2, 3*, *9*, 10, *11*
pvalueWeightedEst, 11

RocheBT, 12
row_min, 13
row_which_max, 13

shuffle_topm, 14