

dFDRscore

June 4, 2015

dFDRscore	<i>Function to transform <code>fdr</code> into scores according to log-likelihood ratio between the true positives and the false positivies and/or after controlling false discovery rate</i>
-----------	---

Description

dFDRscore is supposed to take as input a vector of `fdr`, which are transformed into scores according to according to log-likelihood ratio between the true positives and the false positivies. Also if the FDR threshold is given, it is used to make sure that `fdr` below threshold are considered significant and thus scored positively. Instead, those `fdr` above the given threshold are considered insignificant and thus scored negatively.

Usage

```
dFDRscore(fdr, fdr.threshold = NULL, scatter = F)
```

Arguments

<code>fdr</code>	a vector containing a list of input <code>fdr</code>
<code>fdr.threshold</code>	the given FDR threshold. By default, it is set to <code>NULL</code> , meaning there is no constraint. If given, those <code>fdr</code> with the FDR below threshold are considered significant and thus scored positively. Instead, those <code>fdr</code> with the FDR above given threshold are considered insignificant and thus scored negatively
<code>scatter</code>	logical to indicate whether the scatter graph of scores against p-values should be drawn. Also indicated is the score corresponding to the given FDR threshold (if any)

Value

- `scores`: a vector of scores

Note

none

See Also

[dSVDsignif](#), [dNetPipeline](#)

Examples

```
# 1) generate data with an iid matrix of 1000 x 9
data <- cbind(matrix(rnorm(1000*3,mean=0,sd=1), nrow=1000, ncol=3),
matrix(rnorm(1000*3,mean=0.5,sd=1), nrow=1000, ncol=3),
matrix(rnorm(1000*3,mean=-0.5,sd=1), nrow=1000, ncol=3))

# 2) calculate the significance according to SVD
# using "fdr" significance
fdr <- dSVDsignif(data, signif="fdr", num.permutation=10)

# 3) calculate the scores according to the fitted BUM and fdr=0.01
# no fdr threshold
scores <- dFDRscore(fdr)
# using fdr threshold of 0.01
scores <- dFDRscore(fdr, fdr.threshold=0.1, scatter=TRUE)
```