Package ‘VisualizeSimon2Stage’

June 28, 2022

Type Package
Title Visualize Simon's Two-Stage Design
Version 0.1.0
Date 2022-06-26
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 7.2.0
NeedsCompilation no
Repository CRAN
Date/Publication 2022-06-28 07:20:05 UTC

R topics documented:

VisualizeSimon2Stage-package .................................................. 2
print.ph2simon ................................................................. 2
show,Simon_oc-method ....................................................... 3
show,Simon_pr-method ....................................................... 3
Simon_oc ................................................................. 4
Simon_oc-class ............................................................ 5
Simon_pr ................................................................. 5
Simon_pr-class ............................................................ 6
Index

VisualizeSimon2Stage-package

Visualize Simon's Two-Stage Design

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.

print.ph2simon

Print ph2simon Object

Description

Print ph2simon object, overwriting clinfun::print.ph2simon

Usage

## S3 method for class 'ph2simon'
print(x, ...)

Arguments

x ph2simon object

... potential parameters, currently not in use

Value

print.ph2simon does not have a return value

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')
show,Simon_oc-method

---

### Show Simon_oc Object

**Description**

Show Simon_oc object

**Usage**

```r
## 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**

```r
## 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_oc

Operating Characteristics of Simon’s Two-Stage Design

Description

...

Usage

Simon_oc(
  prob,  
  simon,  
  type = c("minimax", "optimal", "n1", "maximax"),  
  N,  
  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).
N         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, N = 1e3L)

Simon_oc-class S4 class Simon_oc

Description

S4 class Simon_oc

Slots

prob named numeric vector of length \( p \), true response rate(s)
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.
eN numeric vector of length \( p \), expected sample sizes by simulation

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)
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 means observing \( > r_1 \) positive response.
**Details**

Parameters nomenclature of \( n_1, n, r_1 \) 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(n1 = 15L, r1 = 3L, n = 24L, r = 7L, prob = c(.2, .3))

---

**Description**

S4 class Simon_pr

**Slots**

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

\( eN \) numeric vector of length \( p \), expected sample size(s)

\( \text{prob} \) numeric vector of length \( p \), true response rate(s)
Index

* package
  VisualizeSimon2Stage-package, 2

character, 4
integer, 4, 5
matrix, 6
numeric, 4–6
ph2simon, 2, 4, 6
print.ph2simon, 2, 2

show, 3
show,Simon_oc-method, 3
show,Simon_pr-method, 3
Simon_oc, 3, 4, 4, 5
Simon_oc-class, 5
Simon_pr, 3, 5, 6
Simon_pr-class, 6

VisualizeSimon2Stage-package, 2