Package 'VisualizeSimon2Stage'

November 25, 2022

Type Package

Title Visualize Simon's Two-Stage Design

Version 0.1.1

Date 2022-11-25

Author Tingting Zhan [aut, cre, cph]

Maintainer Tingting Zhan <tingtingzhan@gmail.com>

Description Functions for visualizing the probabilities of early termination, fail and success of Simon's two-stage design. Functions for evaluating and visualizing the operating characteristics of Simon's two-stage design.

License GPL-2

Imports methods

Encoding UTF-8

Language en-US

Depends R (>= 4.2.0), ggplot2

Suggests clinfun

RoxygenNote 7.2.2

NeedsCompilation no

Repository CRAN

Date/Publication 2022-11-25 18:40:02 UTC

R topics documented:

`autoplot.ph2simon` *Plot A Simon's Two-Stage Design*

Description

Plot ph2simon object.

Usage

```
## S3 method for class 'ph2simon'
autoplot(object, ...)
```

Arguments

`object`	ph2simon object
`...`	potential parameters, currently not in use

Value

autoplot.ph2simon returns a ggplot object

Examples

```
library(clinfun)
(x = ph2simon(pu = .2, pa = .4, ep1 = .05, ep2 = .1))
class(x)
autoplot(x, type = 'minimax')
autoplot(x, type = 'optimal')
autoplot(x, type = 'n1')
autoplot(x, type = 'maximax')

# example from user feedback
(des = ph2simon(pu = .05, pa = .3, ep1 = .05, ep2 = .2))
autoplot(des, type = 'optimal')
autoplot(des, type = 'minimax')
```

`show,Simon_oc-method` *Show Simon_oc Object*

Description

Show Simon_oc object

Usage

```
## S4 method for signature 'Simon_oc'
show(object)
```

Arguments

`object`	Simon_oc object

Value

The show method for Simon_oc object does not have a returned value.

`show,Simon_pr-method` *Show Simon_pr Object*

Description

Show Simon_pr object

Usage

```
## S4 method for signature 'Simon_pr'
show(object)
```

Arguments

`object`	Simon_pr object

Value

The show method for Simon_pr object does not have a returned value.

simon_design-class *S4 Class simon_design*

Description

S4 Class simon_design

Slots

`n1,n` integer scalars, stage-1 sample size n_1 and total sample size n

`r1,r` integer scalars, number of positive response in stage-1 r_1 and overall r required **exclusive**. In other words, passing Stage 1 indicates observing $> r_1$ positive responses, and rejecting H_0 indicates observing $> r$ positive responses.

Simon_oc *Operating Characteristics of Simon's Two-Stage Design*

Description

..

Usage

```
Simon_oc(
  prob,
  simon,
  type = c("minimax", "optimal", "n1", "maximax"),
  R,
  n1 = stop("must provide `n1`"),
  n = stop("must provide `n`"),
  r1 = stop("must provide `r1`"),
  r = stop("must provide `r`"),
  ...
)
```

Arguments

`prob`	**named** numeric vector, true response rate(s)
`simon`	ph2simon object
`type`	character scalar, either `'minimax'` for Simon's two-stage design with minimum total sample size (default), `'optimal'` for minimum expected total sample size **under** p_0, `'n1'` for minimum stage-1 sample size, or `'maximax'` for maximum total sample size (as provided by user).
`R`	integer scalar, number of simulations

`n1, n` (optional) integer scalars, stage 1 sample size n_1 and total sample size n. Will be overridden if `simon` is given

`r1, r` (optional) integer scalars, number of positive response in Stage 1 r_1 and overall r required **exclusive**. In other words, passing Stage 1 means observing $> r_1$ positive response. Will be overridden if `simon` is given

`...` potential parameters, currently not in use

Details

..

Value

Simon_oc returns Simon_oc object

References

doi:10.1016/01972456(89)900159

Examples

```
library(clinfun)
(x = ph2simon(pu = .2, pa = .4, ep1 = .05, ep2 = .1))
Simon_oc(prob = c(A = .3, B = .2, C = .15), simon = x, type = 'minimax', R = 1e3L)
Simon_oc(prob = c(A = .3, B = .2, C = .15), simon = x, type = 'optimal', R = 1e3L)
```

Simon_oc-class *S4 Class Simon_oc*

Description

S4 Class Simon_oc

Slots

`.Data` Simon_pr object

`maxResp` integer vector of length p, the frequencies of each regime having maximum response. The summation of `maxResp` is the number of simulation copies.

`Simon_maxResp` integer vector of length p, the frequencies of each regime having maximum response and success in Simon's two-stage trial.

`Simon_pr` *Probabilities of Simon's Two-Stage Design*

Description

Probability of frail (i.e., early termination), fail (to reject the null) and success (to reject the null) of a Simon's two-stage design, at given true response rate(s).

Usage

```
Simon_pr(prob, n1, n, r1, r)
```

Arguments

`prob`	numeric vector, true response rate(s) p
`n1, n`	integer scalars, stage-1 sample size n_1 and total sample size n
`r1, r`	integer scalars, number of positive response in stage-1 r_1 and overall r required **exclusive**. In other words, passing Stage 1 indicates observing $> r_1$ positive responses, and rejecting H_0 indicates observing $> r$ positive responses.

Details

Given the Simon's two-stage design with n_1, r_1, n and r, for a response rate p, we have the number of Stage-1 positive responses $X_1 \sim \text{Binom}(n_1, p)$ and the number of Stage-2 positive responses $X_2 \sim \text{Binom}(n - n_1, p)$. Obviously X_1 and X_2 are independent.

The probability of early termination is $\Pr(X_1 \leq r_1)$.

The probability of failure to reject H_0 is

$$\sum_{s_1=r_1+1}^{n_1} \Pr(X_1 = s_1) \cdot \Pr(X_2 \leq (r - s_1))$$

The probability of rejection of H_0 is

$$\sum_{s_1=r_1+1}^{n_1} \Pr(X_1 = s_1) \cdot \Pr(X_2 > (r - s_1))$$

Parameters nomenclature of `n1`, `n`, `r1` and `r` follows that of PASS and ph2simon.

Value

Simon_pr returns Simon_pr object.

References

doi:10.1016/01972456(89)900159

Examples

```
Simon_pr(prob = c(.2, .4), n1 = 15L, r1 = 3L, n = 24L, r = 7L)
```

`Simon_pr-class` *S4 Class Simon_pr*

Description

Probabilities of early termination, failure and success, of Simon's Two-Stage Design.

Slots

`.Data` $l \times 3$ numeric matrix, probability of frail (i.e., early termination), fail (to reject the null) and success (to reject the null), at each response rate p given in `@prob`

`eN` numeric vector of length l, expected sample size(s) $\mathrm{E}(N)$

`prob` numeric vector of length l, response rate(s) p

`simon` simon_design object

Index