Package ‘ldbounds’

July 22, 2025

Type Package

Title Lan-DeMets Method for Group Sequential Boundaries

Version 2.0.2

Date 2023-12-01

Author Charlie Casper, Thomas Cook and Oscar A. Perez. Based on FORTRAN program ld98.

Maintainer Charlie Casper <charlie.casper@hsc.utah.edu>

Description Computations related to group sequential boundaries. Includes calculation of bounds using the Lan-DeMets alpha spending function approach. Based on FORTRAN program ld98 implemented by Reboussin, et al. (2000) <doi:10.1016/s0197-2456(00)00057-x>.

Imports graphics, stats

License GPL (>= 2)

Suggests R.rsp

VignetteBuilder R.rsp

NeedsCompilation no

Repository CRAN

Date/Publication 2023-12-02 00:40:10 UTC

Contents

commonbounds *Commonly Used Group Sequential Boundaries*

Description

'commonbounds' determines group sequential boundaries of the well-known O'Brien-Fleming and Pocock types. These can be used as guidelines for early stopping of the trial. For Haybittle-Peto type boundaries, see function 'lastbound'. Note: these are NOT the alpha-spending versions of these boundaries.

Usage

```
commonbounds(looks, t=(1:looks)/looks, t2=t, iuse="OF", alpha=0.05, sides=2)
```

Arguments

`looks`	A number of equally spaced analysis times.
`t`	a vector of analysis times, if `looks` is not provided. If times are provided and they are not equally spaced, there will be a warning.
`t2`	the second time scale, usually in terms of amount of accumulating information. By default, same as the equally spaced analysis times or the analysis times corresponding to `t`.
`iuse`	the type of bounds: O'Brien-Fleming ("OF") or Pocock ("PK") If two-sided bounds are calculated, the same type will be used for each of the two boundaries. If a vector of length two is given, the two values will be used for lower and upper bounds, respectively. Details of specification are given below.
`alpha`	Type I error(s). In two-sided situations, alpha can be a single value, indicating symmetric type I error control (half of alpha for each boundary). If a vector of length two is given, this corresponds to the amount allocated to the lower and upper boundaries, respectively. The total alpha must be greater than 0 and less than or equal to 1.
`sides`	Designates one- or two-sided bounds.

Details

This function calculates boundaries corresponding to traditional O'Brien-Fleming or Pocock boundaries. Note that these are not the alpha spending function versions of the boundaries. For those, use `ldBounds`.

Value

'commonbounds' returns an object of 'class' '"ldBounds"'.

For details about this object class, see the documentation for the `ldBounds` function.

Author(s)

Charlie Casper <charlie.casper@hsc.utah.edu>

References

Reboussin, D. M., DeMets, D. L., Kim, K. M., and Lan, K. K. G. (2000) Computations for group sequential boundaries using the Lan-DeMets spending function method. *Controlled Clinical Trials*, 21:190-207.

DeMets, D. L. and Lan, K. K. G. (1995) *Recent Advances in Clinical Trial Design and Analysis*, Thall, P. F. (ed.). Boston: Kluwer Academic Publishers.

Lan, K. K. G. and DeMets, D. L. (1983) Discrete sequential boundaries for clinical trials. *Biometrika*, 70:659-63.

See Also

Generic functions `summary.ldBounds` and `plot.ldPower`.

`ldBounds` for boundaries that use the alpha spending approach.

`ldPower` for exit probabilities given boundaries OR drift (effect) given power OR confidence interval OR adjusted p-value.

Examples

```
## From Reboussin, et al. (2000)

time <- seq(0.2,1,length=5)
obf.bd <- commonbounds(t=time)
summary(obf.bd)
plot(obf.bd)

# Equivalent bounds to above
obf.bd2 <- commonbounds(5)
```

`condpower` *Conditional Power Given Interim Results*

Description

'condpower' determines conditional power, given interim results and hypothesized treatment effect. This is commonly used to evaluate the futility of an ongoing trial.

Usage

```
condpower(z.crit, z.val, accr, outcome.type, par.c, par.t=NULL, N, sigma=NULL)
```

Arguments

`z.crit`	the critical value (on the z-value scale) that will be used at the end of the trial to determine whether the experimental treatment is superior to control.
`z.val`	the current test statistic (on the z-value scale) using interim data. A positive value means that the experimental arm is estimated to have a more favorable outcome.
`accr`	the amount of information accrued at the time of the interim analysis. For binary and continuous outcomes, this is the number of subjects randomized. For a survival outcome, this is the number of events that have been observed.
`outcome.type`	The type of outcome: binary ("bin"), continuous ("mean"), or survival ("surv").
`par.c`	the hypothesized parameter value in the control group (when par.t is specified) or the hypothesized treatment effect (when par.t is missing). For a binary outcome, this is the proportion of failures in the control group. For a continuous outcome, this is the mean in the control group or the difference in means. For a survival outcome, this is the hazard in the control group or the hazard ratio.
`par.t`	the hypothesized parameter value in the experimental group. Leave null if par.c is the difference in means or the hazard ratio.
`N`	the total target sample size (or number of events for survival) in the trial.
`sigma`	the assumed standard deviation (continuous outcome).

Details

This function calculates simple conditional power. The user specifies either the hypothesized parameter in each group (required for binary outcome) or the hypothesized treatment effect (difference for continuous outcome or hazard ratio for survival outcome).

Value

'condpower' returns a numeric value corresponding to the conditional probability.

Author(s)

Charlie Casper <charlie.casper@hsc.utah.edu>

Examples

```
# Binary outcome
# Assumptions used for power calculations in design of study
# Poor outcome 25% in control group
# Poor outcome 15% in experimental group
# Total sample size 900
# Interim analysis at 300 subjects
# Final critical value 2.0 (to account for group sequential
# monitoring)
# At interim, intervention doing better with z-statistic 0.067

condpower(2.0,0.067,300,"bin",0.25,0.15,900)
```

lastbound *Final Boundary Given Earlier Boundaries*

Description

'lastbound' determines the final boundary value, given earlier values. This can be used, for example, to create Haybittle-Peto boundaries that have the correct overall alpha.

Usage

```
lastbound(t, t2, alpha=0.05, sides=2, za=NULL, zb)
```

Arguments

t	a vector of analysis times or a number of analysis times. If the number of analyses is specified, they are assumed to be equally spaced. The last analysis time corresponds to the boundary value that is being calculated.
t2	the second time scale, usually in terms of amount of accumulating information. By default, same as the analysis times corresponding to `t`.
alpha	Type I error(s). In two-sided situations, alpha can be a single value, indicating symmetric type I error control (half of alpha for each boundary). If a vector of length two is given, this corresponds to the amount allocated to the lower and upper boundaries, respectively. The total alpha must be greater than 0 and less than or equal to 1.
sides	Designates one- or two-sided bounds.
za	the vector of lower boundaries, not including the final analysis time.
zb	the vector of upper boundaries, not including the final analysis time.

Details

This function calculates the final boundary value when all other boundaries have been specified.

Value

'lastbound' returns an object of 'class' '"ldBounds"'.

For details about this object class, see the documentation for the `ldBounds` function.

Author(s)

Charlie Casper <charlie.casper@hsc.utah.edu>

References

Reboussin, D. M., DeMets, D. L., Kim, K. M., and Lan, K. K. G. (2000) Computations for group sequential boundaries using the Lan-DeMets spending function method. *Controlled Clinical Trials*, 21:190-207.

DeMets, D. L. and Lan, K. K. G. (1995) *Recent Advances in Clinical Trial Design and Analysis*, Thall, P. F. (ed.). Boston: Kluwer Academic Publishers.

Lan, K. K. G. and DeMets, D. L. (1983) Discrete sequential boundaries for clinical trials. *Biometrika*, 70:659-63.

See Also

Generic functions `summary.ldBounds` and `plot.ldPower`.

`ldBounds` for boundaries that use the alpha spending approach.

`commonbounds` for boundaries that do not use alpha spending.

`ldPower` for exit probabilities given boundaries OR drift (effect) given power OR confidence interval OR adjusted p-value.

Examples

```
# Haybittle-Peto boundary with 3 looks (two-sided)
hpb <- lastbound(3,zb=c(3,3))
summary(hpb)
plot(hpb)
```

---

ldBounds — *Group Sequential Boundaries Using Spending Functions*

Description

'ldBounds' determines group sequential boundaries for interim analyses of accumulating data in clinical trials using the Lan-DeMets alpha spending function method. These can be used as guidelines for early stopping of the trial.

Usage

```
ldBounds(t, t2, iuse=1, asf=NULL, alpha=0.05, phi=rep(1,length(alpha)),
       sides=2, ztrun=rep(Inf,length(alpha)))
```

Arguments

`t`	a vector of analysis times or a number of analysis times. If the number of analyses is specified, they are assumed to be equally spaced. Analysis times must be increasing and in (0,1].
`t2`	the second time scale, usually in terms of amount of accumulating information. By default, same as `t` or the analysis times corresponding to `t` looks.

`iuse`	the type of alpha spending function(s) to use for bounds. If two-sided bounds are calculated and `iuse` is a single value, the same type will be used for each of the two boundaries. If a vector of length two is given, the two values will be used for lower and upper bounds, respectively. Details of specification are given below.
`asf`	a list of one or two functions to be used as alpha spending function(s). Used with `iuse`=5 (See below).
`alpha`	Type I error(s). In two-sided situations, alpha can be a single value, indicating symmetric type I error control (half of alpha for each boundary). If a vector of length two is given, this corresponds to the amount allocated to the lower and upper boundaries, respectively. The total alpha must be greater than 0 and less than or equal to 1. When `iuse`=5, any alpha specified is ignored (see details).
`phi`	a vector of values used when `iuse`=3 or 4 (See below).
`sides`	Designates one- or two-sided bounds.
`ztrun`	a vector of values specifying where to truncate lower and upper boundaries, respectively. Default is no truncation.

Details

This is based on a Fortran program, 'ld98', by Reboussin, DeMets, Kim, and Lan. It has some advantages, like making use of probability distributions in R and the ability to specify any valid spending function without changing the program.

`iuse` values of 1 and 2 correspond to alpha spending functions which give O'Brien Fleming and Pocock type boundaries, respectively. A value of 3 is the power family. Here, the spending function is αt^{ϕ}, where ϕ must be greater than 0. A value of 4 is the Hwang-Shih-DeCani family, with spending function $\alpha(1 - e^{-\phi t})/(1 - e^{-\phi})$, where ϕ cannot be 0.

With `iuse`=5, the user will specify any alpha spending function as `asf`. Such a function asf() must be of class 'function', satisfy asf(t)$\leq$1 for t in (0,1), and be strictly increasing. Alpha will be derived as asf(1). If two spending functions are specified, they represent the lower and upper boundary spending functions, respectively, with the values at time 1 designating the lower and upper alphas, respectively. Currently, this option cannot be used for one side of the boundary with one of the other options for the other side. In other words, the user may define one spending function for a one-sided boundary or two for a two-sided boundary, symmetric or asymmetric, but cannot define one spending function and select the other from `iuse` 1 through 4.

Value

'ldBounds' returns an object of 'class' '"ldBounds"'.

An object of class '"ldBounds"' is a list containing the following components:

`bounds.type`	the type of bounds: 1 is 'one-sided', 2 is 'two-sided symmetric', and 3 is 'two-sided asymmetric'. For non-alpha-spending function boundaries (see 'common-bounds' function) 4, 5, and 6 correspond, respectively, to types 1 through 3 above.
`spending.type`	the type(s) of spending function. A descriptive version of the value(s) used for `iuse`

`time`	the original time scale.
`time2`	the second (information) time scale.
`alpha`	the alpha(s) used.
`overall.alpha`	if two-sided, the sum of the two alphas. If one-sided, just alpha.
`lower.bounds`	the vector of lower boundaries calculated. Should be ignored if one-sided.
`upper.bounds`	the vector of upper boundaries calculated.
`exit.pr`	the vector of cumulative exit probabilities at each analysis.
`diff.pr`	the vector of exit probabilities accumulated at each analysis.
`nom.alpha`	the nominal alpha level to be used at each analysis (i.e., bound has been crossed if the p-value at that time is less than this value). Not given for two-sided asymmetric boundaries.

Author(s)

Charlie Casper <charlie.casper@hsc.utah.edu>, Thomas Cook <cook@biostat.wisc.edu>, and Oscar A. Perez

References

Reboussin, D. M., DeMets, D. L., Kim, K. M., and Lan, K. K. G. (2000) Computations for group sequential boundaries using the Lan-DeMets spending function method. *Controlled Clinical Trials*, 21:190-207.

Fortran program 'ld98' by the same authors as above.

DeMets, D. L. and Lan, K. K. G. (1995) *Recent Advances in Clinical Trial Design and Analysis*, Thall, P. F. (ed.). Boston: Kluwer Academic Publishers.

Lan, K. K. G. and DeMets, D. L. (1983) Discrete sequential boundaries for clinical trials. *Biometrika*, 70:659-63.

See Also

Generic functions `summary.ldBounds` and `plot.ldPower`.

`ldPower` for exit probabilities given boundaries OR drift (effect) given power OR confidence interval OR adjusted p-value.

`commonbounds` for some commonly used boundaries that do not use alpha spending.

Examples

```
## From Reboussin, et al. (2000)

#t <- seq(0.2,1,length=5)
#obf.bd <- ldBounds(t)
#summary(obf.bd)
#plot(obf.bd)

# Equivalent bounds to above
obf.bd2 <- ldBounds(5)
```

```
t <- c(0.2292,0.3333,0.4375,0.5833,0.7083,0.8333)
t2 <- c(56,77,126,177,247,318)
power.fam <- ldBounds(t,t2,iuse=3,alpha=0.05)
summary(power.fam)
```

---

ldBounds-defunct — *Defunct functions in ldBounds*

Description

These functions are defunct and no longer available:

- `bounds`: Removed. Use `ldBounds`.
- `drift`: Removed. Use `ldPower`.

---

ldPower — *Power and Other Probabilities for Group Sequential Boundaries*

Description

'ldPower' calculates drift (effect), confidence interval for drift, adjusted p-value, or power and other probabilities given drift for specified group sequential boundaries for interim analyses of accumulating data in clinical trials.

Usage

```
ldPower(t, za=NULL, zb=NULL, t2=t, pow=NULL, drift=NULL,
      conf=NULL, method=NULL, pvaltime=NULL,
      zval=zb[length(zb)])
```

Arguments

`t`	a vector of analysis times or an 'ldBounds' object (from either the 'ldBounds' or 'commonbounds' function). If a vector of analysis times, must be increasing and in (0,1].
`za`	the vector of lower boundaries. If not specified, made symmetric to `zb`.
`zb`	the vector of upper boundaries.
`t2`	the second time scale, usually in terms of amount of accumulating information. By default, same as `t`.
`pow`	the desired power when drift is not specified.
`drift`	the true drift (i.e. treatment effect when t=1). Default is 0 when `pow`, `conf`, and `method` are also left unspecified.

conf	the confidence level when a confidence interval for drift is wanted.
method	the type of adjusted p-value desired. Possible values are 'SW' (stage-wise) and 'LR' (likelihood ratio).
pvaltime	the analysis time at which the final Z-statistic was observed and an adjusted p-value is desired.
zval	the final observed Z-statistic (i.e. when trial is stopped). Used for confidence interval or ajusted p-value. Default is final upper boundary value.

Details

This is based on a Fortran program, 'ld98', by Reboussin, DeMets, Kim, and Lan. It has some advantages, like making use of probability distributions in R. Only one of `pow`, `drift`, `conf`, or `pval` is to be specified and `zval` is only used in the latter two cases.

If `t` is an 'ldBounds' object, `za`, `zb`, `t`, and `t2` are already defined and should not be specified.

Value

'ldPower' returns an object of 'class' '"ldPower"'.

An object of class '"ldPower"' is a list containing the following components:

type	Type of computation performed: 1 is drift given power, 2 is exit probabilities given drift, 3 is confidence interval for drift given final Z-statistic, and 4 is adjusted p-value given final Z-statistic.
time	the original time scale.
time2	the second (information) time scale.
lower.bounds	the vector of lower boundaries given.
upper.bounds	the vector of upper boundaries given.
power	the power. If power is given, it is returned here. If drift is given, the resulting power is calculated.
drift	the drift. If drift is given, it is returned here. If power is given, the drift resulting in given power is calculated.
lower.probs	the vector of exit probabilities across the lower boundary. Returned if power or drift is given.
upper.probs	the same for upper boundary.
exit.probs	the probability at each analysis of crossing the boundary. The sum of `lower.probs` and `upper.probs`.
cum.exit	the cumulative probability of crossing.
conf.level	the desired confidence level, if given.
final.zvalue	the final Z statistic, if given.
conf.interval	the confidence interval for drift, if `conf` and `zval` are given.
p.ordering	the ordering specified for p-value calculation (if given).
p.value	the adjusted p-value if `pval`, `pvaltime`, and `zval` are given.

Author(s)

Charlie Casper <charlie.casper@hsc.utah.edu>, Thomas Cook <cook@biostat.wisc.edu>, and Oscar A. Perez

References

Reboussin, D. M., DeMets, D. L., Kim, K. M., and Lan, K. K. G. (2000) Computations for group sequential boundaries using the Lan-DeMets spending function method. *Controlled Clinical Trials*, 21:190-207.

Fortran program 'ld98' by the same authors as above.

DeMets, D. L. and Lan, K. K. G. (1995) *Recent Advances in Clinical Trial Design and Analysis*, Thall, P. F. (ed.). Boston: Kluwer Academic Publishers.

Lan, K. K. G. and DeMets, D. L. (1983) Discrete sequential boundaries for clinical trials. *Biometrika*, 70:659-63.

See Also

Generic functions `summary.ldPower` and `plot.ldPower`.

`ldBounds` for computation of boundaries using alpha spending function method.

`commonbounds` for boundaries that do not use alpha spending.

Examples

```
## From Reboussin, et al. (2000)

t <- c(0.13,0.4,0.69,0.9,0.98,1)
upper <- c(5.3666,3.7102,2.9728,2.5365,2.2154,1.9668)
bound.pr <- ldPower(t,zb=upper,drift=3.242)
summary(bound.pr)

t <- c(0.2292,0.3333,0.4375,0.5833,0.7083,0.8333)
power.fam <- ldBounds(t,iuse=3,alpha=0.05)
bound.ci <- ldPower(power.fam,conf=0.95,zval=2.82)
bound.p <- ldPower(power.fam,method="LR",pvaltime=5,zval=2.82)
summary(bound.ci)
summary(bound.p)
plot(bound.ci)

obf.bd <- ldBounds(5)
obf.dr <- ldPower(obf.bd,pow=0.9)
summary(obf.dr)
```

plot.ldPower *Plot for Group Sequential Boundaries*

Description

Plot of the sequential boundaries for objects of class '"ldBounds"' or '"ldPower"'.

Usage

```
## S3 method for class 'ldPower'
plot(x, scale = "z", main = NULL, xlab = NULL, ylab =
      NULL, xlim, ylim, las=1, pch=19, type="o", add=F, ...)
```

Arguments

`x`	an object of class '"ldBounds"' or '"ldPower"'.
`scale`	whether the y-axis should use the Z-value (default, "z") or the B-value ("b") scale.
`main`	an overall title for the plot: see `title`.
`xlab`	a title for the x axis: see `title`.
`ylab`	a title for the y axis: see `title`.
`xlim, ylim, las, pch, type, ...`	graphical parameters passed to `plot`.
`add`	logical, whether to add to an existing plot or draw a new plot.

Author(s)

Charlie Casper `<charlie.casper@hsc.utah.edu>`, Thomas Cook `<cook@biostat.wisc.edu>`, and Oscar A. Perez

References

Reboussin, D. M., DeMets, D. L., Kim, K. M., and Lan, K. K. G. (2000) Computations for group sequential boundaries using the Lan-DeMets spending function method. *Controlled Clinical Trials*, 21:190-207.

Fortran program 'ld98' by the same authors as above.

DeMets, D. L. and Lan, K. K. G. (1995) *Recent Advances in Clinical Trial Design and Analysis*, Thall, P. F. (ed.). Boston: Kluwer Academic Publishers.

Lan, K. K. G. and DeMets, D. L. (1983) Discrete sequential boundaries for clinical trials. *Biometrika*, 70:659-63.

See Also

Generic functions `summary.ldBounds` and `summary.ldPower`, `ldBounds` to calculate sequential boundaries, `ldPower` for exit probabilities given boundaries OR drift (effect) given power OR confidence interval OR adjusted p-value.

Examples

```
## See 'ldBounds' or 'ldPower'.
```

`summary.ldBounds` *Summary for Group Sequential Boundaries*

Description

'summary' method for class '"ldBounds"'.

Usage

```
## S3 method for class 'ldBounds'
summary(object, digit = 5, ...)
## S3 method for class 'summary.ldBounds'
print(x, digit = 5, ...)
```

Arguments

`object`	an object of class '"ldBounds"', a result of a call to `ldBounds` or `commonbounds`.
`x`	an object of class '"summary.ldBounds"', a result of a call to `summary.ldBounds`.
`digit`	the number of significant digits to use when printing.
`...`	further arguments passed to or from other methods.

Value

The function 'summary.ldBounds' returns a list of summary values of the group sequential boundary calculations given in 'object'.

Author(s)

T. Charles Casper <charlie.casper@hsc.utah.edu> and Oscar A. Perez

References

Reboussin, D. M., DeMets, D. L., Kim, K. M., and Lan, K. K. G. (2000) Computations for group sequential boundaries using the Lan-DeMets spending function method. *Controlled Clinical Trials*, 21:190-207.

Fortran program 'ld98' by the same authors as above.

DeMets, D. L. and Lan, K. K. G. (1995) *Recent Advances in Clinical Trial Design and Analysis*, Thall, P. F. (ed.). Boston: Kluwer Academic Publishers.

Lan, K. K. G. and DeMets, D. L. (1983) Discrete sequential boundaries for clinical trials. *Biometrika*, 70:659-63.

See Also

`ldBounds` for computation of boundaries using alpha spending function method. `ldPower` for exit probabilities given boundaries OR drift (effect) given power OR confidence interval OR adjusted p-value.

Examples

```
## See function 'ldBounds'
```

---

summary.ldPower *Summary for Computations Related to Group Sequential Boundaries*

Description

'summary' method for class '"ldPower"'.

Usage

```
## S3 method for class 'ldPower'
summary(object, ...)
## S3 method for class 'summary.ldPower'
print(x, digit = 5, ...)
```

Arguments

`object`	an object of class '"ldPower"', a result of a call to `ldPower`.
`x`	an object of class '"summary.ldPower"', a result of a call to `summary.ldPower`.
`digit`	the number of significant digits to use when printing.
`...`	further arguments passed to or from other methods.

Value

The function 'summary.ldPower' returns a list of summary values of the group sequential boundary calculations given in 'object'.

Author(s)

T. Charles Casper <charlie.casper@hsc.utah.edu> and Oscar A. Perez

References

Reboussin, D. M., DeMets, D. L., Kim, K. M., and Lan, K. K. G. (2000) Computations for group sequential boundaries using the Lan-DeMets spending function method. *Controlled Clinical Trials*, 21:190-207.

Fortran program 'ld98' by the same authors as above.

DeMets, D. L. and Lan, K. K. G. (1995) *Recent Advances in Clinical Trial Design and Analysis*, Thall, P. F. (ed.). Boston: Kluwer Academic Publishers.

Lan, K. K. G. and DeMets, D. L. (1983) Discrete sequential boundaries for clinical trials. *Biometrika*, 70:659-63.

See Also

`ldBounds` for computation of boundaries using alpha spending function method. `ldPower` for exit probabilities given boundaries OR drift (effect) given power OR confidence interval OR adjusted p-value.

Examples

```
## See function 'ldPower'
```

Index