

Package ‘howmany’

February 20, 2015

Type Package

Title A lower bound for the number of correct rejections

Version 0.3-1

Date 2012-06-01

Author Nicolai Meinshausen

Maintainer Nicolai Meinshausen <meinshausen@stats.ox.ac.uk>

Imports graphics, stats

Description When testing multiple hypotheses simultaneously, this package provides functionality to calculate a lower bound for the number of correct rejections (as a function of the number of rejected hypotheses), which holds simultaneously -with high probability- for all possible number of rejections. As a special case, a lower bound for the total number of false null hypotheses can be inferred. Dependent test statistics can be handled for multiple tests of associations. For independent test statistics, it is sufficient to provide a list of p-values.

License GPL

URL <http://www.stats.ox.ac.uk/~meinshau/>

Repository CRAN

Date/Publication 2012-06-01 16:06:00

NeedsCompilation no

R topics documented:

howmany	2
howmany_dependent	3
lowerbound	5
plot.howmany	6
print.howmany	7
summary.howmany	8

Index	9
--------------	----------

howmany	<i>Number of correct rejections, for independent test statistics</i>
---------	--

Description

Lower bounds for the number of correct rejections, for independent test statistics.

Usage

```
howmany(pvalues, alpha = 0.05, cutoff = 0.05/length(pvalues), m=length(pvalues))
```

Arguments

pvalues	a numeric vector of p-values
alpha	the level, a scalar in [0,1]
cutoff	a scalar in [0,1]
m	total number of tests made

Details

When testing multiple hypotheses simultaneously (test statistics are supposed to be independent), a quantity of interest is the number of correctly rejected hypotheses. Given a list of p-values, the function provides a lower bound for the number of correct rejections, which is simultaneously valid for all possible number of rejections. The bound is monotonically increasing with the number of made rejections.

The level is asymptotically valid (for a large number of tested hypotheses). To ensure better small sample behaviour, it is recommended to truncate p-values by setting a non-zero value of cutoff. For a value c of cutoff, p-values below c are set to c .

For computational efficiency, only the most significant p-values can be supplied and the total number of tests made must then be given with the argument m .

Value

An object of class `howmany`, for which `summary`, `plot`, and `print` methods are available.

The lower bound for the number of correct rejections (as a function of the number of rejections) can be accessed with the function [lowerbound](#).

Author(s)

Nicolai Meinshausen, <nicolai@stat.berkeley.edu>

References

- N. Meinshausen and J. Rice (2006) "Estimating the proportion of false discoveries among a large number of independently tested hypotheses", *Annals of Statistics* 34(1), 373-393
- N. Meinshausen (2006) "False discovery control for multiple tests of association under general dependence", *Scandinavian Journal of Statistics* 33(2), 227-237
- N. Meinshausen and P. Buhlmann (2005) "Lower bounds for the number of false null hypotheses for multiple testing of associations", *Biometrika* 92(4), 893-907

See Also

[lowerbound](#) for extracting the number of correct rejections (as a function of the number of made rejections).

[howmany_dependent](#) for similar functionality for multiple tests of associations, where test statistics can be dependent.

Examples

```
## create a list of pvalues,
## of which 1000 are uniform on [0,1]
## (1000 true null hypotheses),
## and 200 follow a (truncated) chi-squared distribution
## (200 false null hypotheses).
pvalues <- c( runif(300), pmin(1,0.05*rchisq(50,df=1)) )

## compute object of class 'howmany' and print the result
(object <- howmany(pvalues))

## extract the lower bound
(lower <- lowerbound(object))

## plot the result
plot(object)

## for comparison: number of rejections with Bonferroni's correction
(bonf <- sum( pvalues < (0.05/1200) ))
```

howmany_dependent *Number of correct rejections, for dependent test statistics*

Description

Lower bounds for the number of correct rejections, for multiple tests of associations with dependent test statistics.

Usage

```
howmany_dependent(X, Y, alpha = 0.05, test = wilcox.test,
                  alternative = "two.sided", n.permutation=round(20/alpha) )
```

Arguments

X	a n*p matrix, where each of the p columns contains n observations
Y	a factor or numerical vector of length n, containing a binary class variable
alpha	the level, a scalar in [0,1]
test	the test to be used
alternative	an alternative for the test, supplied as an argument to function test
n.permutation	the number of permutations to use to determine the bounding function

Details

For multiple tests of associations (with possibly dependent test statistics), a lower bound for the number of correct rejections is calculated, which is valid simultaneously for all possible number of rejections and under arbitrary dependence between test statistics. The bound is monotonically increasing with the number of made rejections.

The matrix X contains the observations, while Y contains binary class labels. For each hypothesis $k=1, \dots, p$, a p-value is calculated internally according to supplied function test, with first argument $X[Y==0, k]$, second argument $X[Y==1, k]$, and additional argument alternative (if the class labels are not 0 and 1, they are converted accordingly). The object returned by test has to have a component `p.value`, containing (perhaps unsurprisingly) the p-value of the corresponding test.

The focus of the current implementation is on portability, not speed, and computations might take some time.

Value

An object of class `howmany`, for which `summary`, `plot`, and `print` methods are available.

The lower bound for the number of correct rejections (as a function of the number of rejections) can be accessed with the function [lowerbound](#).

Author(s)

Nicolai Meinshausen, <nicolai@stat.berkeley.edu>

References

- N. Meinshausen and J. Rice (2006) "Estimating the proportion of false discoveries among a large number of independently tested hypotheses", *Annals of Statistics* 34(1), 373-393
- N. Meinshausen (2006) "False discovery control for multiple tests of association under general dependence", *Scandinavian Journal of Statistics* 33(2), 227-237
- N. Meinshausen and P. Bühlmann (2005) "Lower bounds for the number of false null hypotheses for multiple testing of associations", *Biometrika* 92(4), 893-907

See Also

[lowerbound](#) for extracting the number of correct rejections (as a function of the number of made rejections).

[howmany](#) for similar functionality for multiple tests of associations, where test statistics are independent.

Examples

```
## Warning: running example might take a
## few minutes of computing time...

## create observation matrix X
## for p=200 hypotheses with n=40 observations
p <- 200
n <- 40
Indep <- matrix( rnorm(p*n) , ncol= p )
C <- diag(p); C <- C+matrix( 0.01*rbinom(p^2,1,0.2) , ncol=p )
X <- Indep%*%C

## create binary class variables Y
Y <- c( rep(0,round(n/2)), rep(1,n-round(n/2)) )

## 100 false null hypotheses (random effects)
for (k in 1:100){ X[Y==1, k] <- X[Y==1, k] + rnorm(1) }

## compute object of class 'howmany' and print the result
(object <- howmany_dependent(X,Y))

## extract the lower bound
(lower <- lowerbound(object))

## plot the result
plot(object)

## for comparison: number of rejections with Bonferroni correction
(bonf <- sum(object$pvalues<0.05/p))
```

lowerbound

The lower bound for the number of correct rejections

Description

Extracts the lower bound for the number of false rejections (as a function of the total number of made rejections) from an object of class 'howmany'.

Usage

```
lowerbound(object)
```

Arguments

object an object of class 'howmany'

Value

A numerical vector, giving the lower bound for the number of correct rejections as a function of the number of made rejections.

Author(s)

Nicolai Meinshausen, <nicolai@stat.math.ethz.ch>

See Also

[howmany](#) and [howmany_dependent](#) for multiple tests with independent and dependent test statistics respectively.

```
plot.howmany
```

Plot of an object of class 'howmany'

Description

Displays information stored in objects of class howmany

Usage

```
## S3 method for class 'howmany'
plot(x, ...)
```

Arguments

x an object of class howmany
 ... additional arguments, currently un-implemented

Details

There are four plots.

The upper-left plot displays the empirical distribution function of p-values along with the linear function and the bounding function at the chosen level alpha (empirical distributions functions, where only true null hypotheses are present, are below the bounding function with high probability 1-alpha).

The upper-right plot displays the same information as the upper-left plot, yet with the linear component subtracted.

The lower-left plot shows the lower bound for the number of correct rejections as a function of the number of rejected hypotheses.

The lower-right plot shows the lower bound for the number of correct rejections divided by the number of made rejections.

See Also

[howmany](#) and [howmany_dependent](#); [summary.howmany](#)

print.howmany

Print an object of class 'howmany'

Description

Displays summary information about an object of class howmany

Usage

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

Arguments

x	an object of class howmany
...	additional arguments, currently un-implemented

Details

Currently, invokes the summary method

See Also

[howmany](#) and [howmany_dependent](#); [summary.howmany](#) and [plot.howmany](#)

summary.howmany	<i>Summary of an object of class 'howmany'</i>
-----------------	--

Description

Displays summary information about an object of class howmany

Usage

```
## S3 method for class 'howmany'  
summary(object,...)
```

Arguments

object	an object of class howmany
...	additional arguments, currently un-implemented

See Also

[howmany](#) and [howmany_dependent](#); [plot.howmany](#)

Index

*Topic **htest**

- howmany, [2](#)
- howmany_dependent, [3](#)
- lowerbound, [5](#)
- plot.howmany, [6](#)
- print.howmany, [7](#)
- summary.howmany, [8](#)

howmany, [2](#), [5–8](#)

howmany_dependent, [3](#), [3](#), [6–8](#)

lowerbound, [2–5](#), [5](#)

plot.howmany, [6](#), [7](#), [8](#)

print.howmany, [7](#)

summary.howmany, [7](#), [8](#)