Package 'pwrFDR'

January 14, 2025

Version 3.2.4

Title FDR Power

Imports flextable, ggplot2, methods, mvtnorm, stats, stringr, TableMonster

Description Computing Average and TPX Power under various BHFDR type sequential procedures. All of these procedures involve control of some summary of the distribution of the FDP, e.g. the proportion of discoveries which are false in a given experiment. The most widely known of these, the BH-FDR procedure, controls the FDR which is the mean of the FDP. A lesser known procedure, due to Lehmann and Romano, controls the FDX, or probability that the FDP exceeds a user provided threshold. This is less conservative than FWE control procedures but much more conservative than the BH-FDR proceudre. This package and the references supporting it introduce a new procedure for controlling the FDX which we call the BH-FDX procedure. This procedure iteratively identifies, given alpha and lower threshold delta, an alpha* less than alpha at which BH-FDR guarantees FDX control. This uses asymptotic approximation and is only slightly more conservative than the BH-FDR procedure. Likewise, we can think of the power in multiple testing experiments in terms of a summary of the distribution of the True Positive Proportion (TPP), the portion of tests truly non-null distributed that are called significant. The package will compute power, sample size or any other missing parameter required for power defined as (i) the mean of the TPP (average power) or (ii) the probability that the TPP exceeds a given value, lambda, (TPX power) via asymptotic approximation. All supplied theoretical results are also obtainable via simulation. The suggested approach is to narrow in on a design via the theoretical approaches and then make final adjustments/verify the results by simulation. The theoretical results are described in Izmirlian, G (2020) Statistics and Probability letters, ``<doi:10.1016/j.spl.2020.108713>", and an applied paper describing the methodology with a simulation study is in preparation.
See citation(``pwrFDR").

License GPL (>= 2)

LazyLoad yes

NeedsCompilation no

Author Grant Izmirlian [aut, cre]

Maintainer Grant Izmirlian <izmirlig@mail.nih.gov>

Repository CRAN

Date/Publication 2025-01-14 17:10:02 UTC

Contents

`arg.vals` *Extracts the full argument list and call attribute.*

Description

Extracts the full argument list and call attribute from an object of class `pwr`, which is the result of a call to `pwrFDR`.

Usage

```
arg.vals(object)
```

Arguments

`object`	An object of class 'pwr', which is the result of a call to `pwrFDR`

Value

A list with a `call` component and one component for each of the possible arguments, `effect.size`, `n.sample`,`r.1`, `alpha`, `N.tests`, `lambda`, `FDP.control.method`, `delta`, `groups`, `type`, `grpj.per.grp1`, `method` and `control`, with defaults filled in.

Author(s)

Grant Izmirlian <izmirlian at nih dot gov>

References

Izmirlian G. (2020) Strong consistency and asymptotic normality for quantities related to the Benjamini-Hochberg false discovery rate procedure. Statistics and Probability Letters; 108713, <doi:10.1016/j.spl.2020.108713>.

Izmirlian G. (2017) Average Power and λ-power in Multiple Testing Scenarios when the Benjamini-Hochberg False Discovery Rate Procedure is Used. <arXiv:1801.03989>

Jung S-H. (2005) Sample size for FDR-control in microarray data analysis. Bioinformatics; 21:3097-3104.

Liu P. and Hwang J-T. G. (2007) Quick calculation for sample size while controlling false discovery rate with application to microarray analysis. Bioinformatics; 23:739-746.

Lehmann E. L., Romano J. P.. Generalizations of the familywise error rate. Ann. Stat.. 2005;33(3):1138-1154.

Romano Joseph P., Shaikh Azeem M.. Stepup procedures for control of generalizations of the familywise error rate. Ann. Stat.. 2006;34(4):1850-1873.

Examples

```
rslt <- pwrFDR(effect.size = 0.79, n.sample = 46, r.1 = 2000/54675, alpha = 0.15,
               N.tests = 1000, FDP.control.method = "Auto")

arg.vals(rslt)
```

`backsolve.seFDPoalpha` *Find missing argument giving required se[FDP]/alpha (or se[TPP]/average.power)*

Description

backsolve.seFDPoalpha finds the missing argument, one of 'N.tests', 'r.1', 'n.sample' or 'effect size' giving the specified value of se[FDP]/alpha under the BH-FDR procedure.

backsolve.seTPPoavgpwr finds the missing argument, one of 'N.tests', 'r.1', 'n.sample' or 'effect size' giving the specified value of se[TPP]/average.power under the BH-FDR procedure.

Usage

```
backsolve.seFDPoalpha(seFDPoalpha, effect.size, n.sample, r.1, alpha, groups = 2, N.tests,
                  type = "balanced", grpj.per.grp1 = 1, distopt = 1, rho, k.bs)

backsolve.seTPPoavgpwr(seTPPoavgpwr, effect.size, n.sample, r.1, alpha, groups = 2,
                N.tests, type = "balanced", grpj.per.grp1 = 1, distopt = 1, rho,
                     k.bs)
```

Arguments

`seFDPoalpha`	In backsolve.seFDPoalpha, the user specified value of se[FDP]/alpha
`seTPPoavgpwr`	In backsolve.seTPPoavgpwr, the user specified value of se[TPP]/average.power
`effect.size`	The effect size (mean over standard deviation) for test statistics having non-zero means. Assumed to be a constant (in magnitude) over non-zero mean test statistics.
`n.sample`	The number of experimental replicates. Required for calculation of power
`r.1`	The proportion of simultaneous tests that are non-centrally located
`alpha`	The false discovery rate (in the BH case) or the upper bound on the probability that the FDP exceeds delta (BHFDX and Romano case)
`groups`	The number of experimental groups to compare. Must be integral and >=1. The default value is 2.
`N.tests`	The number of simultaneous hypothesis tests.
`type`	A character string specifying, in the groups=2 case, whether the test is 'paired', 'balanced', or 'unbalanced' and in the case when groups >=3, whether the test is 'balanced' or 'unbalanced'. The default in all cases is 'balanced'. Left unspecified in the one sample (groups=1) case.
`grpj.per.grp1`	Required when type="unbalanced", specifies the group 0 to group 1 ratio in the two group case, and in the case of 3 or more groups, the group j to group 1 ratio, where group 1 is the group with the largest effect under the alternative hypothesis.

distopt	Test statistic distribution in among null and alternatively distributed sub-populations. distopt=0 gives normal (2 groups), distop=1 gives t- (2 groups) and distopt=2 gives F- (2+ groups)
rho	This can be done under the assumption of tests that are correlated identically in pair within blocks of given size.
k.bs	When 'rho' is specified, the common block-size for correlated test statistics.

Value

A numeric vector having components

<missing argument>	Value of missing argument giving required se[FDP]/alpha (backsolve.seFDPoalpha) or se[TPP]/average.power (backsolve.seTPPoavgpwr).
average.power	The average power at the given set of conditions
se.VoR/se.ToM	The standard error of the FDP (backsolve.seFDPoalpha) or standard error of the TPP (backsolve.seTPPoavgpwr).
value	Value returned by the solver. Should be near zero if a solution was found.

Author(s)

Grant Izmirlian Jr <izmirlian at nih dot gov>

References

Izmirlian G. (2020) Strong consistency and asymptotic normality for quantities related to the Benjamini-Hochberg false discovery rate procedure. Statistics and Probability Letters; 108713, <doi:10.1016/j.spl.2020.108713>

Izmirlian G. (2017) Average Power and λ-power in Multiple Testing Scenarios when the Benjamini-Hochberg False Discovery Rate Procedure is Used. <arXiv:1801.03989>

Jung S-H. (2005) Sample size for FDR-control in microarray data analysis. Bioinformatics; 21:3097-3104.

Kluger D. M., Owen A. B. (2023) A central limit theorem for the Benjamini-Hochberg false discovery proportion under a factor model. Bernoulli; xx:xxx-xxx.

Liu P. and Hwang J-T. G. (2007) Quick calculation for sample size while controlling false discovery rate with application to microarray analysis. Bioinformatics; 23:739-746.

Lehmann E. L., Romano J. P.. Generalizations of the familywise error rate. Ann. Stat.. 2005;33(3):1138-1154.

Romano Joseph P., Shaikh Azeem M.. Stepup procedures for control of generalizations of the familywise error rate. Ann. Stat.. 2006;34(4):1850-1873.

Examples

```
backsolve.seFDPoalpha(seFDPoalpha=0.50, n.sample=50, alpha=0.05, effect.size=0.8,
                      r.1=0.20)

backsolve.seTPPoavgpwr(seTPPoavgpwr=0.20, n.sample=30, alpha=0.05, effect.size=0.8,
                       r.1=0.20)
```

`basic.tmPrint` *Wrapper to Print a Basic Nicely Formatted Table*

Description

Creates a generic call to print.TableMonster which in turn calls xtable

Usage

```
basic.tmPrint(x, special = NULL, simple = FALSE, dbg = FALSE, ...)
```

Arguments

`x`	Any data.frame object. Here, the result of a call to pwrFDR.
`special`	Special arguments to print.TableMonster. See package documentation.
`simple`	The simplest use case
`dbg`	Set to a value >= 1 for debugging
`...`	Other arguments

Value

The value returned is an invisible version of the argument 'x'.

Author(s)

Grant Izmirlian

`cc.ROC` *Computes the optimal number of controls per case in hypothesis tests involving the ROC. Included here with the intent that it can be used in conjunction with* `pwrFDR` *to allow power/sample size calculation for multiple tests of ROC curve based hypothesis. See details.*

Description

In hypothesis tests of TPR_1 vs TPR_0 at fixed FPR, or FPR_1 vs FPR_0 at fixed TPR, this computes the optimal number of controls per case. Required by `es.ROC`

Usage

```
cc.ROC(FPR0, FPR1 = NULL, TPR0, TPR1 = NULL, b = NULL)
```

Arguments

FPR0	When the TPR is fixed, the FPR under the null. Otherwise the fixed FPR.
FPR1	When the TPR is fixed, the FPR under the alternative. Otherwise left blank.
TPR0	When the FPR is fixed, the TPR under the null. Otherwise the fixed TPR.
TPR1	When the FPR is fixed, the TPR under the alternative. Otherwise left blank.
b	Nominal slope of the ROC at FPR0. Taken to be 1 by default.

Value

The optimal number of controls per case.

Author(s)

Grant Izmirlian <izmirlian at nih dot gov>

References

Pepe M. S., Feng Z, Janes, H Bossuyt P. M. and Potter J. D. Pivotal evaluation of the accuracy of a biomarker used for classification or prediction. Supplement. J Natl Cancer Inst 2008;100: 1432–1438

See Also

es.ROC

Examples

```
cc.ROC(FPR0=0.15, TPR0=0.80, TPR1=0.90)
```

---

cCDF.Rom *Computes the complimentary CDF for the significant call proportion, R_m/m.*

Description

Computes the complimentary CDF for the significant call proportion, R_m/m via asymptotic approximation. Included here mainly for pedagogic purposes.

Usage

```
cCDF.Rom(u, effect.size, n.sample, r.1, alpha, delta, groups = 2, N.tests,
         type = c("paired", "balanced", "unbalanced"), grpj.per.grp1 = NULL,
         FDP.control.method = "BHFDR", distopt,
      control=list(tol=1e-08,max.iter=c(1000,20),sim.level=2,low.power.stop=TRUE,
                   FDP.meth.thresh=FDP.cntl.mth.thrsh.def,verb=FALSE))
```

Arguments

u	A sorted vector of values on the interval [0, 1] for which the cCDF of R_m/m should be computed.
effect.size	The effect size (mean over standard deviation) for test statistics having non-zero means. Assumed to be a constant (in magnitude) over non-zero mean test statistics.
n.sample	The number of experimental replicates. Required for calculation of power
r.1	The proportion of simultaneous tests that are non-centrally located
alpha	The false discovery rate (in the BH case) or the upper bound on the probability that the FDP exceeds delta (Romano case)
delta	If the "FDP.control.method" is set to 'Romano' or 'BHFDX', then the user can set the exceedance thresh-hold for the FDP tail probability control $P\{FDP > \delta\} < \alpha$. The default value is α.
groups	The number of experimental groups to compare. Must be integral and >=1. The default value is 2.
N.tests	The number of simultaneous hypothesis tests.
type	A character string specifying, in the groups=2 case, whether the test is 'paired', 'balanced', or 'unbalanced' and in the case when groups >=3, whether the test is 'balanced' or 'unbalanced'. The default in all cases is 'balanced'. Left unspecified in the one sample (groups=1) case.
grpj.per.grp1	Required when type="unbalanced", specifies the group 0 to group 1 ratio in the two group case, and in the case of 3 or more groups, the group j to group 1 ratio, where group 1 is the group with the largest effect under the alternative hypothesis.
FDP.control.method	A character string specifying how the false discovery proportion (FDP) is to be controlled. You may specify the whole word or any shortened uniquely identifying truncation. "BHFDR": the usual BH-FDR "BHFDX": use asymptotic approximation to the distribution of the FDP to find a smaller FDR which guarantees probability less than alpha that the FDP exceeds alpha. "Romano": use Romano's method which guarantees probability less than alpha that the FDP exceeds alpha. "Auto": in 'FixedPoint' mode, the program will use its own wisdom to determine which choice above to make. The order of conservatism is Romano > BHFDX > BHFDR, but BHFDR offers only expected control while the other two guarantee bounds on the excedance probabilty. If the distribution of the FDP is nearly degenerate, then BHFDR is the best option. Otherwise, if it can be reliably used, BHFDX would be the best choice. The 'effective' denominator, gamma*N.tests, in the CLT determines when the approximation is good enough and the asymptotic standard error of the FDP determines when the distribution is dispersed enough to matter. Use "Auto" to run through these checks and determine the best. A return argument, 'Auto', displays the choice made. See output components and details.

"both": in 'simulation' mode, compute statistics R and T under BHFDX and Romano (in addition to BHFDR). Corresponding statistics are denoted R.st, T.st corresponding to BHFDX control of the FDP, and R.R and T.R corresponding to Romano control of the FDP. If sim.level is set to 2, (default) the statistics R.st.ht and T.st.ht, which are the number rejected and number true positives under BHFDX where r_0 = 1-r_1, gamma, and alpha.star have been estimated from the P-value data and then alpha.star computed from these.

`distopt` Test statistic distribution in among null and alternatively distributed sub-populations. distopt=0 gives normal (2 groups), distop=1 gives t- (2 groups) and distopt=2 gives F- (2+ groups)

`control` Optionally, a list with components with the following components:
'tol' is a convergence criterion used in iterative methods which is set to 1e-8 by default.
'max.iter' is an iteration limit, set to 20 for the iterated function limit and 1000 for all others by default.
'sim.level' sim level 2 (default) stipulates, when FDP.control.method is set to "BHFDX", or "both", R.st.ht and T.st.ht are computed in addition to R.st and T.st (see above).
'low.power.stop' in simulation option, will result in an error message if the power computed via FixedPoint method is too low, which result in no solution for the BHFDX option. Default setting is TRUE. Set to FALSE to over-ride this behavior.
'FDP.meth.thresh' fine-tunes the 'Auto' voodoo (see above). Leave this alone.
'verb' vebosity level.

Value

An object of class `cdf` which contains components

`call` The call which produced the result

`cCDF.Rom` A data frame with columns `u` and `cCDF.Rom`

Author(s)

Grant Izmirlian <izmirlian at nih dot gov>

References

Izmirlian G. (2020) Strong consistency and asymptotic normality for quantities related to the Benjamini-Hochberg false discovery rate procedure. Statistics and Probability Letters; 108713, <doi:10.1016/j.spl.2020.108713>.

Izmirlian G. (2017) Average Power and λ-power in Multiple Testing Scenarios when the Benjamini-Hochberg False Discovery Rate Procedure is Used. <arXiv:1801.03989>

Jung S-H. (2005) Sample size for FDR-control in microarray data analysis. Bioinformatics; 21:3097-3104.

Kluger D. M., Owen A. B. (2023) A central limit theorem for the Benjamini-Hochberg false discovery proportion under a factor model. Bernoulli; xx:xxx-xxx.

Liu P. and Hwang J-T. G. (2007) Quick calculation for sample size while controlling false discovery rate with application to microarray analysis. Bioinformatics; 23:739-746.

Lehmann E. L., Romano J. P.. Generalizations of the familywise error rate. Ann. Stat.. 2005;33(3):1138-1154.

Romano Joseph P., Shaikh Azeem M.. Stepup procedures for control of generalizations of the familywise error rate. Ann. Stat.. 2006;34(4):1850-1873.

See Also

cCDF.ToM cCDF.VoR pwrFDR

Examples

```
library(pwrFDR)

u <- seq(from=0,to=1,len=100000)
rslt <- cCDF.Rom(u=u, effect.size=0.9, n.sample=70, r.1=0.05, alpha=0.15, N.tests=1000,
                 FDP.control.method="Auto")

## plot the result
with(rslt$cCDF.Rom, plot(u, cCDF.Rom, type="s"))

## compute the mean and median as a check
DX <- function(x)c(x[1], diff(x))
.mean. <- with(rslt$cCDF.Rom, sum(cCDF.Rom*DX(u)))
.median. <- with(rslt$cCDF.Rom, u[max(which(cCDF.Rom>0.5))])
```

---

cCDF.ToM *Computes the complimentary CDF for the true positive proportion, T_m/M_m.*

Description

Computes the complimentary CDF for the true positive proportion, T_m/M_m via asymptotic approximation. Included here mainly for pedagogic purposes.

Usage

```
cCDF.ToM(u, effect.size, n.sample, r.1, alpha, delta, groups = 2, N.tests,
         type = c("paired", "balanced", "unbalanced"), grpj.per.grp1 = NULL,
         FDP.control.method = "BHFDR", distopt,
      control=list(tol=1e-08,max.iter=c(1000,20),sim.level=2,low.power.stop=TRUE,
                   FDP.meth.thresh=FDP.cntl.mth.thrsh.def,verb=FALSE))
```

Arguments

u	A sorted vector of values on the interval [0, 1] for which the cCDF of T_m/M_m should be computed.
effect.size	The effect size (mean over standard deviation) for test statistics having non-zero means. Assumed to be a constant (in magnitude) over non-zero mean test statistics.

`n.sample` The number of experimental replicates. Required for calculation of power

`r.1` The proportion of simultaneous tests that are non-centrally located

`alpha` The false discovery rate (in the BH case) or the upper bound on the probability that the FDP exceeds delta (Romano case)

`delta` If the "FDP.control.method" is set to 'Romano' or 'BHFDX', then the user can set the exceedance thresh-hold for the FDP tail probability control $P\{FDP > \delta\} < \alpha$. The default value is α.

`groups` The number of experimental groups to compare. Must be integral and >=1. The default value is 2.

`N.tests` The number of simultaneous hypothesis tests.

`type` A character string specifying, in the groups=2 case, whether the test is 'paired', 'balanced', or 'unbalanced' and in the case when groups >=3, whether the test is 'balanced' or 'unbalanced'. The default in all cases is 'balanced'. Left unspecified in the one sample (groups=1) case.

`grpj.per.grp1` Required when `type="unbalanced"`, specifies the group 0 to group 1 ratio in the two group case, and in the case of 3 or more groups, the group j to group 1 ratio, where group 1 is the group with the largest effect under the alternative hypothesis.

`FDP.control.method`

A character string specifying how the false discovery proportion (FDP) is to be controlled. You may specify the whole word or any shortened uniquely identifying truncation.
"BHFDR": the usual BH-FDR
"BHFDX": use asymptotic approximation to the distribution of the FDP to find a smaller FDR which guarantees probability less than alpha that the FDP exceeds alpha.
"Romano": use Romano's method which guarantees probability less than alpha that the FDP exceeds alpha.
"Auto": in 'FixedPoint' mode, the program will use its own wisdom to determine which choice above to make. The order of conservatism is Romano > BHFDX > BHFDR, but BHFDR offers only expected control while the other two guarantee bounds on the excedance probabilty. If the distribution of the FDP is nearly degenerate, then BHFDR is the best option. Otherwise, if it can be reliably used, BHFDX would be the best choice. The 'effective' denominator, gamma*N.tests, in the CLT determines when the approximation is good enough and the asymptotic standard error of the FDP determines when the distribution is dispersed enough to matter. Use "Auto" to run through these checks and determine the best. A return argument, 'Auto', displays the choice made. See output components and details.
"both": in 'simulation' mode, compute statistics R and T under BHFDX and Romano (in addition to BHFDR). Corresponding statistics are denoted R.st, T.st corresponding to BHFDX control of the FDP, and R.R and T.R corresponding to Romano control of the FDP. If sim.level is set to 2, (default) the statistics R.st.ht and T.st.ht, which are the number rejected and number true positives under BHFDX where r_0 = 1-r_1, gamma, and alpha.star have been estimated from the P-value data and then alpha.star computed from these.

distopt	Test statistic distribution in among null and alternatively distributed sub-populations. distopt=0 gives normal (2 groups), distop=1 gives t- (2 groups) and distopt=2 gives F- (2+ groups)
control	Optionally, a list with components with the following components: 'tol' is a convergence criterion used in iterative methods which is set to 1e-8 by default. 'max.iter' is an iteration limit, set to 20 for the iterated function limit and 1000 for all others by default. 'sim.level' sim level 2 (default) stipulates, when FDP.control.method is set to "BHFDX", or "both", R.st.ht and T.st.ht are computed in addition to R.st and T.st (see above). 'low.power.stop' in simulation option, will result in an error message if the power computed via FixedPoint method is too low, which result in no solution for the BHFDX option. Default setting is TRUE. Set to FALSE to over-ride this behavior. 'FDP.meth.thresh' fine-tunes the 'Auto' voodoo (see above). Leave this alone. 'verb' vebosity level.

Value

An object of class cdf which contains components

call	The call which produced the result
cCDF.ToM	A data frame with columns u and cCDF.ToM

Author(s)

Grant Izmirlian <izmirlian at nih dot gov>

References

Izmirlian G. (2020) Strong consistency and asymptotic normality for quantities related to the Benjamini-Hochberg false discovery rate procedure. Statistics and Probability Letters; 108713, <doi:10.1016/j.spl.2020.108713>.

Izmirlian G. (2017) Average Power and λ-power in Multiple Testing Scenarios when the Benjamini-Hochberg False Discovery Rate Procedure is Used. <arXiv:1801.03989>

Jung S-H. (2005) Sample size for FDR-control in microarray data analysis. Bioinformatics; 21:3097-3104.

Kluger D. M., Owen A. B. (2023) A central limit theorem for the Benjamini-Hochberg false discovery proportion under a factor model. Bernoulli; xx:xxx-xxx.

Liu P. and Hwang J-T. G. (2007) Quick calculation for sample size while controlling false discovery rate with application to microarray analysis. Bioinformatics; 23:739-746.

Lehmann E. L., Romano J. P.. Generalizations of the familywise error rate. Ann. Stat.. 2005;33(3):1138-1154.

Romano Joseph P., Shaikh Azeem M.. Stepup procedures for control of generalizations of the familywise error rate. Ann. Stat.. 2006;34(4):1850-1873.

See Also

cCDF.Rom cCDF.VoR pwrFDR

Examples

```
library(pwrFDR)

u <- seq(from=0,to=1,len=100000)
rslt <- cCDF.ToM(u=u, effect.size=0.9, n.sample=70, r.1=0.05, alpha=0.15, N.tests=1000,
                 FDP.control.method="Auto")

## plot the result
with(rslt$cCDF.ToM, plot(u, cCDF.ToM, type="s"))

## compute the mean and median as a check
DX <- function(x)c(x[1], diff(x))
.mean. <- with(rslt$cCDF.ToM, sum(cCDF.ToM*DX(u)))
.median. <- with(rslt$cCDF.ToM, u[max(which(cCDF.ToM>0.5))])
```

---

cCDF.VoR — *Computes the complimentary CDF for the false discovery proportion, V_m/R_m.*

Description

Computes the complimentary CDF for the false discovery proportion, V_m/R_m via asymptotic approximation. Included here mainly for pedagogic purposes.

Usage

```
cCDF.VoR(u, effect.size, n.sample, r.1, alpha, delta, groups = 2, N.tests,
         type = c("paired", "balanced", "unbalanced"), grpj.per.grp1 = NULL,
         FDP.control.method = "BHFDR", distopt,
      control=list(tol=1e-08,max.iter=c(1000,20),sim.level=2,low.power.stop=TRUE,
                   FDP.meth.thresh=FDP.cntl.mth.thrsh.def,verb=FALSE))
```

Arguments

u	A sorted vector of values on the interval [0, 1] for which the cCDF of T_m/M_m should be computed.
effect.size	The effect size (mean over standard deviation) for test statistics having non-zero means. Assumed to be a constant (in magnitude) over non-zero mean test statistics.
n.sample	The number of experimental replicates. Required for calculation of power
r.1	The proportion of simultaneous tests that are non-centrally located
alpha	The false discovery rate (in the BH case) or the upper bound on the probability that the FDP exceeds delta (Romano case)

`delta` If the "FDP.control.method" is set to 'Romano' or 'BHFDX', then the user can set the exceedance thresh-hold for the FDP tail probability control $P\{FDP > \delta\} < \alpha$. The default value is α.

`groups` The number of experimental groups to compare. Must be integral and >=1. The default value is 2.

`N.tests` The number of simultaneous hypothesis tests.

`type` A character string specifying, in the groups=2 case, whether the test is 'paired', 'balanced', or 'unbalanced' and in the case when groups >=3, whether the test is 'balanced' or 'unbalanced'. The default in all cases is 'balanced'. Left unspecified in the one sample (groups=1) case.

`grpj.per.grp1` Required when `type="unbalanced"`, specifies the group 0 to group 1 ratio in the two group case, and in the case of 3 or more groups, the group j to group 1 ratio, where group 1 is the group with the largest effect under the alternative hypothesis.

`FDP.control.method`

A character string specifying how the false discovery proportion (FDP) is to be controlled. You may specify the whole word or any shortened uniquely identifying truncation.
"BHFDR": the usual BH-FDR
"BHFDX": use asymptotic approximation to the distribution of the FDP to find a smaller FDR which guarantees probability less than alpha that the FDP exceeds alpha.
"Romano": use Romano's method which guarantees probability less than alpha that the FDP exceeds alpha.
"Auto": in 'FixedPoint' mode, the program will use its own wisdom to determine which choice above to make. The order of conservatism is Romano > BHFDX > BHFDR, but BHFDR offers only expected control while the other two guarantee bounds on the excedance probabilty. If the distribution of the FDP is nearly degenerate, then BHFDR is the best option. Otherwise, if it can be reliably used, BHFDX would be the best choice. The 'effective' denominator, gamma*N.tests, in the CLT determines when the approximation is good enough and the asymptotic standard error of the FDP determines when the distribution is dispersed enough to matter. Use "Auto" to run through these checks and determine the best. A return argument, 'Auto', displays the choice made. See output components and details.
"both": in 'simulation' mode, compute statistics R and T under BHFDX and Romano (in addition to BHFDR). Corresponding statistics are denoted R.st, T.st corresponding to BHFDX control of the FDP, and R.R and T.R corresponding to Romano control of the FDP. If sim.level is set to 2, (default) the statistics R.st.ht and T.st.ht, which are the number rejected and number true positives under BHFDX where r_0 = 1-r_1, gamma, and alpha.star have been estimated from the P-value data and then alpha.star computed from these.

`distopt` Test statistic distribution in among null and alternatively distributed sub-populations. distopt=0 gives normal (2 groups), distop=1 gives t- (2 groups) and distopt=2 gives F- (2+ groups)

`control` Optionally, a list with components with the following components:
'tol' is a convergence criterion used in iterative methods which is set to 1e-8 by

default.
'max.iter' is an iteration limit, set to 20 for the iterated function limit and 1000 for all others by default.
'sim.level' sim level 2 (default) stipulates, when FDP.control.method is set to "BHFDX", or "both", R.st.ht and T.st.ht are computed in addition to R.st and T.st (see above).
'low.power.stop' in simulation option, will result in an error message if the power computed via FixedPoint method is too low, which result in no solution for the BHFDX option. Default setting is TRUE. Set to FALSE to over-ride this behavior.
'FDP.meth.thresh' fine-tunes the 'Auto' voodoo (see above). Leave this alone.
'verb' vebosity level.

Value

An object of class `cdf` which contains components

`call`	The call which produced the result
`cCDF.VoR`	A data frame with columns `u` and `cCDF.VoR`

Author(s)

Grant Izmirlian <izmirlian at nih dot gov>

References

Izmirlian G. (2020) Strong consistency and asymptotic normality for quantities related to the Benjamini-Hochberg false discovery rate procedure. Statistics and Probability Letters; 108713, <doi:10.1016/j.spl.2020.108713>.

Izmirlian G. (2017) Average Power and λ-power in Multiple Testing Scenarios when the Benjamini-Hochberg False Discovery Rate Procedure is Used. <arXiv:1801.03989>

Jung S-H. (2005) Sample size for FDR-control in microarray data analysis. Bioinformatics; 21:3097-3104.

Kluger D. M., Owen A. B. (2023) A central limit theorem for the Benjamini-Hochberg false discovery proportion under a factor model. Bernoulli; xx:xxx-xxx.

Liu P. and Hwang J-T. G. (2007) Quick calculation for sample size while controlling false discovery rate with application to microarray analysis. Bioinformatics; 23:739-746.

Lehmann E. L., Romano J. P.. Generalizations of the familywise error rate. Ann. Stat.. 2005;33(3):1138-1154.

Romano Joseph P., Shaikh Azeem M.. Stepup procedures for control of generalizations of the familywise error rate. Ann. Stat.. 2006;34(4):1850-1873.

See Also

`cCDF.Rom` `cCDF.ToM` `pwrFDR`

Examples

```
library(pwrFDR)

u <- seq(from=0,to=1,len=100000)
rslt <- cCDF.VoR(u=u, effect.size=0.9, n.sample=70, r.1=0.05, alpha=0.15, N.tests=1000,
                 FDP.control.method="Auto")

## plot the result
with(rslt$cCDF.VoR, plot(u, cCDF.VoR, type="s"))

## compute the mean and median as a check
DX <- function(x)c(x[1], diff(x))
.mean. <- with(rslt$cCDF.VoR, sum(cCDF.VoR*DX(u)))
.median. <- with(rslt$cCDF.VoR, u[max(which(cCDF.VoR>0.5))])
```

CDF.Pval *CDF of pooled (H0 and HA) population p-values*

Description

Computes the CDF of the pooled population p-values under the mixture model, e.g. the p-values are i.i.d. with CDF a mixture between a uniform (CDF in the null distributed population) and a concave function (CDF in the non-null distributed population).

Usage

```
CDF.Pval(u, effect.size, n.sample, r.1, groups=2, type="balanced",
         grpj.per.grp1=1, distopt, control)
```

Arguments

u	Argument of the CDF. Result will be Pr(P_i <= u)
effect.size	The effect size (mean over standard deviation) for test statistics having non-zero means. Assumed to be a constant (in magnitude) over non-zero mean test statistics.
n.sample	The number of experimental replicates.
r.1	The proportion of all test statistics that are distributed under HA.
groups	The number of experimental groups to compare. Default value is 2.
type	A character string specifying, in the groups=2 case, whether the test is 'paired', 'balanced', or 'unbalanced' and in the case when groups >=3, whether the test is 'balanced' or 'unbalanced'. The default in all cases is 'balanced'. Left unspecified in the one sample (groups=1) case.
grpj.per.grp1	Required when type="unbalanced", specifies the group 0 to group 1 ratio in the two group case, and in the case of 3 or more groups, the group j to group 1 ratio, where group 1 is the group with the largest effect under the alternative hypothesis.

distopt	Test statistic distribution in among null and alternatively distributed sub-populations. distopt=0 gives normal (2 groups), distop=1 gives t- (2 groups) and distopt=2 gives F- (2+ groups)
control	Optionally, a list with components with the following components: 'groups', used when distop=3 (F-dist), specifying number of groups. 'tol' is a convergence criterion used in iterative methods which is set to 1e-8 by default 'max.iter' is an iteration limit, set to 1000 by default

Details

Computes the CDF of the pooled population p-values under the mixture model, e.g. the p-values are i.i.d. with CDF a mixture between a uniform (CDF in the null distributed population) and a concave function (CDF in the non-null distributed population). If Fc_0 is the cCDF of a test statistic under H0 and Fc_A is the cCDF of a test statistic under HA then the CDF of the P-values is

`G(u) = (1-r) u + r Fc_A(Fc_0^{-1}(u))`

The limiting positve call fraction, lim_m V_m/m = gamma (a.s.) is the solution to the equation

G(gamma alpha) = gamma

where alpha is the nominal FDR.

Value

A list with components

call	The call which produced the result
u	The argument that was passed to the function
CDF.Pval	The value of the CDF

Author(s)

Grant Izmirlian <izmirlian at nih dot gov>

References

Izmirlian G. (2020) Strong consistency and asymptotic normality for quantities related to the Benjamini-Hochberg false discovery rate procedure. Statistics and Probability Letters; 108713, <doi:10.1016/j.spl.2020.108713>.

Izmirlian G. (2017) Average Power and λ-power in Multiple Testing Scenarios when the Benjamini-Hochberg False Discovery Rate Procedure is Used. arXiv:1801.03989

Genovese, C. and L. Wasserman. (2004) A stochastic process approach to false discovery control. Annals of Statistics. 32 (3), 1035-1061.

See Also

CDF.Pval.HA

Examples

```
## First calculate an average power for a given set of parameters
rslt.avgp <- pwrFDR(effect.size=0.79, n.sample=46, r.1=2000/54675, alpha=0.15)

## Now verify that G( gamma alpha ) = gamma

gma <- rslt.avgp$gamma
alpha <- rslt.avgp$call$alpha

G.gma.a <- CDF.Pval(u=gma*alpha, r.1=2000/54675, effect.size=0.79, n.sample=46)$CDF.Pval$CDF.Pval

c(G.of.gamma.alpha=G.gma.a, gamma=gma)
```

`CDF.Pval.apsi.eq.u` *Calculates the fixed point for the Romano procedure.*

Description

Calculates the fixed point for the Romano procedure, e.g. finds u which solves u = G(psi(u, d) a) where G is the common p-value CDF, and psi(u, d) = u d/(1 - (1-a) u). Essentially an internal function and included at the user level for pedagogic purposes.

Usage

```
CDF.Pval.apsi.eq.u(effect.size, n.sample, r.1, alpha, delta, groups, type,
                   grpj.per.grp1, distopt, control)
```

Arguments

`effect.size`	The effect size (mean over standard deviation) for test statistics having non-zero means. Assumed to be a constant (in magnitude) over non-zero mean test statistics.
`n.sample`	The number of experimental replicates. Required for calculation of power
`r.1`	The proportion of simultaneous tests that are non-centrally located
`alpha`	The upper bound on the probability that the FDP exceeds delta.
`delta`	The exceedance thresh-hold for the FDP tail probability control method (BHFDX or Romano) $P\{FDP > \delta\} < \alpha$. The default value is α.
`groups`	The number of experimental groups to compare. Must be integral and >=1. The default value is 2.
`type`	A character string specifying, in the groups=2 case, whether the test is 'paired', 'balanced', or 'unbalanced' and in the case when groups >=3, whether the test is 'balanced' or 'unbalanced'. The default in all cases is 'balanced'. Left unspecified in the one sample (groups=1) case.

grpj.per.grp1	Required when type="unbalanced", specifies the group 0 to group 1 ratio in the two group case, and in the case of 3 or more groups, the group j to group 1 ratio, where group 1 is the group with the largest effect under the alternative hypothesis.
distopt	Test statistic distribution in among null and alternatively distributed sub-populations. distopt=0 gives normal (2 groups), distop=1 gives t- (2 groups) and distopt=2 gives F- (2+ groups)
control	Optionally, a list with components with the following components: 'tol' is a convergence criterion used in iterative methods which is set to 1e-8 by default. 'max.iter' is an iteration limit, set to 20 for the iterated function limit and 1000 for all others by default. 'sim.level' sim level 2 (default) stipulates, when FDP.control.method is set to "BHFDX", or "both", R.st.ht and T.st.ht are computed in addition to R.st and T.st (see above). 'low.power.stop' in simulation option, will result in an error message if the power computed via FixedPoint method is too low, which result in no solution for the BHFDX option. Default setting is TRUE. Set to FALSE to over-ride this behavior. 'FDP.meth.thresh' fine-tunes the 'Auto' voodoo (see above). Leave this alone. 'verb' vebosity level.

Value

An object of class cdf which contains components

call	The call which produced the result
gamma	The fixed point for the Romano method.

Author(s)

Grant Izmirlian <izmirlian at nih dot gov>

References

Izmirlian G. (2020) Strong consistency and asymptotic normality for quantities related to the Benjamini-Hochberg false discovery rate procedure. Statistics and Probability Letters; 108713, <doi:10.1016/j.spl.2020.108713>.

Izmirlian G. (2017) Average Power and λ-power in Multiple Testing Scenarios when the Benjamini-Hochberg False Discovery Rate Procedure is Used. <arXiv:1801.03989>

Jung S-H. (2005) Sample size for FDR-control in microarray data analysis. Bioinformatics; 21:3097-3104.

Liu P. and Hwang J-T. G. (2007) Quick calculation for sample size while controlling false discovery rate with application to microarray analysis. Bioinformatics; 23:739-746.

Lehmann E. L., Romano J. P.. Generalizations of the familywise error rate. Ann. Stat.. 2005;33(3):1138-1154.

Romano Joseph P., Shaikh Azeem M.. Stepup procedures for control of generalizations of the familywise error rate. Ann. Stat.. 2006;34(4):1850-1873.

See Also

CDF.Pval.au.eq.u

Examples

```
## An example showing that the Romano method is more conservative than the BHFDX method
## which is in turn more conservative than the BH-FDR method based upon ordering of the
## significant call proportions, R_m/m

## First find alpha.star for the BH-CLT method at level alpha=0.15
a.st.BHFDX <-controlFDP(effect.size=0.8,r.1=0.05,N.tests=1000,n.sample=70,alpha=0.15)$alpha.star

## now find the significant call fraction under the BH-FDR method at level alpha=0.15
gamma.BHFDR <- CDF.Pval.au.eq.u(effect.size = 0.8, n.sample = 70, r.1 = 0.05, alpha=0.15)

## now find the significant call fraction under the Romano method at level alpha=0.15
gamma.romano <- CDF.Pval.apsi.eq.u(effect.size = 0.8, n.sample = 70, r.1 = 0.05, alpha=0.15)

## now find the significant call fraction under the BH-CLT method at level alpha=0.15
gamma.BHFDX <- CDF.Pval.au.eq.u(effect.size = 0.8, n.sample = 70, r.1 = 0.05, alpha=a.st.BHFDX)
```

`CDF.Pval.au.eq.u` *Function which solves the implicit equation u = G(u alpha)*

Description

Function which solves the implicit equation u = G(u alpha) where G is the pooled P-value CDF and alpha is the FDR

Usage

```
CDF.Pval.au.eq.u(effect.size, n.sample, r.1, alpha, groups, type,
                 grpj.per.grp1, distopt, control)
```

Arguments

`effect.size`	The per statistic effect size
`n.sample`	The per statistic sample size
`r.1`	The proportion of Statistics distributed according to the alternative distribution
`alpha`	The false discovery rate.
`groups`	Number of experimental groups from which the test statistic is calculated
`type`	A character string specifying, in the groups=2 case, whether the test is 'paired', 'balanced', or 'unbalanced' and in the case when groups >=3, whether the test is 'balanced' or 'unbalanced'. The default in all cases is 'balanced'. Left unspecified in the one sample (groups=1) case.

grpj.per.grp1	Required when type="unbalanced", specifies the group 0 to group 1 ratio in the two group case, and in the case of 3 or more groups, the group j to group 1 ratio, where group 1 is the group with the largest effect under the alternative hypothesis.
distopt	Test statistic distribution in among null and alternatively distributed sub-populations. distopt=0 gives normal (2 groups), distop=1 gives t- (2 groups) and distopt=2 gives F- (2+ groups)
control	Optionally, a list with components with the following components: 'groups', used when distop=3 (F-dist), specifying number of groups. 'max.iter' is an iteration limit, set to 1000 by default

Value

A list with a single component,

gamma	The solution of the implicit equation u = G(u alpha), where G is the pooled P-value CDF. This represents the infinite tests limiting proportion of hypothesis tests that are called significant by the BH-FDR procedure at alpha.

Author(s)

Grant Izmirlian <izmirlian at nih dot gov>

References

Izmirlian G. (2020) Strong consistency and asymptotic normality for quantities related to the Benjamini-Hochberg false discovery rate procedure. Statistics and Probability Letters; 108713, <doi:10.1016/j.spl.2020.108713>.

Izmirlian G. (2017) Average Power and λ-power in Multiple Testing Scenarios when the Benjamini-Hochberg False Discovery Rate Procedure is Used. <arXiv:1801.03989>

Jung S-H. (2005) Sample size for FDR-control in microarray data analysis. Bioinformatics; 21:3097-3104.

Liu P. and Hwang J-T. G. (2007) Quick calculation for sample size while controlling false discovery rate with application to microarray analysis. Bioinformatics; 23:739-746.

See Also

CDF.Pval.apsi.eq.u

Examples

```
## An example showing that the Romano method is more conservative than the BHFDX method
## which is in turn more conservative than the BH-FDR method based upon ordering of the
## significant call proportions, R_m/m

## First find alpha.star for the BH-CLT method at level alpha=0.15
a.st.BHFDX <-controlFDP(effect.size=0.8,r.1=0.05,N.tests=1000,n.sample=70,alpha=0.15)$alpha.star

## now find the significant call fraction under the BH-FDR method at level alpha=0.15
gamma.BHFDR <- CDF.Pval.au.eq.u(effect.size = 0.8, n.sample = 70, r.1 = 0.05, alpha=0.15)
```

```
 ## now find the significant call fraction under the Romano method at level alpha=0.15
gamma.romano <- CDF.Pval.apsi.eq.u(effect.size = 0.8, n.sample = 70, r.1 = 0.05, alpha=0.15)

 ## now find the significant call fraction under the BH-CLT method at level alpha=0.15
gamma.BHFDX <- CDF.Pval.au.eq.u(effect.size = 0.8, n.sample = 70, r.1 = 0.05, alpha=a.st.BHFDX)
```

`CDF.Pval.HA` *CDF of p-values for test statistics distribted under HA.*

Description

Computes the CDF of p-values for test statistics distribted under HA.

Usage

```
CDF.Pval.HA(u, effect.size, n.sample, r.1, groups = 2, type="balanced",
            grpj.per.grp1=1, distopt, control)
```

Arguments

`u`	Argument of the CDF. Result will be Pr(P_i <= u)
`effect.size`	The effect size (mean over standard deviation) for test statistics having non-zero means. Assumed to be a constant (in magnitude) over non-zero mean test statistics.
`n.sample`	The number of experimental replicates.
`r.1`	The proportion of all test statistics that are distributed under HA.
`groups`	The number of experimental groups to compare. Default value is 2.
`type`	A character string specifying, in the groups=2 case, whether the test is 'paired', 'balanced', or 'unbalanced' and in the case when groups >=3, whether the test is 'balanced' or 'unbalanced'. The default in all cases is 'balanced'. Left unspecified in the one sample (groups=1) case.
`grpj.per.grp1`	Required when type="unbalanced", specifies the group 0 to group 1 ratio in the two group case, and in the case of 3 or more groups, the group j to group 1 ratio, where group 1 is the group with the largest effect under the alternative hypothesis.
`distopt`	Test statistic distribution in among null and alternatively distributed sub-populations. distopt=0 gives normal (2 groups), distop=1 gives t- (2 groups) and distopt=2 gives F- (2+ groups)
`control`	Optionally, a list with components with the following components: 'groups', used when distop=3 (F-dist), specifying number of groups. 'tol' is a convergence criterion used in iterative methods which is set to 1e-8 by default 'max.iter' is an iteration limit, set to 20 for function iteration and 1000 for all others by default 'distop', specifying the distribution family of the central and non-centrally located sub-populations. =1 gives normal (2 groups) =2 gives t- (2 groups) and =3 gives F- (2+ groups)

Details

Computes the CDF of p-values for test statistics distribted under HA. If Fc_0 is the cCDF of a test statistic under H0 and Fc_A is the cCDF of a test statistic under HA then the CDF of a P-value for a test statistic distributed under HA is

`G_A(u) = Fc_A(Fc_0^{-1}(u))`

The limiting true positive fraction is the infinite simultaneous tests average power,

lim_m T_m/M_m = average.power (a.s.),

which is used to approximate the average power for finite 'm', is G_1 at gamma alpha:

G_1(gamma alpha) = average.pwer

where alpha is the nominal FDR and gamma = lim_m R_m/m (a.s.) is the limiting positive call fraction.

Value

A list with components

`call`	The call which produced the result
`u`	The argument that was passed to the function
`CDF.Pval.HA`	The value of the CDF

Author(s)

Grant Izmirlian <izmirlian at nih dot gov>

References

Izmirlian G. (2020) Strong consistency and asymptotic normality for quantities related to the Benjamini-Hochberg false discovery rate procedure. Statistics and Probability Letters; 108713, <doi:10.1016/j.spl.2020.108713>.

Izmirlian G. (2017) Average Power and λ-power in Multiple Testing Scenarios when the Benjamini-Hochberg False Discovery Rate Procedure is Used. <arXiv:1801.03989>

Genovese, C. and L. Wasserman. (2004) A stochastic process approach to false discovery control. Annals of Statistics. 32 (3), 1035-1061.

See Also

`CDF.Pval`

Examples

```
## First calculate an average power for a given set of parameters
rslt.avgp <- pwrFDR(effect.size=0.79, n.sample=42, r.1=0.05, alpha=0.15)

## Now verify that G_A( gamma f ) = average.power

gma <- rslt.avgp$gamma
alpha <- rslt.avgp$call$alpha
```

```
GA.gma.alpha <- CDF.Pval.HA(u=gma*alpha, r.1=0.05, effect.size=0.79, n.sample=42)

c(G.gm.alpha=GA.gma.alpha$CDF.Pval.HA$CDF.Pval.HA, average.power=rslt.avgp$average.power)
```

`controlFDP` *Helper function for the BHFDX FDP control method*

Description

Helper function for the BHFDX FDP control method. Calculates a reduced FDR required to bound the the false discovery proportion in probability using asymptotic approximation.

Usage

```
controlFDP(effect.size, n.sample, r.1, alpha, delta, groups = 2,
  N.tests, type, grpj.per.grp1, corr.struct, control, formula, data, distopt)
```

Arguments

`effect.size`	The effect size (mean over standard deviation) for test statistics having non-zero means. Assumed to be a constant (in magnitude) over non-zero mean test statistics.
`n.sample`	The number of experimental replicates.
`r.1`	The proportion of simultaneous tests that are non-centrally located
`alpha`	The upper bound on the probability that the FDP exceeds delta.
`delta`	The exceedance thresh-hold for the FDP tail probability control method (BHFDX or Romano) $P\{FDP > \delta\} < \alpha$. The default value is α.
`groups`	The number of experimental groups to compare. Default value is 2.
`N.tests`	The number of simultaneous hypothesis tests.
`type`	A character string specifying, in the groups=2 case, whether the test is 'paired', 'balanced', or 'unbalanced' and in the case when groups >=3, whether the test is 'balanced' or 'unbalanced'. The default in all cases is 'balanced'. Left unspecified in the one sample (groups=1) case.
`grpj.per.grp1`	Required when `type`="unbalanced", specifies the group 0 to group 1 ratio in the two group case, and in the case of 3 or more groups, the group j to group 1 ratio, where group 1 is the group with the largest effect under the alternative hypothesis.
`corr.struct`	Specifies a block correlation structure between test statistics which is used in both the simulation routine, and in the computations based upon asymptotic approximation, e.g. the AFDX control and the ATPP method. Its form is specified via the following named elements. "type": "CS-Blocks" for compound symmetry within blocks or "Toeplitz-Blocks" for toeplitz within blocks. "block.size": the size of the correlated blocks "rho": When type="CS-Blocks", then 'rho' is a correlation of length 1 when type="Toeplitz-Blocks", then 'rho' is a vector of correlations of length 'block.size' - 1

control	Optionally, a list with components with the following components: 'groups', used when distop=3 (F-dist), specifying number of groups. 'tol' is a convergence criterion used in iterative methods which is set to 1e-8 by default 'max.iter' is an iteration limit, set to 20 for the iterated function limit and 1000 for all others by default 'sim.level' sim level 2 results in more detail at the expense of slightly more computational time. 'low.power.stop' in simulation option, will result in an error message if the power computed via FixedPoint method is too low, which result in no solution for the BHFDX option. Default setting is TRUE. Set to FALSE to over-ride. 'FDP.meth.thresh' fine-tunes the 'Auto' voodoo (see above). Leave this alone. 'verb' vebosity level.
formula	Optionally, the function can be used to _estimate_ f* from a given dataset of sorted p-values. In this case we specify formula, which is a formula of the form `pval~1` where 'pval' is the name of the p-value variable in the dataset, `dataset` (see
data	The name of the dataset.
distopt	Test statistic distribution in among null and alternatively distributed sub-populations. distopt=0 gives normal (2 groups), distop=1 gives t- (2 groups) and distopt=2 gives F- (2+ groups)

Details

Uses a CLT for the FDP to calculate a reduced alpha required to bound the the false discovery rate in probability...e.g. finds alpha* so that when the BH-FDR procedure is controlled at alpha*, we ensure that

Pr(V_m/R_m > (1-r) alpha) < (1-r) alpha

where 'alpha' is the original false discovery rate and 'r' is the proportion of non-null distributed test statistics.

Value

alpha.star	The reduced alpha required to bound the FDP in probability
obj	Objective value at 'alpha.star'... should be close to 0
L.star	The bound on the FDP, should be (1-r) f. See above.
P.star	The probability that the FDP is greater than L.star. See above.
average.power	Resulting average power.
c.g	The BH-FDR threshold on the scale of the test statistics.
gamma	The proportion of all 'm' tests declared significant.
objective	Result of optimization yielding the 'average.power'.
err.III	Mass on the wrong side of the threshold.
sigma.rtm.SoM	Asymptotic variance of the true positive fraction.

Author(s)

Grant Izmirlian <izmirlian at nih dot gov>

References

Izmirlian G. (2020) Strong consistency and asymptotic normality for quantities related to the Benjamini-Hochberg false discovery rate procedure. Statistics and Probability Letters; 108713, <doi:10.1016/j.spl.2020.108713>

Izmirlian G. (2017) Average Power and λ-power in Multiple Testing Scenarios when the Benjamini-Hochberg False Discovery Rate Procedure is Used. <arXiv:1801.03989>

Kluger D. M., Owen A. B. (2023) A central limit theorem for the Benjamini-Hochberg false discovery proportion under a factor model. Bernoulli; xx:xxx-xxx.

See Also

pwrFDR

Examples

```
## at alpha=0.15 and other parameters, it takes n.sample=46 replicates for
## average power > 80%
pwr.46.15 <- pwrFDR(alpha=0.15, r.1=0.03, N.tests=1000, effect.size=0.79, n.sample=46)

## when there are 'only' N.tests=1000 simultaneous tests, the distribution of the
## false discovery fraction, FDP, is not so highly spiked at the alpha=0.15
## You need to set the alpha down to alpha=0.0657 to ensure that  Pr( T/J > 0.145 ) < 0.0657
fstr <- controlFDP(alpha=0.15, r.1=0.03, N.tests=1000, effect.size=0.8, n.sample=46)

## at all the above settings, with alpha=0.0657 at an n.sample of 46, we only have 69%
## average power.
pwr.46.0657 <- pwrFDR(alpha=0.065747, r.1=0.03, N.tests=1000, effect.size=0.79, n.sample=46)

## it'll cost 7 more replicates to get the average power up over 80%.
pwr.53.0657 <- pwrFDR(alpha=0.065747, r.1=0.03, N.tests=1000, effect.size=0.8, n.sample=53)

## it costs only 8.75% more to get it right!
```

`criterion` *BH-FDR and Romano Criterion*

Description

Compute BH-FDR step up criterion, or Romano step-down criterion

Usage

```
criterion(alpha, delta, N.tests, FDP.control.method = c("BHFDR", "Romano"))
```

Arguments

alpha	The false discovery rate (in the BH case) or the upper bound on the probability that the FDP exceeds lambda (Romano case)
delta	If the "FDP.control.method" is set to 'Romano' then the user can set the exceedance thresh-hold for the FDP tail probability control $P\{FDP > \delta\} < \alpha$. The default value is α.
N.tests	The number of simultaneous hypothesis tests.
FDP.control.method	A character string specifying how the false discovery proportion (FDP) is to be controlled. You may specify the whole word or any shortened uniquely identifying truncation. "BHFDR": the usual BH-FDR "Romano": use Romano's method which guarantees probability less than alpha that the FDP exceeds alpha.

Value

The step down or step up criterion, which is a vector of length N.tests

Author(s)

Grant Izmirlian <izmirlian at nih dot gov>

References

Benjamini Y, Hochberg Y. Controlling the false discovery rate - a practical and powerful approach to multiple testing. J. R. Stat. Soc. Ser. B Stat. Methodol. 1995; 57(1):289-300.

Romano J.P. and Shaikh A.M. On stepdown control of the false discovery proportion. IMS Lecture Notes–Monograph Series. 2006; 49:33-50. DOI: 10.1214/074921706000000383.

Examples

```
library(pwrFDR)

crit.b <- criterion(N.tests=1000, alpha=0.15, FDP.control.method="BHFDR")
crit.r <- criterion(N.tests=1000, alpha=0.15, FDP.control.method="Romano")
crit.r.17 <- criterion(N.tests=1000, alpha=0.15, delta=0.17, FDP.control.method="Romano")
matplot(1:1000, cbind(crit.b, crit.r, crit.r.17), type="l", lty=1, col=2:4)
```

---

detail *The detail extraction function for simulated power objects*

Description

Objects created by the pwrFDR function with option method=="simulation" are returned with an attribute named detail. This is its extractor function

Usage

```
detail(obj)
```

Arguments

obj	An object created by the pwrFDR function with option method=="simulation".

Value

A list with components

reps	A data frame of n.sim rows containing the results of the simulations as columns: M1, the # of non-null distributed statistics, R, the # of statistical tests rejected under the BH-FDR procedure, and T, the # of true positives. Depending on the value of FDP.control.method, additional values are returned: FDP.control.method: "BHFDR": nothing further "BHFDX": In addition to M1, R and T, we compute R.st and T.st, the # of tests rejected under the BHFDX procedure, and among those tests rejected, the # of true positives. "Romano": In addition to M1, R and T, we compute R.R and T.R, the # of tests rejected under Romano's procedure, and among those tests rejected, the # of true positives. "both": If 'both' is specified then counts of all rejected tests and of true positive tests under BHFDX and under Romano are returned along with those counts for BHFDr.
X	A single simulation replicate of the m raw test statistics, included as a sanity check

Author(s)

Grant Izmirlian izmirlig at mail dot nih dot gov

---

dists *The Distribution family object*

Description

The pwrFDR package currently incorporates 3 distribution types, normal, t and F. The first two of these are strictly for statistics formed from two group comparison while the third is for statistics formed from the omnibus test of any difference among an arbitrary number of groups >=2. The structure is general and user expandable. One must specify the density, CDF and quantile function for a given distribution and its parameters under the null and under the alternative. These parameters must be expressions to be evaluated inside the kernel of the power program, functions of the arguments n.sample, groups and effect.size. This is not used directly by the user at all unless she (he) wants to add a distribution type.

Format

A data frame with 3 observations on the following 6 variables.

`pars0` a list vector having components 'c(nd, p1, p2, ...)' where 'nd' is the distribution number starting with 0, and p1, p2, ..., are paramters of the distribution, which are functions of 'n.sample', 'groups' and 'effect.size' as mentioned above. These must be expressed as a call e.g. as.call(expression(c, nd, p1, p2, ...)) etc. 'pars0' are the parameters under the null.

`pars1` a list vector. See directly above. Parameters under the alternative.

`minv` a list vector with components given the values -Inf or 0, which will be used to decide if the two sided corrections are used or not.

`ddist` a list vector with components set to functions, each one computing the probability density function corresponding to the particular distribution. A function of arguments 'x' and 'par'. See details below.

`pdist` a list vector with components set to the functions, each one computing the cumulative distribution function corresponding to the particular distribution. A function of arguments 'x' and 'par'. See details below.

`qdist` a list vector with components set to the functions, each one computing the quantile function (inverse cumulative distribution function) corresponding to the particular distribution. A function of arguments 'x' and 'par'. See details below.

`rdist` a list vector with components set to the functions, each one capable of simulating a specified number of replicates corresponding to the particular distribution. A function of arguments 'n' and 'par'. See details below.

Details

`dists` is a data.frame with components `pars0`, `pars1`, `minv`, `ddist`, `pdist`, `qdist` and `rdist`. For the three available distribution options, "normal", "t" and "f", the components `pars0` and `pars1` take the following form:

	pars0	pars1
1.	pars0	pars1
2.	c(0,ncp=0,sd=1)	c(0,ncp=.NCP.,sd=1)
3.	c(1,ncp=0,ndf=.DF.)	c(1,ncp=.NCP.,ndf=.DF.)
4.	c(2,ncp=0,ndf1=groups-1,ndf2=.DF.)	c(2,ncp=.NCP.^2,ndf1=groups-1,ndf2=.DF.)

The component `minv` gives the minumum value of the support set of the distribution. For the above named three available distribution options, `minv` is set to the values -Inf, -Inf and 0, respectively. The components `ddist`, `pdist`, `qdist`, and `rdist` contain functions defining the density, CDF, quantile, and simulator function, respectively. For the above named three available distribution options, `ddist` takes the following form:

1. ddist
2. function (x, par) dnorm(x, mean = par[2], sd = par[3])
3. function (x, par) dt(x, ncp = par[2], df = par[3])

4. function (x, par) df(x, ncp = par[2], df1 = par[3], df2 = par[4])
exept for rdist, which has arguments 'n' and 'par'

The components `pdist` and `qdist` are nearly identical to the component `ddist`, but with pnorm, pt, pf and qnorm, qt, qf replacing dnorm, dt and df, respectively.
The variables, `.NCP.` and `.DF.` named above are defined within the functions in which `ddist` is used based upon corresponding expressions, `NCP` and `DF`. These expressions currently contain 3 component expressions, one for each of the available test types, "paired", "balanced" and "unbalanced".
`NCP` is currently defined:

1. NCP
expression(n.sample^0.5*effect.size,(n.sample/groups)^0.5*effect.size,
((n.sample-1)/(1+sum((n.sample-1)/(nii.sample-1))))^0.5*effect.size)

and `DF` is currently defined:

1. DF
expression(n.sample - 1, groups * (n.sample - 1),
groups^2*(n.sample-1)/(1+sum((n.sample-1)/(nii.sample-1))))

Source

This isn't 'data' data, its a kind of a 'family' object.

Izmirlian G. (2020) Strong consistency and asymptotic normality for quantities related to the Benjamini-Hochberg false discovery rate procedure. Statistics and Probability Letters; <doi:10.1016/j.spl.2020.108713>

Izmirlian G. (2017) Average Power and λ-power in Multiple Testing Scenarios when the Benjamini-Hochberg False Discovery Rate Procedure is Used. <arXiv:1801.03989>

`es.ROC` *Computes the equivalent Z-test effect size in hypothesis tests involving the ROC. Included here with the intent that it can be used in conjunction with* `pwrFDR` *to allow power/sample size calculation for multiple tests of ROC curve based hypothesis. See details.*

Description

In hypothesis tests of TPR_1 vs TPR_0 at fixed FPR, or FPR_1 vs FPR_0 at fixed TPR, this computes the equivalent Z-test effect size. This can then be passed the `effect.size` argument in a call to `pwrFDR` or `controlFDP`

Usage

```
es.ROC(FPR0, FPR1 = NULL, TPR0, TPR1 = NULL, b = NULL)
```

Arguments

FPR0	When the TPR is fixed, the FPR under the null. Otherwise the fixed FPR.
FPR1	When the TPR is fixed, the FPR under the alternative. Otherwise left blank.
TPR0	When the FPR is fixed, the TPR under the null. Otherwise the fixed TPR.
TPR1	When the FPR is fixed, the TPR under the alternative. Otherwise left blank.
b	Nominal slope of the ROC at FPR0. Taken to be 1 by default.

Value

The equivalent Z-test effect size in a hypothesis test for difference in TPR at fixed FPR or difference in FPR at fixed TPR

Author(s)

Grant Izmirlian <izmirlian at nih dot gov>

References

Pepe M. S., Feng Z, Janes, H Bossuyt P. M. and Potter J. D. Pivotal evaluation of the accuracy of a biomarker used for classification or prediction. Supplement. J Natl Cancer Inst 2008;100: 1432–1438

See Also

cc.ROC

Examples

```
es.ROC(FPR0=0.15, TPR0=0.80, TPR1=0.90)
```

---

gentempfilenm *Generate a tempfile name*

Description

Generates a tempfile name with an optional user specified prefix and suffix Result is a character string

Usage

```
gentempfilenm(prfx = "temp", sfx = ".txt")
```

Arguments

prfx	prefix for the file name, e.g. "temp"
sfx	suffix (file extension) for the file name, e.g. ".txt"

Value

a character string containing the randomly generated name of the tempfile.

Author(s)

Grant Izmirlian izmirlig at mail dot nih dot gov

---

if.0.rm *A helper function– remove if zero.*

Description

A helper function– remove if zero. Included at the user level because it's useful for setting up batch jobs.

Usage

```
if.0.rm(x)
```

Arguments

x	A numeric vector.

Value

A numeric vector, equal to the input vector, x, except with 0's removed.

Author(s)

Grant Izmirlian <izmirlian at nih dot gov>

`if.na.x` *A helper function – substitute 'NA's with a specified 'x'.*

Description

A helper function – substitute 'NA's with a specified 'x'. Included at the user level because it's useful for setting up batch jobs.

Usage

```
if.na.x(x, x0 = FALSE)
```

Arguments

`x`	A numeric or boolean vector.
`x0`	Value with which to replace `NA`'s. Defaults to 0 or `FALSE`.

Value

A numeric vector, equal to the input vector, `x`, except with `NA`'s replaced by the value, `x0`, which the user suplied.

Author(s)

Grant Izmirlian <izmirlian at nih dot gov>

`if.y.z` *A helper function – substitute* `y`*'s with a specified 'z'.*

Description

A helper function – in a numeric vector, substitute values equal to 'y' with user specified 'z'. Included at the user level because it's useful for setting up batch jobs.

Usage

```
if.y.z(x, y = 0, z = 1)
```

Arguments

`x`	A numeric, character or boolean vector
`y`	The valued to be swapped out
`z`	The value which replaces swapped out values

Value

A numeric, character or boolean vector, equal to the input vector, x, except with occurences y replaced with the value z

Author(s)

Grant Izmirlian <izmirlian at nih dot gov>

`join.tbl` *Combine pwrFDR Results*

Description

Joins pwrFDR objects into a single table.

Usage

```
join.tbl(...)
```

Arguments

...	obj1, obj2, ... each being the result of a call to pwrFDR. See the example below.

Value

The table of joined pwrFDR objects as a data.frame

Author(s)

Grant Izmirlian

Examples

```
rslt.avgp.r15 <- pwrFDR(effect.size = 0.79, average.power=0.80, r.1 = 0.15, alpha = 0.15)
rslt.avgp.r10 <- update(rslt.avgp.r15, r.1 = 0.10)
rslt.avgp.r05 <- update(rslt.avgp.r15, r.1 = 0.05)

join.tbl(rslt.avgp.r15, rslt.avgp.r10, rslt.avgp.r05)
```

logit *Computes the logit transform*

Description

Computes the logit transform for objects of type numeric and objects of class "pwr".

Usage

```
logit(mu)
```

Arguments

mu	A real number on the interval [0, 1]

Value

A numeric equal to the logit of mu, a real number.

Author(s)

Grant Izmirlian <izmirlian at mail.nih.gov>

logitInv *Computes the inverse logit transform*

Description

Computes the inverse logit transform for objects of type numeric and objects of class "pwr".

Usage

```
logitInv(eta)
```

Arguments

eta	Any real number

Value

A numeric equal to the logit inverse of mu, a real number on the interval [0, 1]

Author(s)

Grant Izmirlian <izmirlian at mail.nih.gov>

nna *A helper function– turns a missing column into 'NA's inside of a with statement*

Description

A helper function– turns a missing column into 'NA's inside of a with statement. Included at the user level because its useful in setting up batch jobs, especially since the 'pwrFDR' return argument list varies depending on the manner called.

Usage

```
nna(x)
```

Arguments

x	A named numeric vector component of a data frame which may or may not be present.

Value

Either the values in the component x of the data.frame or NA\'s of equal length

Author(s)

Grant Izmirlian <izmirlian at nih dot gov>

Examples

```
sim.1 <- pwrFDR(effect.size=0.8, n.sample=60, lambda=0.90, r.1=0.05, N.tests=450, alpha=0.15,
                method="sim", FDP.control.method="BHFDX")

sim.2 <- pwrFDR(effect.size=0.8, n.sample=60, lambda=0.90, r.1=0.05, N.tests=450, alpha=0.15,
                method="sim", FDP.control.method="both", control=list(sim.level=2))

with(detail(sim.1)$reps, cbind(R.st/100, nna(R.R)/450))

with(detail(sim.2)$reps, cbind(R.st/100, nna(R.R)/450))
```

paste *The paste operator*

Description

A binary operator shortcut for paste(x,y)

Usage

```
x %,% y
```

Arguments

x	a character string
y	a character string

Value

The concatenated character string

Author(s)

Grant Izmirlian <izmirlian@nih.gov>

Examples

```
library(pwrFDR)
"var" %,% (1:10)
```

pwrFDR *Ensemble power or sample size under selected control of the FDP*

Description

This is a function for calculating two differing notions of power, or deriving sample sizes for specified requisite power in multiple testing experiments under a variety of methods for control of the distribution of the False Discovery Proportion (FDP). More specifically, one can choose to control the FDP distribution according to control of its (i) mean, e.g. the usual BH-FDR procedure, or via the probability that it exceeds a given value, delta, via (ii) the Romano procedure, or via (iii) my procedure based upon asymptotic approximation. Likewise, we can think of the power in multiple testing experiments in terms of a summary of the distribution of the True Positive Proportion (TPP). The package will compute power, sample size or any other missing parameter required for power based upon (i) the mean of the TPP which is the average power (ii) the probability that the TPP exceeds a given value, lambda, via my asymptotic approximation procedure. The theoretical results are described in Izmirlian, G. (2020), and an applied paper describing the methodology with a simulation study is in preparation.

Usage

```
pwrFDR(effect.size, n.sample, r.1, alpha, delta = NULL, groups = 2, N.tests,
    average.power, TPX.power, lambda, type = c("paired","balanced","unbalanced"),
    grpj.per.grp1=NULL, corr.struct=list(type=c("CS-Blocks","Toeplitz-Blocks"),
                                         block.size=NA, rho = NA),
    FDP.control.method = c("BHFDR","BHFDX","Romano","Auto","both","Holm","Hochberg",
                           "Bonferroni"),
    distopt=1, method = c("Theoretical","simulation"), n.sim = 1000, temp.file,
    control=list(tol=1e-08, max.iter=c(1000, 20), sim.level=2, low.power.stop=TRUE,
            FDP.meth.thresh=FDP.cntl.mth.thrsh.def, ast.le.a=TRUE, verb=FALSE,
                   show.footer=TRUE))
```

Arguments

effect.size	The effect size (mean over standard deviation) for test statistics having non-zero means. Assumed to be a constant (in magnitude) over non-zero mean test statistics.
n.sample	The number of experimental replicates. Required for calculation of power
r.1	The proportion of simultaneous tests that are non-centrally located
alpha	The false discovery rate (in the BH case) or the upper bound on the probability that the FDP exceeds delta (BHFDX and Romano case)
delta	If the "FDP.control.method" is set to 'Romano' or 'BHFDX', then the user can set the exceedance thresh-hold for the FDP tail probability control $P\{FDP > \delta\} < \alpha$. The default value is α.
groups	The number of experimental groups to compare. Must be integral and >=1. The default value is 2.
N.tests	The number of simultaneous hypothesis tests.
average.power	The desired average power. Sample size calculation requires specification of either 'average.power' or 'TPX.power'.
TPX.power	The desired tp-power (see details for explanation). Sample size calculation requires specification of either 'average.power' or 'TPX.power'.
lambda	The tp-power threshold, required when calculating the tp-power (see details for explanation) or when calculating the sample size required for tp-power.
type	A character string specifying, in the groups=2 case, whether the test is 'paired', 'balanced', or 'unbalanced' and in the case when groups >=3, whether the test is 'balanced' or 'unbalanced'. The default in all cases is 'balanced'. Left unspecified in the one sample (groups=1) case.
grpj.per.grp1	Required when type="unbalanced", specifies the group 0 to group 1 ratio in the two group case, and in the case of 3 or more groups, the group j to group 1 ratio, where group 1 is the group with the largest effect under the alternative hypothesis.
corr.struct	Specifies a block correlation structure between test statistics which is used in both the simulation routine, and in the computations based upon asymptotic approximation, e.g. the AFDX control and the ATPP method. Its form is specified via the following named elements.

"type": a string taking the value "CS-Blocks" for compound compound symmetry within blocks or "Toeplitz-Blocks" for toeplitz within blocks, respectively.
"block.size": the size of the correlated blocks
"rho": When type="CS-Blocks", then 'rho' is a correlation of length 1 when type="Toeplitz-Blocks", then 'rho' is a vector of correlations of length 'block.size' - 1

`FDP.control.method`

A character string specifying how the false discovery proportion (FDP) is to be controlled. You may specify the whole word or any shortened uniquely identifying truncation.
"BHFDR": the usual BH-FDR
"BHFDX": guarantees control of the FDX e.g. probability that the FDP exceeds delta will be less than alpha. Uses asymptotic approximationto the distribution of the FDP to find alpha* such that the BHFDR procedure controls the FDX at alpha. Only slightly more conservative than BHFDR control at alpha. Allows dependence which must be specified. Not guaranteed to have a solution in all cases, in which case Romano's procedure is used as a fall-back.
"Romano": use Romano's method which guarantees control of the FDX, e.g. the probability that the FDP exceeds delta will be less than alpha.
"Auto": in 'FixedPoint' mode, the program will use its own wisdom to determine which choice above to make. The order of conservatism is
Bonferroni > Holm > Hochberg > Romano > BHFDX > BHFDR,
but BHFDR offers only expected control, the most conservative 3 control the FWER and the other two guarantee bounds on the excedance probabilty. If the distribution of the FDP is nearly degenerate, then BHFDR is the best option. Otherwise, if it can be reliably used, BHFDX would be the best choice. The 'effective' denominator, gamma*N.tests, in the CLT determines when the approximation is good enough and the asymptotic standard error of the FDP determines when the distribution is dispersed enough to matter. Use "Auto" to run through these checks and determine the best. A return argument, 'Auto', displays the choice made. See output components and details.
"both": in 'simulation' mode, compute statistics R and T under BHFDX and Romano (in addition to BHFDR). Corresponding statistics are denoted R.st, T.st corresponding to BHFDX control of the FDP, and R.R and T.R corresponding to Romano control of the FDP. If sim.level is set to 2, (default) the statistics R.st.ht and T.st.ht, which are the number rejected and number true positives under BHFDX where r_0 = 1-r_1, gamma, and alpha.star have been estimated from the P-value data and then alpha.star computed from these.
"Holm": Use Holm's step-down procedure which guarantees control of the family-wise error rate (FWER) "Hochberg": Use Hochberg's step-up procedure which guarantees control of the FWER if the tests are independent. Less conservative than Holm and Bonferroni but more conservative than all other procedures here.
"Bonferroni": Use the Bonferroni procedure. Guarantees control of the FWER. the most conservative of procedures offered here.

`distopt` Test statistic distribution in among null and alternatively distributed sub-populations. distopt=0 gives normal (2 groups), distop=1 gives t- (2 groups) and distopt=2 gives F- (2+ groups)

`method` Specify the method whereby the average power is calculated. You may specify the whole word or any unqiuely indentifying truncation.
"Theoretical": for all procedures but Bonferroni, use the fixed point method, e.g., first find the solution to the equation u = G(alpha u) where G is the CDF of the pooled P-values. This solution gives 'gamma', the positive proportion. The average power and TPX power can then be determined e.g. average.power = G_1(gamma alpha), where G_1 is the CDF of the P-values corresponding to statistics drawn under H_A. TPX power is determined via asymptotic approximation.
"simulation": uses brute force simulation to determine the average power and TPX power.

`control` Optionally, a list with components with the following components:
'tol' is a convergence criterion used in iterative methods which is set to 1e-8 by default.
'max.iter' is an iteration limit, set to 20 for the iterated function limit and 1000 for all others by default.
'sim.level' sim level 2 (default) stipulates, when FDP.control.method is set to "BHFDX", or "both", R.st.ht and T.st.ht are computed in addition to R.st and T.st (see above).
'low.power.stop' in simulation option, will result in an error message if the power computed via FixedPoint method is too low, which result in no solution for the BHFDX option. Default setting is TRUE. Set to FALSE to over-ride this behavior.
'FDP.meth.thresh' fine-tunes the 'Auto' voodoo (see above). Leave this alone.
'ast.le.a' leaving this at the default value TRUE forces 'alpha.star', the solution under FDP.method.control="BHFDX", to be less than the specified 'alpha'.
'verb' vebosity level.
'show.footer' optionally, show a textual output

`n.sim` If 'simulation' method is chosen you may specify number of simulations. Default is 1000.

`temp.file` If 'simulation' method is chosen you may specify a tempfile where the current simulation replicate is updated. Very usefull for batch runs. You can use the included utility 'gentempfilenm'

Details

This function will compute one of a variety of ensemble powers under a given choice of FDP control methods. The underlying model assumes that the m simultaneous test statistics are i.i.d., each being formed from k samples which can be paired (k=2), balanced or unbalanced (k>=2), k=1,2,..., and distributed according to one of the available relevent distribution types (see above). The location parameter for each of the statistical tests is either 0 (null hypothesis) or a specified constant effect size (alternative hypothesis), with the identity of these two possibilities in each of the m cases being an i.i.d. unmeasured latent bernouli variable with density r.1, the mixing proportion. The m simultaneous statistical tests partition into those which are distributed according to the alternative, numbering M_m, and those distributed according to the NULL, numbering m-M_m. Once a selected thresholding method is applied, the m statistics can also be partitioned into those which are called significant, numbering R_m, and those which are not, numbering m-R_m. Each of the test statistics is thus given two labels, alternative hypothesis membership and whether

a significant call was made. Of the R_m significant calls, T_m are true positives and V_m are false positives. This results in the following table.

1.		rej H0	acc H0	row Total
2.	H0 is FALSE	T_m	M.1-T	M_m
3.	H0 is TRUE	R_m-T_m	(m-M_m)-(R_m-T_m)	m - M_m
4.	col Total	R	m-R_m	m

The ratio of the false positive count to the significant call count, V_m/R_m, is called the False Discovery Proportion (FDP). Thresholding methods which result in the most reproducibility seek to control the FDP distribution. The most well known is the Benjamini-Hochberg False Discovery Rate (BH-FDR) procedure. It guarantees that the FDR, which is the expected FDP, will be less than a stipulated alpha

$E[V_m/R_m] < \alpha$

While it is true that for large m, the distribution of the FDP, V_m/M_m will become spiked at its mean, $(1-r_1)\alpha$, in many commonly occuring situations, there will still be non-negligible dispersion in the distribution of the FDP. For this reason, any validity promised by the BH-FDR procedure does not actually apply on a case to case basis, and individual FDP's may differ non-negligibly from the FDR. For this reason, the function supplies two other methods of FDP control in addition to FDP.control.method="BHFDR". These two alternate methods, FDP.control.method="Romano" and FDP.control.method="BHFDX" guarantee control of the tail probability of the FDP distribution:

$P\{V_m/R_m > \delta\} < \alpha$

The lower bound δ is left arbitrary for greater flexibility, $\delta = \alpha$ being the default. There is also an automatic option, FDP.control.method="Auto", which lets the function decide which of the three FDP control methods is the most advisable in a given situation. The two tail probability control options are preferred when the standard error of the FDP exceeds a cutoff given in the default 'control' settings:

se[V_m/R_m] / alpha > FDP.cntl.mth.thrsh.def[1]

The default is 10%. When the standard error to alpha ratio is 10% or less then the BHFDR, being the least conservative, is preferred. When the se to alpha ratio is 10% or more, then Romano and BHFDX are decided between, with the BHFDX (asymptotic approximation) being less conservative than Romano and therefor preferred if the CLT approximation is adequate. This will be the case provided m is large enough,

$m \geq$ FDP.cntl.mth.thrsh.def[2]. The default is 50.

The concept of ensemble power for the purposes of this function, concern the distribution of the true positive proportion (TPP), T_m/M_m. The most well known is the average power, which is the expected value of the TPP, which is called the true positve rate (TPR):

$E[T_m/M_m] = averagepower$

For large m, the distribution of the TPP will be spiked at its mean, which is the asymptotic average power. This is used in the function in the average power computation. As was the case for the FDP, there are many commonly occuring situations when the distribution of the TPP will still be non-negligibly dispersed. For this reason, we provide an alternate notion of power which is based upon the tail probability of the TPP distribution:

$P\{T_m/M_m > \lambda\} = tp - power$

This is computed via asymptotic approximation and also requires that m be large enough: $m > 50$. The user decides when the tp-power is to be preferred. A good check is to look at the ratio of the se[TPP] to average power ratio which is the sigma.rtm.TPP/average.power/N.tests^0.5 If this ratio is unacceptibly large (10% or so) than the tp-power is preferred.

For the "FixedPoint" method (default) and for any specified choice of FDP.control method, the function can be used in the following ways:

1. Specify 'n.sample', 'effect.size', 'r.1' and 'alpha'. Calculates 'average power'

2. Specify 'n.sample', 'effect.size', 'r.1', 'alpha' and 'lambda'. 'N.tests' is also required. The function wil calculate the 'TPX.power' in addition to the 'average power'.

3. Specify the 'average.power' or the pair 'TPX.power' and 'lambda'. Specify all but one of the parameters, 'n.sample', 'effect.size', 'r.1' and 'alpha'. The function will calculate the value of the missing parameter required for the specified 'average power' or 'tp-power'. Note: a solution is guaranteed for missing 'n.sample' and missing 'effect.size', but not necessarily for missing 'r.1' or 'alpha'.

The Holm, Hochberg and Bonferroni procedures for controlling the FWER under the assumption of independent test statistics are also provided for sake of completeness.

Value

An object of class "pwr" with with components including:

`call`	The call which produced the result
`average.power`	Resulting average power.
`TPX.power`	When 'lambda' is specified, the tp-power is also computed
`L.eq`	The lambda at which the tp-power and average-power are equal.
`n.sample`	If 'n.sample' is missing from the argument list, then the sample size required for the specified average- or lambda- power.
`alpha.star`	If 'FDP.control.method' was set to "BHFDX" or it resulted from the "Auto" setting, the alpha at which the probability that the FDP exceeds alpha.star is less than or equal to the originally specified alpha.
`c.g`	The FDP control method threshold on the scale of the test statistics.
`gamma`	The proportion of all 'm' tests declared significant.
`objective`	Result of optimization yielding the average or tp- power.
`err.III`	Mass on the wrong side of the threshold.
`sigma.rtm.ToM`	Asymptotic standard deviation of the true positive fraction.
`Auto`	If 'FDP.control.method' was set to "Auto", this returns the resulting choice (a string) which was made internally.
`se.by.a`	The ratio of the standard error of the FDP to alpha, the nominal FDR, which gives an indication of the dispersion of its distribution relative to the nominal FDP. Used by the "Auto" specification.
`gma.Ntsts`	the effective denominator in the CLT asymptotic approximation to the distribution of the FDP, which equals the positive proportion, 'gamma', times the number of simultaneous tests, 'm'.
`detail`	The extractor function, `detail`, will return simulation replicates. See the linked documentation

Author(s)

Grant Izmirlian <izmirlian at nih dot gov>

References

Izmirlian G. (2020) Strong consistency and asymptotic normality for quantities related to the Benjamini-Hochberg false discovery rate procedure. Statistics and Probability Letters; 108713, <doi:10.1016/j.spl.2020.108713>

Izmirlian G. (2017) Average Power and λ-power in Multiple Testing Scenarios when the Benjamini-Hochberg False Discovery Rate Procedure is Used. <arXiv:1801.03989>

Jung S-H. (2005) Sample size for FDR-control in microarray data analysis. Bioinformatics; 21:3097-3104.

Kluger D. M., Owen A. B. (2023) A central limit theorem for the Benjamini-Hochberg false discovery proportion under a factor model. Bernoulli; xx:xxx-xxx.

Liu P. and Hwang J-T. G. (2007) Quick calculation for sample size while controlling false discovery rate with application to microarray analysis. Bioinformatics; 23:739-746.

Lehmann E. L., Romano J. P.. Generalizations of the familywise error rate. Ann. Stat.. 2005;33(3):1138-1154.

Romano Joseph P., Shaikh Azeem M.. Stepup procedures for control of generalizations of the familywise error rate. Ann. Stat.. 2006;34(4):1850-1873.

See Also

pwrFDR.grid controlFDP

Examples

```
## Example 1a: average power

   rslt.avgp <- pwrFDR(effect.size=0.79, n.sample=46, r.1=2000/54675, alpha=0.15)
   rslt.avgp

## Example 1b: average power, FDP.control.method set to "Auto", N.tests=1000

  rslt.avgp.auto <- pwrFDR(effect.size = 0.79, n.sample = 46, r.1 = 2000/54675, alpha = 0.15,
                           N.tests = 1000, FDP.control.method = "Auto")
   rslt.avgp.auto

## Example 1c: average power, FDP.control.method set to "Auto", N.tests=2000

   rslt.avgp.auto <- update(rslt.avgp.auto, N.tests = 2000)
   rslt.avgp.auto

## Example 1d: tp-power

   rslt.lpwr <- pwrFDR(effect.size=0.79, n.sample=46, r.1=2000/54675,
                       alpha=0.15, lambda=0.80, N.tests=54675)
   rslt.lpwr

## Example 1e: sample size required for given average power
```

```
   rslt.ss.avgp <- pwrFDR(effect.size=0.79, average.power=0.82,
                          r.1=2000/54675, alpha=0.15)
   rslt.ss.avgp

## Example 1f: sample size required for given tp-power

   rslt.ss.lpwr <- pwrFDR(effect.size=0.79, TPX.power=0.82, lambda=0.80,
                          r.1=2000/54675, alpha=0.15, N.tests=54675)
   rslt.ss.lpwr

## Example 1g: simulation

   rslt.sim <- update(rslt.avgp, method="sim", n.sim=500, N.tests=1000)
   rslt.sim

## Example 1h: simulation

   rslt.sim <- update(rslt.avgp, method="sim", FDP.control.method="both",
                      n.sim=500, N.tests=1000)
   rslt.sim

## Example 2: methods for adding, subtracting, multiplying, dividing, exp, log,
## logit and inverse logit

   rslt.avgp - rslt.sim
   logit(rslt.avgp)       ## etc

## Example 3: Compare the asymptotic distribution of T/M with kernel
##            density estimate from simulated data

   pdf <- with(detail(rslt.sim)$reps, density(T/M1))

   med <- with(detail(rslt.sim)$reps, median(T/M1))
   avg <- rslt.sim$average.power
   sd <- rslt.sim$se.ToM

   rng.x <- range(pdf$x)
   rng.y <- range(c(pdf$y, dnorm(pdf$x, mean=avg, sd=sd)))

   plot(rng.x, rng.y, xlab="u", ylab="PDF for T/M", type="n")
   with(pdf, lines(x, y))
   lines(rep(rslt.sim$average.power, 2), rng.y, lty=2)
   lines(pdf$x, dnorm(pdf$x, mean=avg, sd=sd), lty=3)
```

---

pwrFDR.grid *Evaluate* pwrFDR *on a grid.*

Description

Function for evaluating pwrFDR on a factorial design of possible parameters.

Usage

```
pwrFDR.grid(effect.size, n.sample, r.1, alpha, delta, groups, N.tests,
            average.power, TPX.power, lambda, type, grpj.per.grp1,
            corr.struct, FDP.control.method, distopt, control)
```

Arguments

effect.size	A vector of effect sizes to be looped over. The effect size (mean over standard deviation) for test statistics having non-zero means. Assumed to be a constant (in magnitude) over non-zero mean test statistics.
n.sample	A vector of sample sizes to be looped over. The sample size is the number of experimental replicates. Required for calculation of power
r.1	A vector of mixing proportions to be looped over. The mixing proportion is the proportion of simultaneous tests that are non-centrally located
alpha	The false discovery rate (in the BH case) or the upper bound on the probability that the FDP exceeds delta (BHFDX and Romano case)
delta	If the "FDP.control.method" is set to 'Romano' or 'BHFDX', then this optional argument can be set to the exceedance thresh-hold in defining the FDP-tp: $P\{FDP > \delta\} < \alpha$. The default value is α.
groups	The number of experimental groups to compare. Must be integral and >=1. The default value is 2.
N.tests	The number of simultaneous hypothesis tests.
average.power	The desired average power. Calculation of sample size, effect size mixing proportion or alpha requires specification of either 'average.power' or 'TPX.power'.
TPX.power	The desired tp-power (see pwrFDR documentation). Calculation of sample size, effect size mixing proportion or alpha requires specification of either 'average.power' or 'TPX.power'.
lambda	The tp-power threshold, required when calculating the tp-power (see pwrFDR documentation) or when calculating the sample size, effect size mixing proportion or alpha required for tp-power.
type	A character string specifying, in the groups=2 case, whether the test is 'paired', 'balanced', or 'unbalanced' and in the case when groups >=3, whether the test is 'balanced' or 'unbalanced'. The default in all cases is 'balanced'. Left unspecified in the one sample (groups=1) case.
grpj.per.grp1	Required when type="unbalanced", specifies the group 0 to group 1 ratio in the two group case, and in the case of 3 or more groups, the group j to group 1 ratio, where group 1 is the group with the largest effect under the alternative hypothesis.
corr.struct	Specifies a block correlation structure between test statistics which is used in both the simulation routine, and in the computations based upon asymptotic approximation, e.g. the AFDX control and the ATPP method. Its form is specified via the following named elements. "type": "CS-Blocks" for compound symmetry within blocks, or "Toeplitz-Blocks" for toeplitz blocks. "block.size": the size of the correlated blocks "rho": When type="CS-Blocks",

then 'rho' is a correlation of length 1 when type="Toeplitz-Blocks", then 'rho' is a vector of correlations of length 'block.size' - 1

`FDP.control.method`

A character string specifying how the false discovery proportion (FDP) is to be controlled. You may specify the whole word or any shortened uniquely identifying truncation.
"BHFDR": the usual BH-FDR
"BHFDX": guarantees control of the FDX e.g. probability that the FDP exceeds delta will be less than alpha. Uses asymptotic approximationto the distribution of the FDP to find alpha* such that the BHFDR procedure controls the FDX at alpha. Only slightly more conservative than BHFDR control at alpha. Allows dependence which must be specified. Not guaranteed to have a solution in all cases, in which case Romano's procedure is used as a fall-back.
"Romano": use Romano's method which guarantees control of the FDX, e.g. the probability that the FDP exceeds delta will be less than alpha.
"Auto": in 'FixedPoint' mode, the program will use its own wisdom to determine which choice above to make. The order of conservatism is
Bonferroni > Holm > Hochberg > Romano > BHFDX > BHFDR,
but BHFDR offers only expected control, the most conservative 3 control the FWER and the other two guarantee bounds on the excedance probabilty. If the distribution of the FDP is nearly degenerate, then BHFDR is the best option. Otherwise, if it can be reliably used, BHFDX would be the best choice. The 'effective' denominator, gamma*N.tests, in the CLT determines when the approximation is good enough and the asymptotic standard error of the FDP determines when the distribution is dispersed enough to matter. Use "Auto" to run through these checks and determine the best. A return argument, 'Auto', displays the choice made. See output components and details.
"both": in 'simulation' mode, compute statistics R and T under BHFDX and Romano (in addition to BHFDR). Corresponding statistics are denoted R.st, T.st corresponding to BHFDX control of the FDP, and R.R and T.R corresponding to Romano control of the FDP. If sim.level is set to 2, (default) the statistics R.st.ht and T.st.ht, which are the number rejected and number true positives under BHFDX where r_0 = 1-r_1, gamma, and alpha.star have been estimated from the P-value data and then alpha.star computed from these.
"Holm": Use Holm's step-down procedure which guarantees control of the family-wise error rate (FWER)
"Hochberg": Use Hochberg's step-up procedure which guarantees control of the FWER if the tests are independent. Less conservative than Holm and Bonferroni but more conservative than all other procedures here.
"Bonferroni": Use the Bonferroni procedure. Guarantees control of the FWER. the most conservative of procedures offered here.

`distopt` Test statistic distribution in among null and alternatively distributed sub-populations. distopt=0 gives normal (2 groups), distop=1 gives t- (2 groups) and distopt=2 gives F- (2+ groups)

`control` Optionally, a list with components with the following components:
'tol' is a convergence criterion used in iterative methods which is set to 1e-8 by default.
'max.iter' is an iteration limit, set to 20 for the iterated function limit and 1000

for all others by default.
'sim.level' sim level 2 (default) stipulates, when FDP.control.method is set to "BHFDX", or "both", R.st.ht and T.st.ht are computed in addition to R.st and T.st (see above).
'low.power.stop' in simulation option, will result in an error message if the power computed via FixedPoint method is too low, which result in no solution for the BHFDX option. Default setting is TRUE. Set to FALSE to over-ride this behavior.
'FDP.meth.thresh' fine-tunes the 'Auto' voodoo (see above). Leave this alone.
'verb' vebosity level.
'ast.le.a' leaving this at the default value TRUE forces 'alpha.star', the solution under FDP.method.control="BHFDX", to be less than the specified 'alpha'.

Details

Arguments may be specified as vectors of possible values or can be set to a single constant value.

Value

A list having two components:

`conditions`	A data.frame with one column for each argument listing the distinct settings for all parameters.
`results`	A list with components objects of class `pwr`, the results of the calls to `pwrFDR`

Author(s)

Grant Izmirlian <izmirlian at nih dot gov>

References

Izmirlian G. (2020) Strong consistency and asymptotic normality for quantities related to the Benjamini-Hochberg false discovery rate procedure. Statistics and Probability Letters; 108713, <doi:10.1016/j.spl.2020.108713>

Izmirlian G. (2017) Average Power and λ-power in Multiple Testing Scenarios when the Benjamini-Hochberg False Discovery Rate Procedure is Used. <arXiv:1801.03989>

Jung S-H. (2005) Sample size for FDR-control in microarray data analysis. Bioinformatics; 21:3097-3104.

Kluger D. M., Owen A. B. (2023) A central limit theorem for the Benjamini-Hochberg false discovery proportion under a factor model. Bernoulli; xx:xxx-xxx.

Liu P. and Hwang J-T. G. (2007) Quick calculation for sample size while controlling false discovery rate with application to microarray analysis. Bioinformatics; 23:739-746.

Lehmann E. L., Romano J. P.. Generalizations of the familywise error rate. Ann. Stat.. 2005;33(3):1138-1154.

Romano Joseph P., Shaikh Azeem M.. Stepup procedures for control of generalizations of the familywise error rate. Ann. Stat.. 2006;34(4):1850-1873.

See Also

pwrFDR.grid controlFDP

Examples

```
tst <- pwrFDR.grid(effect.size=c(0.6,0.9), n.sample=c(50,60,70), r.1=0.4+0.2*(0:1),
                   alpha=0.05+0.05*(0:3), N.tests=1000, FDP.control.method="Auto")
```

---

sd.rtm.Rom *Extractor function for asymptotic sd[R_m/m] under selected FDP control method*

Description

A function which extracts the asymptotic standard deviation for the postive call proportion, R_m/m, under the selected FDP control method from the supplied pwr object, which is the result of a call to the main function, pwrFDR.

Usage

```
sd.rtm.Rom(object)
```

Arguments

object	An object of class, pwr, which is the result of a call to the main function, pwrFDR

Details

The siginificant call proportion (SCP), R_m/m, under the selected FDP control method, is directly related to the ensemble power, which in turn, is determined by the effect size for tests distributed under the alternative, the sample size, the proportion of tests which are distributed according to the alternative and the size, alpha, in the selected FDP control method. Its asymptotic standard error, e.g. the asymptotic standard deviation over the square root of the number of simultaneous tests, m, gives an indication of the range of values one can expect for the significant call proportion. The standard deviations of the ratios R_m/m, T_m/M_m, and V_m/R_m are used internally in control of the distribution of V_m/R_m for the BHFDX FDP control method, and in calculation of the tail probability power for T_m/M_m.

Value

Returns the asymptotic standard deviation of the significant call proportion, sd[R_m/m], as an unnamed numeric.

Author(s)

Grant Izmirlian <izmirlig at mail dot nih dot gov>

References

Izmirlian G. (2020) Strong consistency and asymptotic normality for quantities related to the Benjamini-Hochberg false discovery rate procedure. Statistics and Probability Letters; 108713, <doi:10.1016/j.spl.2020.108713>.

Izmirlian G. (2017) Average Power and λ-power in Multiple Testing Scenarios when the Benjamini-Hochberg False Discovery Rate Procedure is Used. <arXiv:1801.03989>

Kluger D. M., Owen A. B. (2023) A central limit theorem for the Benjamini-Hochberg false discovery proportion under a factor model. Bernoulli; xx:xxx-xxx.

See Also

sd.rtm.VoR sd.rtm.ToM

Examples

```
rslt.BHFDR  <- pwrFDR(effect.size=0.79, n.sample=46, r.1=0.05, alpha=0.15)
rslt.Auto.1 <- pwrFDR(effect.size=0.79, n.sample=46, r.1=0.05, alpha=0.15, N.tests=51,
                      FDP.control.method="Auto")
rslt.Auto.2 <- pwrFDR(effect.size=0.79, n.sample=46, r.1=0.05, alpha=0.15, N.tests=49,
                      FDP.control.method="Auto")

## Asymptotic standard deviation of positive call proportion under BHFDR
sdrtmRomBHFDR  <- sd.rtm.Rom(rslt.BHFDR)

## Asymptotic standard deviation of positive call proportion under BHFDX
sdrtmRomAuto1 <- sd.rtm.Rom(rslt.Auto.1)

## Asymptotic standard deviation of positive call proportionunder Romano
sdrtmRomAuto2 <- sd.rtm.Rom(rslt.Auto.2)
```

---

sd.rtm.ToM *Extractor function for asymptotic sd[T_m/M_m] under selected FDP control method*

Description

A function which extracts the asymptotic standard deviation for the true positive proportion, T_m/M_m, under the selected FDP control method from the supplied pwr object, which is the result of a call to the main function, pwrFDR.

Usage

```
sd.rtm.ToM(object)
```

Arguments

object	An object of class, pwr, which is the result of a call to the main function, pwrFDR

Details

The true positive proportion (TPP), T_m/M_m, is the proportion of all test statistics distributed according to the alternative that are declared significant by the selected FDP control method. Whereas the ensemble type I error in the multiple testing experiment is handled via control of the distribution of the FDP, V_m/R_m, the ensemble power is optimized via the distribution of the TPP. The most commonly used ensemble power is based upon the expected TPP, or true postive rate, E[TPP], which also called the average power. In situations of just adequate power or near adequate power, especially when testing less than 1000 simultaneous tests or so, the distribution of the TPP will be non-negligiby dispersed and this means that the TPP in a given multiple testing experiment for which sample size was based on the average power will likely not be close to the promised average power. For this reason, it is preferable to use a concept of ensemble power which is based upon the excedance probability for the TPP, or tail probability of the TPP (tp-TPP).

$P(TPP > \lambda) >= 1 - \epsilon_{II}$

This package uses asymptotic approximation to derive the tp-TPP ensemble power under any one of the avaialbe FDP control methods, BHFDR, BHFDX or Romano.

Value

Returns the asymptotic standard deviation of the true postive proportion, sd[T_m/M_m], as an unnamed numeric.

Author(s)

Grant Izmirlian <izmirlig at mail dot nih dot gov>

References

Izmirlian G. (2020) Strong consistency and asymptotic normality for quantities related to the Benjamini-Hochberg false discovery rate procedure. Statistics and Probability Letters; 108713, <doi:10.1016/j.spl.2020.108713>.

Izmirlian G. (2017) Average Power and λ-power in Multiple Testing Scenarios when the Benjamini-Hochberg False Discovery Rate Procedure is Used. <arXiv:1801.03989>

Kluger D. M., Owen A. B. (2023) A central limit theorem for the Benjamini-Hochberg false discovery proportion under a factor model. Bernoulli; xx:xxx-xxx.

See Also

sd.rtm.Rom sd.rtm.VoR

Examples

```
rslt.BHFDR  <- pwrFDR(effect.size=0.79, n.sample=46, r.1=0.05, alpha=0.15)
rslt.Auto.1 <- pwrFDR(effect.size=0.79, n.sample=46, r.1=0.05, alpha=0.15, N.tests=51,
                      FDP.control.method="Auto")
rslt.Auto.2 <- pwrFDR(effect.size=0.79, n.sample=46, r.1=0.05, alpha=0.15, N.tests=49,
                      FDP.control.method="Auto")

## Asymptotic standard deviation under BHFDR
sdrtmToMBHFDR  <- sd.rtm.ToM(rslt.BHFDR)
```

```
## Asymptotic standard deviation under BHFDX
sdrtmToMAuto1 <- sd.rtm.ToM(rslt.Auto.1)

## Asymptotic standard deviation under Romano
sdrtmToMAuto2 <- sd.rtm.ToM(rslt.Auto.2)
```

`sd.rtm.VoR` *Extractor function for asymptotic sd[V_m/R_m] under selected FDP control method*

Description

A function which extracts the asymptotic standard deviation for the false discovery proportion, V_m/R_m, under the selected FDP control method from the supplied `pwr` object, which is the result of a call to the main function, pwrFDR.

Usage

```
sd.rtm.VoR(object)
```

Arguments

`object`	An object of class, `pwr`, which is the result of a call to the main function, pwrFDR

Details

The false discovery proportion (FDP), V_m/R_m, under the selected FDP control method, is the proportion of null distributed test statistics that were deemed significant calls by the FDP control method. The most well known of available FDP methods is the Benjamini-Hochberg False Discovery Rate (BH-FDR) procedure. It ensures that the expected value of the FDP will be less than alpha, E[FDP] < alpha. The other two included FDP control methods, "Romano" and "BHFDX", control the probability that the FDP exceeds a given value, delta:

$P(V_m/R_m > \delta) < \alpha$

In most cases, the choice $\delta = \alpha$ is appropriate but δ is a distinct parameter to allow greater flexibility. The choice "Auto" will select the most appropriate choice from the three, BHFDR, BHFDX and Romano. If the asymptotic standard error, sd.rtm.VoR/m^0.5 is greater than a control parameter (default value 10%), then one of the choices "BHFDX" or "Romano" will be made. As the "Romano" FDP control method is more conservative, there is a preference for the "BHFDX" method, which can be used if the number of simultaneous tests, m, is larger than 50. All of this is handled internally within the function `pwrFDR`. These extractor functions exist to allow the user 'under the hood'.

Value

Returns the asymptotic standard deviation of the false discovery proportion, sd[V_m/R_m], as an un-named numeric.

Author(s)

Grant Izmirlian <izmirlig at mail dot nih dot gov>

References

Izmirlian G. (2020) Strong consistency and asymptotic normality for quantities related to the Benjamini-Hochberg false discovery rate procedure. Statistics and Probability Letters; 108713, <doi:10.1016/j.spl.2020.108713>.

Izmirlian G. (2017) Average Power and λ-power in Multiple Testing Scenarios when the Benjamini-Hochberg False Discovery Rate Procedure is Used. <arXiv:1801.03989>

Kluger D. M., Owen A. B. (2023) A central limit theorem for the Benjamini-Hochberg false discovery proportion under a factor model. Bernoulli; xx:xxx-xxx.

See Also

sd.rtm.Rom sd.rtm.ToM

Examples

```
rslt.BHFDR  <- pwrFDR(effect.size=0.79, n.sample=46, r.1=0.05, alpha=0.15)
rslt.Auto.1 <- pwrFDR(effect.size=0.79, n.sample=46, r.1=0.05, alpha=0.15, N.tests=51,
                      FDP.control.method="Auto")
rslt.Auto.2 <- pwrFDR(effect.size=0.79, n.sample=46, r.1=0.05, alpha=0.15, N.tests=49,
                      FDP.control.method="Auto")

## Asymptotic standard deviation under BHFDR
sdrtmVoRBHFDR  <- sd.rtm.VoR(rslt.BHFDR)

## Asymptotic standard deviation under BHFDX
sdrtmVoRAuto1 <- sd.rtm.VoR(rslt.Auto.1)

## Asymptotic standard deviation under Romano
sdrtmVoRAuto2 <- sd.rtm.VoR(rslt.Auto.2)
```

%over% *Division operator with divide by zero clobbering*

Description

x %over% y = x/y when y!=0, equals 0 when y==0.

Usage

```
x %over% y
```

Arguments

x, y	Numeric or complex vectors or objects that can be coerced to such.

Value

x/y when y!=0, otherwise 0.

Author(s)

Grant Izmirlian izmirlig at mail dot nih dot gov

Index