Package: BinGSD (via r-universe)

August 23, 2024

Title Calculation for Single Arm Group Sequential Test with Binary Endpoint

Version 0.0.1

Description Consider an at-most-K-stage group sequential design with only an upper bound for the last analysis and non-binding lower bounds.With binary endpoint, two kinds of test can be applied, asymptotic test based on normal distribution and exact test based on binomial distribution. This package supports the computation of boundaries and conditional power for single-arm group sequential test with binary endpoint, via either asymptotic or exact test. The package also provides functions to obtain boundary crossing probabilities given the design.

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

Imports methods (>= 3.5.0), mvtnorm (>= 1.0-11), stats (>= 3.5.0)

Depends R (>= 3.5.0)

License GPL-3

NeedsCompilation no

Author Lei Wang [aut, cre, cph], BeiGene, Ltd. (Nasdaq: BGNE, HKEX: 06160) [cph]

Maintainer Lei Wang <slimewanglei@163.com>

Date/Publication 2019-10-30 17:00:18 UTC

Repository https://xiaomangmang.r-universe.dev

RemoteUrl https://github.com/cran/BinGSD

RemoteRef HEAD

RemoteSha 94b7b5de408aa79152c26a23335fedb10096f374

Contents

asymcp 2
asymdesign 3
asymprob 6
exactcp 8
exactdesign 9
exactprob 12

Index **14**

`asymcp` *Conditional power computation using asymptotic test.*

Description

Compute conditional power of single-arm group sequential design with binary endpoint based on asymptotic test, given the interim result.

Usage

```
asymcp(d, p_1, i, z_i)
```

Arguments

`d`	An object of the class asymdesign or asymprob.
`p_1`	A scalar or vector representing response rate or probability of success under the alternative hypothesis. The value(s) should be within (p_0,1).
`i`	Index of the analysis at which the interim statistic is given. Should be an integer ranges from 1 to K-1. i will be rounded to its nearest whole value if it is not an integer.
`z_i`	The interim statistic at analysis i.

Details

Conditional power quantifies the conditional probability of crossing the upper bound given the interim result z_i, $1 \leq i < K$. Having inherited sample sizes and boundaries from `asymdesign` or `asymprob`, given the interim statistic at ith analysis z_i, the conditional power is defined as

$$\alpha_{i,K}(p|z_i) = P_p(Z_K \geq u_K, Z_{K-1} > l_{K-1}, \ldots, Z_{i+1} > l_{i+1} | Z_i = z_i)$$

With asymptotic test, the test statistic at analysis k is $Z_k = \hat{\theta}_k \sqrt{n_k/p/(1-p)} = (\sum_{s=1}^{n_k} X_s/n_k - p_0)\sqrt{n_k/p/(1-p)}$, which follows the normal distribution $N(\theta\sqrt{n_k/p/(1-p)}, 1)$ with $\theta = p - p_0$. In practice, p in Z_k can be substituted with the sample response rate $\sum_{s=1}^{n_k} X_s/n_k$.

The increment statistic $Z_k\sqrt{n_k/p/(1-p)} - Z_{k-1}\sqrt{n_{k-1}/p/(1-p)}$ also follows a normal distribution independently of $Z_1, \ldots, Z_{k-1}$. Then the conditional power can be easily obtained using a procedure similar to that for unconditional boundary crossing probabilities.

Value

A list with the elements as follows:

- K: As in d.
- n.I: As in d.
- u_K: As in d.
- lowerbounds: As in d.
- i: i used in computation.
- z_i: As input.
- cp: A matrix of conditional powers under different response rates.
- p_1: As input.
- p_0: As input.

Reference

- Alan Genz et al. (2018). mvtnorm: Multivariate Normal and t Distributions. R package version 1.0-11.

See Also

asymprob, asymdesign, exactcp.

Examples

```
I=c(0.2,0.4,0.6,0.8,0.99)
beta=0.2
betaspend=c(0.1,0.2,0.3,0.3,0.2)
alpha=0.05
p_0=0.3
p_1=0.5
K=4.6
tol=1e-6
tt1=asymdesign(I,beta,betaspend,alpha,p_0,p_1,K,tol)
tt2=asymprob(p_1=c(0.4,0.5,0.6,0.7,0.8,0.9),d=tt1)
asymcp(tt1,p_1=c(0.4,0.5,0.6,0.7,0.8,0.9),1,2)
asymcp(tt2,p_1=c(0.4,0.5,0.6,0.7,0.8,0.9),3,2.2)
```

---

asymdesign *Boundary and sample size computation using asymptotic test.*

Description

Calculate boundaries and sample sizes of single-arm group sequential design with binary endpoint based on asymptotic test.

Usage

```
asymdesign(I, beta = 0.3, betaspend, alpha = 0.05, p_0, p_1, K,
  tol = 1e-06)
```

Arguments

I	The information fractions at each analysis. For binary endpoints, the information fraction for analysis k is equal to n_k/n_K, where n_k is the sample size available at analysis k and n_K is the sample size available at the last analysis or the maximum sample size. Should be a positive increasing vector of length K or K-1. If I has K elements among which the last one is not 1, then I will be standardized so that the last information fraction is 1. If I has K-1 elements, the last element in I must be less than 1.
beta	The desired overall type II error level. Should be a scalar within the interval (0,0.5]. Default value is 0.3, that is, power=0.7.
betaspend	The proportions of beta spent at each analysis. Should be a vector of length K with all elements belong to [0,1]. If the sum of all elements in betaspend is not equal to 1, betaspend will be standardized.
alpha	The desired overall type I error level. Should be a scalar within the interval (0,0.3]. Default is 0.05.
p_0	The response rate or the probability of success under null hypothesis. Should be a scalar within (0,1).
p_1	The response rate or the probability of success under alternative hypothesis. Should be a scalar within (p_0,1).
K	The maximum number of analyses, including the interim and the final. Should be an integer within (1,20]. K will be rounded to its nearest whole number if it is not an integer.
tol	The tolerance level which is essentially the maximum acceptable difference between the desired type II error spending and the actual type II error spending, when computing the boundaries using asymptotic test. Should be a positive scalar no more than 0.01. The default value is 1e-6.

Details

Suppose $X_1, X_2, \ldots$ are binary outcomes following Bernoulli distribution $b(1, p)$, in which 1 stands for the case that the subject responds to the treatment and 0 otherwise. Consider a group sequential test with K planned analyses, where the null and alternative hypotheses are $H_0 : p = p_0$ and $H_1 : p = p_1$ respectively. Note that generally p_1 is greater than p_0. For $k < K$, the trial stops if and only if the test statistic Z_k crosses the futility boundary, that is, $Z_k <= l_k$. The lower bound for the last analysis l_K is set to be equal to the last and only upper bound u_K to make a decision. At the last analysis, the null hypothesis will be rejected if $Z_K >= u_K$.

The computation of lower bounds except for the last one is implemented with u_K fixed, thus the derived lower bounds are non-binding. Furthermore, the overall type I error will not be inflated if the trial continues after crossing any of the interim lower bounds, which is convenient for the purpose of monitoring. Let the sequence of sample sizes required at each analysis be $n_1, n_2, \ldots, n_K$.

For binomial endpoint, the Fisher information equals $n_k/p/(1-p)$ which is proportional to n_k. Accordingly, the information fraction available at each analysis is equivalent to n_k/n_K.

For a p_0 not close to 1 or 0, with a large sample size, the test statistic at analysis k is $Z_k = \hat{\theta}_k\sqrt{n_k/p/(1-p)} = (\sum_{s=1}^{n_k} X_s/n_k - p_0)\sqrt{n_k/p/(1-p)}$, which follows the normal distribution $N(\theta\sqrt{n_k/p/(1-p)}, 1)$ with $\theta = p - p_0$. In practice, p in Z_k can be substituted with the sample response rate $\sum_{s=1}^{n_k} X_s/n_k$.

Under the null hypothesis, $\theta = 0$ and Z_k follows a standard normal distribution. During the calculation, the only upper bound u_K is firstly derived under H_0, without given n_K. Thus, there is no need to adjust u_K for different levels of n_K. Following East, given u_K, compute the maximum sample size n_K under H_1. The rest sample sizes can be obtained by multiplying information fractions and n_K. The lower boundaries for the first $K-1$ analyses are sequentially determined by a search method. The whole searching procedure stops if the overall type II error does not excess the desired level or the times of iteration excess 30. Otherwise, increase the sample sizes until the type II error meets user's requirement.

The multiple integrals of multivariate normal density functions are conducted with `pmvnorm` in R package mvtnorm. Through a few transformations of the integral variables, `pmvnorm` turns the multiple integral to the product of several univariate integrals, which greatly reduces the computational burden of sequentially searching for appropriate boundaries.

Value

An object of the class asymdesign. This class contains:

- I: I used in computation.
- beta: As input.
- betaspend: The desired type II error spent at each analysis used in computation.
- alpha: As input.
- p_0: As input.
- p_1: As input.
- K: K used in computation.
- tol: As input.
- n.I: A vector of length K which contains sample sizes required at each analysis to achieve desired type I and type II error requirements. n.I equals sample size for the last analysis times the vector of information fractions.
- u_K: The upper boundary for the last analysis.
- lowerbounds: A vector of length K which contains lower boundaries for each analysis. Note that the lower boundaries are non-binding.
- problow: Probabilities of crossing the lower bounds under H_1 or the actual type II error at each analysis.
- probhi: Probability of crossing the last upper bound under H_0 or the actual type I error.
- power: power of the group sequential test with the value equals 1-sum(problow).

Reference

- Cytel Inc. East Version 6.4.1 Manual. 2017.
- Alan Genz et al. (2018). mvtnorm: Multivariate Normal and t Distributions. R package version 1.0-11.

See Also

asymprob, asymcp, exactdesign.

Examples

```
I=c(0.2,0.4,0.6,0.8,0.99)
beta=0.2
betaspend=c(0.1,0.2,0.3,0.3,0.2)
alpha=0.05
p_0=0.3
p_1=0.5
K=4.6
tol=1e-6
tt1=asymdesign(I,beta,betaspend,alpha,p_0,p_1,K,tol)
```

---

asymprob *Boundary crossing probabilities computation using asymptotic test.*

Description

Calculate boundary crossing probabilities of single-arm group sequential design with binary endpoint based on asymptotic test.

Usage

```
asymprob(K = 0, p_0, p_1, n.I, u_K, lowerbounds, d = NULL)
```

Arguments

K	The maximum number of analyses, including the interim and the final. Should be an integer within (1,20]. K will be rounded to its nearest whole number if it is not an integer. The default is 0.
p_0	The response rate or the probability of success under null hypothesis. Should be a scalar within (0,1).
p_1	A scalar or vector representing response rate or probability of success under the alternative hypothesis. The value(s) should be within (p_0,1). It is a mandatory input.
n.I	A vector of length K which contains sample sizes required at each analysis. Should be a positive and increasing sequence.

u_K	The upper boundary for the last analysis.
lowerbounds	Non-decreasing lower boundaries for each analysis. With length K, the last lower bound must be identical to u_K. With length K-1, the last element must be no greater than u_K and u_K will be automatically added into the sequence.
d	An object of the class asymdesign.

Details

This function calculates probabilities of crossing the upper or the lower boundaries under null hypothesis and a set of alternative hypotheses. With K=0 (as default), d must be an object of class asymdesign. Meanwhile, other arguments except for p_1 will be inherited from d and the input values will be ignored. With K!=0, the probabilities are derived from the input arguments. In this circumstance, all arguments except for d are required.

The computation is based on the single-arm group sequential asymptotic test described in asymdesign. Therefore, for the output matrix of upper bound crossing probabilities, the values for the first K-1 analyses are zero since there is only one upper bound for the last analysis.

Value

An object of the class asymprob. This class contains:

- p_0: As input with d=NULL or as in d.
- p_1: As input.
- K: K used in computation.
- n.I: As input with d=NULL or as in d.
- u_K: As input with d=NULL or as in d.
- lowerbounds: lowerbounds used in computation.
- problow: Probabilities of crossing the lower bounds at each analysis.
- probhi: Probability of crossing the upper bounds at each analysis.

Reference

- Alan Genz et al. (2018). mvtnorm: Multivariate Normal and t Distributions. R package version 1.0-11.

See Also

asymdesign, asymcp, exactprob.

Examples

```
I=c(0.2,0.4,0.6,0.8,0.99)
beta=0.2
betaspend=c(0.1,0.2,0.3,0.3,0.2)
alpha=0.05
p_0=0.3
p_1=0.5
```

```
K=4.6
tol=1e-6
tt1=asymdesign(I,beta,betaspend,alpha,p_0,p_1,K,tol)
asymprob(p_1=c(0.4,0.5,0.6,0.7,0.8,0.9),d=tt1)
asymprob(K=5,p_0=0.4,p_1=c(0.5,0.6,0.7,0.8),n.I=c(15,20,25,30,35),u_K=1.65,
lowerbounds=c(-1.2,-0.5,0.2,0.8,1.65))
```

`exactcp` *Conditional power computation using exact test.*

Description

Compute conditional power of single-arm group sequential design with binary endpoint based on binomial distribution.

Usage

```
exactcp(d, p_1, i, z_i)
```

Arguments

`d`	An object of the class exactdesign or exactprob.
`p_1`	A scalar or vector representing response rate or probability of success under the alternative hypothesis. The value(s) should be within (p_0,1).
`i`	Index of the analysis at which the interim statistic is given. Should be an integer ranges from 1 to K-1. i will be rounded to its nearest whole value if it is not an integer.
`z_i`	The interim statistic at analysis i.

Details

Conditional power quantifies the conditional probability of crossing the upper bound given the interim result z_i, $1 \leq i < K$. Having inherited sample sizes and boundaries from `exactdesign` or `exactprob`, given the interim statistic at ith analysis z_i, the conditional power is defined as

$\alpha_{i,K}(p|z_i) = P_p(Z_K \geq u_K, Z_{K-1} > l_{K-1}, \ldots, Z_{i+1} > l_{i+1}|Z_i = z_i)$

With exact test, the test statistic at analysis k is $Z_k = \sum_{s=1}^{n_k} X_s$ which follows binomial distribution $b(n_k, p)$. Actually, Z_k is the total number of responses up to the kth analysis.

The increment statistic $Z_k - Z_{k-1}$ also follows a binomial distribution $b(n_k - n_{k-1}, p)$ independently of $Z_1, \ldots, Z_{k-1}$. Then the conditional power can be easily obtained using the same procedure for deriving unconditional boundary crossing probabilities.

Note that $Z_1, \ldots, Z_K$ is a non-decreasing sequence, thus the conditional power is 1 when the interim statistic $z_i >= u_K$.

Value

A list with the elements as follows:

- K: As in d.
- n.I: As in d.
- u_K: As in d.
- lowerbounds: As in d.
- i: i used in computation.
- z_i: As input.
- cp: A matrix of conditional powers under different response rates.
- p_1: As input.
- p_0: As input.

Reference

- Christopher Jennison, Bruce W. Turnbull. Group Sequential Methods with Applications to Clinical Trials. Chapman and Hall/CRC, Boca Raton, FL, 2000.

See Also

exactprob, asymcp, exactdesign.

Examples

```
I=c(0.2,0.4,0.6,0.8,0.99)
beta=0.2
betaspend=c(0.1,0.2,0.3,0.3,0.2)
alpha=0.05
p_0=0.3
p_1=0.5
K=4.6
tol=1e-6
tt1=asymdesign(I,beta,betaspend,alpha,p_0,p_1,K,tol)
tt2=exactdesign(tt1)
tt3=exactprob(p_1=c(0.4,0.5,0.6,0.7,0.8,0.9),d=tt2)
exactcp(tt2,p_1=c(0.4,0.5,0.6,0.7,0.8,0.9),1,2)
exactcp(tt3,p_1=c(0.4,0.5,0.6,0.7,0.8,0.9),3,19)
```

---

exactdesign *Compute sample size and boundaries using exact binomial distribution*

Description

Compute sample size and boundaries of single-arm group sequential design with binary endpoint using exact binomial distribution

Usage

```
exactdesign(d)
```

Arguments

d	An object of the class asymdesign.

Details

Suppose $X_1, X_2, \ldots$ are binary outcomes following Bernoulli distribution $b(1, p)$, in which 1 stands for the case that the subject responds to the treatment and 0 otherwise. Consider a group sequential test with K planned analyses, where the null and alternative hypotheses are $H_0 : p = p_0$ and $H_1 : p = p_1$ respectively. Note that generally p_1 is greater than p_0. For $k < K$, the trial stops if and only if the test statistic Z_k crosses the futility boundary, that is, $Z_k <= l_k$. The lower bound for the last analysis l_K is set to be equal to the last and only upper bound u_K to make a decision. At the last analysis, the null hypothesis will be rejected if $Z_K >= u_K$.

The computation of lower bounds except for the last one is implemented with u_K fixed, thus the derived lower bounds are non-binding. Furthermore, the overall type I error will not be inflated if the trial continues after crossing any of the interim lower bounds, which is convenient for the purpose of monitoring. Let the sequence of sample sizes required at each analysis be $n_1, n_2, \ldots, n_K$. For binomial endpoint, the Fisher information equals $n_k/p/(1-p)$ which is proportional to n_k. Accordingly, the information fraction available at each analysis is equivalent to n_k/n_K.

With exact test, the test statistic at analysis k is $Z_k = \sum_{s=1}^{n_k} X_s$ which follows binomial distribution $b(n_k, p)$. Actually, Z_k is the total number of responses up to the kth analysis.

Under the null hypothesis, Z_k follows a binomial distribution $b(n_k, p_0)$. While under the alternative hypothesis, Z_k follows $b(n_k, p_1)$. It may involve massive computation to simultaneously find proper n_K and u_K. In fact, the sample sizes obtained from asymptotic test ought to be close to those from exact test. Thus, we adopt n_K from asymptotic test as the starting value. The starting value of u_K is computed given the n_K. Iteratively update u_K and n_K until errors are limited to certain amount.

Like asymdesign, the lower boundaries for the first $K-1$ analyses are sequentially determined by a search method. However, if the actual overall type II error exceeds the desired level, not only sample sizes but also all the boundaries are updated, since the binomial distribution under H_0 involves with sample size.

Due to the discreteness of binomial distribution, in exact test, the type I and type II error actually spent at each analysis may not approximate the designated amount. With the only one upper bound, the whole type I error is spent at the final analysis. From some simulation studies, though not presented here, we found that carrying over unused type II error has minor influence on the resulting boundaries and sample sizes. However, in an attempt to reduce the false positive rate, we decided to recycle the unspent amount of desired type II error. Thus, the elements of betaspend in an exactdesign object may be greater than the amount pre-specified by the user.

Value

An object of the class exactdesign. This class contains:

- I: I used in computation, as in d.
- beta: The desired overall type II error level, as in d.

- betaspend: The desired type II error spent at each analysis used in computation, as in d.
- alpha: The desired overall type I error level, as in d.
- p_0: The response rate or the probability of success under null hypothesis, as in d.
- p_1: The response rate or the probability of success under alternative hypothesis, as in d.
- K: K used in computation, as in d.
- n.I: A vector of length K which contains sample sizes required at each analysis to achieve desired type I and type II error requirements. n.I equals sample size for the last analysis times the vector of information fractions.
- u_K: The upper boundary for the last analysis.
- lowerbounds: A vector of length K which contains lower boundaries for each analysis. Note that the lower boundaries are non-binding.
- problow: Probabilities of crossing the lower bounds under H_1 or the actual type II error at each analysis.
- probhi: Probability of crossing the last upper bound under H_0 or the actual type I error.
- power: power of the group sequential test with the value equals 1-sum(problow).

Reference

- Christopher Jennison, Bruce W. Turnbull. Group Sequential Methods with Applications to Clinical Trials. Chapman and Hall/CRC, Boca Raton, FL, 2000.

See Also

exactprob, exactcp, asymdesign.

Examples

```
I=c(0.2,0.4,0.6,0.8,0.99)
beta=0.2
betaspend=c(0.1,0.2,0.3,0.3,0.2)
alpha=0.05
p_0=0.3
p_1=0.5
K=4.6
tol=1e-6
tt1=asymdesign(I,beta,betaspend,alpha,p_0,p_1,K,tol)
tt2=exactdesign(tt1)
```

`exactprob` *Boundary crossing probabilities computation using exact test.*

Description

Calculate boundary crossing probabilities of single-arm group sequential design with binary endpoint using binomial distribution

Usage

```
exactprob(K = 0, p_0, p_1, n.I, u_K, lowerbounds, d = NULL)
```

Arguments

`K`	The maximum number of analyses, including the interim and the final. Should be an integer within (1,20]. K will be rounded to the nearest whole number if it is not an integer. The default is 0.
`p_0`	The response rate or the probability of success under null hypothesis. Should be a scalar within (0,1).
`p_1`	A scalar or vector representing response rate or probability of success under the alternative hypothesis. The value(s) should be within (p_0,1). It is a mandatory input.
`n.I`	A vector of length K which contains sample sizes required at each analysis. Should be a positive and increasing sequence.
`u_K`	The upper boundary for the last analysis.
`lowerbounds`	Non-decreasing lower boundaries for each analysis, in which each element is no less than -1 (no lower bound). With length K, the last lower bound must be identical to u_K. With length K-1, the last element must be no greater than u_K and u_K will be automatically added into the sequence. Note the lower bound must be less than the corresponding sample size.
`d`	An object of the class exactdesign.

Details

This function is similar to `asymprob` except that the former uses binomial distribution and the latter uses the normal asymptotic distribution. With `K=0` (as default), `d` must be an object of class exactdesign. Meanwhile, other arguments except for `p_1` will be inherited from `d` and the input values will be ignored. With `K!=0`, the probabilities are derived from the input arguments. In this circumstance, all the arguments except for `d` are required.

The computation is based on the single-arm group sequential exact test described in `exactdesign`. Therefore, for the output matrix of upper bound crossing probabilities, the values for the first K-1 analyses are zero since there is only one upper bound for the last analysis.

Value

An object of the class exactprob. This class contains:

- p_0: As input with d=NULL or as in d.
- p_1: As input.
- K: K used in computation.
- n.I: As input with d=NULL or as in d.
- u_K: As input with d=NULL or as in d.
- lowerbounds: lowerbounds used in computation.
- problow: Probabilities of crossing the lower bounds at each analysis.
- probhi: Probability of crossing the upper bounds at each analysis.

Reference

- Christopher Jennison, Bruce W. Turnbull. Group Sequential Methods with Applications to Clinical Trials. Chapman and Hall/CRC, Boca Raton, FL, 2000.
- Keaven M. Anderson, Dan (Jennifer) Sun, Zhongxin (John) Zhang. gsDesign: An R Package for Designing Group Sequential Clinical Trials. R package version 3.0-1.

Note

The calculation of boundary crossing probabilities here borrowed strength from the source code of function gsBinomialExact in package gsDesign and we really appreciate their work.

See Also

exactdesign, exactcp, asymprob.

Examples

```
I=c(0.2,0.4,0.6,0.8,0.99)
beta=0.2
betaspend=c(0.1,0.2,0.3,0.3,0.2)
alpha=0.05
p_0=0.3
p_1=0.5
K=4.6
tol=1e-6
tt1=asymdesign(I,beta,betaspend,alpha,p_0,p_1,K,tol)
tt2=exactdesign(tt1)
tt3=exactprob(p_1=c(0.4,0.5,0.6,0.7,0.8,0.9),d=tt2)
tt3=exactprob(K=5,p_0=0.4,p_1=c(0.5,0.6,0.7,0.8),n.I=c(15,20,25,30,35),u_K=15,
lowerbounds=c(3,5,10,12,15))
```

Index

asymcp, 2, *6, 7*, *9*
asymdesign, *2, 3*, 3, *7*, *10, 11*
asymprob, *2, 3*, *6*, 6, *12, 13*

exactcp, *3*, 8, *11*, *13*
exactdesign, *6*, *8, 9*, 9, *12, 13*
exactprob, *7–9*, *11*, 12

pmvnorm, *5*