

Package ‘exactci’

February 14, 2012

Type Package

Title Exact P-values and Matching Confidence Intervals for simple Discrete Parametric Cases

Version 1.1-0.1

Date 2010-07-30

Author M.P. Fay

Maintainer M.P. Fay <mfay@niaid.nih.gov>

Depends stats

Description Calculates exact tests and confidence intervals for one-sample binomial and one- or two-sample Poisson cases.

License GPL-2

LazyLoad yes

Repository CRAN

Date/Publication 2010-07-30 15:58:10

R topics documented:

exactci-package	2
binom.exact	3
binomControl	5
exactbinomPlot	6
exactci-internal	7
exactpoissonPlot	8
poisson.exact	10

Index	12
--------------	-----------

Description

Calculates exact binomial and Poisson tests giving matching confidence intervals. There are 3 different methods for defining the two-sided p-values.

Details

Package:	exactci
Type:	Package
Version:	1.0
Date:	2010-01-04
License:	GPL2
LazyLoad:	yes

Although [binom.test](#) and [poisson.test](#) give exact tests and confidence intervals, for two-sided tests the confidence intervals (CI) are not formed by inverting the tests. Thus, there may be test-CI inconsistencies whereby the test rejects but the confidence interval contains the null parameter. The `exactci` package eliminates many of these test-CI inconsistencies for two-sided tests, by outputting the matching confidence interval with each test. The package uses one of three different methods for defining the two-sided p-value. The main functions of the package are [binom.exact](#) and [poisson.exact](#) which follow the same format as `binom.test` and `poisson.test` except have the option 'tmethod' to define the two-sided method for calculating the p-values, and give matching confidence intervals (i.e., ones that come from the inversion of the p-values).

Author(s)

M.P. Fay

Maintainer: Mike Fay <mfay@niaid.nih.gov>

References

- Blaker, H. (2000) Confidence curves and improved exact confidence intervals for discrete distributions. *Canadian Journal of Statistics* 28: 783-798.
- Fay, M. P. (2010). Confidence intervals that Match Fisher's exact and Blaker's exact tests. *Bio-statistics*. 11:373-374.
- Fay, M.P. (2010). Two-sided Exact Tests and Matching Confidence Intervals for Discrete Data. *R Journal* 2(1): 53-58.
- Hirjim K. F. (2006). *Exact analysis of discrete data*. Chapman and Hall/CRC. New York.

See Also

For comparisons of two binomial groups see [exact2x2](#)

Examples

```
## Note binom.test calculates p-values using principle of minimum likelihood
## while it calculates the central confidence intervals. That is why the
## inferences do not match in this example.
binom.test(10,12,p=20000/37877)
binom.exact(10,12,p=20000/37877,tsmethod="minlike")
binom.exact(10,12,p=20000/37877,tsmethod="central")
## We also allow the method studied in Blaker (2000)
binom.exact(10,12,p=20000/37877,tsmethod="blaker")
```

binom.exact	<i>Exact tests with matching confidence intervals for single binomial parameter</i>
-------------	---

Description

Calculates exact p-values and confidence intervals for a single binomial parameter. This is different from `binom.test` only when `alternative='two.sided'`, in which case `binom.exact` gives three choices for tests based on the `'tsmethod'` option. The resulting p-values and confidence intervals will match.

Usage

```
binom.exact(x, n, p = 0.5,
  alternative = c("two.sided", "less", "greater"),
  tsmethod = c("central", "minlike", "blaker"),
  conf.level = 0.95,
  control=binomControl(),plot=FALSE)
```

Arguments

x	number of successes, or a vector of length 2 giving the numbers of successes and failures, respectively.
n	number of trials, ignored if x has length 2.
p	hypothesized probability of success.
alternative	indicates the alternative hypothesis and must be one of "two.sided", "greater" or "less". You can specify just the initial letter.
tsmethod	indicates the method for a two-sided alternative hypothesis and must be one of "minlike", "central" or "blaker". You can specify just the initial letter.
conf.level	confidence level for the returned confidence interval.
control	list with settings to avoid problems with ties, etc, should not need to change this for normal use, see binomControl
plot	logical, do basic plot of p-value function by null hypothesis value, see exactbinomPlot for more plot options

Details

Traditionally, hypothesis tests and confidence intervals are treated separately. A more unified approach suggested by Hirji (2006) is to use the same p-value function to create confidence intervals. There is essentially only one way to calculate one-sided p-values and confidence intervals so these methods are the same in `binom.test` and `binom.exact`. However, there are three main ways that `binom.exact` allows for defining two-sided p-values.

`minlike`: sum probabilities of all likelihoods equal or less than observed
`central`: double minimum one-sided p-value
`blaker`: combine smaller observed tail probability with opposite tail not greater than observed tail

The 'minlike' method is the p-value that has been used in `binom.test`, and 'blaker' is described in Blaker (2000) or Hirji (2006), where it is called the 'combined tails' method. Once the p-value function is defined we can invert the test to create 'matching' confidence intervals defined as the smallest interval that contains all parameter values for which the two-sided hypothesis test does not reject. There are some calculation issues for the 'minlike' and 'blaker' methods which are the same as for exact tests for 2x2 tables (see Fay, 2010).

Value

An object of class 'htest': a list with items

<code>p.value</code>	p-value
<code>conf.int</code>	confidence interval, see attributes 'conf.level' and perhaps 'conf.limit.prec'
<code>statistic</code>	number of successes
<code>parameter</code>	number of trials
<code>estimate</code>	observed proportion of success
<code>null.value</code>	null hypothesis probability of success, 'p'
<code>alternative</code>	a character string describing alternative hypothesis
<code>method</code>	a character string describing method
<code>data.name</code>	a character string giving the names of the data

Note

The 'central' method gives the Clopper-Pearson intervals, and the 'minlike' method gives confidence intervals proposed by Stern (1954) (see Blaker, 2000). The 'blaker' method is guaranteed to be more powerful than the 'central' method (see Blaker, 2000, Corollary 1), but both the 'blaker' method and 'minlike' method may have some undesirable properties. For example, there are cases where adding an additional Bernoulli observation REGARDLESS OF THE RESPONSE will increase the p-value, see Vos and Hudson (2008). The 'central' method does not have those undesirable properties.

The Blyth-Still-Casella intervals given in `StatXact` (and not by `binom.exact`) are the shortest possible intervals, but those intervals are not nested. This means that the Blyth-Still-Casella intervals are not guaranteed to have the 95 percent interval contain the 90 percent interval. See Blaker (2000) Theorem 2.

Author(s)

M.P. Fay

References

Blaker, H. (2000) Confidence curves and improved exact confidence intervals for discrete distributions. *Canadian Journal of Statistics* 28: 783-798.

Fay, M. P. (2010). Confidence intervals that Match Fisher's exact and Blaker's exact tests. *Biostatistics*. 11:373-374.

Fay, M.P. (2010). Two-sided Exact Tests and Matching Confidence Intervals for Discrete Data. *R Journal* 2(1): 53-58.

Hirjim K. F. (2006). Exact analysis of discrete data. Chapman and Hall/CRC. New York.

Stern, T (1954). Some remarks on confidence and fiducial limits. *Biometrika*, 275-278.

Vos, P.W. and Hudson, S. (2008). Problems with binomial two-sided tests and the associated confidence intervals. *Aust. N.Z. J. Stat.* 50: 81-89.

See Also

[binom.test](#), for two-sample exact binomial tests see [exact2x2](#)

Examples

```
## Notice how binom.test p-value is given by tsmethod='minlike'
## but the confidence interval is given by tsmethod='central'
## in binom.exact p-values and confidence intervals match
binom.test(10,12,p=20000/37877)
binom.exact(10,12,p=20000/37877,tsmethod="minlike")
binom.exact(10,12,p=20000/37877,tsmethod="central")
binom.exact(10,12,p=20000/37877,tsmethod="blaker")
## two-sided methods are also available
## as in binom.test
```

binomControl

Tuning parameters for binom.exact function

Description

This function produces a list of tuning parameters used in the calculations done by [binom.exact](#) and [poisson.exact](#). These will not need to be changed by most ordinary users.

Usage

```
binomControl(relErr=1+1e-07, tol=.00001, pRange=c(1e-10, 1-1e-10))
```

Arguments

relErr	value very close to 1, used in calculation of two-sided p-values
tol	value very close to 0, used in calculation of two-sided confidence intervals
pRange	range close to [0,1], but excluding the endpoints, used in calculation of two-sided confidence intervals

Details

See the code for `fisher.test`, where the term `relErr` is hard-coded into the function. The purpose is to avoid problems with ties. It serves the same purpose in this package and probably need not be changed. The value `tol` indicates the tolerance for the precision of the confidence limits. The value `pRange` is input into `uniroot` to give bounds when searching for confidence limits. For poisson limits `pRange` is transformed using the `qgamma` function (see code in `exactpoissonCI`).

Value

A list with containing the following components:

relErr	a number larger than 1
tol	a number greater than 0
pRange	a vector with 2 elements between 0 and 1, exclusive

exactbinomPlot	<i>Plot p-value function for one binomial response.</i>
----------------	---

Description

Plots p-values as a function of different point null hypothesis values for p. For two-sided p-values, can plot three types of p-values: the minimum likelihood method (default for `binom.test`), the central method (i.e., twice the one-sided exact p-values), and Blaker's exact.

Usage

```
exactbinomPlot(x, n, p = NULL, ndiv = 1000, tsmethod = "central", pRange = c(0, 1),
  dolines = FALSE, dopoints = TRUE, doci=TRUE, alternative=c("two.sided", "less", "greater"),
  relErr=1+10^(-7), conf.level=.95, alphaline=TRUE, newplot = TRUE, ...)
```

Arguments

x	number of successes, or a vector of length 2 giving the numbers of successes and failures, respectively
n	number of trials, ignored if x has length 2
p	null values of p for plot, if NULL divides pRange into ndiv pieces
ndiv	number of pieces to divide up range of x-axis
tsmethod	two-sided method for p-value calculation, either "minlike", "blaker" or "central"

pRange	range for plotting null hypothesis values of p
dolines	logical, add lines to a plot?
dopoints	logical, add points to a plot?
doci	logical, add lines for confidence interval?
alternative	type of alternative for p-values
relErr	number close to 1, avoids problems with ties, see binomControl
conf.level	confidence level for use when doci=TRUE
alphaline	logical, if doci=TRUE should a line be drawn at significance level
newplot	logical, start a new plot?
...	values passed to plot, points, or lines statement

See Also

[binom.exact](#)

Examples

```
## plot central two-sided p-values (double one-sided p-values)
## for 5 positive responses out of 17 tries
exactbinomPlot(5,17)
## add blakers exact p-values
## pch option acts on points, lty acts on ci lines
exactbinomPlot(5,17,tsmethod="blaker",col="blue",pch=".",lty=2,newplot=FALSE)

## can plot one-sided p-values, tsmethod is ignored
exactbinomPlot(5,17,alternative="less")
```

exactci-internal	<i>Internal functions, not to be called directly</i>
------------------	--

Description

These functions give p-values for exact binomial or poisson test. Not to be called directly, use [binom.exact](#) or [poisson.exact](#)

Usage

```
exactbinomPvals(x, n, p, relErr = 1 + 10^(-7), tsmethod = "minlike")
exactpoissonPval(x, T=1, r=1, relErr=1+1e-07, tsmethod="central")
exactpoissonPvals(x, T=1, r=1, relErr=1+1e-07, tsmethod="central")
exactbinomCI(x, n, tsmethod="minlike", conf.level=.95, tol=.00001,
  pRange=c(1e-10,1-1e-10))
exactpoissonCI(x, tsmethod="minlike", conf.level=.95, tol=.00001,
  pRange=c(1e-10,1-1e-10))
```

Arguments

x	number of successes or counts, vectors not allowed
n	number at risk, vectors not allowed
p	binomial parameter for null hypothesis, may be vector
relErr	used in calculation to avoid ties, slightly bigger than 1
tsmethod	two-sided method, one of "minlike", "blaker" ("central" only allowed for exactpoissonPval or exactpoissonPvals)
T	number at risk or person-years at risk, vectors not allowed
r	rate parameter for null hypothesis, null is $E(x*T)=r$, vectors only allowed for exactpoissonPvals
conf.level	number between 0 and 1 for level of confidence interval
pRange	range to search for confidence intervals, between 0 and 1 (even for poisson where it is transformed to a 0 to Inf-like range)
tol	tolerance for precision of confidence interval, very small number

Details

The function `exactbinomPvals` tests point null hypotheses for a single binomial observation. The function `exactpoissonPvals` tests point null hypotheses for a single Poisson observation. To get p-values for the two-sample Poisson test save results from `exactpoissonPlot`. The functions `exactbinomCI` and `exactpoissonCI` calculate the "minlike" and "blaker" confidence intervals.

Value

Returns either a confidence interval with attributes giving precision, or a pvalue (`exactpoissonPval`), or a list with pvals and parameters (r,T for poisson and p for binomial).

See Also

[poisson.exact](#) and [binom.exact](#)

Examples

```
exactbinomPvals(3,10,c(.3,.4,.5),tsmethod="minlike")
```

`exactpoissonPlot`

Plot p-value function for single or pair of poisson responses.

Description

Plots p-values as a function of different point null hypothesis values for rate. For two-sided p-values, can plot three types of p-values: the minimum likelihood method (default for `poisson.test`), the central method (i.e., twice the one-sided exact p-values), and Blaker's exact.

Usage

```
exactpoissonPlot(x,
  T=1,
  r=NULL,
  ndiv=1000,
  tsmethod="central",
  rRange=NULL,
  dolog=TRUE,
  dolines=FALSE,
  dopoints=TRUE,
  doci=TRUE,
  alternative = c("two.sided", "less", "greater"),
  relErr=1 + 10^(-7),
  conf.level=.95,
  alphaline=TRUE,
  newplot=TRUE,...)
```

Arguments

x	number of events. A vector of length one or two
T	time base for event count. A vector of length one or two
r	null values of rate for plot, if NULL divides rRange into ndiv pieces
ndiv	number of pieces to divide up range of x-axis
tsmethod	two-sided method for p-value calculation, either "minlike", "blaker" or "central"
rRange	range for plotting null hypothesis values of rate, if null then uses confidence interval to determine range
dolog	logical, plot horizontal axis in log scale?
dolines	logical, add lines to a plot?
dopoints	logical, add points to a plot?
doci	logical, add lines for confidence interval?
alternative	type of alternative for p-values
relErr	number close to 1, avoids problems with ties, see binomControl
conf.level	confidence level for use when doci=TRUE
alphaline	logical, if doci=TRUE should line be drawn at significance level
newplot	logical, start a new plot?
...	values passed to plot, points, or lines statement

Value

Does graph or adds lines or points. Returns (invisibly, see [invisible](#)) a list with elements r (null hypothesis values) and p.value (associated p-values).

See Also

[binom.exact](#)

Examples

```
## single Poisson response
exactpoissonPlot(2,17877)
```

poisson.exact

Exact Poisson tests with Matching Confidence Intervals

Description

Performs an exact test of a simple null hypothesis about the rate parameter in Poisson distribution, or for the ratio between two rate parameters. This is different from [poisson.test](#) in that 3 different types of exact two-sided tests (and the matching confidence intervals) are offered. The one-sided tests are the same as in [poisson.test](#).

Usage

```
poisson.exact(x, T = 1, r = 1,
  alternative = c("two.sided", "less", "greater"), tsmethod=c("central", "minlike", "blaker"),
  conf.level = 0.95, control=binomControl(), plot=FALSE)
```

Arguments

x	number of events. A vector of length one or two.
T	time base for event count. A vector of length one or two.
r	hypothesized rate or rate ratio
alternative	indicates the alternative hypothesis and must be one of "two.sided", "greater" or "less". You can specify just the initial letter.
tsmethod	character giving two-sided method, one of "central", "minlike" or "blaker", ignored if alternative not equal "two.sided"
conf.level	confidence level for the returned confidence interval.
control	list with settings to avoid problems with ties, etc, should not need to change this for normal use, see binomControl
plot	logical, plot p-value function? For finer control on plot see exactpoissonPlot

Details

Confidence intervals are computed similarly to those of [binom.exact](#) in the one-sample case, in that there are three two-sided options depending on the `tsmethod`. For the one-sample case the default intervals use `tsmethod="central"` giving the Garwood (1936) exact central confidence intervals. For the two-sample case we condition on the total counts and then use binomial methods, see Lehmann and Romano (2005) for that motivation and `vignette("exactci")` for description of the three different two-sided methods for calculating p-values and confidence intervals.

Value

A list with class "htest" containing the following components:

statistic	the number of events (in the first sample if there are two.)
parameter	the corresponding expected count
p.value	the p-value of the test.
conf.int	a confidence interval for the rate or rate ratio.
estimate	the estimated rate or rate ratio.
null.value	the rate or rate ratio under the null, r .
alternative	a character string describing the alternative hypothesis.
method	the character string "Exact Poisson test" or "Comparison of Poisson rates" as appropriate.
data.name	a character string giving the names of the data.

Note

The rate parameter in Poisson data is often given based on a "time on test" or similar quantity (person-years, population size). This is the role of the T argument.

References

- Fay, M.P. (2010). Two-sided Exact Tests and Matching Confidence Intervals for Discrete Data. *R Journal* 2(1): 53-58.
- Garwood, F (1936). Fiducial limits for the Poisson distribution. *Biometrika*, 437-442.
- Lehmann, EL, and Romano, JP (2005). *Testing Statistical Hypotheses*, third edition. Springer:New York.

See Also

[poisson.test](#), [exactpoissonPlot](#),

Examples

```
### Suppose you have observed rates of 2 out of 17877 in group A
### and 10 out of 20000 in group B
### poisson.test gives non-matching confidence intervals
### i.e., p-value using 'minlike' criteria but confidence interval using 'central' criteria
poisson.test(c(2,10),c(17877,20000))
### poisson.exact gives matching CI to the p-values
### defaults to 'central' two-sided method
poisson.exact(c(2,10),c(17877,20000))
### other options
poisson.exact(c(2,10),c(17877,20000),tsmethod="minlike")
poisson.exact(c(2,10),c(17877,20000),tsmethod="blaker")
```

Index

*Topic **\textasciitildekwd1**

exactci-internal, 7

*Topic **\textasciitildekwd2**

exactci-internal, 7

*Topic **hplot**

exactbinomPlot, 6

exactpoissonPlot, 8

*Topic **htest**

binom.exact, 3

exactci-package, 2

poisson.exact, 10

*Topic **misc**

binomControl, 5

*Topic **package**

exactci-package, 2

binom.exact, 2, 3, 5, 7–10

binom.test, 2–5

binomControl, 3, 5, 7, 9, 10

exact2x2, 2, 5

exactbinomCI (exactci-internal), 7

exactbinomPlot, 3, 6

exactbinomPvals (exactci-internal), 7

exactci (exactci-package), 2

exactci-internal, 7

exactci-package, 2

exactpoissonCI (exactci-internal), 7

exactpoissonPlot, 8, 10, 11

exactpoissonPval (exactci-internal), 7

exactpoissonPvals (exactci-internal), 7

invisible, 9

poisson.exact, 2, 5, 7, 8, 10

poisson.test, 2, 10, 11