Package 'EurosarcBayes'

October 12, 2022

Type Package

Title Bayesian Single Arm Sample Size Calculation Software

Version 1.1

Date 2017-11-15

Author Peter Dutton

Maintainer Peter Dutton <dutton.peter@gmail.com>

Description Bayesian sample size calculation software and examples for EuroSARC clinical trials which utilise Bayesian methodology. These trials rely on binomial based endpoints so the majority of programs found here relate to this sort of endpoint. Developed as part of the EuroSARC FP7 grant.

License GPL-2

Depends shiny, VGAM, data.table, plyr, methods, clinfun

RoxygenNote 6.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2017-11-15 18:09:23 UTC

R topics documented:

`EurosarcBayes-package` *Bayesian sample size calculation software*

Description

Bayesian sample size calculation software and examples for Eurosarc clinical trials which utilise Bayesian methodology for binary endpoints (response/no-response). These trials rely on binomial based endpoints so the majority of programs found here relate to this sort of endpoint. Interim analyses are permitted for most designs. Developed as part of the EuroSARC FP7 grant.

Details

Package:	EurosarcBayes
Type:	Package
Version:	1.0
Date:	2015-11-18
License:	None

This package contains functions and some corresponding shiny versions of them for a user interface approach to sample size calculation and some examples.

There are both frequentist and Bayesian sample size optimisation programs contained here. Both versions are capable of computing frequentist and Bayesian properties of the given approach. This should allow for easy comparison between approaches.

List of user friendly shiny apps:

`shiny_binom_single_onestage`
`shiny_binom_single_twostage`
`shiny_LINES_posterior`

function naming convention:

Functions are named in the following way:

- freq_ or bayes_ denoting a frequentist or Bayesian designs.
- binom_ indicating a binomial endpoint.
- one_ or two_ indicating one or two endpoints.
- methodname_ indicating the approach used.
- onestage, twostage or nstage. Program for the number of stages. If the program is designed for any number of stages this has been ommited.

For example freq_binom_one_simons_twostage is a function for designing a frequentist single endpoint binomial trial using Simons two stage design.

One endpoint designs:

`freq_binom_one_onestage`: Finds the smallest sample size for a frequentist trial given the design parameters.

`freq_binom_one_simons_twostage`: Returns Simon's two stage designs with frequentist and bayesian properties of the designs. Options to return both the optimal and minmax designs.

`freq_binom_one_landemets`: Returns designs based on Lan-DeMets alpha spending approach using the O'Brien-Fleming alpha spending function (Lan and DeMets 1995, O'Brien and Fleming 1979).

`bayes_binom_one_postprob_onestage`: Finds the smallest sample size of a Bayesian trial given the design parameters.

`bayes_binom_one_postprob_nstage`: Computes frequentist and Bayesian properties for a trial with given sample sizes at each interim analysis. The posterior probability is used to determine the stopping critical values at interim.

`bayes_binom_one_postlike_nstage`: Computes frequentist and Bayesian properties for a trial with given sample sizes at each interim analysis. The posterior predictive probabilities are used to determine the stopping critical values at interim.

Two endpoint designs:

The two endpoint designs assume that two endpoints are independent.

`freq_binom_two_singlestage`: Finds the smallest sample size for a frequentist trial with two binary endpoints given the design parameters. Exact errors are computed so there is no issue of multiplicity.

`freq_binom_two_bryantday_twostage`: Returns Bryant and Day's two-stage designs with frequentist and bayesian properties of the designs. Options to return both the optimal and minmax designs (Bryant and Day 1995).

`bayes_binom_two_postprob`: Computes frequentist and Bayesian properties for a trial with two binary endpoints and given sample sizes at each interim analysis. Posterior probabilities are used to determine the stopping critical values at interim.

`bayes_binom_two_postlike`: Computes frequentist and Bayesian properties for a trial with two binary endpoints and given sample sizes at each interim analysis. Posterior predictive probabilities are used to determine the stopping critical values at interim.

`bayes_binom_two_loss`: Computes frequentist and Bayesian properties for a trial with two binary endpoints and given sample sizes at each interim analysis. A Bayesian loss function approach is used to determine the stopping critical values at all analyses (Chen 2009).

Author(s)

Peter Dutton

Maintainer: Peter Dutton <dutton.peter@gmail.com>

References

Simon R. Optimal two-stage designs for phase II clinical trials. Control Clin Trials 1989; 10: 1-10.

Bryant J, Day R. Incorporating toxicity considerations into the design of two-stage phase II clinical trials. Biometrics 1995; 51: 1372-1383.

DeMets, D. L. and G. Lan (1995). The alpha spending function approach to interim data analyses. Cancer Treat Res 75: 1-27.

O'Brien, P. C. and T. R. Fleming (1979). A Multiple Testing Procedure for Clinical Trials. Biometrics 35(3): 549-556.

Chen Y, Smith BJ. Adaptive group sequential design for phase II clinical trials: a Bayesian decision theoretic approach. Stat Med 2009; 28: 3347-3362.

`bayes_binom_one_postlike_nstage`

Single arm, any stage, cut-point calculator us posterior predictive distribution of a successful trial occurring to make the cuts.

Description

Generate cut-points given interim analyses at set numbers of patients for Bayesian posterior likelihood approach to stopping early for futility or efficacy

Usage

```
bayes_binom_one_postlike_nstage(reviews, prob.success, prob.failure,
 eta, zeta, p0, p1, prior.a=1e-6, prior.b=1e-6, round=TRUE, warn=TRUE)
```

Arguments

`reviews`	Vector of sample sizes to perform analysis at
`p0`	Probability of success under the null hypothesis
`p1`	Probability of success under the alternative hypothesis
`eta`	The smallest probability that p is less than p1 which is allowed to stop for futility
`zeta`	The smallest probability that p is greater than p0 which is allowed to stop for efficacy
`prob.success,prob.failure`	The probability of success and failure required to stop early at interim analysis
`prior.a, prior.b`	The prior parameters for the beta prior distribution
`round`	Optionally round the probability outputs to 3 significant figures
`warn`	Turn off warnings for designs which are not optimal

Value

Returns an object of class trialDesign_binom_one

See Also

bayes_binom_one_postprob_onestage

Examples

```
reviews=c(7,18)
prob.success=c(0.9)
prob.failure=c(0.9)
eta=0.9
zeta=0.9
p0=0.1
p1=0.3
prior.a=1e-6
prior.b=1e-6
bayes_binom_one_postlike_nstage(reviews,prob.success,prob.failure,
eta,zeta,p0,p1,prior.a,prior.b)
```

---

bayes_binom_one_postprob_nstage

Single arm, any stage, cut-point calculator using posterior probabilities to make the cuts.

Description

Generate cut-points given interim analyses at set numbers of patients for Bayesian posterior probability approach to stopping early for futility or efficacy

Usage

```
bayes_binom_one_postprob_nstage(reviews, eta, zeta, p0, p1,
  prior.a=0, prior.b=0, h0=p0, h1=p1, round=TRUE, warn=TRUE)
```

Arguments

reviews	Vector of sample sizes to perform analysis at
p0	Probability of success under the null hypothesis
p1	Probability of success under the alternative hypothesis
eta	The smallest probability that p is less than p1 which is allowed to stop for futility
zeta	The smallest probability that p is greater than p0 which is allowed to stop for efficacy
h0,h1	Optional values to be used if the properties of the design should be based on hypotheses which do not use the last values of p0 and p1.

prior.a,prior.b	The prior parameters for the beta prior distribution
round	Optionally round the probability outputs to 3 significant figures
warn	Turn off warnings for designs which are not optimal

Value

Returns an object of class trialDesign_binom_one

See Also

bayes_binom_one_postprob_onestage

Examples

```
reviews=c(7,18)
eta=c(0.9,0.9)
zeta=c(0.9,0.9)
p0=0.1
p1=0.3
prior.a=0
prior.b=0
bayes_binom_one_postprob_nstage(reviews,eta,zeta,p0,p1,prior.a,
prior.b)
```

---

bayes_binom_one_postprob_onestage

Bayesian single-arm single-endpoint minimum sample size

Description

Generate minimum sample size for the Bayesian single-endpoint single-arm trial. Also provided a shiny app to evaluate the same thing with both frequentist and Bayesian methods side by side.

Usage

```
bayes_binom_one_postprob_onestage(p0, p1, eta, zeta, prior.a,
prior.b, round=TRUE)

shiny_binom_single_onestage()
```

Arguments

p0	Probability of success under the null hypothesis
p1	Probability of success under the alternative hypothesis
eta	The smallest probability that p is less than p1 which is allowed to stop for futility

zeta	The smallest probability that p is greater than p0 which is allowed to stop for efficacy
prior.a,prior.b	The prior parameters for the beta prior distribution
round	Optionally round the probability outputs to 3 significant figures

Value

Returns an object of class trialDesign_binom_one

See Also

bayes_binom_one_postprob_nstage

Examples

```
p0=0.1
p1=0.3
eta=c(0.9)
zeta=c(0.9)
prior.a=0
prior.b=0
bayes_binom_one_postprob_onestage(p0,p1,eta,zeta,prior.a,prior.b)
```

---

bayes_binom_two_loss *Bayesian, single arm, two endpoint trial designs, using loss functions to make decisions*

Description

Computes the decision rules for a single arm, two endpoint bayesian trial using a region of acceptable designs and loss functions to make decisions. This program assumes that the two endpoints are independent. A number of region spaces are provided. This function has the option of providing pre-existing decision matrices to skip this section if you wish to run additional simulations on an already computed design.

Usage

```
bayes_binom_two_loss(t, r, reviews, pra, prb, pta, ptb,
 l_alpha_beta, l_alpha_c, stage_after_trial, fun.integrate,
 efficacy_critical_value, toxicity_critical_value,
 futility_critical_value, no_toxicity_critical_value, decision=NULL,
 W=NULL, fun.graph=NULL, ...)
```

Arguments

t,r	A vector of the probability of response and toxicity for the simulation scenarios used to compute frequentist properties. The print function requires the first to be the alternative hypothesis and subsequent entries to be the three null hypotheses. This can be run with any scenario when not using the print method
reviews	A vector of the number of patients each interim and final analysis will occur at
pra,prb,pta,ptb	Numeric values for the beta prior distribution to be used
l_alpha_beta,l_alpha_c	The two loss function variables weighting between stopping early for futility or efficacy and continuing the trial
fun.integrate	function used to integrate the probability of being in the region of interest given the posterior distributions of the data and prior information
stage_after_trial	Optional argument for censored stages after the trial has completed. This is likely to create a region of inclusiveness upon concluding the trial
futility_critical_value, efficacy_critical_value, toxicity_critical_value, no_toxicity_critical_value	Four values, for the critical values to be used as thresholds for the posterior distribution
decision	Optional input the decision matrices from a previous run to perform additional frequentist simulations on the design.
W	Optional input the posterior probabilities from a previous run to perform additional frequentist simulations on the design.
fun.graph	Optional function printing a graph of the region of interest. No region is plotted if this is left blank
...	Options passed to the integration function

Details

Returns an object of S4 class trialDesign_binom_two-class. This has plot and print methods. For comparison between designs saved as trialDesign_binom_two objects there is a print function for the S3 class list_trialDesign_binom_two.

The following region spaces are included in the package: tradeoff_square_integrate tradeoff_square_graph tradeoff_ratio_intercepts tradeoff_linear_graph tradeoff_ratio_integrate tradeoff_ratio_graph tradeoff_ellipse_integrate tradeoff_ellipse_graph

Value

Returns an object of class trialDesign_binom_two

References

Chen Y, Smith BJ. Adaptive group sequential design for phase II clinical trials: a Bayesian decision theoretic approach. Stat Med 2009; 28: 3347-3362.

See Also

bayes_binom_two_postprob, bayes_binom_two_postlike

Integration functions and corresponding graphs: tradeoff_square_integrate,tradeoff_ellipse_integrate,tradeoff_

Examples

```
# modelled toxicity probability
t=c(0.1,0.1,0.3,0.3)
# modelled response probability
r=c(0.35,0.2,0.2,0.35)

reviews=c(10,15,20,25,30,35,40)
stage_after_trial=40

# uniform prior
pra=1;prb=1;pta=1;ptb=1

efficacy_critical_value=0.2
futility_critical_value=0.35
toxicity_critical_value=0.1
no_toxicity_critical_value=0.3

# alpha/beta ratio
l_alpha_beta=3
# cost of continuing compared to cost of alpha
l_alpha_c=750

efficacy_region_min=0.2
toxicity_region_max=0.3

########################################
# square region
s=bayes_binom_two_loss(t,r,reviews,pra,prb,pta,ptb,l_alpha_beta,
l_alpha_c,stage_after_trial,fun.integrate=tradeoff_square_integrate,
fun.graph=tradeoff_square_graph,efficacy_critical_value,
toxicity_critical_value,futility_critical_value,
no_toxicity_critical_value,efficacy_region_min=efficacy_region_min,
toxicity_region_max=toxicity_region_max)

plot(s)


########################################
# ellipse region
efficacy_region_min=0.2
efficacy_region_max=0.35
toxicity_region_min=0.1
toxicity_region_max=0.3


s=bayes_binom_two_loss(t,r,reviews,pra,prb,pta,ptb,l_alpha_beta,
l_alpha_c,stage_after_trial,fun.integrate=tradeoff_ellipse_integrate,
```

```
fun.graph=tradeoff_ellipse_graph,efficacy_critical_value,
toxicity_critical_value,futility_critical_value,
no_toxicity_critical_value,efficacy_region_min=efficacy_region_min,
toxicity_region_max=toxicity_region_max,
efficacy_region_max=efficacy_region_max,
toxicity_region_min=toxicity_region_min)


plot(s)
```

`bayes_binom_two_postlike` *Bayesian, single arm, two endpoint trial designs.*

Description

Computes the decision rules for a single arm, two endpoint bayesian trial using the likelihood of success to make decisions. This program assumes that the two endpoints are independent.

Usage

```
bayes_binom_two_postlike(t, r, reviews, pra, prb, pta, ptb,
 efficacy_critical_value, efficacy_prob_stop, toxicity_critical_value,
 toxicity_prob_stop, int_combined_prob, int_futility_prob,
 int_toxicity_prob, int_efficacy_prob, futility_critical_value,
 no_toxicity_critical_value)
```

Arguments

`t,r`	A vector of the probability of response and toxicity for the simulation scenarios used to compute frequentist properties. The print function requires the first to be the alternative hypothesis and subsequent entries to be the three null hypotheses. This can be run with any scenario when not using the print method
`reviews`	A vector of the number of patients each interim and final analysis will occur at
`pra,prb,pta,ptb`	Numeric values for the beta prior distribution to be used
`futility_critical_value, efficacy_critical_value, toxicity_critical_value, no_toxicity_critical_value`	Four values, for the critical values to be used as thresholds for the posterior distribution
`int_combined_prob, int_futility_prob, int_toxicity_prob, int_efficacy_prob`	Probabilities to stop at interim analyses
`efficacy_prob_stop, toxicity_prob_stop`	Values or vectors of the probability required to stop at this interim analysis. If you do not wish to stop due to a rule set this to 1 at that analysis. If you wish to ignor a rule when stopping set this to 0 at that analysis

Details

Returns an object of S4 class trialDesign_binom_two-class. This has plot and print methods. For comparison between designs saved as trialDesign_binom_two objects there is a print function for the S3 class list_trialDesign_binom_two.

Value

Returns an object of class trialDesign_binom_two

See Also

bayes_binom_two_postprob, bayes_binom_two_postlike,bayes_binom_two_loss

Examples

```
# modelled toxicity probability
t=c(0.1,0.1,0.3,0.3)
# modelled response probability
r=c(0.35,0.2,0.2,0.35)

reviews=c(10,15,20,25,30,35,40)

# uniform prior
pra=1;prb=1;pta=1;ptb=1

# End of trial stopping rules for success
efficacy_critical_value=0.2
efficacy_prob_stop=0.9
toxicity_critical_value=0.2
toxicity_prob_stop=0.8

# interim required probability to stop
int_combined_prob=0.99
int_futility_prob=1
int_toxicity_prob=1
int_efficacy_prob=0.99

# unused in the design for comparison to previous design
futility_critical_value=0.35
no_toxicity_critical_value=0.3

s=bayes_binom_two_postlike(t,r,reviews,pra,prb,pta,ptb,
efficacy_critical_value,efficacy_prob_stop,toxicity_critical_value,
toxicity_prob_stop,int_combined_prob,int_futility_prob,
int_toxicity_prob,int_efficacy_prob,futility_critical_value,
no_toxicity_critical_value)

s

plot(s)
```

bayes_binom_two_postprob *Bayesian, single arm, two endpoint trial design, using posterior probability to make decisions.*

Description

Computes the decision rules for a single arm, two endpoint bayesian trial using posterior probabilities to generate the decision rules. This program assumes that the two endpoints are independent.

Usage

```
bayes_binom_two_postprob(t, r, reviews, pra, prb, pta, ptb,
futility_critical_value, futility_prob_stop, efficacy_critical_value,
efficacy_prob_stop, toxicity_critical_value, toxicity_prob_stop,
no_toxicity_critical_value, no_toxicity_prob_stop)
```

Arguments

t,r	A vector of the probability of response and toxicity for the simulation scenarios used to compute frequentist properties. The print function requires the first to be the alternative hypothesis and subsequent entries to be the three null hypotheses. This can be run with any scenario when not using the print method
reviews	A vector of the number of patients each interim and final analysis will occur at
pra,prb,pta,ptb	Numeric values for the beta prior distribution to be used
futility_critical_value, efficacy_critical_value, toxicity_critical_value, no_toxicity_critical_value	Four values, for the critical values to be used as thresholds for the posterior distribution
futility_prob_stop, efficacy_prob_stop, toxicity_prob_stop, no_toxicity_prob_stop	Values or vectors of the probability required to stop at this interim analysis. If you do not wish to stop due to a rule set this to 1 at that analysis. If you wish to ignor a rule when stopping set this to 0 at that analysis

Details

Returns an object of S4 class trialDesign_binom_two-class. This has plot and print methods. For comparison between designs saved as trialDesign_binom_two objects there is a print function for the S3 class list_trialDesign_binom_two.

Value

Returns an object of class trialDesign_binom_two

See Also

bayes_binom_two_postprob, bayes_binom_two_postlike,bayes_binom_two_loss

Examples

```
# modelled toxicity probability
t=c(0.1,0.1,0.3,0.3)
# modelled response probability
r=c(0.35,0.2,0.2,0.35)

reviews=c(10,15,20,25,30,35,40)

# uniform prior
pra=1;prb=1;pta=1;ptb=1

futility_critical_value=0.35
futility_prob_stop=c(0.95,0.95,0.95,0.95,0.95,0.95,0)

efficacy_critical_value=0.2
efficacy_prob_stop=c(1,1,0.95,0.95,0.95,0.95,0.9)

toxicity_critical_value=0.1
toxicity_prob_stop=c(0.95,0.95,0.95,0.95,0.95,0.95,0.95)

no_toxicity_critical_value=0.3
toxicity_prob_stop=c(0.95,0.95,0.95,0.95,0.95,0.95,0.95)

s=bayes_binom_two_postprob(t,r,reviews,pra,prb,pta,ptb,
futility_critical_value,futility_prob_stop,efficacy_critical_value,
efficacy_prob_stop,toxicity_critical_value,toxicity_prob_stop,
no_toxicity_critical_value,toxicity_prob_stop)

s

plot(s)
```

`binom_one_alpha` *Single arm, exact p-value calculator for single or multi-stage binomial trials.*

Description

P-value (alpha) for single arm binomial clinical trials. This is done exactly accounting for all interim analysis prior to stopping the trial.

Usage

```
binom_one_alpha(result.success, result.n, p0, failure, success, n)
```

Arguments

`result.success`	total successes at the end of the trial
`result.n`	total patients at the end of the trial

p0	Probability of success under H0
failure	A vector of the number of failures required to stop for futility, if not able to stop NA or a character string should be provided
success	A vector of the number of successes required to stop for efficacy, if not able to stop NA or a character string should be provided
n	A vector of the total number of patients to recruit up to each stage of the trial

See Also

binom_one_power, binom_one_assurance

Examples

```
# Simon's two stage design
failure=c(0,3)
success=c(NA,4)
n=c(7,18)
p0=0.1

result.success=4
result.n=18

# without accounting for interim analysis when calculating
# the p-value
1-pbinom(result.success-1,result.n,p0)
# account for interim analysis
binom_one_alpha(result.success,result.n,p0,failure,success,n)
```

---

binom_one_assurance *Single arm, assurance calculator for single or multi-stage binomial trials.*

Description

Computes the assurance of a given trial design given a prior assurance distribution.

Usage

```
binom_one_assurance(failure, success, n, ass.dist,
 type="continuous", lower=0, upper=1, ...)

plot_binomassurance(failure, success, n, ass.dist,type="continuous",
ndivisions=1000, xlim=c(0,1), xaxs="i", yaxs="i", ylim=NULL,
main="Assurance distribution", col="red", col.fill="green", lwd=2,
xlab="Probability of successful treatment",
ylab="Prior assurance probability" ,...)
```

Arguments

failure	A vector of the number of failures required to stop for futility, if not able to stop NA or a character string should be provided
success	A vector of the number of successes required to stop for efficacy, if not able to stop NA or a character string should be provided
n	A vector of the total number of patients to recruit up to each stage of the trial
ass.dist	Distribution of prior probability for assurance. May be different to prior information.
type	Tells the program you are passing it a continuous distribution ("continuous") or a discrete distribution ("discrete") for the assurance distribution
ndivisions	The number of points calculated for the plot
lower, upper	Range of the distribution to use
col.fill	Colour of the true positive results in the graph
xlim, xaxs, yaxs, ylim, main, col, lwd, xlab, ylab	Different defaults for plotting parameters
...	Additional plotting parameters to pass to plot function

See Also

binom_one_power, binom_one_alpha

Examples

```
# Simon's two stage design
failure=c(0,3)
success=c(NA,4)
n=c(7,18)
p0=0.1
p1=0.3

# continuous assurance distribution
ass.dist = function(p) dbeta(p,4,18)

# assurance
binom_one_assurance(failure,success,n,ass.dist)

# plot
plot_binomassurance(failure,success,n,ass.dist,ndivisions=1000)

# discrete assurance distribution
ass.dist = matrix(c(0.2,0.3,0.4,0.3,0.4,0.3),ncol=2)

# assurance
binom_one_assurance(failure,success,n,ass.dist,type="discrete")

# plot
plot_binomassurance(failure,success,n,ass.dist,type="discrete",
                    ndivisions=1000)
```

binom_one_power *Single arm, power calculator for single or multi-stage binomial trials.*

Description

Computes the power of a given trial design given the probability of success of treatment p.

Usage

```
binom_one_power(p,failure,success,n)

plot_binom_one_power(failure, success, n, ndivisions=1000,
xlim=c(0,1), xaxs="i", yaxs="i", ylim=c(0,1.1),
main="Power curve for a single arm binomial trial design",
xlab="Probability of successful treatment",
ylab="Probability of successful trial",
p=NULL, alpha=NULL, power=NULL, col.error="red", ...)
```

Arguments

p	Probability of success to compute power for
failure	A vector of the number of failures required to stop for futility, if not able to stop NA or a character string should be provided
success	A vector of the number of successes required to stop for efficacy, if not able to stop NA or a character string should be provided
n	A vector of the total number of patients to recruit up to each stage of the trial
ndivisions	The number of points calculated for the plot
col.error	Colour of type II errors in the plot
alpha, power	Plotted as lines if provided
xlim, ylim, xaxs, yaxs, main, xlab, ylab	Different defaults for plotting parameters
...	Additional plotting parameters to pass to plot function

See Also

binom_one_alpha, binom_one_assurance

Examples

```
# Simon's two stage design
failure=c(0,3)
success=c(NA,4)
n=c(7,18)
p0=0.1
p1=0.3
```

```
# power
binom_one_power(p1,failure,success,n)
# type 1 error (alpha)
binom_one_power(p0,failure,success,n)

# plot
plot_binom_one_power(failure,success,n,ndivisions=1000,p=c(p0,p1),
 alpha=0.1,power=0.8)
```

`binom_two_bryantday-class` *Class* `"binom_two_bryantday"`

Description

This is the S4 class for constructing Bryant and Day designs. This can be converted to the standard format using `properties`.

Objects from the Class

Objects can be created by calls of the form `new("binom_two_bryantday", ...)`.

Slots

`optimal`**:** Object of class `"data.frame"`, single row data.frame containing the optimal design under H0

`minmax`**:** Object of class `"data.frame"`, Single row data.frame containing the minmax design under H0

`all.fit`**:** Object of class `"data.frame"`, A data.frame containing all designs which satisfy the required alpha and power specified for the trial

Methods

properties `signature(x = "binom_two_bryantday")`(x, t, r, pra, prb, pta, ptb, futility_critical_value = 0.2, efficacy_critical_value = 0.35, toxicity_critical_value = 0.3, no_toxicity_critical_value = 0.1)

x Class object which you wish to get properties for

t,r A vector of the probability of response and toxicity for the simulation scenarios used to compute frequentist properties. The print function requires the first to be the alternative hypothesis and subsequent entries to be the three null hypotheses. This can be run with any scenario when not using the print method

reviews A vector of the number of patients each interim and final analysis will occur at

pra, prb, pta, ptb Numeric values for the beta prior distribution to be used

futility_critical_value, efficacy_critical_value, toxicity_critical_value, no_toxicity_critical_value Four values, for the critical values to be used as thresholds for the posterior distribution

References

Bryant J, Day R. Incorporating toxicity considerations into the design of two-stage phase II clinical trials. Biometrics 1995; 51: 1372-1383.

Examples

```
showClass("binom_two_bryantday")
```

---

binom_two_singlestage-class

Class "binom_two_singlestage"

Description

This class is created from the function freq_binom_two_singlestage. This is an intermediate stage to generate an object of class trialDesign_binom_two.

Objects from the Class

Objects can be created by calls of the form new("binom_two_singlestage", ...).

Slots

optimal: Object of class "data.frame", Optimal trial design

output: Object of class "data.frame", list of all acceptable trial designs

Methods

properties signature(x = "binom_two_singlestage")(x, t, r, pra, prb, pta, ptb, futility_critical_value = 0.2, efficacy_critical_value = 0.35, toxicity_critical_value = 0.3, no_toxicity_critical_value = 0.1)

x Class object which you wish to get properties for

t,r A vector of the probability of response and toxicity for the simulation scenarios used to compute frequentist properties. The print function requires the first to be the alternative hypothesis and subsequent entries to be the three null hypotheses. This can be run with any scenario when not using the print method

reviews A vector of the number of patients each interim and final analysis will occur at

pra, prb, pta, ptb Numeric values for the beta prior distribution to be used

futility_critical_value, efficacy_critical_value, toxicity_critical_value, no_toxicity_critical_value Four values, for the critical values to be used as thresholds for the posterior distribution

Returns an object of class trialDesign_binom_two.

Examples

```
showClass("binom_two_singlestage")
```

`freq_binom_one_landemets` *Single arm, two stage, Binomial sample size calculator*

Description

Sample size calculation for single arm, multistage trials using the alpha spending approach to reduce type I and type II error rates. This implementation uses the O'Brien-Fleming alpha spending function for this purpose.

Usage

```
freq_binom_one_landemets(reviews, p0, p1, r=c(p0,p1),
alpha=0.1, beta=0.1, prior.a=0, prior.b=0)
```

Arguments

`reviews`	A vector of the number of patients to perform interim analysis at
`p0`	Probability of success under the H0
`p1`	Probability of success under the H1
`r`	A vector of probabilities used to perform simulations from
`alpha`	The largest allowed value for the frequentist type one error
`beta`	The smallest allowed value for the frequentist type two error
`prior.a, prior.b`	Prior parameters for the beta prior

Value

Returns an object of class `trialDesign_binom_one`

References

DeMets, D. L. and G. Lan (1995). The alpha spending function approach to interim data analyses. Cancer Treat Res 75: 1-27.

O'Brien, P. C. and T. R. Fleming (1979). A Multiple Testing Procedure for Clinical Trials. Biometrics 35(3): 549-556.

Examples

```
reviews=c(11,22,33,44)
p0=0.2
p1=0.35
r=c(0.2,0.35)
alpha=0.1
beta=0.2
freq_binom_one_landemets(reviews,p0,p1,r,alpha,beta)
```

freq_binom_one_onestage *Bayesian single-arm single-endpoint minimum sample size*

Description

Generate minimum sample size for the frequentist single-endpoint single-arm trial. Also provided a shiny app to evaluate the same thing with both frequentist and Bayesian methods side by side.

Usage

```
freq_binom_one_onestage(p0, p1, alpha, power, prior.a=0, prior.b=0,
 round=TRUE)

shiny_binom_single_onestage()
```

Arguments

p0	Probability of success under H0
p1	Probability of success under H1
alpha	The largest allowed value for the frequentist type one error
power	The smallest allowed frequentist power
prior.a,prior.b	The prior parameters for the beta prior distribution
round	Optionally round the probability outputs to 3 significant figures

Value

Returns an object of class trialDesign_binom_one

Examples

```
p0=0.1
p1=0.3
alpha=0.1
power=0.8
prior.a=0
prior.b=0
freq_binom_one_onestage(p0,p1,alpha,power,prior.a,prior.b)
```

`freq_binom_one_simons_twostage` *Single arm, two stage, Binomial sample size calculator*

Description

Sample size calculation for single arm, two stage designs (Simon's optimal and minmax designs) where stoping early for futility is permitted. Returns frequentist and Bayesian properties for the designs.

A shiny app is also provided. This is interactive for Simon's two stage design and also describes a number of multistage designs for the same problem.

Usage

```
freq_binom_one_simons_twostage(p0, p1, alpha, power, prior.a=0,
prior.b=0, nmax=100, round=TRUE, method="optimal")

shiny_binom_single_twostage()
```

Arguments

`p0`	Probability of success under H0
`p1`	Probability of success under H1
`alpha`	The largest allowed value for the frequentist type one error
`power`	The smallest allowed frequentist power
`prior.a,prior.b`	The prior parameters for the beta prior distribution
`nmax`	The maximum sample size to search up to
`round`	Optionally round the probability outputs to 3 significant figures
`method`	Defining the method of optimisation. Either "optimal" or "minmax"

Details

freq_binom_one_simons_twostage is a wrapper function. It uses `ph2simon` from the clinfun package to generate optimal sample sizes for the the frequentist single arm, two stage designs. Frequentist and Bayesian properties are then calculated using properties_binom_one and then optimal and minimax designs are returned.

Value

Returns an object of class `trialDesign_binom_one`

References

Simon R. (1989). Optimal Two-Stage Designs for Phase II Clinical Trials. Controlled Clinical Trials 10, 1-10.

See Also

ph2simon

Examples

```
p0=0.2
p1=0.35
alpha=0.1
power=0.8
freq_binom_one_simons_twostage(p0,p1,alpha,power)
```

`freq_binom_two_bryantday_twostage`

Single arm, two independent endpoint extension to Simons two-stage design

---

Description

This function searches for solutions to a single arm two-stage two-endpoint trial first proposed by Bryant and Day (1995). The two endpoints are assumed independent. A wrapper function to compute the Bayesian properties is also provided.

Usage

```
freq_binom_two_bryantday_twostage(r0=0.2, r1=0.35, t0=0.3, t1=0.1,
  alpha.r, power, nrange, alpha.t=alpha.r)
```

Arguments

`r0,r1`	Probability of success under H0 and H1
`t0,t1`	Probability of toxicity under H0 and H1
`alpha.r`	Probability of a false positive trial if the response H0 is true and toxicity is either H0 or H1
`alpha.t`	Probability of a false positive trial if the toxicity H0 is true and response is either H0 or H1
`power`	Probability of true positive trial result assuming H1 is true
`nrange`	A vector of the total number of patients to recruit up to each stage of the trial

Value

Returns an object of class `binom_two_bryantday`. This can be transformed into an object of class `trialDesign_binom_two` using properties (see `properties`) and supplying the necessary values.

References

Simon R. Optimal two-stage designs for phase II clinical trials. Control Clin Trials 1989; 10: 1-10.

Bryant J, Day R. Incorporating toxicity considerations into the design of two-stage phase II clinical trials. Biometrics 1995; 51: 1372-1383.

Examples

```
r1=0.3
r0=0.1
t0=0.3
t1=0.1

power=0.8
alpha=0.1

#################################################################
# Bryant and Day, two stage
nrange=20:27
out=freq_binom_two_bryantday_twostage(r0,r1,t0,t1,alpha,power,nrange)

#################################################################
## Frequentist simulations
# modelled toxicity probability
t=c(0.1,0.3,0.1,0.3)
# modelled response probability
r=c(0.3,0.1,0.1,0.3)

## Bayesian uniform prior
pra=1;prb=1;pta=1;ptb=1
## bayesian cutoffs
futility_critical_value=0.3
efficacy_critical_value=0.1
toxicity_critical_value=0.1
no_toxicity_critical_value=0.3

byrant_day_optimal=properties(out,t,r,pra,prb,pta,ptb,
                              futility_critical_value,efficacy_critical_value,
                              toxicity_critical_value,no_toxicity_critical_value,
                              target="optimal")

byrant_day_minmax=properties(out,t,r,pra,prb,pta,ptb,
                             futility_critical_value,efficacy_critical_value,
                             toxicity_critical_value,no_toxicity_critical_value,
                             target="minmax")

bayes_table=list(byrant_day_optimal=byrant_day_optimal,
                 byrant_day_minmax=byrant_day_minmax)
class(bayes_table)=c("list_trialDesign_binom_two",class(bayes_table))

bayes_table
```

`freq_binom_two_singlestage` *Frequentist single-arm two-endpoint trial sample size*

Description

Generate minimum sample size for the frequentist two-endpoint single-arm trial.

Usage

```
freq_binom_two_singlestage(r0, r1, t0, t1, power, alpha.r, nmax = 100,
alpha.t = alpha.r, nmin = 1, adjust = TRUE)
```

Arguments

`r0`	Null hypothesis for the response endpoint
`r1`	Alternative hypothesis for the response endpoint
`t0`	Null hypothesis for the toxicity endpoint
`t1`	Alternative hypothesis for the toxicity endpoint
`power`	Required power for the trial design
`alpha.r`	The maximum size for the type one error for the response endpoint
`nmax`	Maximum sample size to look for solutions. Default 100
`alpha.t`	The maximum size for the type one error for the toxicity endpoint. Optional
`nmin`	Minimum sample size to look for solution, Default 1
`adjust`	Boolean about whether to adjust for multiple endpoints or not

Value

Returns an object of class binom_two_singlestage. This can be transformed into an object of class `trialDesign_binom_two` using properties (see `properties`) and supplying the necessary values.

See Also

`properties`

Examples

```
r1=0.35
r0=0.2
t0=0.3
t1=0.1

power=0.8
alpha=0.1
```

```
nmax=50
out_single=freq_binom_two_singlestage(r0,r1,t0,t1,power,alpha,nmax,adjust=TRUE)

#######################################
# Information for properties
## Frequentist simulations
# modelled toxicity probability
t=c(0.1,0.3,0.1,0.3)
# modelled response probability
r=c(0.35,0.2,0.2,0.35)

## Bayesian uniform prior
pra=1;prb=1;pta=1;ptb=1
## bayesian cutoffs
futility_critical_value=0.35
efficacy_critical_value=0.2
toxicity_critical_value=0.1
no_toxicity_critical_value=0.3

single_stage=properties(out_single,t,r,pra,prb,pta,ptb,futility_critical_value,
efficacy_critical_value,toxicity_critical_value,no_toxicity_critical_value)

single_stage
```

`print.list_trialDesign_binom_two` *Bayesian, single arm, two endpoint trial design, using posterior probability to make decisions.*

Description

This class is used to compare designs and methodologies frequentist and bayesian properties. To use it create a list of trial designs of class trialDesign_binom_two and assign the class as list_trialDesign_binom_two (`class(x)=c("list_trialDesign_binom_two",class(x))`).

Usage

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

Arguments

`x`	A list of the S4 class object bayes_binom_two_postprob
`...`	Standard arguments to pass to print

See Also

bayes_binom_two_postprob, bayes_binom_two_postlike, bayes_binom_two_loss, freq_binom_two_bryantday_twos

Examples

```
## Frequentist simulations
# modelled toxicity probability
t=c(0.1,0.3,0.1,0.3)
# modelled response probability
r=c(0.35,0.2,0.2,0.35)

## Bayesian uniform prior
pra=1;prb=1;pta=1;ptb=1
## bayesian cutoffs
futility_critical_value=0.35
efficacy_critical_value=0.2
toxicity_critical_value=0.1
no_toxicity_critical_value=0.3

#################################################################
# Frequentist methods
#################################################################
# Single stage

r1=0.35
r0=0.2
t0=0.3
t1=0.1

power=0.8
alpha=0.1

nmax=50
out_single=freq_binom_two_singlestage(r0,r1,t0,t1,power,alpha,nmax,
adjust=TRUE)

single_stage=properties(out_single,t,r,pra,prb,pta,ptb,
futility_critical_value,efficacy_critical_value,
toxicity_critical_value,no_toxicity_critical_value)

print(single_stage)

#################################################################
# Bayesian posterior probability approach


# analysis at
reviews=c(44)
# Stopping rules at each analysis
futility_prob_stop=0.9
efficacy_prob_stop=0.9
toxicity_prob_stop=0.9
no_toxicity_prob_stop=0.9

bayes_prob_single=bayes_binom_two_postprob(t,r,reviews,pra,prb,pta,
ptb,futility_critical_value,futility_prob_stop,
```

```
efficacy_critical_value,efficacy_prob_stop,
toxicity_critical_value,toxicity_prob_stop,
no_toxicity_critical_value,no_toxicity_prob_stop)

bayes_prob_single


# analysis at
reviews=c(10,17,24,30,37,44)
# Stopping rules at each analysis
futility_prob_stop=c(0.95,0.95,0.95,0.95,0.95,0.9)
efficacy_prob_stop=c(1,1,0.95,0.95,0.95,0.9)
toxicity_prob_stop=c(0.95,0.95,0.95,0.95,0.95,0.9)
no_toxicity_prob_stop=c(1,1,0.95,0.95,0.95,0.9)

bayes_prob_six=bayes_binom_two_postprob(t,r,reviews,pra,prb,pta,
ptb,futility_critical_value,futility_prob_stop,
efficacy_critical_value,efficacy_prob_stop,
toxicity_critical_value,toxicity_prob_stop,
no_toxicity_critical_value,no_toxicity_prob_stop)

plot(bayes_prob_six)


################################################################
# Bayesian posterior likelihood approach
################################################################
reviews=c(11,17,24,30,37,44)

efficacy_prob_stop=0.9
toxicity_prob_stop=0.9

# interim required probability to stop
int_combined_prob=0.95
int_futility_prob=1
int_toxicity_prob=1
int_efficacy_prob=0.95

bayes_like_six=bayes_binom_two_postlike(t,r,reviews,pra,prb,pta,
ptb,efficacy_critical_value,efficacy_prob_stop,
toxicity_critical_value,toxicity_prob_stop,int_combined_prob,
int_futility_prob,int_toxicity_prob,int_efficacy_prob,
futility_critical_value,no_toxicity_critical_value)

plot(bayes_like_six)

################################################################
## Table of all designs
################################################################
tble=list(single_stage=single_stage,bayes_prob_single=bayes_prob_single,
bayes_prob_six=bayes_prob_six,bayes_like_six=bayes_like_six)

class(tble)=c("list_trialDesign_binom_two",class(tble))
```

```
tble
################################################################
```

`properties-methods` *~~ Methods for Function* `properties` *in Package* **EurosarcBayes** *~~*

Description

~~ Methods for function `properties` in package **EurosarcBayes** ~~

Methods:

`signature(x = "ANY")`

`signature(x = "binom_two_bryantday")`

`signature(x = "binom_two_singlestage")`

`properties_binom_one` *Properties for single-arm single binomial endpoint trial design*

Description

Get frequentist and Bayesian properties for a single-arm single binomial endpoint trial design.

Usage

```
properties_binom_one(failure = NULL, success = NULL, reviews = NULL,
p0, p1, prior.a = 0, prior.b = 0, round = TRUE, cutpoints = NULL)
```

Arguments

`failure`	Vector of failures needed to stop the trial
`success`	Vector of successes needed to stop the trial
`reviews`	Vector of the number of patients at each analysis
`p0`	probability of success under H0
`p1`	probability of success under H1
`prior.a,prior.b`	beta prior parameters
`round`	Option whether to round results or not
`cutpoints`	Alternative usage, this replaces failure, success and reviews with a data.frame with columns of the same names

Value

Returns an object of class `trialDesign_binom_one`.

shiny_LINES_posterior *LINES prior-posterior distribution with posterior probabilities*

Description

This is a shiny app for the LINES trial. This trial is a dual endpoint design with both response and toxicity used to make informed decisions at interim anaylsis. This app provides an interactive way of updating the posterior distribution, as well as change the prior distributions.

Usage

```
shiny_LINES_posterior()
```

tradeoff ellipse_integrate *Functions for integration for Bayesian loss methodology*

Description

An integral and graph for an acceptable region for the bayesian loss function approach (see bayes_binom_two_loss)

Usage

```
tradeoff_ellipse_integrate(ar, br, at, bt, efficacy_region_min,
toxicity_region_max, efficacy_region_max, toxicity_region_min)

tradeoff_ellipse_graph(input)
```

Arguments

ar, br	Parameters for the posterior distribution for response
at, bt	Parameters for the posterior distribution for toxicity
efficacy_region_min	Smallest acceptable efficacy
toxicity_region_max	Largest acceptable toxicity
efficacy_region_max	Point where no more tradeoff occurs for efficacy
toxicity_region_min	Point where no more tradeoff occurs for toxicity
input	A list values needed for the graph. It is expecting max.patients, efficacy_region_min, toxicity_region_max and will error without

Value

Returns value of the integration.

References

Chen Y, Smith BJ. Adaptive group sequential design for phase II clinical trials: a Bayesian decision theoretic approach. Stat Med 2009; 28: 3347-3362.

See Also

bayes_binom_two_loss

Integration functions and corresponding graphs: tradeoff_square_integrate,tradeoff_ellipse_integrate,tradeoff_

Examples

```
# modelled toxicity probability
t=c(0.1,0.1,0.3,0.3)
# modelled response probability
r=c(0.35,0.2,0.2,0.35)

reviews=c(10,15,20,25,30,35,40)
stage_after_trial=40

# uniform prior
pra=1;prb=1;pta=1;ptb=1

efficacy_critical_value=0.2
futility_critical_value=0.35
toxicity_critical_value=0.1
no_toxicity_critical_value=0.3

# alpha/beta ratio
l_alpha_beta=3
# cost of continuing compared to cost of alpha
l_alpha_c=750

efficacy_region_min=0.2
toxicity_region_max=0.3


########################################
# ellipse region
efficacy_region_min=0.2
efficacy_region_max=0.35
toxicity_region_min=0.1
toxicity_region_max=0.3


s=bayes_binom_two_loss(t,r,reviews,pra,prb,pta,ptb,l_alpha_beta,
l_alpha_c,stage_after_trial,fun.integrate=tradeoff_ellipse_integrate,
fun.graph=tradeoff_ellipse_graph,efficacy_critical_value,
```

```
toxicity_critical_value,futility_critical_value,
no_toxicity_critical_value,efficacy_region_min=efficacy_region_min,
toxicity_region_max=toxicity_region_max,
efficacy_region_max=efficacy_region_max,
toxicity_region_min=toxicity_region_min)


plot(s)
```

`tradeoff linear` *Functions for integration for Bayesian loss methodology*

Description

An integral and graph for an acceptable region for the bayesian loss function approach (see `bayes_binom_two_loss`)

Usage

```
tradeoff_linear_integrate(ar, br, at, bt, efficacy_region_min,
toxicity_region_max, efficacy_region_max, toxicity_region_min)

tradeoff_linear_graph(input)
```

Arguments

`ar, br`	Parameters for the posterior distribution for response
`at, bt`	Parameters for the posterior distribution for toxicity
`efficacy_region_min`	Smallest acceptable efficacy
`toxicity_region_max`	Largest acceptable toxicity
`efficacy_region_max`	Point where no more tradeoff occurs for efficacy
`toxicity_region_min`	Point where no more tradeoff occurs for toxicity
`input`	A list values needed for the graph. It is expecting max.patients, efficacy_region_min, toxicity_region_max and will error without

Value

Returns value of the integration.

References

Chen Y, Smith BJ. Adaptive group sequential design for phase II clinical trials: a Bayesian decision theoretic approach. Stat Med 2009; 28: 3347-3362.

See Also

bayes_binom_two_loss

Integration functions and corresponding graphs: tradeoff_square_integrate,tradeoff_ellipse_integrate,tradeoff_

Examples

```
# modelled toxicity probability
t=c(0.1,0.1,0.3,0.3)
# modelled response probability
r=c(0.35,0.2,0.2,0.35)

reviews=c(10,15,20,25,30,35,40)
stage_after_trial=40

# uniform prior
pra=1;prb=1;pta=1;ptb=1

efficacy_critical_value=0.2
futility_critical_value=0.35
toxicity_critical_value=0.1
no_toxicity_critical_value=0.3

# alpha/beta ratio
l_alpha_beta=3
# cost of continuing compared to cost of alpha
l_alpha_c=750

efficacy_region_min=0.2
toxicity_region_max=0.3

########################################
# linear region
efficacy_region_min=0.2
efficacy_region_max=0.35
toxicity_region_min=0.1
toxicity_region_max=0.3

s=bayes_binom_two_loss(t,r,reviews,pra,prb,pta,ptb,l_alpha_beta,
l_alpha_c,stage_after_trial,fun.integrate=tradeoff_linear_integrate,
fun.graph=tradeoff_linear_graph,efficacy_critical_value,
toxicity_critical_value,futility_critical_value,
no_toxicity_critical_value,efficacy_region_min=efficacy_region_min,
toxicity_region_max=toxicity_region_max,
efficacy_region_max=efficacy_region_max,
toxicity_region_min=toxicity_region_min)

plot(s)
```

tradeoff ratio *Functions for integration for Bayesian loss methodology*

Description

An integral and graph for an acceptable region for the bayesian loss function approach (see bayes_binom_two_loss). tradeoff_ratio_intercepts computes the intercepts of the odd ratio curve with the limits.

Usage

```
tradeoff_ratio_intercepts(R_min,R_max,T_min,T_max,theta=0)

tradeoff_ratio_integrate(ar, br, at, bt, efficacy_region_min,
toxicity_region_max, efficacy_region_max, toxicity_region_min,
theta, intercepts)

tradeoff_ratio_graph(input)
```

Arguments

R_min, R_max, T_min, T_max	Limis for tradeoff region
theta	odd ratio for efficacy-toxicity tradeoff
intercepts	The values returned from tradeoff_ratio_intercepts
ar, br	Parameters for the posterior distribution for response
at, bt	Parameters for the posterior distribution for toxicity
efficacy_region_min	Smallest acceptable efficacy
toxicity_region_max	Largest acceptable toxicity
efficacy_region_max	Point where no more tradeoff occurs for efficacy
toxicity_region_min	Point where no more tradeoff occurs for toxicity
input	A list values needed for the graph. It is expecting max.patients, efficacy_region_min, toxicity_region_max and will error without

Value

Returns value of the integration.

References

Chen Y, Smith BJ. Adaptive group sequential design for phase II clinical trials: a Bayesian decision theoretic approach. Stat Med 2009; 28: 3347-3362.

See Also

bayes_binom_two_loss

Integration functions and corresponding graphs: tradeoff_square_integrate,tradeoff_ellipse_integrate,tradeoff_

Examples

```
# modelled toxicity probability
t=c(0.1,0.1,0.3,0.3)
# modelled response probability
r=c(0.35,0.2,0.2,0.35)

reviews=c(10,15,20,25,30,35,40)
stage_after_trial=40

# uniform prior
pra=1;prb=1;pta=1;ptb=1

efficacy_critical_value=0.2
futility_critical_value=0.35
toxicity_critical_value=0.1
no_toxicity_critical_value=0.3

# alpha/beta ratio
l_alpha_beta=3
# cost of continuing compared to cost of alpha
l_alpha_c=750

efficacy_region_min=0.2
toxicity_region_max=0.3

########################################
# odds ratio region
efficacy_region_min=0.2
efficacy_region_max=0.35
toxicity_region_min=0.1
toxicity_region_max=0.3

theta= 0.275/0.725 * 0.8/0.2

intercepts=tradeoff_ratio_intercepts(efficacy_region_min,
efficacy_region_max,toxicity_region_min,toxicity_region_max,theta)

s=bayes_binom_two_loss(t,r,reviews,pra,prb,pta,ptb,l_alpha_beta,
l_alpha_c,stage_after_trial,fun.integrate=tradeoff_ratio_integrate,
fun.graph=tradeoff_ratio_graph,efficacy_critical_value,
toxicity_critical_value,futility_critical_value,
no_toxicity_critical_value,efficacy_region_min=efficacy_region_min,
toxicity_region_max=toxicity_region_max,
efficacy_region_max=efficacy_region_max,
toxicity_region_min=toxicity_region_min,
theta=theta,intercepts=intercepts)
```

```
plot(s)
```

`tradeoff square` *Functions for integration for Bayesian loss methodology*

Description

An integral and graph for an acceptable region for the bayesian loss function approach (see `bayes_binom_two_loss`)

Usage

```
tradeoff_square_integrate(ar, br, at, bt, efficacy_region_min,
toxicity_region_max)

tradeoff_square_graph(input)
```

Arguments

`ar, br`	Parameters for the posterior distribution for response
`at, bt`	Parameters for the posterior distribution for toxicity
`efficacy_region_min`	Smallest acceptable efficacy
`toxicity_region_max`	Largest acceptable toxicity
`input`	A list values needed for the graph. It is expecting max.patients, efficacy_region_min, toxicity_region_max and will error without

Value

Returns value of the integration.

References

Chen Y, Smith BJ. Adaptive group sequential design for phase II clinical trials: a Bayesian decision theoretic approach. Stat Med 2009; 28: 3347-3362.

See Also

`bayes_binom_two_loss`

Integration functions and corresponding graphs: `tradeoff_square_integrate`,`tradeoff_ellipse_integrate`,`tradeoff_`

Examples

```
# modelled toxicity probability
t=c(0.1,0.1,0.3,0.3)
# modelled response probability
r=c(0.35,0.2,0.2,0.35)

reviews=c(10,15,20,25,30,35,40)
stage_after_trial=40

# uniform prior
pra=1;prb=1;pta=1;ptb=1

efficacy_critical_value=0.2
futility_critical_value=0.35
toxicity_critical_value=0.1
no_toxicity_critical_value=0.3

# alpha/beta ratio
l_alpha_beta=3
# cost of continuing compared to cost of alpha
l_alpha_c=750

efficacy_region_min=0.2
toxicity_region_max=0.3

########################################
# square region
s=bayes_binom_two_loss(t,r,reviews,pra,prb,pta,ptb,l_alpha_beta,
l_alpha_c,stage_after_trial,fun.integrate=tradeoff_square_integrate,
fun.graph=tradeoff_square_graph,efficacy_critical_value,
toxicity_critical_value,futility_critical_value,
no_toxicity_critical_value,efficacy_region_min=efficacy_region_min,
toxicity_region_max=toxicity_region_max)

plot(s)
```

---

trialDesign_binom_one-class

Class "trialDesign_binom_one"

Description

This is the s4 class for Binomial one endpoint designs with frequentist and Bayesian properties.

Objects from the Class

Objects can be created by calls of the form `new("trialDesign_binom_one", ...)`.

Slots

`reviews`: Object of class `"numeric"`, numeric vector of the number of patients an interim analysis will occur at

`success`: Object of class `"numeric"`, number of successed needed to stop for efficacy

`failure`: Object of class `"numeric"`, number of failures needed to stop for futility

`eta`, `zeta`: Object of class `"numeric"`, bayesiand properties of the design

`alpha`, `power`, `exp.p0`, `exp.p1`: Object of class `"numeric"`, frequentist properties of the design including the expected sampel size under H0 and H1

`p0`, `p1`: Object of class `"numeric"`, the probabilities used for H0 and H1 respectively

Methods

plot `signature(x = "trialDesign_binom_one", y = "ANY")`: ...

print `signature(x = "trialDesign_binom_one")`: ...

show `signature(object = "trialDesign_binom_one")`: ...

Examples

```
showClass("trialDesign_binom_one")
```

---

trialDesign_binom_two-class

Class `"trialDesign_binom_two"`

Description

This is the s4 class for Binomial two endpoint designs with frequentist and Bayesian properties.

Objects from the Class

Objects can be created by calls of the form `new("trialDesign_binom_two", ...)`.

Slots

`reviews`: Object of class `"numeric"`, a vector of the number of patients each interim analysis will occur at

`data`: Object of class `"data.frame"`, exact simulation values for each scenario

`cutpoints`: Object of class `"data.frame"`, the most extreme cutpoints when cause the trial to stop on their own (ignoring interaction with the other endpoint)

`precision`: Object of class `"numeric"`, should be a vector of ones confirming probability is all accounted for

`decision`: Object of class `"list"`, a list of matrices for the decisions to be made at each interim analysis

`post.futility`, `post.efficacy`, `post.toxicity`, `post.no_toxicity`: Object of class `"numeric"`, The posterior probabilities of the design

`graph`: Object of class `"list"`, an optional argument to pass to plot for the ploting of the first graph

data Exact simulation values for each scenario

Methods

plot `signature(x = "trialDesign_binom_two", y = "ANY")`: ...

print `signature(x = "trialDesign_binom_two")`: ...

show `signature(object = "trialDesign_binom_two")`: ...

Examples

```
showClass("trialDesign_binom_two")
```

Index