Package 'twilight'

November 12, 2026

Version 1.88.0

Title Estimation of local false discovery rate

Description In a typical microarray setting with gene expression data observed under two conditions, the local false discovery rate describes the probability that a gene is not differentially expressed between the two conditions given its corrresponding observed score or p-value level. The resulting curve of p-values versus local false discovery rate offers an insight into the twilight zone between clear differential and clear non-differential gene expression. Package 'twilight' contains two main functions: Function twilight.pval performs a two-condition test on differences in means for a given input matrix or expression set and computes permutation based p-values. Function twilight performs a stochastic downhill search to estimate local false discovery rates and effect size distributions. The package further provides means to filter for permutations that describe the null distribution correctly. Using filtered permutations, the influence of hidden confounders could be diminished.

Depends R (>= 2.10)

Imports Biobase, graphics, grDevices, splines, stats

Suggests golubEsets (>= 1.4.2), vsn (>= 1.7.2)

License GPL (>= 2)

URL http://compdiag.molgen.mpg.de/software/twilight.shtml

biocViews Microarray, DifferentialExpression, MultipleComparison

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

git_branch RELEASE_3_23

git_last_commit 3b91675

git_last_commit_date 2026-04-28

Repository Bioconductor 3.23

Date/Publication 2026-11-11

Author Stefanie Senger [cre, aut] (ORCID:
<https://orcid.org/0000-0003-4144-1040>)

Maintainer Stefanie Senger <stefanie.scheid@gmx.de>

Contents

exfdr 2
expval 3
plot.twilight 4
print.twilight 5
twilight 6
twilight.combi 8
twilight.filtering 9
twilight.getlambda 11
twilight.getmatrix 12
twilight.permute.pair 13
twilight.permute.unpair 14
twilight.pval 15
twilight.teststat 17

Index **20**

`exfdr` *Example of twilight result*

Description

Application of function `twilight` on twilight object `data(expval)`.

The function call was `exfdr <- twilight(expval,B=1000)`.

Usage

```
data(exfdr)
```

Format

A twilight object.

References

Scheid S and Spang R (2004): A stochastic downhill search algorithm for estimating the local false discovery rate, *IEEE TCBB* **1(3)**, 98–108.

Scheid S and Spang R (2005): twilight; a Bioconductor package for estimating the local false discovery rate, *Bioinformatics* **21(12)**, 2921–2922.

Scheid S and Spang R (2006): Permutation filtering: A novel concept for significance analysis of large-scale genomic data, in: Apostolico A, Guerra C, Istrail S, Pevzner P, and Waterman M

(Eds.): *Research in Computational Molecular Biology: 10th Annual International Conference, Proceedings of RECOMB 2006, Venice, Italy, April 2-5, 2006*. Lecture Notes in Computer Science vol. 3909, Springer, Heidelberg, pp. 338-347.

`expval` *Example of twilight.pval result*

Description

Application of function `twilight.pval` on leukemia data set of Golub et al. (1999), as given in `data(Golub_Merge)` in `library(golubEsets)`.

First step was the variance-stabilizing normalization of Huber et al. (2002) in `library(vsn)`: `golubNorm <- justvsn(Golub_Merge)`.

The function call was then `expval <- twilight.pval(golubNorm,id)` with `id <- as.numeric(Golub_Merge$ALL.AML)`.

Usage

```
data(expval)
```

Format

A twilight object.

References

Golub TR, Slonim DK, Tamayo P, Huard C, Gaasenbeek M, Mesirov JP, Coller H, Loh ML, Downing JR, Caligiuri MA, Bloomfield CD and Lander ES (1999): Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring, *Science* **286**, 531–537.

Huber W, von Heydebreck A, Sultmann H, Poustka A and Vingron M (2002): Variance stabilization applied to microarray data calibration and to the quantification of differential expression, *Bioinformatics* **18**, suppl. 1, S96–S104.

Scheid S and Spang R (2004): A stochastic downhill search algorithm for estimating the local false discovery rate, *IEEE TCBB* **1(3)**, 98–108.

Scheid S and Spang R (2005): twilight; a Bioconductor package for estimating the local false discovery rate, *Bioinformatics* **21(12)**, 2921–2922.

Scheid S and Spang R (2006): Permutation filtering: A novel concept for significance analysis of large-scale genomic data, in: Apostolico A, Guerra C, Istrail S, Pevzner P, and Waterman M (Eds.): *Research in Computational Molecular Biology: 10th Annual International Conference, Proceedings of RECOMB 2006, Venice, Italy, April 2-5, 2006*. Lecture Notes in Computer Science vol. 3909, Springer, Heidelberg, pp. 338-347.

Tusher VG, Tibshirani R and Chu G (2001): Significance analysis of mircroarrays applied to the ionizing response, *PNAS* **98(9)**, 5116–5121.

plot.twilight *Plot function for twilight objects*

Description

Interface to plotting of twilight objects. Produces one of five possible plots.

Usage

```
## S3 method for class 'twilight'
plot(x, which = NULL, grayscale = FALSE, legend = TRUE, ...)
```

Arguments

x	An object of class twilight.
which	A character string specifying the plot to be made.
grayscale	Logical value. Specifying whether plots should be colored or grayscaled. Only necessary for "scores" and "fdr".
legend	Logical value. Produces a legend for "scores" and "effectsize". A legend for "fdr" is only available if bootstrap estimates exist.
...	Additional graphical arguments.

Details

Option which="scores" plots the expected versus the observed test statistics and draws confidence lines calculated from permutations. This plot is similar to plots in Tusher et al. (2001).

Option which="qvalues" plots q-values versus the number of rejected hypotheses.

Option which="fdr" plots p-values versus 1 - local false discovery rate, that is the conditional probability of being significant given the corresponding p-value, plus bootstrap estimates if computed. Bottom ticks are 1%-quantiles of p-values.

Option which="volcano" results in the volcano plot, that is observed score versus 1 - local false discovery rate. Bottom ticks are 1%-quantiles of scores.

Option which="effectsize" plots the observed fold change equivalent score distribution overlaid by the estimated effect size distribution, that is distribution of scores under the alternative. This plot is only available if function twilight.pval was applied with method="fc" as fold change equivalent scores are computed from log ratios.

Option which="table" tabulates histogram "effectsize".

Value

No value is returned except for "table".

Author(s)

Stefanie Scheid

References

Scheid S and Spang R (2004): A stochastic downhill search algorithm for estimating the local false discovery rate, *IEEE TCBB* **1(3)**, 98–108.

Scheid S and Spang R (2005): twilight; a Bioconductor package for estimating the local false discovery rate, *Bioinformatics* **21(12)**, 2921–2922.

Scheid S and Spang R (2006): Permutation filtering: A novel concept for significance analysis of large-scale genomic data, in: Apostolico A, Guerra C, Istrail S, Pevzner P, and Waterman M (Eds.): *Research in Computational Molecular Biology: 10th Annual International Conference, Proceedings of RECOMB 2006, Venice, Italy, April 2-5, 2006*. Lecture Notes in Computer Science vol. 3909, Springer, Heidelberg, pp. 338-347.

Tusher VG, Tibshirani R and Chu G (2001): Significance analysis of mircroarrays applied to the ionizing response, *PNAS* **98(9)**, 5116–5121.

See Also

`twilight.pval`, `twilight`

Examples

```
### contains a twilight object created by function twilight
data(exfdr)
plot(exfdr)
```

`print.twilight` *Print function for twilight objects*

Description

Extract and print information about a twilight object.

Usage

```
## S3 method for class 'twilight'
print(x, ...)
```

Arguments

`x`	Input object of class `twilight`.
`...`	Additional printing arguments.

Value

No value is returned.

Author(s)

Stefanie Scheid

References

Scheid S and Spang R (2004): A stochastic downhill search algorithm for estimating the local false discovery rate, *IEEE TCBB* **1(3)**, 98–108.

Scheid S and Spang R (2005): twilight; a Bioconductor package for estimating the local false discovery rate, *Bioinformatics* **21(12)**, 2921–2922.

Scheid S and Spang R (2006): Permutation filtering: A novel concept for significance analysis of large-scale genomic data, in: Apostolico A, Guerra C, Istrail S, Pevzner P, and Waterman M (Eds.): *Research in Computational Molecular Biology: 10th Annual International Conference, Proceedings of RECOMB 2006, Venice, Italy, April 2-5, 2006*. Lecture Notes in Computer Science vol. 3909, Springer, Heidelberg, pp. 338-347.

See Also

`plot.twilight`

Examples

```
### contains a twilight object created by function twilight
data(exfdr)
print(exfdr)
```

`twilight` *Estimation of the local false discovery rate*

Description

The function performs the successive exclusion procedure (SEP) as described in Scheid and Spang (2004).

Usage

```
twilight(xin, lambda = NULL, B = 0, boot.ci = 0.95, clus = NULL, verbose = TRUE)
```

Arguments

`xin`	Numerical vector of p-values or a `twilight` object.
`lambda`	Numerical value denoting the penalty factor. If not specified, the function searchs for an appropriate regularization parameter.
`B`	Numerical value specifying the number of bootstrap samples. If not specified, no bootstrap estimates are calculated.
`boot.ci`	Numerical value denoting the probability value for bootstrap confidence intervals of local false discovery rate and prior pi0.
`clus`	A list or numerical value to be passed to `makeCluster(clus,...)` in `library(snow)`. If specified, bootstrapping is performed in parallel. No checks on `clus` are performed. Please make sure that `makeCluster(clus)` works properly in your environment.
`verbose`	Logical value for message printing.

Details

In short, the successive exclusion procedure divides the set of p-values into two parts. The first part is chosen such that it resembles a uniform distribution while containing as many p-values as possible. This set of p-values represents the set of p-values derived from non-induced genes. The height of the uninform distribution is a natural estimate for the mixture parameter pi0. The p-value not contained in the uniform part serve as representatives of p-values derived from induced genes. Their distribution is the basis of the final estimate of the local false discovery rate.

NOTE: Library(snow) has to be loaded manually. It is not loaded as part of 'suggests' or 'depends' because it is only available under UNIX. If twilight does not work with the current version of snow, please send a report.

Value

Returns a `twilight` object consisting of a `data.frame` named `result` with variables

`pvalue`	Sorted input vector.
`qvalue`	q-values computed as described in Storey and Tibshirani (2003) with new estimate `pi0`.
`index`	Index of the original ordering.
`fdr`	Local false discovery rate averaged over 10 runs of SEP.
`mean.fdr`	Bootstrap estimate of local false discovery rate.
`lower.fdr`	Lower `boot.ci`-bootstrap confidence bound.
`upper.fdr`	Upper `boot.ci`-bootstrap confidence bound.

Values are sorted by `pvalue`.

Note

Additional output consists of

`lambda`	Regularization parameter.
`pi0`	SEP estimate of prior probability.
`boot.pi0`	Bootstrap estimate and `boot.ci`-bootstrap confidence bounds.
`boot.ci`	Passes `boot.ci` for plotting.
`effect`	Histogram of effect size distributions averaged over 10 runs of SEP.

If `xin` is of class `twilight`, the remaining slots are filled with corresponding input values. If `xin` is not of class `twilight`, these slots remain empty.

Author(s)

Stefanie Scheid

References

Scheid S and Spang R (2004): A stochastic downhill search algorithm for estimating the local false discovery rate, *IEEE TCBB* **1(3)**, 98–108.

Scheid S and Spang R (2005): twilight; a Bioconductor package for estimating the local false discovery rate, *Bioinformatics* **21(12)**, 2921–2922.

Scheid S and Spang R (2006): Permutation filtering: A novel concept for significance analysis of large-scale genomic data, in: Apostolico A, Guerra C, Istrail S, Pevzner P, and Waterman M (Eds.): *Research in Computational Molecular Biology: 10th Annual International Conference, Proceedings of RECOMB 2006, Venice, Italy, April 2-5, 2006*. Lecture Notes in Computer Science vol. 3909, Springer, Heidelberg, pp. 338-347.

Storey JD and Tibshirani R (2003): Statistical significance for genomewide studies, *PNAS* **100(16)**, 9440–9445.

See Also

`twilight.pval`, `plot.twilight`, `exfdr`

Examples

```
### twilight object created with B=1000 bootstrap samples
data(exfdr)
print(exfdr)
plot(exfdr)
```

`twilight.combi` *All permutations of a binary vector*

Description

For a given binary input vector, the function completely enumerates all possible permutations.

Usage

```
twilight.combi(xin, pin, bin)
```

Arguments

`xin`	Binary input vector, e.g. class labels.
`pin`	Logical value. `TRUE` if samples are paired, `FALSE` if not.
`bin`	Logical value. `TRUE` if permutations should be balanced, `FALSE` if not.

Details

Please note, that the resulting permutations are always as "balanced" as possible. The balancing is done for the smaller subsample. If its sample size is odd, say 5, `twilight.combi` computes all permutations with 2 or 3 samples unchanged. In the paired case, the output matrix contains only one half of all permutations. The second half is simply 1-`output` which leads to the same absolute test statistics in a paired test.

Value

Returns a matrix where each row contains one permuted vector. Note that even for balanced permutations, the first row always contains the original vector. If the number of rows exceeds 10000, NULL is returned.

Author(s)

Stefanie Scheid

References

Scheid S and Spang R (2004): A stochastic downhill search algorithm for estimating the local false discovery rate, *IEEE TCBB* **1(3)**, 98–108.

Scheid S and Spang R (2005): twilight; a Bioconductor package for estimating the local false discovery rate, *Bioinformatics* **21(12)**, 2921–2922.

Scheid S and Spang R (2006): Permutation filtering: A novel concept for significance analysis of large-scale genomic data, in: Apostolico A, Guerra C, Istrail S, Pevzner P, and Waterman M (Eds.): *Research in Computational Molecular Biology: 10th Annual International Conference, Proceedings of RECOMB 2006, Venice, Italy, April 2-5, 2006*. Lecture Notes in Computer Science vol. 3909, Springer, Heidelberg, pp. 338-347.

See Also

twilight.permute.pair, twilight.permute.unpair

Examples

```
x <- c(rep(0,4),rep(1,3))
y <- twilight.combi(x,pin=FALSE,bin=FALSE)
```

---

twilight.filtering *Permutation filtering*

Description

The function call invokes the filtering for permutations of class labels that produce a set of complete null scores. Depending on the test setting, the algorithm iteratively generates valid permutations of the class labels and computes scores. These are transformed to pooled p-values and each set of permutation p-values is tested for uniformity. Permutations with acceptable uniform p-value distributions are kept. The search stops if either the number num.perm of wanted permutations is reached or if the number of possible unique(!) permutations is smaller than num.perm. The default values are similar to function twilight.pval but please note the details below.

Usage

```
twilight.filtering(xin, yin, method = "fc", paired = FALSE, s0 = 0, verbose = TRUE, num.perm = 1000, num.
```

Arguments

xin	Either an expression set (ExpressionSet) or a data matrix with rows corresponding to features and columns corresponding to samples.
yin	A numerical vector containing class labels. The higher label denotes the case, the lower label the control samples to test case vs. control. For correlation scores, yin can be any numerical vector of length equal to the number of samples.
method	Character string: "fc" for fold change equivalent test (that is log ratio test), "t" for t-test, and "z" for Z-test. With "pearson" or "spearman", the test statistic is either Pearson's correlation coefficient or Spearman's rank correlation coefficient.
paired	Logical value. Depends on whether the samples are paired. Ignored if method="pearson" or method="spearman".
s0	Fudge factor for variance correction in the Z-test. Takes effect only if method="z". If s0=0: The fudge factor is set to the median of the pooled standard deviations.
verbose	Logical value for message printing.
num.perm	Number of permutations. Within twilight.pval, num.perm is set to B.
num.take	Number of permutations kept in each step of the iterative filtering. Within twilight.pval, num.take is set to the minimum of 50 and ceiling(num.perm/20).

Details

See vignette.

Value

Returns a matrix with permuted input labels yin as rows. Please note that this matrix is already translated into binary labels for two-sample testing or to ranks if Spearman's correlation was chosen. The resulting permutation matrix can be directly passed into function twilight.pval. Please note that the first row always contains the original input yin to be consistent with the other permutation functions in package twilight.

Author(s)

Stefanie Scheid

References

Scheid S and Spang R (2004): A stochastic downhill search algorithm for estimating the local false discovery rate, *IEEE TCBB* **1(3)**, 98–108.

Scheid S and Spang R (2005): twilight; a Bioconductor package for estimating the local false discovery rate, *Bioinformatics* **21(12)**, 2921–2922.

Scheid S and Spang R (2006): Permutation filtering: A novel concept for significance analysis of large-scale genomic data, in: Apostolico A, Guerra C, Istrail S, Pevzner P, and Waterman M (Eds.): *Research in Computational Molecular Biology: 10th Annual International Conference, Proceedings of RECOMB 2006, Venice, Italy, April 2-5, 2006*. Lecture Notes in Computer Science vol. 3909, Springer, Heidelberg, pp. 338-347.

See Also

twilight.pval

Examples

```
## Not run:
### Leukemia data set of Golub et al. (1999).
library(golubEsets)
data(Golub_Train)

### Variance-stabilizing normalization of Huber et al. (2002).
library(vsn)
golubNorm <- justvsn(Golub_Train)

### A binary vector of class labels.
id <- as.numeric(Golub_Train$ALL.AML)

### Do an unpaired t-test.
### Let's have a quick example with 50 filtered permutations only.
### With num.take=10, we only need 5 iteration steps.
yperm <- twilight.filtering(golubNorm,id,method="t",num.perm=50,num.take=10)
dim(yperm)

### Let's check that the filtered permutations really produce uniform p-value distributions.
### The first row is the original labeling, so we try the second permutation.
yperm <- yperm[-1,]
b <- twilight.pval(golubNorm,yperm[1,],method="t",yperm=yperm)
hist(b$result$pvalue)

## End(Not run)
```

---

twilight.getlambda *Internal function to compute regularization parameter lambda*

Description

The SEP method optionally includes a penalty term in order to estimate the percentage of non-significant differences/genes.

Usage

```
twilight.getlambda(xin, verbose = TRUE)
```

Arguments

xin	A numerical vector containing p-values.
verbose	Logical value for message printing.

Details

The function finds a suitable regularization parameter λ. For a sequence of lambdas, the objective function of SEP is computed for subsamples of the $p - value$ vector `xin`. The final estimate is chosen based upon a Wilcoxon test comparison between objective function values of $\lambda = 0$ and each $\lambda > 0$. The penalized objective function should not differ a lot from the unpenalized one. Therefore, the highest lambda that leads to a non-significant difference in means is chosen.

Value

Returns numerical value.

Author(s)

Stefanie Scheid

References

Scheid S and Spang R (2004): A stochastic downhill search algorithm for estimating the local false discovery rate, *IEEE TCBB* **1(3)**, 98–108.

Scheid S and Spang R (2005): twilight; a Bioconductor package for estimating the local false discovery rate, *Bioinformatics* **21(12)**, 2921–2922.

Scheid S and Spang R (2006): Permutation filtering: A novel concept for significance analysis of large-scale genomic data, in: Apostolico A, Guerra C, Istrail S, Pevzner P, and Waterman M (Eds.): *Research in Computational Molecular Biology: 10th Annual International Conference, Proceedings of RECOMB 2006, Venice, Italy, April 2-5, 2006*. Lecture Notes in Computer Science vol. 3909, Springer, Heidelberg, pp. 338-347.

See Also

`twilight`

---

twilight.getmatrix *Internal function to extract the gene expression matrix from an expression set*

Description

If the input object is of class `ExpressionSet`, the function extracts the gene expression matrix.

Usage

```
twilight.getmatrix(xin)
```

Arguments

`xin`	Input object of class `ExpressionSet`.

Value

Returns the matrix of gene expression values.

Author(s)

Stefanie Scheid

References

Scheid S and Spang R (2004): A stochastic downhill search algorithm for estimating the local false discovery rate, *IEEE TCBB* **1(3)**, 98–108.

Scheid S and Spang R (2005): twilight; a Bioconductor package for estimating the local false discovery rate, *Bioinformatics* **21(12)**, 2921–2922.

Scheid S and Spang R (2006): Permutation filtering: A novel concept for significance analysis of large-scale genomic data, in: Apostolico A, Guerra C, Istrail S, Pevzner P, and Waterman M (Eds.): *Research in Computational Molecular Biology: 10th Annual International Conference, Proceedings of RECOMB 2006, Venice, Italy, April 2-5, 2006*. Lecture Notes in Computer Science vol. 3909, Springer, Heidelberg, pp. 338-347.

See Also

`twilight.pval`

---

twilight.permute.pair *Permutation matrix of paired class labels*

Description

The function returns a matrix where each row is a (un)balanced permutation of the input twosample class labels.

Usage

```
twilight.permute.pair(v, m, bal = TRUE)
```

Arguments

`v`	A binary vector representing class labels in original order. Pairs must be in the same order.
`m`	A numerical value giving the number of permutations.
`bal`	Logical value. Results in balanced or unbalanced permutations.

Value

Returns a matrix where each row contains one permuted vector of class labels. Note that even for balanced permutations, the first row always contains the original vector.

Author(s)

Stefanie Scheid

References

Scheid S and Spang R (2004): A stochastic downhill search algorithm for estimating the local false discovery rate, *IEEE TCBB* **1(3)**, 98–108.

Scheid S and Spang R (2005): twilight; a Bioconductor package for estimating the local false discovery rate, *Bioinformatics* **21(12)**, 2921–2922.

Scheid S and Spang R (2006): Permutation filtering: A novel concept for significance analysis of large-scale genomic data, in: Apostolico A, Guerra C, Istrail S, Pevzner P, and Waterman M (Eds.): *Research in Computational Molecular Biology: 10th Annual International Conference, Proceedings of RECOMB 2006, Venice, Italy, April 2-5, 2006*. Lecture Notes in Computer Science vol. 3909, Springer, Heidelberg, pp. 338-347.

See Also

`twilight.permute.unpair`, `twilight.combi`

`twilight.permute.unpair`

Permutation matrix of unpaired class labels

Description

The function returns a matrix where each row is a (un)balanced permutation of the input twosample class labels.

Usage

```
twilight.permute.unpair(v, m, bal = TRUE)
```

Arguments

`v`	A binary vector representing class labels in original order.
`m`	A numerical value giving the number of permutations.
`bal`	Logical value. Results in balanced or unbalanced permutations.

Value

Returns a matrix where each row contains one permuted vector of class labels. Note that even for balanced permutations, the first row always contains the original vector.

Author(s)

Stefanie Scheid

References

Scheid S and Spang R (2004): A stochastic downhill search algorithm for estimating the local false discovery rate, *IEEE TCBB* **1(3)**, 98–108.

Scheid S and Spang R (2005): twilight; a Bioconductor package for estimating the local false discovery rate, *Bioinformatics* **21(12)**, 2921–2922.

Scheid S and Spang R (2006): Permutation filtering: A novel concept for significance analysis of large-scale genomic data, in: Apostolico A, Guerra C, Istrail S, Pevzner P, and Waterman M (Eds.): *Research in Computational Molecular Biology: 10th Annual International Conference, Proceedings of RECOMB 2006, Venice, Italy, April 2-5, 2006*. Lecture Notes in Computer Science vol. 3909, Springer, Heidelberg, pp. 338-347.

See Also

`twilight.permute.pair`, `twilight.combi`

`twilight.pval` *Compute p-values from expression sets*

Description

A function to compute two-sample t, Z and fold change equivalent test statistics (paired or unpaired) and correlation coefficients. Based on permutations, expected test statistics as given in Tusher et al. (2001) and empirical p-values are computed. Additional output are q-values computed as given in Storey and Tibshirani (2003). The resulting object is of class `twilight` and can be passed to functions `twilight` or `plot.twilight`.

Usage

```
twilight.pval(xin, yin, method = "fc", paired = FALSE, B = 1000, yperm = NULL, balance = FALSE, quant.ci
```

Arguments

`xin`	Either an expression set (`ExpressionSet`) or a data matrix with rows corresponding to features and columns corresponding to samples.
`yin`	A numerical vector containing class labels. The higher label denotes the case, the lower label the control samples to test case vs. control. For correlation scores, `yin` can be any numerical vector of length equal to the number of samples.
`method`	Character string: `"fc"` for fold change equivalent test (that is log ratio test), `"t"` for t-test, and `"z"` for Z-test. With `"pearson"` or `"spearman"`, the test statistic is either Pearson's correlation coefficient or Spearman's rank correlation coefficient.
`paired`	Logical value. Depends on whether the samples are paired. Ignored if `method="pearson"` or `method="spearman"`.
`B`	Numerical value specifying the number of permutations.

yperm	Optional matrix containing in each row a permutation of the class labels in binary(!) format for two-sample testing. For computation of correlation scores, the rows of yperm have to contain the appropriate values or ranks. Use this argument carefully! If yperm is specified, no other permutation will be done. Please note that the first row of yperm MUST be the input vector yin. Otherwise, the p-value calculation will be incorrect.
balance	Logical value. Depends on whether balanced or unbalanced permutations should be done. Ignored if method="pearson" or method="spearman".
quant.ci	Probability value for confidence lines. Lines are symmetric and denote the quant.ci-quantile of maximal absolute differences between each permutatin and the expected scores.
s0	Fudge factor for variance correction in the Z-test. Takes effect only if method="z". If s0=NULL: The fudge factor is set to the median of the pooled standard deviations.
verbose	Logical value for message printing.
filtering	Logical value for filtering for permutations of class labels that produce a set of complete null scores. Invokes function twilight.filtering. If yperm is specified, no filtering will be done. Note that the filtering is done on unbalanced permutations even if balance=TRUE.

Details

Please see vignette for detailed information.

Value

Returns a twilight object consisting of a data.frame named result with variables

observed	Observed test statistics.
expected	Mean of order statistics of the permutation statistics.
candidate	Binary vector. "1" for observations exceeding the confidence lines.
pvalue	Empirical p-values from two-sided hypothesis tests.
qvalue	q-values computed as described in Storey and Tibshirani (2003).
index	Index of the original ordering.

Values are sorted by absolute observed scores.

Note

Additional output consists of

ci.line	Quantile corresponding to quant.ci, passed for plotting.
pi0	Estimated prior probability.
s0	Estimated fudge factor if method="z".
call	Character string of function arguments.
quant.ci	Passes quant.ci for plotting.

The remaining slots are left empty for function `twilight`.

Author(s)

Stefanie Scheid

References

Scheid S and Spang R (2004): A stochastic downhill search algorithm for estimating the local false discovery rate, *IEEE TCBB* **1(3)**, 98–108.

Scheid S and Spang R (2005): twilight; a Bioconductor package for estimating the local false discovery rate, *Bioinformatics* **21(12)**, 2921–2922.

Scheid S and Spang R (2006): Permutation filtering: A novel concept for significance analysis of large-scale genomic data, in: Apostolico A, Guerra C, Istrail S, Pevzner P, and Waterman M (Eds.): *Research in Computational Molecular Biology: 10th Annual International Conference, Proceedings of RECOMB 2006, Venice, Italy, April 2-5, 2006*. Lecture Notes in Computer Science vol. 3909, Springer, Heidelberg, pp. 338-347.

Storey JD and Tibshirani R (2003): Statistical significance for genomewide studies, *PNAS* **100(16)**, 9440–9445.

Tusher VG, Tibshirani R and Chu G (2001): Significance analysis of mircroarrays applied to the ionizing response, *PNAS* **98(9)**, 5116–5121.

See Also

`twilight`, `plot.twilight`, `twilight.combi`, `twilight.filtering`, `expval`, `twilight.teststat`

Examples

```
### twilight object created from Golub data set
data(expval)
print(expval)
plot(expval)
```

`twilight.teststat` *Interface to the test statistics provided within 'twilight'*

Description

A function to compute two-sample t, Z and fold change equivalent test statistics (paired or unpaired) and correlation coefficients.

Usage

```
twilight.teststat(xin, yin, method = "fc", paired = FALSE, s0 = NULL)
```

Arguments

xin	Either an expression set (ExpressionSet) or a data matrix with rows corresponding to features and columns corresponding to samples.
yin	A numerical vector containing class labels. The higher label denotes the case, the lower label the control samples to test case vs. control. For correlation scores, yin can be any numerical vector of length equal to the number of samples.
method	Character string: "fc" for fold change equivalent test (that is log ratio test), "t" for t-test, and "z" for Z-test of Tusher et al. (2001). With "pearson" or "spearman", the test statistic is either Pearson's correlation coefficient or Spearman's rank correlation coefficient.
paired	Logical value. Depends on whether the samples are paired. Ignored if method="pearson" or method="spearman".
s0	Fudge factor for variance correction in the Z-test. Takes effect only if method="z". If s0=NULL: The fudge factor is set to the median of the pooled standard deviations.

Details

Please see vignette for detailed information.

Value

Returns a list with two components: a numerical vector of observed test statistics observed. Each entry corresponds to one row of the input data matrix. Also, the estimated fudge factor s0 is returned. In any other case except method="z", s0 is zero.

Author(s)

Stefanie Scheid

References

Scheid S and Spang R (2004): A stochastic downhill search algorithm for estimating the local false discovery rate, *IEEE TCBB* **1(3)**, 98–108.

Scheid S and Spang R (2005): twilight; a Bioconductor package for estimating the local false discovery rate, *Bioinformatics* **21(12)**, 2921–2922.

Scheid S and Spang R (2006): Permutation filtering: A novel concept for significance analysis of large-scale genomic data, in: Apostolico A, Guerra C, Istrail S, Pevzner P, and Waterman M (Eds.): *Research in Computational Molecular Biology: 10th Annual International Conference, Proceedings of RECOMB 2006, Venice, Italy, April 2-5, 2006*. Lecture Notes in Computer Science vol. 3909, Springer, Heidelberg, pp. 338-347.

Tusher VG, Tibshirani R and Chu G (2001): Significance analysis of mircroarrays applied to the ionizing response, *PNAS* **98(9)**, 5116–5121.

See Also

twilight.pval

Examples

```
### Z-test on random values
M <- matrix(rnorm(20000),nrow=1000)
id <- c(rep(1,10),rep(0,10))
stat <- twilight.teststat(M,id,method="z")

### Pearson correlation
id <- 1:20
stat <- twilight.teststat(M,id,method="pearson")
```

Index

∗ **datagen**
twilight.combi, 8
twilight.filtering, 9
twilight.permute.pair, 13
twilight.permute.unpair, 14
∗ **datasets**
exfdr, 2
expval, 3
∗ **hplot**
plot.twilight, 4
∗ **internal**
twilight.getlambda, 11
twilight.getmatrix, 12
∗ **nonparametric**
twilight, 6
twilight.pval, 15
twilight.teststat, 17
∗ **print**
print.twilight, 5

exfdr, 2, *8*
expval, 3, *17*

plot.twilight, 4, *6*, *8*, *17*
print.twilight, 5

twilight, *5*, 6, *12*, *17*
twilight.combi, 8, *14, 15*, *17*
twilight.filtering, 9, *17*
twilight.getlambda, 11
twilight.getmatrix, 12
twilight.permute.pair, *9*, 13, *15*
twilight.permute.unpair, *9*, *14*, 14
twilight.pval, *5*, *8*, *11*, *13*, 15, *18*
twilight.teststat, *17*, 17