Package ‘BlakerCI’

October 12, 2022

Type Package

Title Blaker's Binomial and Poisson Confidence Limits

Version 1.0-6

Date 2019-04-29

Author Jan Klaschka

Maintainer Jan Klaschka <klaschka@cs.cas.cz>

Description Fast and accurate calculation of Blaker's binomial and Poisson confidence limits (and some related stuff).

License GPL-3

LazyLoad yes

NeedsCompilation no

Repository CRAN

Date/Publication 2019-04-29 19:00:03 UTC

R topics documented:

`BlakerCI-package` *Blaker's binomial and Poisson confidence limits*

Description

Fast and accurate calculation of Blaker's binomial and Poisson confidence limits.

Details

Package:	BlakerCI
Type:	Package
Version:	1.0-6
Date:	2019-04-29
License:	GPL-3
LazyLoad:	yes

Acknowledgement

This work was accomplished with institutional support RVO:67985807. The same was true for the previous version 1.0-5.

Versions up to 1.0-2 were supported by grant 205/09/1079 from The Czech Science Foundation.

Author(s)

Jan Klaschka <klaschka@cs.cas.cz>

Maintainer: Jan Klaschka <klaschka@cs.cas.cz>

Examples

```
binom.blaker.limits(3,10) # [1] 0.08726443 0.61941066
poisson.blaker.limits(3)  # [1] 0.8176914 8.5597971
```

`binom.blaker.acc` *Blaker's binomial acceptability function, optionally unimodalized.*

Description

Calculates values of the acceptability function for the binomial distribution (see function `acceptbin` in Blaker (2000)) in a sequence of points (for, e.g., plotting purposes). The acceptability function may optionally be “unimodalized”, i.e. replaced with the smallest greater or equal unimodal function.

Usage

```
binom.blaker.acc(x, n, p, type = c("orig", "unimod"),
      acc.tol = 1e-10, ...)
```

Arguments

`x`	number of successes.
`n`	number of trials.
`p`	vector (length 1 allowed) of hypothesized binomial parameters (between 0 and 1). In case of more than one point, an increasing sequence required.
`type`	for `type = "orig"`, original acceptability function calculated. For `type = "unimod"`, smallest unimodal function greater or equal to the acceptability function calculated instead.
`acc.tol`	numerical tolerance (relevant only for `type = "unimod"`).
`...`	additional arguments to be passed to `binom.blaker.acc.single.p`; in fact, just `maxiter` (see `BlakerCI-internal`).

Details

For `type = "orig"`, essentially the same is calculated as – for single points – by `acceptbin` function from Blaker (2000).

Single values of the "unimodalized" acceptability function (for `type = "unimod"`) are computed by an iterative numerical algorithm implemented in internal function `binom.blaker.acc.single.p`. The function cited is called just once in each of the intervals where the acceptability function is continuous (namely in the leftmost one of those points of `p` that fall into the interval when dealing with points below `x/n`, and the rightmost one when above `x/n`). The rest is done by function `cummax`. This is considerably faster than calling `binom.blaker.acc.single.p` for every point of `p`. Note that applying `cummax` directly to a vector of unmodified acceptability values is even faster and provides a unimodal output; it may, nevertheless, lack accuracy (see Examples).

Value

Vector of acceptability values (with or without unimodalization) in points of `p`.

Note

Inspired by M.P. Fay (2010), mentioning "unavoidable inconsistencies" between tests with non-unimodal acceptability functions and confidence intervals derived from them. When the acceptability functions are unimodalized and the test modified accordingly (i.e. p-values slightly increased in some cases), a perfectly matching test-CI pair is obtained.

Author(s)

Jan Klaschka <klaschka@cs.cas.cz>

References

Blaker, H. (2000) Confidence curves and improved exact confidence intervals for discrete distributions. Canadian Journal of Statistics 28: 783-798.
(Corrigenda: Canadian Journal of Statistics 29: 681.)

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

Examples

```
p <- seq(0,1,length=1001)
acc <- binom.blaker.acc(3,10,p)
acc1 <- binom.blaker.acc(3,10,p,type="unimod")
## The two functions look the same at first glance.
plot(p,acc,type="l")
lines(p,acc1,col="red")
legend(x=.7,y=.8,c("orig","unimod"),col=c("black","red"),lwd=1)

## There is, nevertheless, a difference.
plot(p,acc1-acc,type="l")

## Focussing on the difference about p=0.4:
p <- seq(.395,.405,length=1001)
acc <- binom.blaker.acc(3,10,p)
acc1 <- binom.blaker.acc(3,10,p,type="unimod")
plot(p,acc,type="l",ylim=c(.749,.7495))
lines(p,acc1,col="red")
legend(x=.402,y=.7494,c("orig","unimod"),col=c("black","red"),lwd=1)

## Difference between type="unimod" and mere applying
## cummax to values obtained via type="orig":
p <- seq(0,1,length=1001)
x <- 59
n <- 355
## Upper confidence limit (at 0.95 level) is slightly above 0.209:
binom.blaker.limits(x,n) ## [1] 0.1300807 0.2090809
## Unmodified acceptability value fall below 0.05 at p = .209
## left to the limit (so that the null hypothesis p = .209
## would be rejected despite the fact that p lies within
## the confidence interval):
acc <- binom.blaker.acc(59,355,p)
rbind(p,acc)[,207:211]
##           [,1]       [,2]       [,3]       [,4]       [,5]
## p   0.20600000 0.20700000 0.20800000 0.20900000 0.21000000
## acc 0.06606867 0.05759836 0.05014189 0.04999082 0.04330283
##
## Modified acceptability is above 0.05 at p = 0.05 (so that
## hypothesis p = 0.05 is not rejected by the modified test):
acc1 <- binom.blaker.acc(59,355,p,type="unimod")
rbind(p,acc1)[,207:211]
##            [,1]       [,2]       [,3]       [,4]       [,5]
## p    0.20600000 0.20700000 0.20800000 0.20900000 0.21000000
```

```
## acc1 0.06608755 0.05759836 0.05014189 0.05000009 0.04331409
##
## Applying cummax to unmodified acceptabilities guarantees unimodality
## but lacks accuracy, leaving the value at p = 0.209 below 0.05:
m <- max(which(p <= 59/355))
acc2 <- acc
acc2[1:m] <- cummax(acc2[1:m])
acc2[1001:(m+1)] <- cummax(acc2[1001:(m+1)])
rbind(p,acc2)[,207:211]
##            [,1]       [,2]       [,3]       [,4]       [,5]
## p    0.20600000 0.20700000 0.20800000 0.20900000 0.21000000
## acc2 0.06606867 0.05759836 0.05014189 0.04999082 0.04330283
```

`binom.blaker.limits` *Blaker's binomial confidence limits*

Description

Fast and accurate calculation of Blaker's binomial confidence limits.

Usage

```
binom.blaker.limits(x, n, level = 0.95, tol = 1e-10, ...)
```

Arguments

`x`	number of successes.
`n`	number of trials.
`level`	confidence level.
`tol`	numerical tolerance.
`...`	additional arguments to be passed to `binom.blaker.lower.limit`; in fact, just `maxiter` (see `BlakerCI-internal`).

Details

Note that the Blaker's `(1 - alpha)` confidence interval is the convex hull of the set `C` of those points where the acceptability function (Blaker (2000)) exceeds level `alpha`. The original numerical algorithm from Blaker (2000) is prone, when `C` is a union of disjoint intervals, to skipping a short interval and finding inaccurate over-liberal confidence limits.

Function `binom.blaker.limits` is, by contrast, immune from such failures and yields always as its result the whole confidence interval (Klaschka (2010)).

Value

Length 2 vector – the lower and upper confidence limits.

Note

Package exactci by M. P. Fay includes another algorithm that calculates Blaker's binomial confidence limits (see user-level function binom.exact and internal function exactbinomCI). It is more sophisticated than the original Blaker's one, but considerably slower and sometimes less accurate than that of binom.blaker.limits.

Earlier 2010 versions of the algorithm of binom.blaker.limits were designed independently of (though already existing) M.P. Fay's packages exact2x2 and exactci. Some later modifications, however, have been inspired by Fay's programs.

Lecoutre & Poitevineau (2014) designed another algorithm for the calculation of the Blaker's confidence limits. Despite more abstract theoretical background and broader scope (not confined to the binomial distribution), it is closely analogous to that of binom.blaker.limits.

Author(s)

Jan Klaschka <klaschka@cs.cas.cz>

References

Blaker, H. (2000) Confidence curves and improved exact confidence intervals for discrete distributions. Canadian Journal of Statistics 28: 783-798.
(Corrigenda: Canadian Journal of Statistics 29: 681.)

Klaschka, J. (2010). BlakerCI: An algorithm and R package for the Blaker's binomial confidence limits calculation. Technical report No. 1099, Institute of Computer Science, Academy of Sciences of the Czech Republic, http://hdl.handle.net/11104/0195722.

Lecoutre, B. & Poitevineau J. (2014). New results for computing Blaker's exact confidence interval limits for usual one-parameter discrete distributions. Communications in Statistics - Simulation and Computation, http://dx.doi.org/10.1080/03610918.2014.911900.

See Also

exactci:binom.exact	One of the options yields Blaker's limits. The algorithm is more sophisticated than the original Blaker's one.
propCIs:blakerci	Implementation of the original algorithm from Blaker (2000).
binGroup:binBlaker	Another implementation of the same algorithm.

Examples

```
binom.blaker.limits(3,10) # [1] 0.08726443 0.61941066

## Example of a failure of the original algorithm:
## Requires PropCIs package.
## Tolerance 1e-4 - default in the Blaker's paper.
## Not run:
blakerci(29,99,conf.level=0.95,tolerance=1e-4) ## [1] 0.2096386 0.3923087
## The correct upper limit should be 0.3929\dots,
```

```
## as demonstrated:
## (1) By the same function with a smaller tolerance:
blakerci(29,99,conf.level=0.95,tolerance=1e-7) ## [1] 0.2097022 0.3929079
## (2) By binom.blaker.limits
##     (default confidence limit 0.95, default tolerance 1e-10):
binom.blaker.limits(29,99) ## [1] 0.2097022 0.3929079
## (3) By exactbinomCI function from package exactci
##     (default confidence level, default tolerance):
exactbinomCI(29,99,tsmethod="blaker")[1:2] ## [1] 0.2097 0.3929
## The same function, smaller tolerance:
exactbinomCI(29,99,tsmethod="blaker",tol=1e-8)[1:2]
                                                ## [1] 0.2097022 0.3929079

## Another example of a failure of the original algorithm
## with even as small tolerance as 1e-6:
blakerci(59,355,conf.level=0.95,tolerance=1e-4) ## [1] 0.1299899 0.2085809
blakerci(59,355,conf.level=0.95,tolerance=1e-5) ## [1] 0.1300799 0.2085409
blakerci(59,355,conf.level=0.95,tolerance=1e-6) ## [1] 0.1300799 0.2085349
## Only for tolerance = 1e-7 the result is satisfactory
## and in agreement with binom.blaker.limits:
blakerci(59,355,conf.level=0.95,tolerance=1e-7) ## [1] 0.1300807 0.2090809
binom.blaker.limits(59,355)                     ## [1] 0.1300807 0.2090809

## End(Not run)
```

---

binom.blaker.VHadj.acc — *Blaker's binomial acceptability function with Vos-Hudson adjustment.*

Description

Calculates values of the Vos-Hudson adjusted acceptability function in a sequence of points (for, e.g., plotting purposes). The adjusted acceptability function may optionally be "unimodalized", i.e. replaced with the smallest greater or equal unimodal function.

Usage

```
binom.blaker.VHadj.acc(x, n, p, type = c("orig", "unimod"),
      acc.tol = 1e-10, nmax=n+1000,int.eps=1e-12, ...)
```

Arguments

x	number of successes.
n	number of trials.
p	vector (length 1 allowed) of hypothesized binomial parameters (between 0 and 1). In case of more than one point, an increasing sequence required.

type	for type = "orig", Vos-Hudson adjustment applied to original acceptability function. For type = "unimod", smallest unimodal function greater or equal to the adjusted acceptability function.
acc.tol	numerical tolerance (relevant only for type = "unimod").
nmax	Pairs (y, m) of number of trials and number of successes are allowed to contribute to the Vos-Hudson adjustment for only m up to nmax. Warning is returned when greater numbers of trials are suspected to have influence.
int.eps	Maximum expected error of machine representation of integers calculated from reals via multiplication and division. (Used in order to round numbers correctly if they happen to be integer, e. g. ceiling(xx - int.eps) is calculated instead of ceiling(xx).)
...	additional arguments to be passed to binom.blaker.acc.single.p; in fact, just maxiter (see BlakerCI-internal).

Details

The relationship between the adjusted acceptability function and the adjusted confidence intervals (see binom.blaker.VHadj.limits) is the same as between the unadjusted acceptability function and confidence interval (see binom.blaker.acc, binom.blaker.limits): The confidence interval is the convex hull of the set of those points where the function exceeds 1 - confidence level.

Value

Vector of Vos-Hudson adjusted acceptability values (with or without unimodalization) in points of p.

Warning

(1) Comparing output of the function with that of binom.blaker.acc cannot answer positively the question whether the unadjusted and adjusted functions are identical on an interval (but, up to the numerical accuracy, in the points of p only).

(2) The Warning section of the binom.blaker.VHadj.limits documentation is relevant here, as well.

Author(s)

Jan Klaschka <klaschka@cs.cas.cz>

Examples

```
p <- seq(0,1,length=10001)
acc.adj <- binom.blaker.VHadj.acc(6,13,p)
acc <- binom.blaker.acc(6,13,p)

plot(p,acc.adj,type="l",col="red",ylab="acceptability"
  ,main=paste("Vos-Hudson adjustment of acceptability function"
  ,"for 6 successes in 13 trials"
  , sep="\n")
)
```

```
lines(p,acc,type="l")
legend(x=.7,y=.8,c("unadjusted","adjustment"),col=c("black","red"),lwd=1)

## Plot of differences between the unadjusted and adjusted
## acceptability functions reveals some adjustment details
## hardly visible in the previous graph.

plot(p,acc.adj-acc,type="l",ylab="acceptability difference")

## The narrow peak near 0.215 is close to the
## Blaker's lower 0.95 confidence limit.
##
## Focussing on the neighbourhood of 0.215:

p <- seq(0.21,0.22,length=1001)
acc.adj <- binom.blaker.VHadj.acc(6,13,p)
acc <- binom.blaker.acc(6,13,p)

plot(p,acc.adj,type="l",col="red",ylab="acceptability"
  ,main=paste("A detail of Vos-Hudson adjustment of acceptability function"
  ,"for 6 successes in 13 trials"
  ,sep="\n")
  ,ylim=c(0.02,0.09)
)
lines(p,acc,type="l")
legend(x=.210,y=.08,c("unadjusted","adjustment"),col=c("black","red"),lwd=1)

## The above adjustment results from the fact that, though
## 15 > 13 and 7/15 > 6/13, the acceptability function
## for 7 successes in 15 trials is greater that that for 6 successes
## in 13 trials on a short interval:

acc.7.15 <- binom.blaker.acc(7,15,p)
plot(p,acc,type="l",ylab="acceptability"
  ,main=paste("A detail of acceptability functions"
  ,sep="\n")
  ,ylim=c(0.02,0.09)
)
lines(p,acc.7.15,type="l",col="green")
legend(x=.210,y=.08,c("6 / 13","7 / 15"),col=c("black","green")
  ,title="succ / trials",lwd=1)

## The adjustment shifts the point where the 0.05 level is exceeded,
## i. e. the Blaker's lower 0.95 confidence limit, from 0.2158 to 0.2150.
## (Compare with Examples in binom.blaker.VHadj.limits section.)
```

`binom.blaker.VHadj.limits`

Vos-Hudson adjustment of Blaker's binomial confidence limits

Description

Blaker's binomial confidence limits adjusted so that logical inconsistencies criticized by Vos and Hudson (2008) are avoided.

Usage

```
binom.blaker.VHadj.limits(x, n, level = 0.95, tol = 1e-10, ...)
```

Arguments

`x`	number of successes.
`n`	number of trials.
`level`	confidence level.
`tol`	numerical tolerance.
`...`	additional arguments to be passed to `binom.blaker.VHadj.lower.limit`: `maxiter`, `nmax`, `int.eps` (see `BlakerCI-internal`).

Value

Length 2 vector – the lower and upper (adjusted) confidence limits.

Warning

The stopping rule used is not fully justified:

The Clopper-Pearson `1 - alpha` confidence bounds for `x` successes in `n` trials may be expressed as `qbeta(alpha/2,x,n-x+1)` and `qbeta(1-alpha/2,x+1,n-x)`, and can be generalized this way to real (i. e. not only integer) values of `x`.

The stopping rule used in `binom.blaker.VHadj.limits` relies on the hypothesis that the generalized lower (upper) Clopper-Pearson confidence bounds grow (decrease) whenever the number of trials grows, and the proportion of successes grows (decreases) or remains unchanged (with obvious exceptions in extremes).

Though I firmly trust the hypothesis, I can prove it, so far, just for integer numbers of successes (i. e. for "ordinary" Clopper-Pearson confidence bounds, not the generalized ones), and lack a general proof. Should the hypothesis be invalid, the stopping rule implemented in `binom.blaker.VHadj.limits` would be incorrect, and the process of modifying the Blaker's confidence bounds could be incomplete in some cases.

Note

Vos & Hudson (2008) gave examples of mutually contradictory inferences yielded by some binomial tests and confidence intervals, including the Blaker's confidence interval. Their objections may be interpreted as follows: When the number of trials is increased so that the success proportion increases (decreases) or remains the same, the lower (upper) confidence limit at the same confidence level should not decrease (increase).
The adjustment implemented in `binom.blaker.VHadj.limits` replaces the lower (upper) Blaker's

confidence limit for x successes in n trials with the infimum (supremum) of the Blaker's lower (upper) confidence limits over such pairs y, m that m is not less that n, and y/m is not less (greater) than x/n.

Note that Lecoutre & Poitevineau (2014), refering to the criticism by Vos & Hudson, proposed a modification of the Blaker's confidence limits. Their adjustment, however, eliminates only a subset of "discrepancies" treated by binom.blaker.VHadj.limits, namely nonmonotonicities of upper (lower) Blaker's confidence bounds in the number of trials when the number of successes (failures) remains the same.

Author(s)

Jan Klaschka <klaschka@cs.cas.cz>

References

Vos, P. W. & Hudson, S. (2008). Problems with binomial two-sided tests and the associated confidence intervals. Australian & New Zealand Journal of Statistics 50(1): 81-89.

Lecoutre, B. & Poitevineau, J. (2014). New results for computing Blaker's exact confidence interval limits for usual one-parameter discrete distributions. Communications in Statistics - Simulation and Computation, http://dx.doi.org/10.1080/03610918.2014.911900.

Examples

```
binom.blaker.VHadj.limits(6,13) # [1] 0.2150187 0.7395922

## Note that the lower limit differs from the
## unadjusted version:

binom.blaker.limits(6,13)       # [1] 0.2158050 0.7395922

## The (unadjusted) lower limit was replaced with the
## Blaker's lower limit (both unadjusted and adjusted)
## assigned to 7 successes in 15 trials:

binom.blaker.limits(7,15)       # [1] 0.2150187 0.7096627
binom.blaker.VHadj.limits(7,15) # [1] 0.2150187 0.7096627

## The adjustment avoids a contradiction between
## inferences corresponding to
## 6 successes in 13 trials, and 7 successess in 15 trials:
## Though the latter situation means a higher succes proportion
## in a higher number of trials, it is assigned a smaller
## (unadjusted) Blaker's 95% lower confidence limit.
```

BlakerCI-internal *Internal functions, not expected to be called by the user.*

Description

For binomial distribution: Calculation of the lower Blaker's confidence limit as defined by Blaker (binom.blaker.lower.limit), or with so called Vos-Hudson adjustment (binom.blaker.VHadj.lower.limit); a single acceptability value, optionally "unimodalized" (binom.blaker.acc.single.p).

For Poisson distribution: Calculation of the lower and upper Blaker's confidence limits (poisson.blaker.lower.limit, poisson.blaker.upper.limit); a single acceptability value, optionally "unimodalized" (poisson.blaker.acc.single.p

Usage

```
binom.blaker.lower.limit(x, n, level, tol = 1e-10, maxiter=100)
binom.blaker.VHadj.lower.limit(x,n,level,tol=1e-10,maxiter=100,
      nmax=n+1000,int.eps=1e-10)
binom.blaker.acc.single.p(x, n, p, type = "orig", acc.tol = 1e-10,
      output = "acc", maxiter=100)
poisson.blaker.lower.limit(x, level, tol = 1e-10, maxiter=100)
poisson.blaker.upper.limit(x, level, tol = 1e-10, maxiter=100)
poisson.blaker.acc.single.p(x, p, type = "orig", acc.tol = 1e-10,
      output = "acc", maxiter=100)
```

Arguments

x	number of successes (binomial case), or events (Poisson case).
n	number of trials.
level	confidence level.
tol	numerical tolerance (for the confidence limit).
p	point (binomial or Poisson parameter value) where to calculate the acceptability.
type	"orig", or "unimod" – either unmodified, or unimodalized acceptability (see binom.blaker.acc, poisson.blaker.acc).
acc.tol	numerical tolerance (for the acceptability values when type = "unimod").
output	the acceptability value output (output = "acc", the default) or, instead, an auxiliary integer-valued parameter q1, used for testing whether points belong to the same continuous segment of the acceptability function (output = "q1"), or both (output = "both").
maxiter	Maximum number of interval halving iterations during the search for a confidence limit (binom.blaker.lower.limit, poisson.blaker.lower.limit, poisson.blaker.upper.limit), or a discontinuity point of the acceptability function (binom.blaker.acc.single.p, or poisson.blaker.acc.single.p with type = "unimod"). When the required accuracy is not reached in maxiter steps – typically when too small tol or acc.tol exceeds capabilities of machine arithmetic – last step's result is returned with warning.

nmax	Pairs (y, m) of number of trials and number of successes are allowed to contribute to the Vos-Hudson adjustment for only m up to nmax. Warning is returned when greater numbers of trials are suspect to have influence.
int.eps	Maximum expected error of machine representation of integers calculated via multiplication and division from reals. (Used in order to round numbers correctly if they happen to be integer, e. g. ceiling(xx - int.eps) is calculated instead of ceiling(xx).)

Value

For binom.blaker.lower.limit and binom.blaker.VHadj.lower.limit, a single number – the lower confidence limit.
For binom.blaker.acc.single.p – depending on the output parameter – a single acceptability value, or a single auxiliary integer, or both.

Author(s)

Jan Klaschka <klaschka@cs.cas.cz>

See Also

binom.blaker.limits, binom.blaker.VHadj.limits,
binom.blaker.acc, binom.blaker.VHadj.acc, poisson.blaker.limits, poisson.blaker.acc.

---

poisson.blaker.acc *Blaker's Poisson acceptability function, optionally unimodalized.*

Description

Calculates values of the acceptability function for the Poisson distribution (see Blaker (2000)) in a sequence of points (for, e.g., plotting purposes). The acceptability function may optionally be "unimodalized", i.e. replaced with the smallest greater or equal unimodal function.

Usage

```
poisson.blaker.acc(x, p, type = c("orig", "unimod"),
      acc.tol = 1e-10, ...)
```

Arguments

x	number of events.
p	vector (length 1 allowed) of hypothesized Poisson parameters. In case of more than one point, an increasing sequence required.
type	for type = "orig", original acceptability function calculated. For type = "unimod", smallest unimodal function greater or equal to the acceptability function calculated instead.
acc.tol	numerical tolerance (relevant only for type = "unimod").
...	additional arguments to be passed to poisson.blaker.acc.single.p; in fact, just maxiter (see BlakerCI-internal).

Details

Single values of the "unimodalized" acceptability function (for type = "unimod") are computed by an iterative numerical algorithm implemented in internal function poisson.blaker.acc.single.p. The function cited is called just once in each of the intervals where the acceptability function is continuous (namely in the leftmost one of those points of p that fall into the interval when dealing with points below x, and the rightmost one when above x). The rest is done by function cummax. This is considerably faster than calling poisson.blaker.acc.single.p for every point of p. Note that applying cummax directly to a vector of unmodified acceptability values is even faster and provides a unimodal output; it may, nevertheless, lack accuracy.

Value

Vector of acceptability values (with or without unimodalization) in points of p.

Note

Inspired by M.P. Fay (2010), mentioning "unavoidable inconsistencies" between tests with non-unimodal acceptability functions and confidence intervals derived from them. When the acceptability functions are unimodalized and the test modified accordingly (i.e. p-values slightly increased in some cases), a perfectly matching test-CI pair is obtained.

Author(s)

Jan Klaschka <klaschka@cs.cas.cz>

References

Blaker, H. (2000) Confidence curves and improved exact confidence intervals for discrete distributions. Canadian Journal of Statistics 28: 783-798.
(Corrigenda: Canadian Journal of Statistics 29: 681.)

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

Examples

```
p <- seq(0,10,length=1001)
acc <- poisson.blaker.acc(3,p)
acc1 <- poisson.blaker.acc(3,p,type="unimod")
plot(p,acc,type="l")
lines(p,acc1,col="red")
legend(x=7,y=.8,c("orig","unimod"),col=c("black","red"),lwd=1)

## The two lines -- the unimodalized and original acceptabilities --
## look almost the same but some small differences are slightly
## visible.

## They can be seen better this way:
plot(p,acc1-acc,type="l")

## Focussing on one of them:
```

```
p <- seq(5.05,5.6,length=1001)
acc <- poisson.blaker.acc(3,p)
acc1 <- poisson.blaker.acc(3,p,type="unimod")
plot(p,acc,type="l",ylim=c(.391,.396))
lines(p,acc1,col="red")
legend(x=5.4,y=.395,c("orig","unimod"),col=c("black","red"),lwd=1)
```

---

poisson.blaker.limits *Blaker's Poisson confidence limits*

Description

Fast and accurate calculation of Blaker's Poisson confidence limits.

Usage

```
poisson.blaker.limits(x, level = 0.95, tol = 1e-10, ...)
```

Arguments

x	number of events.
level	confidence level.
tol	numerical tolerance.
...	additional arguments to be passed to poisson.blaker.lower.limit; in fact, just maxiter (see BlakerCI-internal).

Details

Note that the Blaker's (1 - alpha) confidence interval is the convex hull of the set C of those points where the acceptability function (Blaker (2000)) exceeds level alpha. When C is not connected, the algorithm is, analogously to binom.blaker.limits (see its details), immune from leaving out short intervals and making thus the confidence intervals over-liberal.

Value

Length 2 vector – the lower and upper confidence limits.

Note

Package exactci by M. P. Fay includes another algorithm that calculates Blaker's Poisson confidence limits (see user-level function poisson.exact and internal function exactpoissonCI).

Lecoutre & Poitevineau (2014) designed another algorithm for the calculation of the Blaker's confidence limits. It is closely analogous to that of poisson.blaker.limits.

Author(s)

Jan Klaschka <klaschka@cs.cas.cz>

References

Blaker, H. (2000) Confidence curves and improved exact confidence intervals for discrete distributions. Canadian Journal of Statistics 28: 783-798.
(Corrigenda: Canadian Journal of Statistics 29: 681.)

Lecoutre, B. & Poitevineau J. (2014). New results for computing Blaker's exact confidence interval limits for usual one-parameter discrete distributions. Communications in Statistics - Simulation and Computation, http://dx.doi.org/10.1080/03610918.2014.911900.

See Also

`exactci:poisson.exact`	One of the options yields Blaker's limits.

Examples

```
poisson.blaker.limits(3) # [1] 0.8176914 8.5597971
```

Index