Package ‘AGSDest’

February 25, 2019

Type Package
Title Estimation in Adaptive Group Sequential Trials
Version 2.3.2
Date 2019-02-24
Author Niklas Hack [aut, cre], Werner Brannath [aut], Matthias Brueckner [aut]
Maintainer Matthias Brueckner <matthias.brueckner@posteo.de>
Imports stats, graphics, grDevices, ldbounds
Description Calculation of repeated confidence intervals as well as confidence
intervals based on the stage-wise ordering in group sequential designs and
adaptive group sequential designs. For adaptive group sequential designs
the confidence intervals are based on the conditional rejection probability
principle. Currently the procedures do not support the use of futility
boundaries or more than one adaptive interim analysis.
License GPL (>= 2)
RoxygenNote 6.1.1
NeedsCompilation yes
Repository CRAN
Date/Publication 2019-02-25 10:50:04 UTC

R topics documented:

adapt ................................................................. 2
AGSDest ............................................................ 4
AGSTobj ............................................................. 5
as.AGST ............................................................. 10
as.GST .............................................................. 11
cer ................................................................. 11
cp ................................................................. 12
GSTobj ............................................................. 13
plan.GST ........................................................... 17
pvalue ............................................................. 18
seqconfint ......................................................... 21
typelerr .......................................................... 24
adapt

Description
adapt is a function that performs adaptations and plans the secondary group sequential trial. The effect size used for planning the secondary trial is a weighted mean between the interim estimate theta and the initially assumed estimate delta (pT\$delta) of the primary trial.

Usage
adapt(pT, idL, SF, phi, cp, theta = idL$z/(pT$t[idL] * pT$Imax), I2min, I2max, swImax, delta = pT$delta, weight = 1, warn = TRUE)

Arguments
- \texttt{pT} object of the class \texttt{GSTobj}; primary trial design
- \texttt{id} interim data; a list with the variables \texttt{T} and \texttt{z}; list(\texttt{T} = stage of interim analysis, \texttt{z} = interim \texttt{z}-statistic)
- \texttt{SF} spending function for the secondary trial
- \texttt{phi} parameter of spending function for the secondary trial when \texttt{SF}=3 or 4 (See below)
- \texttt{cp} conditional power
- \texttt{theta} new effect size (default: estimate from interim analysis)
- \texttt{I2min} minimal total information of secondary trial
- \texttt{I2max} maximal total information of secondary trial
- \texttt{swImax} maximal incremental information per stage
- \texttt{delta} initially assumed effect size for the primary trial (default: estimate from primary trial)
- \texttt{weight} weight of \texttt{theta} when updating the effect size estimate as weighted mean of \texttt{theta} and \texttt{delta}
- \texttt{warn} option if warnings should be printed to the screen (default: true)

Details
If no adaptation is performed then this indicates that the original plan is kept. In this case \texttt{sT} is set to \texttt{NULL}. If an adaptation is performed \texttt{sT} is a list which contains the following elements:

- \texttt{K} number of stages
- \texttt{a} lower critical bounds of secondary group sequential design (are currently always set to -8)
- \texttt{b} upper critical bounds of secondary group sequential design
- \texttt{t} vector with cumulative information fractions
- \texttt{al} alpha (type I error rate); equal to the conditional type I error rate of the primary trial
SF    spending function
phi   parameter of spending function when SF=3 or 4 (See below)
alab  alpha-absorbing parameter values of secondary group sequential design
als   alpha-values "spent" at each stage of secondary group sequential design
imax  maximum information number
delta effect size used for planning the secondary trial
cp    conditional power

A value of SF=3 is the power family. Here, the spending function is $t^\phi$, where phi must be greater than 0. A value of SF=4 is the Hwang-Shih-DeCani family, with the spending function $(1 - e^{-\phi t})/(1 - e^{-\phi})$, where phi cannot be 0.

Value
adapt returns an object of the class GSTobj; the design of the secondary trial. The adaptation rule is as in the first simulation example of Brannath et al.(2008). If no adaptations are performed, the function returns st = NULL. An object of class GSTobj is a list containing the following components:

st secondary trial

Author(s)
Niklas Hack <niklas.hack@meduniwien.ac.at> and Werner Brannath <werner.brannath@meduniwien.ac.at>

References

See Also
GSTobj, print.GSTobj, plot.GSTobj, plan.GST

Examples
#The following performs an adaptation of the sample size and
#number of interim analyses after the first stage of the primary trial.

pT=plan.GST(K=3,SF=4,phi=-4,alpha=0.05,delta=6,pow=0.9,compute.alab=TRUE,compute.als=TRUE)
id=list(T=1, z=1.090728)

swimax=0.0625

I2min=3*swimax
I2max=3*swimax

st=adapt(pT=pT,id=id,SF=1,phi=0,cp=0.8,theta=5,I2min,I2max,swimax)
Description

The package allows to compute repeated confidence intervals as well as confidence intervals based on the stage-wise ordering in group sequential designs (GSD; see Jennison and Turnbull, 1989; Tsiatis, Rosner, Mehta, 1984) and adaptive group sequential designs (Mehta, Bauer, Posch, Brannath, 2007; Brannath, Mehta, Posch, 2008). For adaptive group sequential designs the confidence intervals are based on the conditional rejection probability principle of Mueller and Schaefer (2001). This principle allows us to perform data dependent changes to the sample size, the spending function, and the number and spacing of interim looks while preserving the overall type I error rate. Currently the procedures do not support the use of futility boundaries as well as more than one adaptive interim analysis. Furthermore, the package is currently restricted to the computation of lower one-sided confidence intervals.

Details

Package: AGSDest
Type: Package Version: 2.2 Date: 2015-01-12 License: GPL Version 2 or later

Main functions:
- adapt: Performs adaptations at an interim analysis of a GSD to the sample size, number of interim stages and spending function based on the conditional power in a GSD at an interim analysis; the result is a secondary trial plan
- GST: Plans a group sequential trial
- cer: Computes the conditional type I error rate (also called conditional rejection probability) of a GSD at an interim analysis
- typeierr: Computes the type I error rate of a GSD
- pvalue: Computes the repeated or stage-wise adjusted p-value for a classical GSD or for a GSD with design adaptations
- seqconfint: Computes the repeated confidence bound and confidence bound based on the stage-wise ordering for a GSD or for a GSD with design adaptations

Subfunctions:
- as.GST: Builds a group sequential trial object
- as.AGST: Builds an adaptive group sequential trial object

Author(s)
Niklas Hack <niklas.hack@meduniwien.ac.at> and Werner Brannath <werner.brannath@meduniwien.ac.at>

References


**Examples**

```r
pt=plan.GST(K=3, SF=4, phi=-4, alpha=0.05, delta=6, pow=0.9, compute.alab=TRUE, compute.als=TRUE)
iD=list(T=1, z=1.090728)
swImax=0.0625
I2min=3*swImax
I2max=3*swImax
st=adapt(pt, iD, iD=1, phi=0, cp=0.8, theta=5, I2min, I2max, swImax)
sto=list(T=2, z=2.393)
AGST <- as.AGST(pT=pt, iD=iD, sT=sT, sto=sto)
```

### The following calculates the stage-wise adjusted p-value
### of a group sequential trial after a design adaptation

```r
pvalue(AGST, type="so")
```

### and the corresponding confidence bound based on the stage-wise ordering.

```r
seqconfint(AGST, type="so")
```

### Both, the p-value and the confidence interval can be calculated by
### the summary function

```r
summary(AGST, ctype="so", ptype="so")
```

### End(Not run)

---

**AGSTobj**

*Adaptive group sequential trial object (AGSTobj)*

---

**Description**

The AGSTobj includes design and outcome of primary and secondary trial.

**Usage**

```r
AGSTobj(x, ...)
```

## S3 method for class 'AGSTobj'

```r
plot(x, main = c("primary trial", "secondary trial"),
```
print.pdf = FALSE, ...)

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

## S3 method for class 'AGSTobj'
summary(object, ctype = c("r", "so"), ptype = c("r", "so"), etype = c("ml", "mu", "cons"), overwrite = FALSE, ...)

## S3 method for class 'summary.AGSTobj'
print(x, ...)

### Arguments

**x**
- object of the class AGSTobj

**...**
- additional arguments.

**main**
- Title of the plots (default: first plot: "primary trial"; second plot: "secondary trial")

**print.pdf**
- option; if TRUE a pdf file is created. Instead of setting print.pdf to TRUE, the user can specify a character string giving the name or the path of the file.

**object**
- object of the class AGSTobj

**ctype**
- confidence type: repeated "r" or stage-wise ordering "so" (default: c("r", "so"))

**ptype**
- p-value type: repeated "r" or stage-wise ordering "so" (default: c("r", "so"))

**etype**
- point estimate: maximum likelihood "ml", median unbiased "mu", or conservative "cons" (default: c("ml", "mu", "cons"))

**overwrite**
- option; if TRUE all old values are deleted and new values are calculated (default: FALSE)

### Details

A AGSTobj object is designed.

The function summary returns an object of class AGSTobj. *ctype* defines the type of confidence interval that is calculated.

"r"  Repeated confidence bound for a GSD with design adaptations
"so"  Confidence bound for a GSD with design adaptation based on the stage-wise ordering

The calculated confidence bounds are saved as:

- cb.r  repeated confidence bound
- cb.so  confidence bound based on the stage-wise ordering

*ptype* defines the type of *p*-value that is calculated.
Repeated p-value for a GSD with design adaptations
Stage-wise adjusted p-value for a GSD with design adaptations

The calculated p-values are saved as:

\[
\begin{array}{ll}
\text{pvalue.r} & \text{repeated p-value} \\
\text{pvalue.so} & \text{stage-wise adjusted p-value}
\end{array}
\]

etype defines the type of point estimate

Maximum likelihood estimate (ignoring the sequential and adaptive nature of the design)
Median unbiased estimate (stage-wise lower confidence bound at level 0.5) for a GSD with design adaptations
Conservative estimate (repeated lower confidence bound at level 0.5) for a GSD with design adaptations

The calculated point estimates are saved as:

\[
\begin{array}{ll}
\text{est.ml} & \text{Maximum likelihood estimate} \\
\text{est.mu} & \text{Median unbiased estimate} \\
\text{est.cons} & \text{Conservative estimate}
\end{array}
\]

The stage-wise adjusted confidence bound, p-value and the median unbiased point estimate can only be calculated at the stage where the trial stops and are only valid if the stopping rule is met.

The repeated confidence bound, repeated p-value, conservative estimate and maximum likelihood estimate can be calculated at every stage of the trial and not just at the stage where the trial stops and are also valid if the stopping rule is not met. For calculating the repeated confidence bounds or p-values the user has to specify stO (secondary trial outcome) in the object AGSTobj (see example below). If the stopping rule is not met in object stO then stage-wise adjusted confidence bounds and p-values will not be computed while a warning message is given when their computation have erroneously been specified.

**Value**

An object of class AGSTobj, which is basically a list with the elements

\[
\begin{array}{ll}
\text{cb.so} & \text{confidence bound based on the stage-wise ordering (stage-wise adjusted confidence bound)} \\
\text{cb.r} & \text{repeated confidence bound} \\
\text{pvalue.so} & \text{p-value based on the stage-wise ordering (stage-wise adjusted p-value)} \\
\text{pvalue.r} & \text{repeated p-value} \\
\text{est.ml} & \text{maximum likelihood estimate} \\
\text{est.mu} & \text{median unbiased point estimate} \\
\text{est.cons} & \text{conservative point estimate}
\end{array}
\]
AGSTobj

pT  number of stages
K   alpha (type I error rate)
a   lower critical bounds of primary group sequential design (are currently always set to -8)
b   upper critical bounds of primary group sequential design
t   vector with cumulative information fraction
SF  spending function (for details see below)
phi parameter of spending function when SF=3 or 4 (for details see below)
als alpha-absorbing parameter values of primary group sequential design
als lower critical bounds of primary group sequential design
Imax maximum information number
delta effect size used for planning the primary trial
cp  conditional power for planning the primary trial
iD  stage of the adaptation
L   z-statistic at adaptive interim analysis
sT  number of stages
al  conditional rejection probability
a   lower critical bounds of secondary group sequential design (are currently always set to -8)
b   upper critical bounds of secondary group sequential design
t   vector with cumulative information fraction
SF  spending function (for details see below)
phi parameter of spending function when SF=3 or 4 (for details see below)
Imax maximum information number
delta effect size used for planning the secondary trial
cp  conditional power for planning the secondary trial
sto stage where trial stops
T   z-statistic at stage where trial stops
z   z-statistic at stage where trial stops

Note

The AGSTobj should always have the same ordering and names as given in the list above or as given in the example. 1. pt, 2. id, 3. st, 4. sto SF defines the spending function. SF = 1 O’Brien and Fleming type spending function of Lan and DeMets (1983) SF = 2 Pocock type spending function of Lan and DeMets (1983) SF = 3 Power family \((c_\alpha * t^\phi)\); phi must be greater than 0. SF = 4 Hwang-Shih-DeCani family\(\frac{(1 - e^{-\phi t})}{(1 - e^{-\phi t})}\), where phi cannot be 0. A value of SF=3 corresponds
AGSTobj

to the power family. Here, the spending function is $t^\phi$, where phi must be greater than 0. A value of SF=4 corresponds to the Hwang-Shih-DeCani family, with the spending function $(1 - e^{-\phi t})/(1 - e^{-\phi})$, where phi cannot be 0. If a path is specified for print.pdf, all \ must be changed to /. If a filename is specified the ending of the file must be (.pdf). In the current version the vector of lower bounds a should be set to rep(-8,K).

Author(s)

Niklas Hack <niklas.hack@meduniwien.ac.at> and Werner Brannath <werner.brannath@meduniwien.ac.at>

See Also

AGSTobj, print.AGSTobj, plot.AGSTobj, summary.AGSTobj

Examples

## not run:
pt]plan.GST(K=3,SF=4,phi=-4,alpha=0.05,delta=6,pow=0.9,compute.alab=TRUE,compute.als=TRUE)

iD=list(T=1, z=1.090728)

swImax=0.0625

I2min=3*swImax
I2max=3*swImax

sT=adapt(pt=pt,iD=iD,SF=1,phi=0,cp=0.8,theta=5,I2min,I2max,swImax)

sTo=list(T=2, z=2.393)

AGST<-as.AGST(pt=pt,iD=iD,sT=sT,sTo=sTo)
AGST
plot(AGST)
AGST<-summary(AGST)
plot(AGST)

##The repeated confidence interval and p-value at an earlier stage
##than the one where the trial stops (T=3).

summary(as.AGST(pt,iD,sT,sTo=list(T=1,z=1.7)),ctype="r",ptype="r")

##If the stage-wise adjusted confidence interval is calculated at this stage,
##the function returns an error message

summary(as.AGST(pt,iD,sT,sTo=list(T=1,z=1.7)),ctype="so",ptype="so")

## End(Not run)
as.AGST

as Adaptive Group Sequential Trial

Description

Function as.AGST builds an adaptive group sequential trial object

Usage

as.AGST(pT, iD, sT, sTo = NULL)

Arguments

pT  object of the class GSTobj; primary trial design
iD  interim data; a list with the variables T and z; list(T = stage of interim analysis, z = interim z-statistic)
sT  object of the class GSTobj; secondary trial design
sTo secondary trial outcome; a list with the variables T and z; list(T = stage where trial stops, z = z-statistic at stage where trial stops)

Value

Returns a list containing the pT, iD, sT and sTo with class=AGSTobj

Author(s)

Niklas Hack <niklas.hack@meduniwien.ac.at> and Werner Brannath <werner.brannath@meduniwien.ac.at>

See Also

AGSTobj

Examples

pT=plan.GST(K=3, SF=4, phi=-4, alpha=0.05, delta=6, pow=0.9, compute.alab=TRUE, compute.als=TRUE)
iD=list(T=1, z=7.090728)
sw1max=0.0625
I2min=3*sw1max
I2max=3*sw1max

sT=adapt(pT=pT, iD=iD, SF=1, phi=0, cp=0.8, theta=5, I2min, I2max, sw1max)
sTo=list(T=2, z=2.393)
AGST <- as.AGST(pT=pT, iD=iD, sT=sT, sTo=sTo)
Description
Function as.GST builds a group sequential trial object

Usage
as.GST(GSd, GSDo)

Arguments
GSD  object of the class GSTobj; group sequential design
GSDo group sequential design outcome; a list with the variables T and z; list(T = stage where trial stops, z = z-statistic at stage where trial stops)

Value
Returns a list containing the GSD and GSDo with class=GSTobj

Author(s)
Niklas Hack <niklas.hack@meduniwien.ac.at> and Werner Brannath <werner.brannath@meduniwien.ac.at>

See Also
GSTobj

Examples
GSD <- plan.GST(K=4, SF=1, phi=0, alpha=0.025, delta=6, pow=0.8, compute.alab=TRUE, compute.als=TRUE)
GSDo <- list(T=2, z=3.1)
GST <- as.GST(GSD=GSD, GSDo=GSDo)
GST

cer
Conditional type I error rate (also called conditional rejection probability)

Description
Calculates the conditional type I error rate of a GSD

Usage
cer(pT, iD)
Arguments

pt object of the class GSTobj; primary trial design
id interim data; a list with the variables T and z; list(T = stage of interim analysis, z = interim z-statistic)

Value

cer conditional type I error rate

Author(s)

Niklas Hack <niklas.hack@meduniwien.ac.at> and Werner Brannath <werner.branath@meduniwien.ac.at>

References


See Also

plan.GST

Examples

##The following calculates the conditional type I error rate ##under the null hypothesis after an adaptation at the second stage ##of the primary trial.
pt=plan.GST(K=4,SF=1,phi=0,alpha=0.025,delta=6,pow=0.8,compute.alab=TRUE,compute.als=TRUE)
cer(pT=pt,id=list(T=2, z=1.09))

---

cp conditional power of a GSD

Description

cp is a function that computes the conditional power of a GSD.

Usage

cp(GSD)

Arguments

GSD object of the class GSTobj or list with the following elements: K = number of stages, a = vector with futility boundaries (not supported yet), b = rejection boundaries, t = vector with information fractions, Imax = maximum information number, delta = effect size used for planning the trial; see example blow.
GSTobj

Value

`cp` returns the conditional power of a GSD.

Author(s)

Niklas Hack <niklas.hack@meduniwien.ac.at> and Werner Brannath <werner.brannath@meduniwien.ac.at>

References

Schoenfeld, D (2001) "A simple Algorithm for Designing Group Sequential Clinical Trials", Biometrics, 27, 972-974

See Also

GSTobj

Examples

## The following calculates the conditional power of a GSD.

```r
gsd <- list(k=4,a=rep(-8,4),b=c(4.333,2.963,2.359,2.014),t=c(0.25,0.5,0.75,1),Imax=0.22,delta=4)

cp(gsd)
```

---

**GSTobj**

*Group sequential trial object (GSTobj)*

**Description**

The GSTobj includes design and outcome of primary trial.

**Usage**

GSTobj(x, ...)

## S3 method for class 'GSTobj'
plot(x, main = "GSD", print.pdf = FALSE, ...)

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

## S3 method for class 'GSTobj'
summary(object, ctype = c("r", "so"), ptype = c("r", "so"),
etype = c("ml", "mu", "cons"), overwrite = FALSE, ...)

## S3 method for class 'summary.GSTobj'
print(x, ...)
Arguments

x  
   object of the class GSTobj
...
   additional arguments.
main  
   Title of the plots (default: "GSD")
print.pdf  
   option; if TRUE a pdf file is created. Instead of setting print.pdf to TRUE, the user can specify a character string giving the name or the path of the file.
object  
   object of the class GSTobj
ctype  
   confidence type: repeated "r" or stage-wise ordering "so" (default: c("r", "so"))
ptype  
   p-value type: repeated "r" or stage-wise ordering "so" (default: c("r", "so"))
etype  
   point estimate: maximum likelihood "ml", median unbiased "mu" or conservative "cons" (default: c("ml", "mu", "cons"))
overwrite  
   option; if TRUE all old values are deleted and new values are calculated (default: FALSE)

Details

A GSTobj object is designed.
The function summary returns an object of class GSTobj.
ctype defines the type of confidence interval that is calculated.

"r"  Repeated confidence bound for a classical GSD
"so"  Confidence bound for a classical GSD based on the stage-wise ordering

The calculated confidence bounds are saved as:

   cb.r  repeated confidence bound
   cb.so  confidence bound based on the stage-wise ordering

ptype defines the type of p-value that is calculated.

"r"  Repeated p-value for a classical GSD
"so"  Stage-wise adjusted p-value for a classical GSD

The calculated p-values are saved as:

   pvalue.r  repeated p-value
   pvalue.so  stage-wise adjusted p-value

etype defines the type of point estimate
"ml"  maximum likelihood estimate (ignoring the sequential nature of the design)
"mu"  median unbiased estimate (stage-wise lower confidence bound at level 0.5) for a classical GSD
"cons" Conservative estimate (repeated lower confidence bound at level 0.5) for a classical GSD

The calculated point estimates are saved as:

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>est.ml</td>
<td>Maximum likelihood estimate</td>
</tr>
<tr>
<td>est.mu</td>
<td>Median unbiased estimate</td>
</tr>
<tr>
<td>est.cons</td>
<td>Conservative estimate</td>
</tr>
</tbody>
</table>

The stage-wise adjusted confidence interval and p-value and the median unbiased point estimate can only be calculated at the stage where the trial stops and is only valid if the stopping rule is met. The repeated confidence interval and repeated p-value, conservative estimate and maximum likelihood estimate can be calculated at every stage of the trial and not just at the stage where the trial stops and is also valid if the stopping rule is not met. For calculating the repeated confidence interval or p-value at any stage of the trial the user has to specify the outcome GSDo in the object GSTobj (see example below).

Value

An object of class GSTobj, is basically a list with the elements

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cb.so</td>
<td>confidence bound based on the stage-wise ordering</td>
</tr>
<tr>
<td>cb.r</td>
<td>repeated confidence bound</td>
</tr>
<tr>
<td>pvalue.so</td>
<td>stage-wise adjusted p-value</td>
</tr>
<tr>
<td>pvalue.r</td>
<td>repeated p-value</td>
</tr>
<tr>
<td>est.ml</td>
<td>maximum likelihood estimate</td>
</tr>
<tr>
<td>est.mu</td>
<td>median unbiased point estimate</td>
</tr>
<tr>
<td>est.cons</td>
<td>conservative point estimate</td>
</tr>
</tbody>
</table>

GSTobj

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>K</td>
<td>number of stages</td>
</tr>
<tr>
<td>a1</td>
<td>alpha (type I error rate)</td>
</tr>
<tr>
<td>a</td>
<td>lower critical bounds of group sequential design (are currently always set to -8)</td>
</tr>
<tr>
<td>b</td>
<td>upper critical bounds of group sequential design</td>
</tr>
<tr>
<td>t</td>
<td>vector with cumulative information fraction</td>
</tr>
<tr>
<td>SF</td>
<td>spending function (for details see below)</td>
</tr>
<tr>
<td>phi</td>
<td>parameter of spending function when SF=3 or 4 (for details see below)</td>
</tr>
<tr>
<td>alab</td>
<td>alpha-absorbing parameter values of group sequential design</td>
</tr>
<tr>
<td>als</td>
<td>alpha-values &quot;spent&quot; at each stage of group sequential design</td>
</tr>
<tr>
<td>Imax</td>
<td>maximum information number</td>
</tr>
</tbody>
</table>
delta  effect size used for planning the primary trial

cp    condition\(\alpha\) power of the trial

GSDo  stage where trial stops

z     \(z\)-statistic at stage where trial stops

Note

\(SF\) defines the spending function. \(SF = 1\) O’Brien and Fleming type spending function of Lan and DeMets (1983) \(SF = 2\) Pocock type spending function of Lan and DeMets (1983) \(SF = 3\) Power family \((c_{\alpha}t^{phi})\). \(phi\) must be greater than 0. \(SF = 4\) Hwang-Shih-DeCani family, \((1-e^{-\phi t})/(1-e^{-\phi})\), where \(phi\) cannot be 0. A value of \(SF=3\) corresponds to the power family. Here, the spending function is \(t^{phi}\), where \(phi\) must be greater than 0. A value of \(SF=4\) corresponds to the Hwang-Shih-DeCani family, with the spending function \((1-e^{-\phi t})/(1-e^{-\phi})\), where \(phi\) cannot be 0. If a path is specified for \(print.pdfs\), all \(\backslash\) must be changed to \(/\). If a filename is specified the ending of the file must be (.pdf). In the current version \(a\) should be set to rep(-8,K)

Author(s)

Niklas Hack <niklas.hack@meduniwien.ac.at> and Werner Brannath <werner.brannath@meduniwien.ac.at>

See Also

\(GSTobj, print.GSTobj, plot.GSTobj, summary.GSTobj\)

Examples

\[
GSD=plan.GST(K=4,SF=1,phi=0,\alpha=0.025,\text{delta}=6,\text{pow}=0.8,\text{compute.alab}=TRUE,\text{compute.als}=TRUE)
\]
\[
GST<-as.GST(GSD,GSDo=list(T=2, z=3.1))
\]
\[
GST
\]
\[
plot(GST)
\]
\[
GST<-summary(GST)
\]
\[
plot(GST)
\]
\[
\text{##The repeated confidence interval, p-value and maximum likelihood estimate}
\]
\[
\text{##at the earlier stage T=1 where the trial stopping rule is not met.}
\]
\[
\text{summary(as.GST(GSD,GSDo=list(T=1, z=0.7)),ctype="r",ptype="r",etyp="ml")}
\]
\[
\text{## Not run:}
\]
\[
\text{##If e.g. the stage-wise adjusted confidence interval is calculated at this stage,}
\]
\[
\text{##the function returns an error message}
\]
\[
\text{summary(as.GST(GSD,GSDo=list(T=1, z=0.7)),ctype="so",etyp="mu")}
\]
\[
\text{## End(Not run)}
\]
Plans a group sequential trial (GST)

Usage

plan.GST(K, t = (1:K)/K, Imax = NULL, SF, phi, alpha, delta = NULL, pow = NULL, compute.alab = TRUE, compute.als = TRUE)

Arguments

K  number of stages

T  vector with the cumulative information fraction (default: (1:K)/K)

Imax  maximum information number (default: NULL)

SF  spending function (for details see below)

phi  parameter of spending function when SF=3 or 4 (See below)

alpha  alpha (type I error rate)

delta  effect size (alternative)(default: NULL)

pow  power (default: NULL)

compute.alab  specify if alpha-absorbing parameter values should be calculated (default: TRUE)

compute.als  specify if alpha-values "spent" at every stage should be calculated (default: TRUE)

Details

The user has to specify either Imax or delta and pow. If all three items are specified, the pre-defined maximum information number is newly calculated from the information for delta and power, and Imax is overwritten.

SF defines the spending function.

SF = 1 O’Brien and Fleming type spending function of Lan and DeMets (1983)

SF = 2 Pocock type spending function of Lan and DeMets (1983)

SF = 3 Power family \(c_\alpha + \phi^t\); phi must be greater than 0

SF = 4 Hwang-Shih-DeCani family; \((1 - e^{-\phi t})/(1 - e^{-\phi})\), where phi cannot be 0

Value

plan.GST returns an object of the class GSTobj. An object of class GSTobj is a list containing the following components:

K  number of stages
a  lower critical bounds of group sequential design (are currently always set to -8)
b  upper critical bounds of group sequential design
t  vector with cumulative information fraction
al  alpha (type I error)
SF  spending function
phi  parameter of spending function when SF=3 or 4 (See below)
Imax  maximum information number
delta  effect size used for planning the primary trial

Author(s)
Niklas Hack <niklas.hack@meduniwien.ac.at> and Werner Brannath <werner.brannath@meduniwien.ac.at>

References

See Also
GSTobj, print.GSTobj, plot.GSTobj

Examples
```r
##The following plans an O'Brien and Flaming group sequential design (GSD)
##with 4 stages and equally spaced looks.
pt <- plan.GST(K=4, SF=1, phi=0, alpha=0.025, delta=6, pow=0.8, compute.alab=TRUE, compute.als=TRUE)

pvalue(pt)
```

Description
Calculates the repeated or stage-wise adjusted p-value of a GSD or a AGSD

Usage
```
pvalue(object, type = c("r", "so"))
```

Arguments
<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>object</td>
<td>object of the class GSTobj or of the class AGSTobj</td>
</tr>
<tr>
<td>type</td>
<td>p-value type: repeated &quot;r&quot;, stage-wise ordering &quot;so&quot; or both &quot;b&quot; (default: &quot;b&quot;)</td>
</tr>
</tbody>
</table>
Details

object can be an object of the class GSTobj or an object of the class AGSTobj. The function identifies the class of the object and calculates the corresponding p-value (classical or adaptive).

If object has class GSTobj, then a p-value for a classical GSD is calculated. type defines the type of confidence interval that is calculated

"r"  Repeated p-value for a classical GSD
"so"  Stage-wise adjusted p-value for a classical GSD

If object has class AGSTobj, then a p-value for a GSD with design adaptation is calculated. type defines the type of confidence interval that is calculated

"r"  Repeated p-value for a GSD with design adaptations
"so"  Stage-wise adjusted p-value for a GSD with design adaptations

Value

The function pvalue returns according to the object the classical or adaptive p-value for the final stage. If the parameter value has the class GSTobj the classical p-value is calculated. If the parameter value has the class AGSTobj the adaptive p-value is calculated.

The calculated p-values are saved as:

pvalue$r$ repeated p-value
pvalue$so$ stage-wise adjusted p-value

Note

The stage-wise adjusted p-value can only be calculated at the stage where the trial stops and is only valid if the stopping rule is met.

The repeated p-value can be calculated at every stage of the trial and not just at the stage where the trial stops and is also valid if the stopping rule is not met.

For calculating the sequential p-values at stage $T$ the user has to specify the outcome GSDo in the object GSTobj or sto (secondary trial outcome) in the object AGSTobj. A trial outcome is a list of the form list=(T=stage of interim analysis, z = interim z-statistic); see the example below.

Author(s)

Niklas Hack <niklas.hack@meduniwien.ac.at> and Werner Brannath <werner.brannath@meduniwien.ac.at>

References


See Also

`AGSTobj`, `GSTobj`

Examples

```r
## The following calculates the repeated p-value of a group sequential trial

## Not run:
GSD=plan.GST(K=4,SF=1,phi=0,alpha=0.025,delta=6,pow=0.8,compute.alab=TRUE,compute.als=TRUE)

GST<-as.GST(GSD=GSD,GSDo=list(T=2, z=3.1))
pvalue(GST,type="r")

## The stage-wise adjusted p-value of a group sequential trial is calculated by
pvalue(GST,type="so")

## The repeated p-value at the earlier stage T=1 where the trial stopping rule is not met.
pvalue(as.GST(GSD=GSD,GSDo=list(T=1,z=0.7)),type="r")

## If the stage-wise adjusted p-value is calculated at this stage,
## the function returns an error message
pvalue(as.GST(GSD,GSDo=list(T=1,z=0.7)),type="so")

## The repeated and the stage-wise adjusted p-value of a
## group sequential trial after a design adaptation is calculated by
pT=plan.GST(K=3,SF=4,phi=-4,alpha=0.05,delta=6,pow=0.9,compute.alab=TRUE,compute.als=TRUE)
iD=list(T=1, z=1.090728)

swImax=0.0625
I2min=3*swImax
I2max=3*swImax

st=adapt(pT=iD, SF=1, phi=0, cp=0.8, theta=5, I2min,I2max,swImax)
sTo=list(T=2, z=2.393)
```
seqconfint

Calculates confidence interval

Description

Calculates the repeated confidence bound or the confidence bound based on the stage-wise ordering of a GSD or a AGSD

Usage

seqconfint(object, type = c("r", "so"), level = NULL)

Arguments

- object: object of the class GSTobj or of the class AGSTobj
- type: confidence type: repeated "r", stage-wise ordering "so" or both "b" (default: "b")
- level: type I error rate (default: NULL)

Details

object can be an object of the class GSTobj or an object of the class AGSTobj. The function identifies the class of the object and calculates the corresponding confidence interval (classical or adaptive).

If object has class GSTobj, then a confidence bound for a classical GSD is calculated. type defines the type of confidence interval that is calculated

"r"  Repeated confidence bound for a classical GSD
"so"  Confidence bound for a classical GSD based on the stage-wise ordering

If object has class AGSTobj, then a confidence bound for a GSD with design adaptation is calculated. type defines the type of confidence interval that is calculated

"r"  Repeated confidence bound for a GSD with design adaptations
"so"  Confidence bound for a GSD with design adaptation based on the stage-wise ordering

By setting level to the value 0.5 the conservative point estimate is calculated. Default is the level of the primary trial.

Value

The function `seqconfint` returns according to the class of `object` the classical or adaptive confidence bound. If `object` has class `GSTobj` the classical confidence bound is calculated. If the parameter value has the class `AGSTobj` the adaptive confidence bound is calculated.

The calculated confidence bounds are saved as:

- `cb.r`: repeated confidence bound
- `cb.so`: confidence bound based on the stage-wise ordering

If the `level` is set to 0.5, the calculated point estimates are:

- `est.mu`: Median unbiased point estimate, based on the stage-wise ordering
- `est.cons`: Flexible, but conservative repeated point estimate

Note

The stage-wise adjusted confidence interval can only be calculated at the stage where the trial stops and is only valid if the stopping rule is met.

The repeated confidence interval can be calculated at every stage of the trial and not just at the stage where the trial stops and is also valid if the stopping rule is not met.

For calculating the sequential confidence intervals at stage `t` the user has to specify the outcome `gsto` in the object `GSTobj` or `sto` (secondary trial outcome) in the object `AGSTobj`. A trial outcome is a list of the form `list(t=stage of interim analysis, z = interim z-statistic)`: see the example below.

Author(s)

Niklas Hack <niklas.hack@meduniwien.ac.at> and Werner Brannath <werner.brannath@meduniwien.ac.at>

References


seqconfint

See Also
AGSTobj, GSTobj

Examples

## The following calculates the repeated confidence bound of a group sequential trial

GSD <- plan.GST(K=4, SF=1, phi=0, alpha=0.025, delta=6, pow=0.8,
               compute.alab=TRUE, compute.als=TRUE)

GST <- as.GST(GSD=GSD, GSDo=list(T=2, z=3.1))
seqconfint(GST, type="r")

## The confidence bound based on the stage-wise ordering of a group sequential trial is calculated by

seqconfint(GST, type="so")

## The repeated confidence interval at the earlier stage T=1 where the
## trial stopping rule is not met.

seqconfint(as.GST(GSD, GSDo=list(T=1, z=0.7)), type="r")

## The repeated confidence bound and the confidence bound
## based on the stage-wise ordering of a group sequential trial
## after a design adaptation is calculated by

pT <- plan.GST(K=3, SF=4, phi=-4, alpha=0.05, delta=6, pow=0.9,
               compute.alab=TRUE, compute.als=TRUE)

iD <- list(T=1, z=1.090728)

swImax <- 0.0625
I2min <- 3*swImax
I2max <- 3*swImax

sT <- adapt(pT=pT, iD=iD, SF=1, phi=0, cp=0.8, theta=5, I2min, I2max, swImax)

sTo <- list(T=2, z=2.393)

AGST <- as.AGST(pT=pT, iD=iD, sT=sT, sTo=sTo)
seqconfint(AGST)

## The repeated confidence interval at the earlier stage T=2 where the
## trial stopping rule is not met.

seqconfint(as.AGST(pT, iD, sT, sTo=list(T=2, z=1.7)), type="r")

## Not run:
## If the stage-wise adjusted confidence interval is calculated at this stage,
## the function returns an error message
Description

`typeIerr` is a function that computes the type I error rate of a GSD.

Usage

```r
typeIerr(GSD)
```

Arguments

- `GSD`: object of the class `GSTobj` or list with the following elements: `K` = number of stages, `a` = vector with futility boundaries (not supported yet), `b` = rejection boundaries, `t` = vector with information fractions; see example below.

Value

`typeIerr` returns the type I error rate of a GSD.

Author(s)

Niklas Hack <niklas.hack@meduniwien.ac.at> and Werner Brannath <werner.brannath@meduniwien.ac.at>

References


See Also

`GSTobj`

Examples

```r
##the following calculates the type I error rate of a GSD
GSD <- list(K=4,a=rep(-8,4),b=c(4.333,2.963,2.359,2.014),
             t=c(0.25,0.5,0.75,1),Imax=0.22)

typeIerr(GSD)
```
# Index

*Topic **datasets**
- AGSDest, 4
- AGSTobj, 5
- GSTobj, 13

*Topic **list**
- AGSDest, 4

*Topic **methods**
- adapt, 2
- AGSDest, 4
- as.AGST, 10
- as.GST, 11
- cer, 11
- cp, 12
- plan.GST, 17
- pvalue, 18
- seqconfint, 21
- typeIerr, 24

adapt, 2
AGSDest, 4
AGSDest-package (AGSDest), 4
AGSTobj, 5, 9, 10, 16, 18, 20, 23
as.AGST, 10
as.GST, 11

cer, 11

cp, 12

GSTobj, 3, 11, 13, 16, 18, 20, 23, 24

plan.GST, 3, 12, 17
plot.AGSTobj, 9
plot.AGSTobj (AGSTobj), 5
plot.GSTobj, 3, 16, 18
plot.GSTobj (GSTobj), 13
print.AGSTobj, 9
print.AGSTobj (AGSTobj), 5
print.GSTobj, 3, 16, 18
print.GSTobj (GSTobj), 13
print.summary.AGSTobj (AGSTobj), 5
print.summary.GSTobj (GSTobj), 13
pvalue, 18

seqconfint, 21
summary.AGSTobj, 9
summary.AGSTobj (AGSTobj), 5
summary.GSTobj, 16
summary.GSTobj (GSTobj), 13

typeIerr, 24