

Package: grouprar (via r-universe)

October 9, 2026

Type Package

Title Group Response Adaptive Randomization for Clinical Trials

Version 0.2.0

Date 2026-10-08

Description Implements group response-adaptive randomization procedures, which include standard (non-group) response-adaptive randomization methods as special cases. The package also handles delayed and missing responses, which broadens its use in real-world trials. It offers functions for simulating a variety of response-adaptive randomization procedures, to help guide the choice of design for a clinical trial, including the doubly adaptive biased coin design and the multi-arm efficient randomized adaptive design (ERADE), k-arm optimal target allocations, group sequential monitoring, and a function that computes allocation probabilities for an ongoing trial. For details of the methods and algorithms, see the following references: Wei, L. J. (1979) <[doi:10.1214/aos/1176344614](https://doi.org/10.1214/aos/1176344614)>; Wei, L. J. and Durham, S. (1978) <[doi:10.1080/01621459.1978.10480109](https://doi.org/10.1080/01621459.1978.10480109)>; Durham, S. D., Flournoy, N. and Li, W. (1998) <[doi:10.2307/3315771](https://doi.org/10.2307/3315771)>; Ivanova, A., Rosenberger, W. F., Durham, S. D. and Flournoy, N. (2000) <<https://www.jstor.org/stable/25053121>>; Bai, Z. D., Hu, F. and Shen, L. (2002) <[doi:10.1006/jmva.2001.1987](https://doi.org/10.1006/jmva.2001.1987)>; Ivanova, A. (2003) <[doi:10.1007/s001840200220](https://doi.org/10.1007/s001840200220)>; Hu, F. and Zhang, L. X. (2004) <[doi:10.1214/aos/1079120137](https://doi.org/10.1214/aos/1079120137)>; Hu, F. and Rosenberger, W. F. (2006, ISBN:978-0-471-65396-7); Zhang, L. X., Chan, W. S., Cheung, S. H. and Hu, F. (2007) <<https://www.jstor.org/stable/26432528>>; Zhang, L. and Rosenberger, W. F. (2006) <[doi:10.1111/j.1541-0420.2005.00496.x](https://doi.org/10.1111/j.1541-0420.2005.00496.x)>; Hu, F., Zhang, L. X., Cheung, S. H. and Chan, W. S. (2008) <[doi:10.1002/cjs.5550360404](https://doi.org/10.1002/cjs.5550360404)>; Tymofyeyev, Y., Rosenberger, W. F. and Hu, F. (2007) <[doi:10.1198/016214506000000906](https://doi.org/10.1198/016214506000000906)>; Hu, F., Zhang, L. X. and He, X. (2009) <[doi:10.1214/08-AOS655](https://doi.org/10.1214/08-AOS655)>; Zhu, H. and Hu, F. (2010) <[doi:10.1214/10-AOS796](https://doi.org/10.1214/10-AOS796)>; Zhai, G., Li, Y.,

Zhang, L. and Hu, F. (2024) <doi:10.1002/sim.10220>; Alkhnefr, N., Hu, F. and Zhai, G. (2025) <doi:10.1177/09622802251362644>.

License GPL (>= 2)
Imports extraDistr, stats
Suggests testthat (>= 3.1.5)
Config/testthat/edition 3
Depends R (>= 3.6.0)
Encoding UTF-8
NeedsCompilation no
Author Guannan Zhai [aut, cre], Feifang Hu [aut, ths]
Maintainer Guannan Zhai <guannanzhai1996@gmail.com>
Repository https://neptune555.r-universe.dev
Date/Publication 2026-10-09 09:17:39 UTC
RemoteUrl https://github.com/cran/grouprar
RemoteRef HEAD
RemoteSha 450af664b350b6346ef2c89c403cca70fe05daf5

Contents

Bai.Hu.Shen.Urn	3
BirthDeathUrn	5
CRDesign	7
DBCD_Bin	9
DBCD_Cont	14
DLRule	18
dyldDBCD_Bin	20
dyldDBCD_Cont	24
GDLRule	29
Group.DBCD_Bin	31
Group.DBCD_Cont	35
Group.dyldDBCD_Bin	40
Group.dyldDBCD_Cont	44
grouprar	48
nextAlloc	50
PolyaUrn	53
print.grouprar	55
RPWRule	56
sqMonitor	59
WeiUrn	60

Index	63
--------------	-----------

Bai.Hu.Shen.Urn

*Bai Hu Shen's Urn***Description**

Bai, Hu, and Shen (2002) proposed an adaptive design for multi-arm clinical trials. The allocation probabilities adapt to the performance of the patients already treated: a success on a treatment increases the chance that the next patient is assigned to it, and a failure moves probability to the other treatments in proportion to their estimated success rates. This function simulates the Bai, Hu, and Shen urn with two-sided hypothesis testing in a clinical trial context.

Usage

```
Bai.Hu.Shen.Urn(k, p, ssn, Y0 = NULL, nsim = 2000, alpha = 0.05,
  test.fun = NULL, typeI = FALSE, seed = NULL)
```

Arguments

k	A positive integer. The number of treatment groups in the trial ($k \geq 2$).
p	A vector of length k with values between 0 and 1. The true success rates of the treatments, used to generate data for the simulations.
ssn	A positive integer. The total number of participants in each simulated trial.
Y0	A vector of length k giving the initial urn composition (number of balls of each treatment type). For instance, if $Y0 = c(1, 1, 1)$, the first patient is assigned to each treatment with probability $Y0 / \text{sum}(Y0)$. If Y0 is NULL (default), it is set to a vector of k ones.
nsim	A positive integer. The number of simulated trials, with a default value of 2000.
alpha	A number between 0 and 1. The significance level of the two-sided test, with a default value of 0.05.
test.fun	An optional function <code>function(outcome, assignment)</code> that returns the p-value of a test for one simulated trial, where <code>outcome</code> holds the observed responses and <code>assignment</code> the treatment labels (integers 1 to k); missing responses are removed first. The null hypothesis is rejected when the p-value is at most alpha. The default NULL uses the built-in two-sided tests: the t-test for two arms and the Wald chi-squared test of equal means for more than two arms (for binary responses, if some arms have no variability, the variances are computed from $(S_k + 1)/(n_k + 2)$ as in Agresti and Caffo, 2000).
typeI	Logical. If TRUE, the simulation is repeated under the null hypothesis, with every arm's success rate set to the average of p, and the rejection rate is reported as type I error. The default is FALSE.
seed	An optional integer passed to set.seed before simulating, so that the results can be reproduced. The default NULL leaves the random number generator unchanged.

Details

Bai, Hu and Shen's urn can be described as follows. An urn initially contains balls of K types, where balls of types $1, 2, \dots, K$ represent treatments $1, 2, \dots, K$. A ball is drawn at random from the urn and, if it is of type k , the next patient is assigned to treatment k . After the response is observed, the urn composition is updated. A success on treatment k adds one ball of type k to the urn. A failure on treatment k adds $\hat{p}_j/(\hat{M} - \hat{p}_k)$ balls of each other type $j \neq k$, where $\hat{p}_j = (S_j + 1)/(N_j + 1)$ is the current estimate of the success rate of treatment j (S_j successes among N_j patients) and $\hat{M} = \hat{p}_1 + \dots + \hat{p}_K$. The estimates use the responses of the previous patients only. This is adaptive design 3 of Bai, Hu and Shen (2002), the design proposed in the paper. (Versions of `grouprar` before 0.2.0 used the true success rates instead, which is their design 2.)

Value

An object of class "`grouprar`", a list that is printed as a short summary (see `print.grouprar`), with the following elements.

<code>method</code>	The name of the procedure.
<code>sample size</code>	The total sample size.
<code>parameter</code>	The true success rates used in the simulations, named <code>pA</code> , <code>pB</code> , ...
<code>propotion</code>	The mean allocation proportion of each arm over the simulations, named <code>treatment A</code> , <code>treatment B</code> , ...
<code>sd of propotion</code>	The standard deviation of the allocation proportion of each arm over the simulations.
<code>failure rate</code>	The mean failure rate over the simulations.
<code>sd of failure rate</code>	The standard deviation of the failure rate (or mean response) over the simulations.
<code>power</code>	The proportion of simulated trials that reject the null hypothesis of equal success rates. Simulations in which the test cannot be computed are dropped.
<code>data: failureRate</code>	The failure rate (or mean response) of each simulated trial.
<code>data: test</code>	The test decision of each simulated trial (1 = reject).
<code>data: assignment</code>	The treatment assignments of the last simulated trial.
<code>data: propotion</code>	A data frame with the allocation proportions of each simulated trial.
<code>data: allocation</code>	An <code>nsim</code> by <code>ssn</code> matrix with the treatment assignments of every simulated trial.
<code>type I error</code>	Only if <code>typeI = TRUE</code> . The rejection rate under the null hypothesis.

References

Bai, Z. D., Hu, F. and Shen, L. (2002). *An adaptive design for multi-arm clinical trials*. Journal of Multivariate Analysis, 81(1), 1-18. doi:10.1006/jmva.2001.1987

Examples

```
## a simple use
bhs.res <- Bai.Hu.Shen.Urn(k = 3,
                           p = c(0.7, 0.8, 0.6),
                           ssn = 200,
                           Y0 = NULL,
                           nsim = 100,
                           alpha = 0.05)

## view the output
bhs.res

## view all simulation settings
bhs.res[["method"]]
bhs.res[["parameter"]]

## view the simulation results
bhs.res[["propotion"]]
bhs.res[["failure rate"]]
bhs.res[["power"]]
bhs.res[["data: assignment"]]
```

BirthDeathUrn

*Birth and Death Urn***Description**

Simulating the birth and death urn procedure (number of arms ≥ 2) with two-sided hypothesis testing in a clinical trial context.

Usage

```
BirthDeathUrn(k, p, ssn, Y0 = NULL, nsim = 2000, alpha = 0.05,
               test.fun = NULL, typeI = FALSE, seed = NULL)
```

Arguments

k	A positive integer. The number of treatment groups in the trial ($k \geq 2$).
p	A vector of length k with values between 0 and 1. The true success rates of the treatments, used to generate data for the simulations.
ssn	A positive integer. The total number of participants in each simulated trial.
Y0	A vector of length k giving the initial number of treatment balls of each type in the urn. One immigration ball is added automatically. For instance, if $Y0 = c(1, 1, 1)$, the urn starts with one ball of each treatment and one immigration ball. If Y0 is NULL (default), it is set to a vector of k ones.
nsim	A positive integer. The number of simulated trials, with a default value of 2000.

<code>alpha</code>	A number between 0 and 1. The significance level of the two-sided test, with a default value of 0.05.
<code>test.fun</code>	An optional function <code>function(outcome, assignment)</code> that returns the p-value of a test for one simulated trial, where <code>outcome</code> holds the observed responses and <code>assignment</code> the treatment labels (integers 1 to k); missing responses are removed first. The null hypothesis is rejected when the p-value is at most <code>alpha</code> . The default NULL uses the built-in two-sided tests: the t-test for two arms and the Wald chi-squared test of equal means for more than two arms (for binary responses, if some arms have no variability, the variances are computed from $(S_k + 1)/(n_k + 2)$ as in Agresti and Caffo, 2000).
<code>typeI</code>	Logical. If TRUE, the simulation is repeated under the null hypothesis, with every arm's success rate set to the average of <code>p</code> , and the rejection rate is reported as type I error. The default is FALSE.
<code>seed</code>	An optional integer passed to <code>set.seed</code> before simulating, so that the results can be reproduced. The default NULL leaves the random number generator unchanged.

Details

The birth and death urn works as follows. Initially the urn contains balls of K treatment types and an immigration ball. A ball is drawn at random with replacement. If it is the immigration ball, one ball of each treatment type is added to the urn, no patient is treated, and the next ball is drawn. This is repeated until a type i ball ($i = 1, \dots, K$) is drawn, and then the patient is assigned to treatment i . After a success a type i ball is added to the urn, and after a failure a type i ball is removed (Hu and Rosenberger, 2006). More details can be found in Ivanova et al. (2000).

When the best success rate is at least 1/2 the urn concentrates on that arm, and the other arms can end with very few patients. The asymptotic tests can then be well above their nominal level under the null hypothesis, especially for $K \geq 3$. Use `typeI = TRUE` to check the type I error.

Value

An object of class "grouprar", a list that is printed as a short summary (see `print.grouprar`), with the following elements.

<code>method</code>	The name of the procedure.
<code>sample size</code>	The total sample size.
<code>parameter</code>	The true success rates used in the simulations, named <code>pA</code> , <code>pB</code> , ...
<code>propotion</code>	The mean allocation proportion of each arm over the simulations, named <code>treatment A</code> , <code>treatment B</code> , ...
<code>sd of propotion</code>	The standard deviation of the allocation proportion of each arm over the simulations.
<code>failure rate</code>	The mean failure rate over the simulations.
<code>sd of failure rate</code>	The standard deviation of the failure rate (or mean response) over the simulations.
<code>power</code>	The proportion of simulated trials that reject the null hypothesis of equal success rates. Simulations in which the test cannot be computed are dropped.

data: failureRate The failure rate (or mean response) of each simulated trial.

data: test The test decision of each simulated trial (1 = reject).

data: assignment The treatment assignments of the last simulated trial.

data: propotion A data frame with the allocation proportions of each simulated trial.

data: allocation An nsim by ssn matrix with the treatment assignments of every simulated trial.

type I error Only if typeI = TRUE. The rejection rate under the null hypothesis.

References

Hu, F. and Rosenberger, W. F. (2006). *The Theory of Response-Adaptive Randomization in Clinical Trials*. John Wiley & Sons.

Ivanova, A., Rosenberger, W. F., Durham, S. D. and Flournoy, N. (2000). *A birth and death urn for randomized clinical trials: asymptotic methods*. Sankhya, Series B, 62(1), 104-118.

Examples

```
## a simple use
bd.res <- BirthDeathUrn(k = 3, p = c(0.6, 0.7, 0.6), ssn = 200, Y0 = NULL,
                        nsim = 100, alpha = 0.05)

## view the output
bd.res

## view all simulation settings
bd.res[["method"]]
bd.res[["parameter"]]

## view the simulation results
bd.res[["propotion"]]
bd.res[["failure rate"]]
bd.res[["power"]]
bd.res[["data: assignment"]]
```

CRDesign

Complete Randomization

Description

Simulating complete randomization with two-sided hypothesis testing in a clinical trial context.

Usage

```
CRDesign(k, p, ssn, nsim = 2000, alpha = 0.05, test.fun = NULL, typeI = FALSE,
         seed = NULL)
```

Arguments

<code>k</code>	A positive integer. The number of treatment groups in the trial ($k \geq 2$).
<code>p</code>	A vector of length <code>k</code> with values between 0 and 1. The true success rates of the treatments, used to generate data for the simulations.
<code>ssn</code>	A positive integer. The total number of participants in each simulated trial.
<code>nsim</code>	A positive integer. The number of simulated trials, with a default value of 2000.
<code>alpha</code>	A number between 0 and 1. The significance level of the two-sided test, with a default value of 0.05.
<code>test.fun</code>	An optional function <code>function(outcome, assignment)</code> that returns the p-value of a test for one simulated trial, where <code>outcome</code> holds the observed responses and <code>assignment</code> the treatment labels (integers 1 to <code>k</code>); missing responses are removed first. The null hypothesis is rejected when the p-value is at most <code>alpha</code> . The default NULL uses the built-in two-sided tests: the t-test for two arms and the Wald chi-squared test of equal means for more than two arms (for binary responses, if some arms have no variability, the variances are computed from $(S_k + 1)/(n_k + 2)$ as in Agresti and Caffo, 2000).
<code>typeI</code>	Logical. If TRUE, the simulation is repeated under the null hypothesis, with every arm's success rate set to the average of <code>p</code> , and the rejection rate is reported as type I error. The default is FALSE.
<code>seed</code>	An optional integer passed to <code>set.seed</code> before simulating, so that the results can be reproduced. The default NULL leaves the random number generator unchanged.

Details

Complete randomization assigns each participant to one of the treatment groups with equal probability, independently of all other participants.

Value

An object of class "grouprar", a list that is printed as a short summary (see `print.grouprar`), with the following elements.

<code>method</code>	The name of the procedure.
<code>sample size</code>	The total sample size.
<code>parameter</code>	The true success rates used in the simulations, named <code>pA</code> , <code>pB</code> , ...
<code>propotion</code>	The mean allocation proportion of each arm over the simulations, named <code>treatment A</code> , <code>treatment B</code> , ...
<code>sd of propotion</code>	The standard deviation of the allocation proportion of each arm over the simulations.
<code>failure rate</code>	The mean failure rate over the simulations.
<code>sd of failure rate</code>	The standard deviation of the failure rate (or mean response) over the simulations.

power	The proportion of simulated trials that reject the null hypothesis of equal success rates. Simulations in which the test cannot be computed are dropped.
data: failureRate	The failure rate (or mean response) of each simulated trial.
data: test	The test decision of each simulated trial (1 = reject).
data: assignment	The treatment assignments of the last simulated trial.
data: propotion	A data frame with the allocation proportions of each simulated trial.
data: allocation	An nsim by ssn matrix with the treatment assignments of every simulated trial.
type I error	Only if typeI = TRUE. The rejection rate under the null hypothesis.

Examples

```
## a simple use
CR.res <- CRDesign(k = 3, p = c(0.7, 0.8, 0.6), ssn = 400, nsim = 100)
## view the output
CR.res

## view all simulation settings
CR.res[["method"]]
CR.res[["parameter"]]
## view the simulation results
CR.res[["propotion"]]
CR.res[["failure rate"]]
CR.res[["power"]]
CR.res[["data: assignment"]]

## estimate the type I error as well, with a reproducible seed
res <- CRDesign(k = 2, p = c(0.6, 0.8), ssn = 100, nsim = 50, typeI = TRUE, seed = 1)
summary(res)
```

Description

Simulating Hu and Zhang's doubly biased coin design with binary response (number of arms ≥ 2) in a clinical trial context. (Inference: two-sided t-test for two arms or chi-square test for more than two arms.)

Usage

```
DBCD_Bin(n0 = 20, p, k, ssn, theta0 = NULL, target.alloc = "RPW", r = 2,
         nsim = 2000, mRate = NULL, alpha = 0.05, allocation = "DBCD",
         erade.alpha = 0.5, lower.bound = 0, monitor = NULL, test.fun = NULL,
         typeI = FALSE, seed = NULL)
```

Arguments

<code>n0</code>	A positive integer and a multiple of k . The number of initial patients assigned by restricted randomization for initial parameter estimation.
<code>p</code>	A vector of length k with values between 0 and 1. The true success rates of the treatments, used to generate data for the simulations.
<code>k</code>	A positive integer. The number of treatment groups in the trial ($k \geq 2$).
<code>ssn</code>	A positive integer. The total number of participants in each simulated trial.
<code>theta0</code>	A vector of length k used to smooth the success-rate estimates, $\hat{p}_k = (S_k + \theta_{0k}) / (N_k + 1)$, where S_k and N_k are the number of successes and patients on treatment k . If NULL (default), all values are 0.5.
<code>target.alloc</code>	Desired allocation proportion. One of "Neyman", "RSIHR", "RPW", "WeisUrn", "OptimalNeyman" or "OptimalRSIHR". The default is "RPW". See Details.
<code>r</code>	A non-negative number. The tuning parameter of Hu and Zhang's allocation function, used when <code>allocation = "DBCD"</code> . Values between 2 and 4 are common. The default value is 2.
<code>nsim</code>	A positive integer. The number of simulated trials, with a default value of 2000.
<code>mRate</code>	A number between 0 and 1 giving the probability that a response is missing. Missingness is simulated completely at random (MCAR), and missing responses are excluded from estimation and testing. The default NULL means no missing responses. As in Zhai et al. (2024), missing responses are excluded from the estimates, the failure rate and the test, while the allocation proportions (both those used by the allocation function and the reported ones) count all enrolled patients.
<code>alpha</code>	A number between 0 and 1. The significance level of the two-sided test, with a default value of 0.05.
<code>allocation</code>	The allocation function. "DBCD" (default) uses the allocation function of Hu and Zhang (2004) with parameter r . "ERADE" uses the efficient randomized adaptive design of Hu, Zhang and He (2009), in its multi-arm version (Alkhnefr, Hu and Zhai, 2025), with parameter <code>erade.alpha</code> . See Details.
<code>erade.alpha</code>	A number between 0 and 1. The degree of randomization of ERADE, used when <code>allocation = "ERADE"</code> . Smaller values push the allocation more strongly towards the target, and values of 1/2 or 2/3 are recommended. The default is 0.5.
<code>lower.bound</code>	A number between 0 and $1/k$. The smallest allowed target proportion of each arm for the targets "OptimalNeyman" and "OptimalRSIHR". The default is 0.
<code>monitor</code>	An optional object created by sqMonitor for group sequential monitoring of a two-arm trial. For each information time t , a look is taken after $\text{ceiling}(t * \text{ssn})$ patients, and the trial stops for efficacy as soon as the Wald statistic

	crosses the boundary of <code>sqBoundary</code> . The first look must come after the <code>n0</code> initial patients. Cannot be combined with <code>test.fun</code> . The default NULL gives a fixed sample design. See Details.
<code>test.fun</code>	An optional function <code>function(outcome, assignment)</code> that returns the p-value of a test for one simulated trial, where <code>outcome</code> holds the observed responses and <code>assignment</code> the treatment labels (integers 1 to k); missing responses are removed first. The null hypothesis is rejected when the p-value is at most α . The default NULL uses the built-in two-sided tests: the t-test for two arms and the Wald chi-squared test of equal means for more than two arms (for binary responses, if some arms have no variability, the variances are computed from $(S_k + 1)/(n_k + 2)$ as in Agresti and Caffo, 2000). Cannot be combined with <code>monitor</code> .
<code>typeI</code>	Logical. If TRUE, the simulation is repeated under the null hypothesis, with every arm's success rate set to the average of p , and the rejection rate is reported as type I error. The default is FALSE.
<code>seed</code>	An optional integer passed to <code>set.seed</code> before simulating, so that the results can be reproduced. The default NULL leaves the random number generator unchanged.

Details

The objective of Hu and Zhang's doubly biased coin design is to allocate patients sequentially while closely approximating the desired allocation proportion, which is a function of certain unknown parameters related to the response variable under each treatment.

The process begins by assigning `n0` patients to treatment groups using restricted randomization and collecting their responses. Initial parameter estimates for the response variable are then obtained for each treatment group. Based on these estimates, the desired allocation proportion is calculated. Then Hu and Zhang's allocation function is applied to determine the probabilities for the next patient to be assigned to each treatment group, which drive the allocation proportion towards the desired one. This process is repeated sequentially for each patient until the predetermined number of patients has been allocated.

This methodology was introduced by Hu and Zhang (2004) in their paper 'Asymptotic properties of doubly adaptive biased coin designs for multitreatment clinical trials'.

Target allocations. With $q_k = 1 - p_k$, "Neyman" and "RSIHR" are proportional to $\sqrt{p_k q_k}$ and $\sqrt{p_k}$. For two arms these are the Neyman allocation and the optimal allocation of Rosenberger et al. (2001), and for more arms they are simple generalizations. "RPW" and "WeisUrn" are proportional to $1/q_k$, the limiting allocation of the randomized play-the-winner rule and of Wei's urn. "OptimalNeyman" and "OptimalRSIHR" are the k -arm optimal allocations of Tymofyeyev, Rosenberger and Hu (2007). They minimize the total sample size and the expected number of failures, respectively, for a fixed noncentrality parameter of the chi-squared test of equal success rates, with every proportion at least `lower bound`. For two arms and `lower bound = 0` they equal "Neyman" and "RSIHR". For three or more arms the optimum without a lower bound can assign no patients to the middle arms, so a positive `lower bound` such as 0.1 is recommended.

Allocation functions. With `allocation = "ERADE"` and $\alpha = \text{erade.alpha}$, arm k receives the next patient (in group designs, each patient of the next group) with probability $\alpha \hat{p}_k$ if it is over-allocated ($N_k/m > \hat{p}_k$), $\hat{p}_k$ if it is on target, and $\hat{p}_k + (1 - \alpha) \sum_{j \in S} \hat{p}_j / |T|$ if it is under-allocated, where S and T are the sets of over- and under-allocated arms, N_k/m is the current allocation proportion and

$\hat{\rho}$ is the estimated target (Alkhnefr, Hu and Zhai, 2025, eq. (1)). For two arms this is the ERADE of Hu, Zhang and He (2009). ERADE attains the lower bound of the asymptotic variance of the allocation proportions, so its allocation is less variable than that of the DBCD.

Sequential monitoring. With `monitor = sqMonitor(t, spend)` (two arms only) the trial is analysed at the information times t . Looks are taken after `ceiling(t * ssn)` patients. At each look the Wald statistic $Z = (\hat{\theta}_1 - \hat{\theta}_2) / \sqrt{\hat{v}_1/n_1 + \hat{v}_2/n_2}$ is computed from all patients enrolled so far, with $\hat{p}_k(1 - \hat{p}_k)$ as variance estimates (missing responses are excluded), and the trial stops and rejects the null hypothesis as soon as $|Z|$ reaches the boundary of `sqBoundary`. Zhu and Hu (2010) showed that under the DBCD the sequential statistics are asymptotically a Brownian motion in the information time, so alpha spending boundaries keep the type I error. Their theory covers two arms and the DBCD, and the same boundaries are used with ERADE. The result then also contains the stopping probability at each look and the expected sample size.

Value

An object of class "grouprar", a list that is printed as a short summary (see `print.grouprar`), with the following elements.

<code>method</code>	The name of the procedure.
<code>sample size</code>	The total sample size.
<code>parameter</code>	The true success rates used in the simulations, named pA, pB, ...
<code>propotion</code>	The mean allocation proportion of each arm over the simulations, named treatment A, treatment B, ...
<code>sd of propotion</code>	The standard deviation of the allocation proportion of each arm over the simulations.
<code>failure rate</code>	The mean failure rate over the simulations.
<code>sd of failure rate</code>	The standard deviation of the failure rate (or mean response) over the simulations.
<code>power</code>	The proportion of simulated trials that reject the null hypothesis of equal success rates. Simulations in which the test cannot be computed are dropped.
<code>data: failureRate</code>	The failure rate (or mean response) of each simulated trial.
<code>data: test</code>	The test decision of each simulated trial (1 = reject).
<code>data: assignment</code>	The treatment assignments of the last simulated trial.
<code>data: propotion</code>	A data frame with the allocation proportions of each simulated trial.
<code>data: allocation</code>	An <code>nsim</code> by <code>ssn</code> matrix with the treatment assignments of every simulated trial (positions after an early stop are NA).
<code>type I error</code>	Only if <code>typeI = TRUE</code> . The rejection rate under the null hypothesis.
<code>boundary</code>	Only if <code>monitor</code> is given. The boundaries for $ Z $ at the looks.
<code>stopping probability</code>	Only if <code>monitor</code> is given. The proportion of simulated trials that stop at each look (the last look is the final analysis).

expected sample size
 Only if monitor is given. The mean number of enrolled patients.

data: stage Only if monitor is given. The look at which each simulated trial stopped.

data: sample size
 Only if monitor is given. The number of patients enrolled in each simulated trial.

References

- Alkhnefr, N., Hu, F. and Zhai, G. (2025). *Efficient randomized adaptive designs for multi-arm clinical trials*. Statistical Methods in Medical Research, 34(9), 1886-1898. doi:[10.1177/09622802251362644](https://doi.org/10.1177/09622802251362644)
- Hu, F. and Zhang, L. X. (2004). *Asymptotic properties of doubly adaptive biased coin designs for multitreatment clinical trials*. The Annals of Statistics, 32(1), 268-301. doi:[10.1214/aos/1079120137](https://doi.org/10.1214/aos/1079120137)
- Hu, F., Zhang, L. X. and He, X. (2009). *Efficient randomized-adaptive designs*. The Annals of Statistics, 37(5A), 2543-2560. doi:[10.1214/08AOS655](https://doi.org/10.1214/08AOS655)
- Rosenberger, W. F., Stallard, N., Ivanova, A., Harper, C. N. and Ricks, M. L. (2001). *Optimal adaptive designs for binary response trials*. Biometrics, 57(3), 909-913. doi:[10.1111/j.0006-341X.2001.00909.x](https://doi.org/10.1111/j.0006-341X.2001.00909.x)
- Tymofyeyev, Y., Rosenberger, W. F. and Hu, F. (2007). *Implementing optimal allocation in sequential binary response experiments*. Journal of the American Statistical Association, 102(477), 224-234. doi:[10.1198/016214506000000906](https://doi.org/10.1198/016214506000000906)
- Zhu, H. and Hu, F. (2010). *Sequential monitoring of response-adaptive randomized clinical trials*. The Annals of Statistics, 38(4), 2218-2241. doi:[10.1214/10AOS796](https://doi.org/10.1214/10AOS796)

Examples

```
res <- DBCD_Bin(n0 = 20, p = c(0.7, 0.8), k = 2, ssn = 300, theta0 = NULL,
               target.alloc = "RPW", r = 2, nsim = 50, mRate = NULL, alpha = 0.05)
res

## ERADE instead of the DBCD allocation function
res.erade <- DBCD_Bin(n0 = 20, p = c(0.7, 0.8), k = 2, ssn = 200, target.alloc = "RSIHR",
                    nsim = 50, allocation = "ERADE", erade.alpha = 0.5, seed = 1)
res.erade

## three arms with the optimal RSIHR target of Tymofyeyev, Rosenberger and Hu (2007)
res.opt <- DBCD_Bin(n0 = 30, p = c(0.5, 0.7, 0.8), k = 3, ssn = 200,
                  target.alloc = "OptimalRSIHR", lower.bound = 0.1, nsim = 20, seed = 1)
res.opt[["propotion"]]

## group sequential monitoring with two interim looks (O'Brien-Fleming-type spending)
res.sq <- DBCD_Bin(n0 = 20, p = c(0.6, 0.8), k = 2, ssn = 200, target.alloc = "RSIHR",
                 nsim = 50, monitor = sqMonitor(c(1/3, 2/3)), seed = 1)
res.sq[["stopping probability"]]
res.sq[["expected sample size"]]

## a user-supplied test: Fisher's exact test
fisher <- function(outcome, assignment)
```

```

    fisher.test(table(factor(assignment, 1:2), factor(outcome, 0:1)))$p.value
res.f <- DBCD_Bin(n0 = 20, p = c(0.6, 0.8), k = 2, ssn = 100, nsim = 20,
                 test.fun = fisher, seed = 1)
res.f[["power"]]

```

DBCD_Cont

Hu and Zhang's Doubly Biased Coin Design with Continuous Response Type

Description

Simulating Hu and Zhang's doubly biased coin design with continuous response (number of arms ≥ 2) in a clinical trial context. (Inference: two-sided t-test for two arms or chi-square test for more than two arms.)

Usage

```

DBCD_Cont(n0 = 20, theta, k, ssn, theta0 = NULL, target.alloc = "Neyman",
          r = 2, nsim = 2000, alpha = 0.05, allocation = "DBCD",
          erade.alpha = 0.5, lower.bound = 0, monitor = NULL, test.fun = NULL,
          typeI = FALSE, seed = NULL)

```

Arguments

n0	A positive integer and a multiple of k. The number of initial patients assigned by restricted randomization for initial parameter estimation.
theta	A numerical vector of length 2k giving the true mean and variance of each treatment, used to generate data for the simulations. For example, with $k = 2$ use $\text{theta} = \text{c}(13, 4.0^2, 15, 2.5^2)$.
k	A positive integer. The number of treatment groups in the trial ($k \geq 2$).
ssn	A positive integer. The total number of participants in each simulated trial.
theta0	Currently unused. Kept for backward compatibility.
target.alloc	Desired allocation proportion. One of "Neyman", "ZR", "OptimalNeyman" or "DaOptimal". The default is "Neyman". See Details.
r	A non-negative number. The tuning parameter of Hu and Zhang's allocation function, used when <code>allocation = "DBCD"</code> . Values between 2 and 4 are common. The default value is 2.
nsim	A positive integer. The number of simulated trials, with a default value of 2000.
alpha	A number between 0 and 1. The significance level of the two-sided test, with a default value of 0.05.
allocation	The allocation function. "DBCD" (default) uses the allocation function of Hu and Zhang (2004) with parameter r. "ERADE" uses the efficient randomized adaptive design of Hu, Zhang and He (2009), in its multi-arm version (Alkhnefr, Hu and Zhai, 2025), with parameter <code>erade.alpha</code> . See Details.

<code>erade.alpha</code>	A number between 0 and 1. The degree of randomization of ERADE, used when <code>allocation = "ERADE"</code> . Smaller values push the allocation more strongly towards the target, and values of 1/2 or 2/3 are recommended. The default is 0.5.
<code>lower.bound</code>	A number between 0 and $1/k$. The smallest allowed target proportion of each arm for the targets "ZR" and "OptimalNeyman". The default is 0.
<code>monitor</code>	An optional object created by <code>sqMonitor</code> for group sequential monitoring of a two-arm trial. For each information time t , a look is taken after $\text{ceiling}(t * \text{ssn})$ patients, and the trial stops for efficacy as soon as the Wald statistic crosses the boundary of <code>sqBoundary</code> . The first look must come after the n_0 initial patients. Cannot be combined with <code>test.fun</code> . The default NULL gives a fixed sample design. See Details.
<code>test.fun</code>	An optional function <code>function(outcome, assignment)</code> that returns the p-value of a test for one simulated trial, where <code>outcome</code> holds the observed responses and <code>assignment</code> the treatment labels (integers 1 to k); missing responses are removed first. The null hypothesis is rejected when the p-value is at most α . The default NULL uses the built-in two-sided tests: the t-test for two arms and the Wald chi-squared test of equal means for more than two arms (for binary responses, if some arms have no variability, the variances are computed from $(S_k + 1)/(n_k + 2)$ as in Agresti and Caffo, 2000). Cannot be combined with <code>monitor</code> .
<code>typeI</code>	Logical. If TRUE, the simulation is repeated under the null hypothesis, with the mean of every arm set to the average of the means in <code>theta</code> and the variance of each arm kept, and the rejection rate is reported as type I error. The default is FALSE.
<code>seed</code>	An optional integer passed to <code>set.seed</code> before simulating, so that the results can be reproduced. The default NULL leaves the random number generator unchanged.

Details

The objective of Hu and Zhang's doubly biased coin design is to allocate patients sequentially while closely approximating the desired allocation proportion, which is a function of certain unknown parameters related to the response variable under each treatment.

The process begins by assigning n_0 patients to treatment groups using restricted randomization and collecting their responses. Initial parameter estimates for the response variable are then obtained for each treatment group. Based on these estimates, the desired allocation proportion is calculated. Then Hu and Zhang's allocation function is applied to determine the probabilities for the next patient to be assigned to each treatment group, which drive the allocation proportion towards the desired one. This process is repeated sequentially for each patient until the predetermined number of patients has been allocated.

This methodology was introduced by Hu and Zhang (2004) in their paper 'Asymptotic properties of doubly adaptive biased coin designs for multitreatment clinical trials'.

Target allocations. "Neyman" is proportional to σ_k . "ZR" is the allocation of Zhang and Rosenberger (2006), which minimizes the total expected response (smaller responses are better, and all means must be positive). For two arms it is their rule (7): $\rho_1 = \sigma_1 \sqrt{\mu_2} / (\sigma_1 \sqrt{\mu_2} + \sigma_2 \sqrt{\mu_1})$ when this assigns more patients to the arm with the smaller mean, and 1/2 otherwise. For three or

more arms it minimizes the total expected response for a fixed noncentrality parameter of the chi-squared test of equal means, with every proportion at least lower.bound, the continuous analogue of Tymofyeyev, Rosenberger and Hu (2007). "OptimalNeyman" minimizes the total sample size under the same constraints and equals "Neyman" for two arms with lower.bound = 0. For three or more arms a positive lower.bound such as 0.1 is recommended with "ZR" and "OptimalNeyman". "DaOptimal" is proportional to $\sigma_k^{4/3}$. While the target cannot be estimated (for example with fewer than two responses in an arm) equal allocation is used.

Allocation functions. With allocation = "ERADE" and $\alpha = \text{erade.alpha}$, arm k receives the next patient (in group designs, each patient of the next group) with probability $\alpha \hat{\rho}_k$ if it is over-allocated ($N_k/m > \hat{\rho}_k$), $\hat{\rho}_k$ if it is on target, and $\hat{\rho}_k + (1 - \alpha) \sum_{j \in S} \hat{\rho}_j / |T|$ if it is under-allocated, where S and T are the sets of over- and under-allocated arms, N_k/m is the current allocation proportion and $\hat{\rho}$ is the estimated target (Alkhnefr, Hu and Zhai, 2025, eq. (1)). For two arms this is the ERADE of Hu, Zhang and He (2009). ERADE attains the lower bound of the asymptotic variance of the allocation proportions, so its allocation is less variable than that of the DBCD.

Sequential monitoring. With monitor = sqMonitor(t, spend) (two arms only) the trial is analysed at the information times t . Looks are taken after ceiling($t * \text{ssn}$) patients. At each look the Wald statistic $Z = (\hat{\theta}_1 - \hat{\theta}_2) / \sqrt{\hat{v}_1/n_1 + \hat{v}_2/n_2}$ is computed from all patients enrolled so far, with the sample variances as variance estimates (missing responses are excluded), and the trial stops and rejects the null hypothesis as soon as $|Z|$ reaches the boundary of sqBoundary. Zhu and Hu (2010) showed that under the DBCD the sequential statistics are asymptotically a Brownian motion in the information time, so alpha spending boundaries keep the type I error. Their theory covers two arms and the DBCD, and the same boundaries are used with ERADE. The result then also contains the stopping probability at each look and the expected sample size.

Value

An object of class "grouprar", a list that is printed as a short summary (see print.grouprar), with the following elements.

method	The name of the procedure.
sample size	The total sample size.
parameter	The true means and variances used in the simulations, named muA, sigma2A, muB, ...
propotion	The mean allocation proportion of each arm over the simulations, named treatment A, treatment B, ...
sd of propotion	The standard deviation of the allocation proportion of each arm over the simulations.
failure rate	The mean response over the simulations (for continuous responses this element holds the mean response, not a failure rate).
sd of failure rate	The standard deviation of the failure rate (or mean response) over the simulations.
power	The proportion of simulated trials that reject the null hypothesis of equal means. Simulations in which the test cannot be computed are dropped.
data: failureRate	The failure rate (or mean response) of each simulated trial.

data: test	The test decision of each simulated trial (1 = reject).
data: assignment	The treatment assignments of the last simulated trial.
data: propotion	A data frame with the allocation proportions of each simulated trial.
data: allocation	An nsim by ssn matrix with the treatment assignments of every simulated trial (positions after an early stop are NA).
type I error	Only if typeI = TRUE. The rejection rate under the null hypothesis.
boundary	Only if monitor is given. The boundaries for $ Z $ at the looks.
stopping probability	Only if monitor is given. The proportion of simulated trials that stop at each look (the last look is the final analysis).
expected sample size	Only if monitor is given. The mean number of enrolled patients.
data: stage	Only if monitor is given. The look at which each simulated trial stopped.
data: sample size	Only if monitor is given. The number of patients enrolled in each simulated trial.

References

- Alkhnefr, N., Hu, F. and Zhai, G. (2025). *Efficient randomized adaptive designs for multi-arm clinical trials*. Statistical Methods in Medical Research, 34(9), 1886-1898. doi:[10.1177/09622802251362644](https://doi.org/10.1177/09622802251362644)
- Hu, F. and Zhang, L. X. (2004). *Asymptotic properties of doubly adaptive biased coin designs for multitreatment clinical trials*. The Annals of Statistics, 32(1), 268-301. doi:[10.1214/aos/1079120137](https://doi.org/10.1214/aos/1079120137)
- Hu, F., Zhang, L. X. and He, X. (2009). *Efficient randomized-adaptive designs*. The Annals of Statistics, 37(5A), 2543-2560. doi:[10.1214/08AOS655](https://doi.org/10.1214/08AOS655)
- Hu, F., Zhang, L. X., Cheung, S. H. and Chan, W. S. (2008). *Doubly adaptive biased coin designs with delayed responses*. Canadian Journal of Statistics, 36(4), 541-559.
- Tymofyeyev, Y., Rosenberger, W. F. and Hu, F. (2007). *Implementing optimal allocation in sequential binary response experiments*. Journal of the American Statistical Association, 102(477), 224-234. doi:[10.1198/016214506000000906](https://doi.org/10.1198/016214506000000906)
- Zhang, L. and Rosenberger, W. F. (2006). *Response-adaptive randomization for clinical trials with continuous outcomes*. Biometrics, 62(2), 562-569. doi:[10.1111/j.15410420.2005.00496.x](https://doi.org/10.1111/j.15410420.2005.00496.x)
- Zhu, H. and Hu, F. (2010). *Sequential monitoring of response-adaptive randomized clinical trials*. The Annals of Statistics, 38(4), 2218-2241. doi:[10.1214/10AOS796](https://doi.org/10.1214/10AOS796)

See Also

See [DBCD_Bin](#) for simulations of Hu and Zhang's doubly biased coin design with binary response.

See [dyldDBCD_Cont](#) for simulations of Hu and Zhang's doubly biased coin design with delayed continuous response.

Examples

```
# A simple use
## Arguments for generating the simulated data
theta = c(13, 4.0^2, 15, 2.5^2)
k = 2
ssn = 88

res <- DBCD_Cont(n0 = 20, theta = theta, k = k, ssn = ssn, theta0 = NULL,
  target.alloc = "Neyman", r = 2, nsim = 100, alpha = 0.05)

# View the output (a list of all results)
res

## three arms with the k-arm ZR target and ERADE
res3 <- DBCD_Cont(n0 = 30, theta = c(13, 16, 15, 6.25, 14, 9), k = 3, ssn = 150,
  target.alloc = "ZR", lower.bound = 0.1, nsim = 30,
  allocation = "ERADE", seed = 1)

summary(res3)
```

DLRule

Drop the loser rule

Description

Simulating the drop-the-loser rule (number of arms ≥ 2) with two-sided hypothesis testing in a clinical trial context.

Usage

```
DLRule(k, p, ssn, Y0 = NULL, nsim = 2000, alpha = 0.05, test.fun = NULL,
  typeI = FALSE, seed = NULL)
```

Arguments

k	A positive integer. The number of treatment groups in the trial ($k \geq 2$).
p	A vector of length k with values between 0 and 1. The true success rates of the treatments, used to generate data for the simulations.
ssn	A positive integer. The total number of participants in each simulated trial.
Y0	A vector of length k giving the initial number of treatment balls of each type in the urn. One immigration ball is added automatically. For instance, if $Y0 = c(1, 1)$, the urn starts with one ball of each treatment and one immigration ball. If Y0 is NULL (default), it is set to a vector of k ones.
nsim	A positive integer. The number of simulated trials, with a default value of 2000.
alpha	A number between 0 and 1. The significance level of the two-sided test, with a default value of 0.05.

<code>test.fun</code>	An optional function <code>function(outcome, assignment)</code> that returns the p-value of a test for one simulated trial, where <code>outcome</code> holds the observed responses and <code>assignment</code> the treatment labels (integers 1 to k); missing responses are removed first. The null hypothesis is rejected when the p-value is at most <code>alpha</code> . The default NULL uses the built-in two-sided tests: the t-test for two arms and the Wald chi-squared test of equal means for more than two arms (for binary responses, if some arms have no variability, the variances are computed from $(S_k + 1)/(n_k + 2)$ as in Agresti and Caffo, 2000).
<code>typeI</code>	Logical. If TRUE, the simulation is repeated under the null hypothesis, with every arm's success rate set to the average of <code>p</code> , and the rejection rate is reported as type I error. The default is FALSE.
<code>seed</code>	An optional integer passed to <code>set.seed</code> before simulating, so that the results can be reproduced. The default NULL leaves the random number generator unchanged.

Details

The drop-the-loser rule can be described as follows. An urn initially contains balls of K treatment types and an immigration ball (type 0). A ball is drawn at random. If a treatment ball of type k is drawn, treatment k is assigned to the subject and the response is observed. If the response is a failure, the ball is not replaced, otherwise it is replaced. If the immigration ball is drawn, no treatment is assigned, and the ball is returned to the urn together with one ball of each treatment type. With two treatments A and B this is the rule of Ivanova (2003).

Value

An object of class "grouprar", a list that is printed as a short summary (see `print.grouprar`), with the following elements.

<code>method</code>	The name of the procedure.
<code>sample size</code>	The total sample size.
<code>parameter</code>	The true success rates used in the simulations, named <code>pA</code> , <code>pB</code> , ...
<code>propotion</code>	The mean allocation proportion of each arm over the simulations, named <code>treatment A</code> , <code>treatment B</code> , ...
<code>sd of propotion</code>	The standard deviation of the allocation proportion of each arm over the simulations.
<code>failure rate</code>	The mean failure rate over the simulations.
<code>sd of failure rate</code>	The standard deviation of the failure rate (or mean response) over the simulations.
<code>power</code>	The proportion of simulated trials that reject the null hypothesis of equal success rates. Simulations in which the test cannot be computed are dropped.
<code>data: failureRate</code>	The failure rate (or mean response) of each simulated trial.
<code>data: test</code>	The test decision of each simulated trial (1 = reject).

data: assignment The treatment assignments of the last simulated trial.

data: propotion A data frame with the allocation proportions of each simulated trial.

data: allocation An nsim by ssn matrix with the treatment assignments of every simulated trial.

type I error Only if typeI = TRUE. The rejection rate under the null hypothesis.

References

Ivanova, A. (2003). *A play-the-winner-type urn design with reduced variability*. *Metrika*, 58, 1-13.
[doi:10.1007/s001840200220](https://doi.org/10.1007/s001840200220)

Examples

```
## a simple use
dl.res <- DLRule(k = 2, p = c(0.7, 0.8), ssn = 200, Y0 = NULL, nsim = 100, alpha = 0.05)

## view the output
dl.res

## view all simulation settings
dl.res[["method"]]
dl.res[["parameter"]]

## view the simulation results
dl.res[["propotion"]]
dl.res[["failure rate"]]
dl.res[["power"]]
dl.res[["data: assignment"]]
```

dyldDBCD_Bin

Hu and Zhang's Doubly Biased Coin Design with Delayed Binary Response

Description

Simulating Hu and Zhang's doubly biased coin design with delayed binary response (number of arms ≥ 2) in a clinical trial context. (Inference: two-sided t-test for two arms or chi-square test for more than two arms.)

Usage

```
dyldDBCD_Bin(n0 = 20, p, k, ssn, ent.param, rspT.dist, rspT.param,
             theta0 = NULL, target.alloc = "RPW", r = 2, nsim = 2000,
             mRate = NULL, alpha = 0.05, allocation = "DBCD",
             erade.alpha = 0.5, lower.bound = 0, test.fun = NULL,
             typeI = FALSE, seed = NULL)
```

Arguments

<code>n0</code>	A positive integer and a multiple of k . The number of initial patients assigned by restricted randomization for initial parameter estimation.
<code>p</code>	A vector of length k with values between 0 and 1. The true success rates of the treatments, used to generate data for the simulations.
<code>k</code>	A positive integer. The number of treatment groups in the trial ($k \geq 2$).
<code>ssn</code>	A positive integer. The total number of participants in each simulated trial.
<code>ent.param</code>	A positive number. The mean time between consecutive patient arrivals. Inter-arrival times are drawn from an exponential distribution with this mean.
<code>rspT.dist</code>	The distribution of the time from enrollment until the response is observed. One of "exponential", "normal" or "uniform".
<code>rspT.param</code>	A numeric vector of parameters for the response-time distributions, one distribution per treatment and response, ordered as (treatment 1 failure, treatment 1 success, treatment 2 failure, treatment 2 success, ...). For "exponential" give the $2k$ means. For "normal" give $2k$ (mean, sd) pairs and for "uniform" give $2k$ (min, max) pairs, that is $4k$ values in total. Normal times below 0 are set to 0. For example, with $k = 2$ and exponential times, <code>rspT.param = c(3, 2, 4, 1)</code> gives mean times 3 and 2 for failures and successes on treatment 1, and 4 and 1 on treatment 2.
<code>theta0</code>	A vector of length k used to smooth the success-rate estimates, $\hat{p}_k = (S_k + \theta_{0k}) / (N_k + 1)$, where S_k and N_k are the number of observed successes and observed responses on treatment k . If NULL (default), all values are 0.5.
<code>target.alloc</code>	Desired allocation proportion. One of "Neyman", "RSIHR", "RPW", "WeisUrn", "OptimalNeyman" or "OptimalRSIHR". The default is "RPW". See Details.
<code>r</code>	A non-negative number. The tuning parameter of Hu and Zhang's allocation function, used when <code>allocation = "DBCD"</code> . Values between 2 and 4 are common. The default value is 2.
<code>nsim</code>	A positive integer. The number of simulated trials, with a default value of 2000.
<code>mRate</code>	A number between 0 and 1 giving the probability that a response is missing. Missingness is simulated completely at random (MCAR), and missing responses are never observed and are excluded from estimation and testing. The default NULL means no missing responses. As in Zhai et al. (2024), missing responses are excluded from the estimates, the failure rate and the test, while the allocation proportions (both those used by the allocation function and the reported ones) count all enrolled patients.
<code>alpha</code>	A number between 0 and 1. The significance level of the two-sided test, with a default value of 0.05.
<code>allocation</code>	The allocation function. "DBCD" (default) uses the allocation function of Hu and Zhang (2004) with parameter r . "ERADE" uses the efficient randomized adaptive design of Hu, Zhang and He (2009), in its multi-arm version (Alkhnefr, Hu and Zhai, 2025), with parameter <code>erade.alpha</code> . See Details.
<code>erade.alpha</code>	A number between 0 and 1. The degree of randomization of ERADE, used when <code>allocation = "ERADE"</code> . Smaller values push the allocation more strongly towards the target, and values of 1/2 or 2/3 are recommended. The default is 0.5.

lower.bound	A number between 0 and $1/k$. The smallest allowed target proportion of each arm for the targets "OptimalNeyman" and "OptimalRSIHR". The default is 0.
test.fun	An optional function <code>function(outcome, assignment)</code> that returns the p-value of a test for one simulated trial, where <code>outcome</code> holds the observed responses and <code>assignment</code> the treatment labels (integers 1 to k); missing responses are removed first. The null hypothesis is rejected when the p-value is at most α . The default NULL uses the built-in two-sided tests: the t-test for two arms and the Wald chi-squared test of equal means for more than two arms (for binary responses, if some arms have no variability, the variances are computed from $(S_k + 1)/(n_k + 2)$ as in Agresti and Caffo, 2000).
typeI	Logical. If TRUE, the simulation is repeated under the null hypothesis, with every arm's success rate set to the average of p , and the rejection rate is reported as type I error. The default is FALSE.
seed	An optional integer passed to <code>set.seed</code> before simulating, so that the results can be reproduced. The default NULL leaves the random number generator unchanged.

Details

Hu and Zhang's doubly biased coin design with delayed binary response uses the following allocation scheme.

(a) Initially, due to limited information about treatment efficacy, the first n_0 patients are assigned to the K treatments using restricted randomization (as described by Rosenberger and Lachin, 2002).

(b) For $m \geq n_0$, patient $(m + 1)$ is allocated to treatment k with probability $p_{m+1,k}$, which depends on the responses available at that time and on the estimated target allocation through the allocation function g_k proposed by Hu and Zhang (2004).

For a more comprehensive description of the procedure, please refer to the paper 'Doubly adaptive biased coin designs with delayed responses' by Hu et al. (2008).

Target allocations. With $q_k = 1 - p_k$, "Neyman" and "RSIHR" are proportional to $\sqrt{p_k q_k}$ and $\sqrt{p_k}$. For two arms these are the Neyman allocation and the optimal allocation of Rosenberger et al. (2001), and for more arms they are simple generalizations. "RPW" and "WeisUrn" are proportional to $1/q_k$, the limiting allocation of the randomized play-the-winner rule and of Wei's urn. "OptimalNeyman" and "OptimalRSIHR" are the k -arm optimal allocations of Tymofeyev, Rosenberger and Hu (2007). They minimize the total sample size and the expected number of failures, respectively, for a fixed noncentrality parameter of the chi-squared test of equal success rates, with every proportion at least `lower.bound`. For two arms and `lower.bound = 0` they equal "Neyman" and "RSIHR". For three or more arms the optimum without a lower bound can assign no patients to the middle arms, so a positive `lower.bound` such as 0.1 is recommended.

Allocation functions. With `allocation = "ERADE"` and $\alpha = \text{erade.alpha}$, arm k receives the next patient (in group designs, each patient of the next group) with probability $\alpha \hat{p}_k$ if it is over-allocated ($N_k/m > \hat{p}_k$), $\hat{p}_k$ if it is on target, and $\hat{p}_k + (1 - \alpha) \sum_{j \in S} \hat{p}_j / |T|$ if it is under-allocated, where S and T are the sets of over- and under-allocated arms, N_k/m is the current allocation proportion and $\hat{p}$ is the estimated target (Alkhnefr, Hu and Zhai, 2025, eq. (1)). For two arms this is the ERADE of Hu, Zhang and He (2009). ERADE attains the lower bound of the asymptotic variance of the allocation proportions, so its allocation is less variable than that of the DBCD.

Value

An object of class "grouprar", a list that is printed as a short summary (see `print.grouprar`), with the following elements.

method	The name of the procedure.
sample size	The total sample size (Total) and the expected number of patients with observed responses (Effective Size).
parameter	The true success rates used in the simulations, named pA, pB, ...
propotion	The mean allocation proportion of each arm over the simulations, named treatment A, treatment B, ...
sd of propotion	The standard deviation of the allocation proportion of each arm over the simulations.
failure rate	The mean failure rate over the simulations.
sd of failure rate	The standard deviation of the failure rate (or mean response) over the simulations.
power	The proportion of simulated trials that reject the null hypothesis of equal success rates. Simulations in which the test cannot be computed are dropped.
data: failureRate	The failure rate (or mean response) of each simulated trial.
data: test	The test decision of each simulated trial (1 = reject).
data: assignment	The treatment assignments of the last simulated trial.
data: propotion	A data frame with the allocation proportions of each simulated trial.
data: allocation	An nsim by ssn matrix with the treatment assignments of every simulated trial.
type I error	Only if typeI = TRUE. The rejection rate under the null hypothesis.
duration	The mean time from the first enrollment to the last observed response. Trials without any observed response have duration NA and are left out of the mean.
sd of duration	The standard deviation of the duration over the simulations.
enrollment duration	The mean time from the first to the last enrollment.
data: duration	The duration of each simulated trial.
data: enrollment	The enrollment duration of each simulated trial.

References

- Alkhnefr, N., Hu, F. and Zhai, G. (2025). *Efficient randomized adaptive designs for multi-arm clinical trials*. Statistical Methods in Medical Research, 34(9), 1886-1898. doi:10.1177/09622802251362644
- Hu, F. and Zhang, L. X. (2004). *Asymptotic properties of doubly adaptive biased coin designs for multitreatment clinical trials*. The Annals of Statistics, 32(1), 268-301. doi:10.1214/aos/1079120137

Hu, F., Zhang, L. X. and He, X. (2009). *Efficient randomized-adaptive designs*. The Annals of Statistics, 37(5A), 2543-2560. doi:10.1214/08AOS655

Hu, F., Zhang, L. X., Cheung, S. H. and Chan, W. S. (2008). *Doubly adaptive biased coin designs with delayed responses*. Canadian Journal of Statistics, 36(4), 541-559. doi:10.1002/cjs.5550360404

Rosenberger, W. F. and Lachin, J. M. (2002). *Randomization in Clinical Trials: Theory and Practice*. John Wiley & Sons.

Rosenberger, W. F., Stallard, N., Ivanova, A., Harper, C. N. and Ricks, M. L. (2001). *Optimal adaptive designs for binary response trials*. Biometrics, 57(3), 909-913. doi:10.1111/j.0006-341X.2001.00909.x

Tymofyeyev, Y., Rosenberger, W. F. and Hu, F. (2007). *Implementing optimal allocation in sequential binary response experiments*. Journal of the American Statistical Association, 102(477), 224-234. doi:10.1198/016214506000000906

Examples

```
# a simple use
## Arguments for generating the simulated data
### For response simulation
p = c(0.6, 0.8)
k = 2
ssn = 100
### for entry time and response time simulation
ent.param = 0.7
rspT.dist = "exponential"
rspT.param = c(1, 1, 3, 1)

## Arguments for the design
n0 = 20
target.alloc = "RSIHR"

res <- dyldDBCD_Bin(n0 = n0, p = p, k = k, ssn = ssn, ent.param = ent.param,
                    rspT.dist = rspT.dist, rspT.param = rspT.param, theta0 = NULL,
                    target.alloc = target.alloc, r = 2, nsim = 100,
                    mRate = NULL, alpha = 0.05)

res
```

dyldDBCD_Cont

Hu and Zhang's Doubly Biased Coin Design with Delayed Continuous Response

Description

Simulating Hu and Zhang's doubly biased coin design with delayed continuous response (number of arms ≥ 2) in a clinical trial context. (Inference: two-sided t-test for two arms or chi-square test for more than two arms.)

Usage

```
dyldDBCD_Cont(n0 = 20, theta, k, ssn, ent.param, rspT.dist, rspT.param,
  target.alloc = "Neyman", r = 2, nsim = 2000, mRate = NULL,
  alpha = 0.05, allocation = "DBCD", erade.alpha = 0.5,
  lower.bound = 0, test.fun = NULL, typeI = FALSE, seed = NULL)
```

Arguments

<code>n0</code>	A positive integer and a multiple of k . The number of initial patients assigned by restricted randomization for initial parameter estimation.
<code>theta</code>	A numerical vector of length $2k$ giving the true mean and variance of each treatment, used to generate data for the simulations. For example, with $k = 2$ use <code>theta = c(13, 4.0^2, 15, 2.5^2)</code> .
<code>k</code>	A positive integer. The number of treatment groups in the trial ($k \geq 2$).
<code>ssn</code>	A positive integer. The total number of participants in each simulated trial.
<code>ent.param</code>	A positive number. The mean time between consecutive patient arrivals. Inter-arrival times are drawn from an exponential distribution with this mean.
<code>rspT.dist</code>	The distribution of the time from enrollment until the response is observed. One of "exponential", "normal" or "uniform".
<code>rspT.param</code>	A numeric vector of parameters for the response-time distribution of each treatment. For "exponential" give the k means. For "normal" give k (mean, sd) pairs and for "uniform" give k (min, max) pairs, that is $2k$ values in total. Normal times below 0 are set to 0. For example, with 3 treatments and normal times with (mean, sd) pairs (3, 2), (2, 1) and (4, 1), use <code>rspT.param = c(3, 2, 2, 1, 4, 1)</code> .
<code>target.alloc</code>	Desired allocation proportion. One of "Neyman", "ZR", "OptimalNeyman" or "DaOptimal". The default is "Neyman". See Details.
<code>r</code>	A non-negative number. The tuning parameter of Hu and Zhang's allocation function, used when <code>allocation = "DBCD"</code> . Values between 2 and 4 are common. The default value is 2.
<code>nsim</code>	A positive integer. The number of simulated trials, with a default value of 2000.
<code>mRate</code>	A number between 0 and 1 giving the probability that a response is missing. Missingness is simulated completely at random (MCAR), and missing responses are never observed and are excluded from estimation and testing. The default NULL means no missing responses. As in Zhai et al. (2024), missing responses are excluded from the estimates, the failure rate and the test, while the allocation proportions (both those used by the allocation function and the reported ones) count all enrolled patients.
<code>alpha</code>	A number between 0 and 1. The significance level of the two-sided test, with a default value of 0.05.
<code>allocation</code>	The allocation function. "DBCD" (default) uses the allocation function of Hu and Zhang (2004) with parameter r . "ERADE" uses the efficient randomized adaptive design of Hu, Zhang and He (2009), in its multi-arm version (Alkhnefr, Hu and Zhai, 2025), with parameter <code>erade.alpha</code> . See Details.

<code>erade.alpha</code>	A number between 0 and 1. The degree of randomization of ERADE, used when <code>allocation = "ERADE"</code> . Smaller values push the allocation more strongly towards the target, and values of 1/2 or 2/3 are recommended. The default is 0.5.
<code>lower.bound</code>	A number between 0 and $1/k$. The smallest allowed target proportion of each arm for the targets "ZR" and "OptimalNeyman". The default is 0.
<code>test.fun</code>	An optional function <code>function(outcome, assignment)</code> that returns the p-value of a test for one simulated trial, where <code>outcome</code> holds the observed responses and <code>assignment</code> the treatment labels (integers 1 to k); missing responses are removed first. The null hypothesis is rejected when the p-value is at most α . The default NULL uses the built-in two-sided tests: the t-test for two arms and the Wald chi-squared test of equal means for more than two arms (for binary responses, if some arms have no variability, the variances are computed from $(S_k + 1)/(n_k + 2)$ as in Agresti and Caffo, 2000).
<code>typeI</code>	Logical. If TRUE, the simulation is repeated under the null hypothesis, with the mean of every arm set to the average of the means in <code>theta</code> and the variance of each arm kept, and the rejection rate is reported as type I error. The default is FALSE.
<code>seed</code>	An optional integer passed to <code>set.seed</code> before simulating, so that the results can be reproduced. The default NULL leaves the random number generator unchanged.

Details

Hu and Zhang's doubly biased coin design with delayed continuous response uses the following allocation scheme.

(a) Initially, due to limited information about treatment efficacy, the first n_0 patients are assigned to the K treatments using restricted randomization (as described by Rosenberger and Lachin, 2002).

(b) For $m \geq n_0$, patient $(m + 1)$ is allocated to treatment k with probability $p_{m+1,k}$, which depends on the responses available at that time and on the estimated target allocation through the allocation function g_k proposed by Hu and Zhang (2004).

For a more comprehensive description of the procedure, please refer to the paper 'Doubly adaptive biased coin designs with delayed responses' by Hu et al. (2008).

Target allocations. "Neyman" is proportional to σ_k . "ZR" is the allocation of Zhang and Rosenberger (2006), which minimizes the total expected response (smaller responses are better, and all means must be positive). For two arms it is their rule (7): $\rho_1 = \sigma_1 \sqrt{\mu_2} / (\sigma_1 \sqrt{\mu_2} + \sigma_2 \sqrt{\mu_1})$ when this assigns more patients to the arm with the smaller mean, and 1/2 otherwise. For three or more arms it minimizes the total expected response for a fixed noncentrality parameter of the chi-squared test of equal means, with every proportion at least `lower.bound`, the continuous analogue of Tymofeyev, Rosenberger and Hu (2007). "OptimalNeyman" minimizes the total sample size under the same constraints and equals "Neyman" for two arms with `lower.bound` = 0. For three or more arms a positive `lower.bound` such as 0.1 is recommended with "ZR" and "OptimalNeyman". "DaOptimal" is proportional to $\sigma_k^{4/3}$. While the target cannot be estimated (for example with fewer than two responses in an arm) equal allocation is used.

Allocation functions. With `allocation = "ERADE"` and $\alpha = \text{erade.alpha}$, arm k receives the next patient (in group designs, each patient of the next group) with probability $\alpha \hat{\rho}_k$ if it is over-allocated ($N_k/m > \hat{\rho}_k$), $\hat{\rho}_k$ if it is on target, and $\hat{\rho}_k + (1 - \alpha) \sum_{j \in S} \hat{\rho}_j / |T|$ if it is under-allocated, where S

and T are the sets of over- and under-allocated arms, N_k/m is the current allocation proportion and $\hat{\rho}$ is the estimated target (Alkhnefr, Hu and Zhai, 2025, eq. (1)). For two arms this is the ERADE of Hu, Zhang and He (2009). ERADE attains the lower bound of the asymptotic variance of the allocation proportions, so its allocation is less variable than that of the DBCD.

Value

An object of class "grouprar", a list that is printed as a short summary (see `print.grouprar`), with the following elements.

method	The name of the procedure.
sample size	The total sample size (Total) and the expected number of patients with observed responses (Effective Size).
parameter	The true means and variances used in the simulations, named muA, sigma2A, muB, ...
propotion	The mean allocation proportion of each arm over the simulations, named treatment A, treatment B, ...
sd of propotion	The standard deviation of the allocation proportion of each arm over the simulations.
failure rate	The mean response over the simulations (for continuous responses this element holds the mean response, not a failure rate).
sd of failure rate	The standard deviation of the failure rate (or mean response) over the simulations.
power	The proportion of simulated trials that reject the null hypothesis of equal means. Simulations in which the test cannot be computed are dropped.
data: failureRate	The failure rate (or mean response) of each simulated trial.
data: test	The test decision of each simulated trial (1 = reject).
data: assignment	The treatment assignments of the last simulated trial.
data: propotion	A data frame with the allocation proportions of each simulated trial.
data: allocation	An nsim by ssn matrix with the treatment assignments of every simulated trial.
type I error	Only if typeI = TRUE. The rejection rate under the null hypothesis.
duration	The mean time from the first enrollment to the last observed response. Trials without any observed response have duration NA and are left out of the mean.
sd of duration	The standard deviation of the duration over the simulations.
enrollment duration	The mean time from the first to the last enrollment.
data: duration	The duration of each simulated trial.
data: enrollment	The enrollment duration of each simulated trial.

References

- Alkhnefr, N., Hu, F. and Zhai, G. (2025). *Efficient randomized adaptive designs for multi-arm clinical trials*. Statistical Methods in Medical Research, 34(9), 1886-1898. doi:[10.1177/09622802251362644](https://doi.org/10.1177/09622802251362644)
- Hu, F. and Zhang, L. X. (2004). *Asymptotic properties of doubly adaptive biased coin designs for multitreatment clinical trials*. The Annals of Statistics, 32(1), 268-301. doi:[10.1214/aos/1079120137](https://doi.org/10.1214/aos/1079120137)
- Hu, F., Zhang, L. X. and He, X. (2009). *Efficient randomized-adaptive designs*. The Annals of Statistics, 37(5A), 2543-2560. doi:[10.1214/08AOS655](https://doi.org/10.1214/08AOS655)
- Hu, F., Zhang, L. X., Cheung, S. H. and Chan, W. S. (2008). *Doubly adaptive biased coin designs with delayed responses*. Canadian Journal of Statistics, 36(4), 541-559. doi:[10.1002/cjs.5550360404](https://doi.org/10.1002/cjs.5550360404)
- Rosenberger, W. F. and Lachin, J. M. (2002). *Randomization in Clinical Trials: Theory and Practice*. John Wiley & Sons.
- Tymofeyev, Y., Rosenberger, W. F. and Hu, F. (2007). *Implementing optimal allocation in sequential binary response experiments*. Journal of the American Statistical Association, 102(477), 224-234. doi:[10.1198/016214506000000906](https://doi.org/10.1198/016214506000000906)
- Zhang, L. and Rosenberger, W. F. (2006). *Response-adaptive randomization for clinical trials with continuous outcomes*. Biometrics, 62(2), 562-569. doi:[10.1111/j.15410420.2005.00496.x](https://doi.org/10.1111/j.15410420.2005.00496.x)

Examples

```
# a simple use
## Arguments for generating the simulated data
### For response simulation
theta = c(13, 4.0^2, 15, 2.5^2)
k = 2
ssn = 88

### for entry time and response time simulation
ent.param = 5
rspT.param = rep(10, 2)
rspT.dist = "exponential"

## Arguments for the design
target.alloc = "Neyman"

res <- dyldBCD_Cont(n0 = 10, theta = theta, k = k, ssn = ssn, ent.param = ent.param,
                   rspT.dist = rspT.dist, rspT.param = rspT.param,
                   target.alloc = target.alloc, r = 2, nsim = 100,
                   mRate = 0.2, alpha = 0.05)

# View the output (a list of all results)
res
```

GDLRule

*Generalized drop-the-loser rule***Description**

Simulating the generalized drop-the-loser rule (number of arms ≥ 2) with two-sided hypothesis testing in a clinical trial context.

Usage

```
GDLRule(k, p, ssn, aK, Y0 = NULL, nsim = 2000, alpha = 0.05, test.fun = NULL,
        typeI = FALSE, seed = NULL)
```

Arguments

k	A positive integer. The number of treatment groups in the trial ($k \geq 2$).
p	A vector of length k with values between 0 and 1. The true success rates of the treatments, used to generate data for the simulations.
ssn	A positive integer. The total number of participants in each simulated trial.
aK	A positive vector of length k. The number of balls of each treatment type added to the urn when the immigration ball is drawn.
Y0	A vector of length k giving the initial number of treatment balls of each type in the urn. One immigration ball is added automatically. For instance, if $Y0 = c(1, 1, 1)$, the urn starts with one ball of each treatment and one immigration ball. If Y0 is NULL (default), it is set to a vector of k ones.
nsim	A positive integer. The number of simulated trials, with a default value of 2000.
alpha	A number between 0 and 1. The significance level of the two-sided test, with a default value of 0.05.
test.fun	An optional function function(outcome, assignment) that returns the p-value of a test for one simulated trial, where outcome holds the observed responses and assignment the treatment labels (integers 1 to k); missing responses are removed first. The null hypothesis is rejected when the p-value is at most alpha. The default NULL uses the built-in two-sided tests: the t-test for two arms and the Wald chi-squared test of equal means for more than two arms (for binary responses, if some arms have no variability, the variances are computed from $(S_k + 1)/(n_k + 2)$ as in Agresti and Caffo, 2000).
typeI	Logical. If TRUE, the simulation is repeated under the null hypothesis, with every arm's success rate set to the average of p, and the rejection rate is reported as type I error. The default is FALSE.
seed	An optional integer passed to set.seed before simulating, so that the results can be reproduced. The default NULL leaves the random number generator unchanged.

Details

Consider an urn containing balls of $K + 1$ types. Balls of types $1, \dots, K$ represent treatments, and balls of type 0 are called immigration balls. When a subject arrives for randomization, a ball is drawn at random. If the ball is of type 0 (an immigration ball), no subject is treated, and the ball is returned to the urn together with $A = a_1 + \dots + a_K$ additional balls, a_k of treatment type $k, k = 1, \dots, K$. If a treatment ball is drawn (say, of type k , for some $k = 1, \dots, K$), the subject is given treatment k . If the observed response of this subject is a success, the ball is replaced, otherwise it is not replaced.

Value

An object of class "grouprar", a list that is printed as a short summary (see `print.grouprar`), with the following elements.

method	The name of the procedure.
sample size	The total sample size.
parameter	The true success rates used in the simulations, named pA, pB, ...
propotion	The mean allocation proportion of each arm over the simulations, named treatment A, treatment B, ...
sd of propotion	The standard deviation of the allocation proportion of each arm over the simulations.
failure rate	The mean failure rate over the simulations.
sd of failure rate	The standard deviation of the failure rate (or mean response) over the simulations.
power	The proportion of simulated trials that reject the null hypothesis of equal success rates. Simulations in which the test cannot be computed are dropped.
data: failureRate	The failure rate (or mean response) of each simulated trial.
data: test	The test decision of each simulated trial (1 = reject).
data: assignment	The treatment assignments of the last simulated trial.
data: propotion	A data frame with the allocation proportions of each simulated trial.
data: allocation	An nsim by ssn matrix with the treatment assignments of every simulated trial.
type I error	Only if typeI = TRUE. The rejection rate under the null hypothesis.

References

Zhang, L. X., Chan, W. S., Cheung, S. H. and Hu, F. (2007). *A generalized drop-the-loser urn for clinical trials with delayed responses*. Statistica Sinica, 17(1), 387-409.

Examples

```
## a simple use
gdl.res <- GDLRule(k = 3, p = c(0.6, 0.7, 0.6),
                  ssn = 200, aK = c(1, 1, 1), Y0 = NULL, nsim = 100, alpha = 0.05)

## view the output
gdl.res

## view all simulation settings
gdl.res[["method"]]
gdl.res[["parameter"]]

## view the simulation results
gdl.res[["propotion"]]
gdl.res[["failure rate"]]
gdl.res[["power"]]
gdl.res[["data: assignment"]]
```

Group.DBCD_Bin

*Group Doubly Biased Coin Design with Binary Response Type***Description**

Simulating the group doubly biased coin design with binary response (number of arms ≥ 2) in a clinical trial context. (Inference: two-sided t-test for two arms or chi-square test for more than two arms.)

Usage

```
Group.DBCD_Bin(n0 = 20, p, k, gsize.param, ssn, theta0 = NULL,
               target.alloc = "RPW", r = 2, nsim = 2000, mRate = NULL,
               alpha = 0.05, allocation = "DBCD", erade.alpha = 0.5,
               lower.bound = 0, monitor = NULL, test.fun = NULL,
               typeI = FALSE, seed = NULL)
```

Arguments

<code>n0</code>	A positive integer and a multiple of k . Whole groups are assigned by restricted randomization until at least $n0$ patients are enrolled, for initial parameter estimation.
<code>p</code>	A vector of length k with values between 0 and 1. The true success rates of the treatments, used to generate data for the simulations.
<code>k</code>	A positive integer. The number of treatment groups in the trial ($k \geq 2$).
<code>gsize.param</code>	A positive number. Group sizes are drawn from a zero-truncated Poisson distribution with rate <code>gsize.param</code> , so every group has at least one patient.

ssn	A positive integer. The total number of participants in each simulated trial.
theta0	A vector of length k used to smooth the success-rate estimates, $\hat{p}_k = (S_k + \theta_{0k})/(N_k + 1)$, where S_k and N_k are the number of successes and patients on treatment k . If NULL (default), all values are 0.5.
target.alloc	Desired allocation proportion. One of "Neyman", "RSIHR", "RPW", "WeisUrn", "OptimalNeyman" or "OptimalRSIHR". The default is "RPW". See Details.
r	A non-negative number. The tuning parameter of Hu and Zhang's allocation function, used when allocation = "DBCD". Values between 2 and 4 are common. The default value is 2.
nsim	A positive integer. The number of simulated trials, with a default value of 2000.
mRate	A number between 0 and 1 giving the probability that a response is missing. Missingness is simulated completely at random (MCAR), and missing responses are excluded from estimation and testing. The default NULL means no missing responses. As in Zhai et al. (2024), missing responses are excluded from the estimates, the failure rate and the test, while the allocation proportions (both those used by the allocation function and the reported ones) count all enrolled patients.
alpha	A number between 0 and 1. The significance level of the two-sided test, with a default value of 0.05.
allocation	The allocation function. "DBCD" (default) uses the allocation function of Hu and Zhang (2004) with parameter r . "ERADE" uses the efficient randomized adaptive design of Hu, Zhang and He (2009), in its multi-arm version (Alkhnefr, Hu and Zhai, 2025), with parameter <code>erade.alpha</code> . See Details.
erade.alpha	A number between 0 and 1. The degree of randomization of ERADE, used when allocation = "ERADE". Smaller values push the allocation more strongly towards the target, and values of 1/2 or 2/3 are recommended. The default is 0.5.
lower.bound	A number between 0 and $1/k$. The smallest allowed target proportion of each arm for the targets "OptimalNeyman" and "OptimalRSIHR". The default is 0.
monitor	An optional object created by sqMonitor for group sequential monitoring of a two-arm trial. For each information time t , a look is taken at the end of the first group in which at least $\text{ceiling}(t * \text{ssn})$ patients are enrolled, and the trial stops for efficacy as soon as the Wald statistic crosses the boundary of sqBoundary . The first look must come after the n_0 initial patients. Cannot be combined with <code>test.fun</code> . The default NULL gives a fixed sample design. See Details.
test.fun	An optional function function(outcome, assignment) that returns the p-value of a test for one simulated trial, where outcome holds the observed responses and assignment the treatment labels (integers 1 to k); missing responses are removed first. The null hypothesis is rejected when the p-value is at most alpha. The default NULL uses the built-in two-sided tests: the t-test for two arms and the Wald chi-squared test of equal means for more than two arms (for binary responses, if some arms have no variability, the variances are computed from $(S_k + 1)/(n_k + 2)$ as in Agresti and Caffo, 2000). Cannot be combined with <code>monitor</code> .

typeI	Logical. If TRUE, the simulation is repeated under the null hypothesis, with every arm's success rate set to the average of p , and the rejection rate is reported as type I error. The default is FALSE.
seed	An optional integer passed to set.seed before simulating, so that the results can be reproduced. The default NULL leaves the random number generator unchanged.

Details

Hu and Zhang's doubly biased coin design (DBCD) adjusts the probability of assigning each patient to a specific treatment group in a clinical trial, based on the responses of all previous patients. The group DBCD is a more practical version of this approach. It updates the allocation probabilities for the patients in each group based on the responses of all preceding groups, either when the data become available or at fixed time intervals (for example weekly or biweekly).

The process begins by assigning the first n_0 patients (possibly the first few groups) to treatment groups using restricted randomization and collecting their responses. Initial parameter estimates for the response variable are then obtained for each treatment group. Based on these estimates, the estimated desired allocation proportion is calculated. Then Hu and Zhang's allocation function is applied to determine the probabilities for the next group of patients to be assigned to each treatment group, which drive the allocation proportion towards the desired one. This process is repeated sequentially for each group until the predetermined number of patients has been allocated.

This methodology was introduced by Zhai, Li, Zhang and Hu (2024) in their paper 'Group response-adaptive randomization with delayed and missing responses'.

Target allocations. With $q_k = 1 - p_k$, "Neyman" and "RSIHR" are proportional to $\sqrt{p_k q_k}$ and $\sqrt{p_k}$. For two arms these are the Neyman allocation and the optimal allocation of Rosenberger et al. (2001), and for more arms they are simple generalizations. "RPW" and "WeisUrn" are proportional to $1/q_k$, the limiting allocation of the randomized play-the-winner rule and of Wei's urn. "OptimalNeyman" and "OptimalRSIHR" are the k -arm optimal allocations of Tymofyeyev, Rosenberger and Hu (2007). They minimize the total sample size and the expected number of failures, respectively, for a fixed noncentrality parameter of the chi-squared test of equal success rates, with every proportion at least `lower.bound`. For two arms and `lower.bound = 0` they equal "Neyman" and "RSIHR". For three or more arms the optimum without a lower bound can assign no patients to the middle arms, so a positive `lower.bound` such as 0.1 is recommended.

Allocation functions. With `allocation = "ERADE"` and $\alpha = \text{erade.alpha}$, arm k receives the next patient (in group designs, each patient of the next group) with probability $\alpha \hat{p}_k$ if it is over-allocated ($N_k/m > \hat{p}_k$), $\hat{p}_k$ if it is on target, and $\hat{p}_k + (1 - \alpha) \sum_{j \in S} \hat{p}_j / |T|$ if it is under-allocated, where S and T are the sets of over- and under-allocated arms, N_k/m is the current allocation proportion and $\hat{p}$ is the estimated target (Alkhnefr, Hu and Zhai, 2025, eq. (1)). For two arms this is the ERADE of Hu, Zhang and He (2009). ERADE attains the lower bound of the asymptotic variance of the allocation proportions, so its allocation is less variable than that of the DBCD.

Sequential monitoring. With `monitor = sqMonitor(t, spend)` (two arms only) the trial is analysed at the information times t . Looks are taken at the end of the first group in which at least $\text{ceiling}(t * \text{ssn})$ patients are enrolled. If one group passes several planned looks, only the last of them is analysed and the alpha of the skipped looks is not spent, which makes the test slightly conservative. At each look the Wald statistic $Z = (\hat{\theta}_1 - \hat{\theta}_2) / \sqrt{\hat{v}_1/n_1 + \hat{v}_2/n_2}$ is computed from all patients enrolled so far, with $\hat{p}_k(1 - \hat{p}_k)$ as variance estimates (missing responses are excluded), and the trial stops and rejects the null hypothesis as soon as $|Z|$ reaches the boundary of [sqBoundary](#). Zhu

and Hu (2010) showed that under the DBCD the sequential statistics are asymptotically a Brownian motion in the information time, so alpha spending boundaries keep the type I error. Their theory covers two arms and the DBCD, and the same boundaries are used with ERADE. The result then also contains the stopping probability at each look and the expected sample size.

Value

An object of class "grouprar", a list that is printed as a short summary (see `print.grouprar`), with the following elements.

method	The name of the procedure.
sample size	The total sample size (Total) and the expected number of patients with observed responses (Effective Size).
parameter	The true success rates used in the simulations, named pA, pB, ...
propotion	The mean allocation proportion of each arm over the simulations, named treatment A, treatment B, ...
sd of propotion	The standard deviation of the allocation proportion of each arm over the simulations.
failure rate	The mean failure rate over the simulations.
sd of failure rate	The standard deviation of the failure rate (or mean response) over the simulations.
power	The proportion of simulated trials that reject the null hypothesis of equal success rates. Simulations in which the test cannot be computed are dropped.
data: failureRate	The failure rate (or mean response) of each simulated trial.
data: test	The test decision of each simulated trial (1 = reject).
data: assignment	The treatment assignments of the last simulated trial.
data: propotion	A data frame with the allocation proportions of each simulated trial.
data: allocation	An nsim by ssn matrix with the treatment assignments of every simulated trial (positions after an early stop are NA).
type I error	Only if typeI = TRUE. The rejection rate under the null hypothesis.
boundary	Only if monitor is given. The boundaries for $ Z $ at the looks.
stopping probability	Only if monitor is given. The proportion of simulated trials that stop at each look (the last look is the final analysis).
expected sample size	Only if monitor is given. The mean number of enrolled patients.
data: stage	Only if monitor is given. The look at which each simulated trial stopped.
data: sample size	Only if monitor is given. The number of patients enrolled in each simulated trial.

References

- Alkhnefr, N., Hu, F. and Zhai, G. (2025). *Efficient randomized adaptive designs for multi-arm clinical trials*. Statistical Methods in Medical Research, 34(9), 1886-1898. doi:[10.1177/09622802251362644](https://doi.org/10.1177/09622802251362644)
- Hu, F. and Zhang, L. X. (2004). *Asymptotic properties of doubly adaptive biased coin designs for multitreatment clinical trials*. The Annals of Statistics, 32(1), 268-301. doi:[10.1214/aos/1079120137](https://doi.org/10.1214/aos/1079120137)
- Hu, F., Zhang, L. X. and He, X. (2009). *Efficient randomized-adaptive designs*. The Annals of Statistics, 37(5A), 2543-2560. doi:[10.1214/08AOS655](https://doi.org/10.1214/08AOS655)
- Rosenberger, W. F., Stallard, N., Ivanova, A., Harper, C. N. and Ricks, M. L. (2001). *Optimal adaptive designs for binary response trials*. Biometrics, 57(3), 909-913. doi:[10.1111/j.0006-341X.2001.00909.x](https://doi.org/10.1111/j.0006-341X.2001.00909.x)
- Tymofeyev, Y., Rosenberger, W. F. and Hu, F. (2007). *Implementing optimal allocation in sequential binary response experiments*. Journal of the American Statistical Association, 102(477), 224-234. doi:[10.1198/016214506000000906](https://doi.org/10.1198/016214506000000906)
- Zhai, G., Li, Y., Zhang, L. and Hu, F. (2024). *Group response-adaptive randomization with delayed and missing responses*. Statistics in Medicine, 43(27), 5047-5059. doi:[10.1002/sim.10220](https://doi.org/10.1002/sim.10220)
- Zhu, H. and Hu, F. (2010). *Sequential monitoring of response-adaptive randomized clinical trials*. The Annals of Statistics, 38(4), 2218-2241. doi:[10.1214/10AOS796](https://doi.org/10.1214/10AOS796)

Examples

```
res <- Group.DBCD_Bin(n0 = 20, p = c(0.65, 0.8), k = 2, gsize.param = 5,
                     ssn = 300, theta0 = NULL, target.alloc = "RPW",
                     r = 2, nsim = 100, mRate = NULL, alpha = 0.05)

res

## monitored group design with Pocock-type spending
res.sq <- Group.DBCD_Bin(n0 = 20, p = c(0.6, 0.8), k = 2, gsize.param = 5, ssn = 200,
                        target.alloc = "RSIHR", nsim = 50,
                        monitor = sqMonitor(c(0.5), spend = "Pocock"), seed = 1)

res.sq
```

Description

Simulating the group doubly biased coin design with continuous response (number of arms ≥ 2) in a clinical trial context. (Inference: two-sided t-test for two arms or chi-square test for more than two arms.)

Usage

```
Group.DBCD_Cont(n0 = 20, theta, k, gsize.param, ssn, theta0 = NULL,
  target.alloc = "Neyman", r = 2, nsim = 2000, mRate = NULL,
  alpha = 0.05, allocation = "DBCD", erade.alpha = 0.5,
  lower.bound = 0, monitor = NULL, test.fun = NULL,
  typeI = FALSE, seed = NULL)
```

Arguments

<code>n0</code>	A positive integer and a multiple of k . Whole groups are assigned by restricted randomization until at least <code>n0</code> patients are enrolled, for initial parameter estimation.
<code>theta</code>	A numerical vector of length $2k$ giving the true mean and variance of each treatment, used to generate data for the simulations. For example, with $k = 2$ <code>theta = c(13, 4.0^2, 15, 2.5^2)</code> .
<code>k</code>	A positive integer. The number of treatment groups in the trial ($k \geq 2$).
<code>gsize.param</code>	A positive number. Group sizes are drawn from a zero-truncated Poisson distribution with rate <code>gsize.param</code> , so every group has at least one patient.
<code>ssn</code>	A positive integer. The total number of participants in each simulated trial.
<code>theta0</code>	Currently unused. Kept for backward compatibility.
<code>target.alloc</code>	Desired allocation proportion. One of "Neyman", "ZR", "OptimalNeyman" or "DaOptimal". The default is "Neyman". See Details.
<code>r</code>	A non-negative number. The tuning parameter of Hu and Zhang's allocation function, used when <code>allocation = "DBCD"</code> . Values between 2 and 4 are common. The default value is 2.
<code>nsim</code>	A positive integer. The number of simulated trials, with a default value of 2000.
<code>mRate</code>	A number between 0 and 1 giving the probability that a response is missing. Missingness is simulated completely at random (MCAR), and missing responses are excluded from estimation and testing. The default NULL means no missing responses. As in Zhai et al. (2024), missing responses are excluded from the estimates, the failure rate and the test, while the allocation proportions (both those used by the allocation function and the reported ones) count all enrolled patients.
<code>alpha</code>	A number between 0 and 1. The significance level of the two-sided test, with a default value of 0.05.
<code>allocation</code>	The allocation function. "DBCD" (default) uses the allocation function of Hu and Zhang (2004) with parameter <code>r</code> . "ERADE" uses the efficient randomized adaptive design of Hu, Zhang and He (2009), in its multi-arm version (Alkhnefr, Hu and Zhai, 2025), with parameter <code>erade.alpha</code> . See Details.
<code>erade.alpha</code>	A number between 0 and 1. The degree of randomization of ERADE, used when <code>allocation = "ERADE"</code> . Smaller values push the allocation more strongly towards the target, and values of 1/2 or 2/3 are recommended. The default is 0.5.
<code>lower.bound</code>	A number between 0 and $1/k$. The smallest allowed target proportion of each arm for the targets "ZR" and "OptimalNeyman". The default is 0.

monitor	An optional object created by <code>sqMonitor</code> for group sequential monitoring of a two-arm trial. For each information time t , a look is taken at the end of the first group in which at least $\text{ceiling}(t * \text{ssn})$ patients are enrolled, and the trial stops for efficacy as soon as the Wald statistic crosses the boundary of <code>sqBoundary</code> . The first look must come after the n_0 initial patients. Cannot be combined with <code>test.fun</code> . The default NULL gives a fixed sample design. See Details.
test.fun	An optional function <code>function(outcome, assignment)</code> that returns the p-value of a test for one simulated trial, where <code>outcome</code> holds the observed responses and <code>assignment</code> the treatment labels (integers 1 to k); missing responses are removed first. The null hypothesis is rejected when the p-value is at most α . The default NULL uses the built-in two-sided tests: the t-test for two arms and the Wald chi-squared test of equal means for more than two arms (for binary responses, if some arms have no variability, the variances are computed from $(S_k + 1)/(n_k + 2)$ as in Agresti and Caffo, 2000). Cannot be combined with <code>monitor</code> .
typeI	Logical. If TRUE, the simulation is repeated under the null hypothesis, with the mean of every arm set to the average of the means in <code>theta</code> and the variance of each arm kept, and the rejection rate is reported as type I error. The default is FALSE.
seed	An optional integer passed to <code>set.seed</code> before simulating, so that the results can be reproduced. The default NULL leaves the random number generator unchanged.

Details

Hu and Zhang's doubly biased coin design (DBCD) adjusts the probability of assigning each patient to a specific treatment group in a clinical trial, based on the responses of all previous patients. The group DBCD is a more practical version of this approach. It updates the allocation probabilities for the patients in each group based on the responses of all preceding groups, either when the data become available or at fixed time intervals (for example weekly or biweekly).

The process begins by assigning the first n_0 patients (possibly the first few groups) to treatment groups using restricted randomization and collecting their responses. Initial parameter estimates for the response variable are then obtained for each treatment group. Based on these estimates, the estimated desired allocation proportion is calculated. Then Hu and Zhang's allocation function is applied to determine the probabilities for the next group of patients to be assigned to each treatment group, which drive the allocation proportion towards the desired one. This process is repeated sequentially for each group until the predetermined number of patients has been allocated.

This methodology was introduced by Zhai, Li, Zhang and Hu (2024) in their paper 'Group response-adaptive randomization with delayed and missing responses'.

Target allocations. "Neyman" is proportional to σ_k . "ZR" is the allocation of Zhang and Rosenberger (2006), which minimizes the total expected response (smaller responses are better, and all means must be positive). For two arms it is their rule (7): $\rho_1 = \sigma_1 \sqrt{\mu_2} / (\sigma_1 \sqrt{\mu_2} + \sigma_2 \sqrt{\mu_1})$ when this assigns more patients to the arm with the smaller mean, and 1/2 otherwise. For three or more arms it minimizes the total expected response for a fixed noncentrality parameter of the chi-squared test of equal means, with every proportion at least `lower.bound`, the continuous analogue of Tymofeyev, Rosenberger and Hu (2007). "OptimalNeyman" minimizes the total sample size

under the same constraints and equals "Neyman" for two arms with lower.bound = 0. For three or more arms a positive lower.bound such as 0.1 is recommended with "ZR" and "OptimalNeyman". "DaOptimal" is proportional to $\sigma_k^{4/3}$. While the target cannot be estimated (for example with fewer than two responses in an arm) equal allocation is used.

Allocation functions. With allocation = "ERADE" and $\alpha = \text{erade.alpha}$, arm k receives the next patient (in group designs, each patient of the next group) with probability $\alpha \hat{\rho}_k$ if it is over-allocated ($N_k/m > \hat{\rho}_k$), $\hat{\rho}_k$ if it is on target, and $\hat{\rho}_k + (1 - \alpha) \sum_{j \in S} \hat{\rho}_j / |T|$ if it is under-allocated, where S and T are the sets of over- and under-allocated arms, N_k/m is the current allocation proportion and $\hat{\rho}$ is the estimated target (Alkhnefr, Hu and Zhai, 2025, eq. (1)). For two arms this is the ERADE of Hu, Zhang and He (2009). ERADE attains the lower bound of the asymptotic variance of the allocation proportions, so its allocation is less variable than that of the DBCD.

Sequential monitoring. With monitor = sqMonitor(t, spend) (two arms only) the trial is analysed at the information times t . Looks are taken at the end of the first group in which at least ceiling($t * \text{ssn}$) patients are enrolled. If one group passes several planned looks, only the last of them is analysed and the alpha of the skipped looks is not spent, which makes the test slightly conservative. At each look the Wald statistic $Z = (\hat{\theta}_1 - \hat{\theta}_2) / \sqrt{\hat{v}_1/n_1 + \hat{v}_2/n_2}$ is computed from all patients enrolled so far, with the sample variances as variance estimates (missing responses are excluded), and the trial stops and rejects the null hypothesis as soon as $|Z|$ reaches the boundary of sqBoundary. Zhu and Hu (2010) showed that under the DBCD the sequential statistics are asymptotically a Brownian motion in the information time, so alpha spending boundaries keep the type I error. Their theory covers two arms and the DBCD, and the same boundaries are used with ERADE. The result then also contains the stopping probability at each look and the expected sample size.

Value

An object of class "grouprar", a list that is printed as a short summary (see `print.grouprar`), with the following elements.

method	The name of the procedure.
sample size	The total sample size (Total) and the expected number of patients with observed responses (Effective Size).
parameter	The true means and variances used in the simulations, named muA, sigma2A, muB, ...
propotion	The mean allocation proportion of each arm over the simulations, named treatment A, treatment B, ...
sd of propotion	The standard deviation of the allocation proportion of each arm over the simulations.
failure rate	The mean response over the simulations (for continuous responses this element holds the mean response, not a failure rate).
sd of failure rate	The standard deviation of the failure rate (or mean response) over the simulations.
power	The proportion of simulated trials that reject the null hypothesis of equal means. Simulations in which the test cannot be computed are dropped.
data: failureRate	The failure rate (or mean response) of each simulated trial.

data: test	The test decision of each simulated trial (1 = reject).
data: assignment	The treatment assignments of the last simulated trial.
data: propotion	A data frame with the allocation proportions of each simulated trial.
data: allocation	An nsim by ssn matrix with the treatment assignments of every simulated trial (positions after an early stop are NA).
type I error	Only if typeI = TRUE. The rejection rate under the null hypothesis.
boundary	Only if monitor is given. The boundaries for $ Z $ at the looks.
stopping probability	Only if monitor is given. The proportion of simulated trials that stop at each look (the last look is the final analysis).
expected sample size	Only if monitor is given. The mean number of enrolled patients.
data: stage	Only if monitor is given. The look at which each simulated trial stopped.
data: sample size	Only if monitor is given. The number of patients enrolled in each simulated trial.

References

- Alkhnefr, N., Hu, F. and Zhai, G. (2025). *Efficient randomized adaptive designs for multi-arm clinical trials*. Statistical Methods in Medical Research, 34(9), 1886-1898. doi:10.1177/09622802251362644
- Hu, F. and Zhang, L. X. (2004). *Asymptotic properties of doubly adaptive biased coin designs for multitreatment clinical trials*. The Annals of Statistics, 32(1), 268-301. doi:10.1214/aos/1079120137
- Hu, F., Zhang, L. X. and He, X. (2009). *Efficient randomized-adaptive designs*. The Annals of Statistics, 37(5A), 2543-2560. doi:10.1214/08AOS655
- Tymofeyev, Y., Rosenberger, W. F. and Hu, F. (2007). *Implementing optimal allocation in sequential binary response experiments*. Journal of the American Statistical Association, 102(477), 224-234. doi:10.1198/016214506000000906
- Zhai, G., Li, Y., Zhang, L. and Hu, F. (2024). *Group response-adaptive randomization with delayed and missing responses*. Statistics in Medicine, 43(27), 5047-5059. doi:10.1002/sim.10220
- Zhang, L. and Rosenberger, W. F. (2006). *Response-adaptive randomization for clinical trials with continuous outcomes*. Biometrics, 62(2), 562-569. doi:10.1111/j.15410420.2005.00496.x
- Zhu, H. and Hu, F. (2010). *Sequential monitoring of response-adaptive randomized clinical trials*. The Annals of Statistics, 38(4), 2218-2241. doi:10.1214/10AOS796

Examples

```
theta = c(13, 4.0^2, 15, 2.5^2)
k = 2
gsize.param = 5
ssn = 120
```

```
res <- Group.DBCD_Cont(n0 = 20, theta = theta, k = k, gsize.param = gsize.param,
                      ssn = ssn, target.alloc = "Neyman", r = 2, nsim = 100,
                      mRate = NULL, alpha = 0.05)

res
```

Group.dyldBCD_Bin

Group Doubly Biased Coin Design with Delayed Binary Response

Description

Simulating the group doubly biased coin design with delayed binary response (number of arms ≥ 2) in a clinical trial context. (Inference: two-sided t-test for two arms or chi-square test for more than two arms.)

Usage

```
Group.dyldBCD_Bin(n0 = 20, p, k, ssn, gsize.param, rspT.dist, rspT.param,
                  theta0 = NULL, target.alloc = "RPW", r = 2, nsim = 2000,
                  eTime = 7, mRate = NULL, alpha = 0.05, allocation = "DBCD",
                  erade.alpha = 0.5, lower.bound = 0, test.fun = NULL,
                  typeI = FALSE, seed = NULL)
```

Arguments

n0	A positive integer and a multiple of k. Whole groups are assigned by restricted randomization until at least n0 patients are enrolled, for initial parameter estimation.
p	A vector of length k with values between 0 and 1. The true success rates of the treatments, used to generate data for the simulations.
k	A positive integer. The number of treatment groups in the trial ($k \geq 2$).
ssn	A positive integer. The total number of participants in each simulated trial.
gsize.param	A positive number. Group sizes are drawn from a zero-truncated Poisson distribution with rate gsize.param, so every group has at least one patient.
rspT.dist	The distribution of the time from enrollment until the response is observed. One of "exponential", "normal" or "uniform".
rspT.param	A numeric vector of parameters for the response-time distributions, one distribution per treatment and response, ordered as (treatment 1 failure, treatment 1 success, treatment 2 failure, treatment 2 success, ...). For "exponential" give the $2k$ means. For "normal" give $2k$ (mean, sd) pairs and for "uniform" give $2k$ (min, max) pairs, that is $4k$ values in total. Normal times below 0 are set to 0. For example, with $k = 2$ and exponential times, <code>rspT.param = c(3, 2, 4, 1)</code> gives mean times 3 and 2 for failures and successes on treatment 1, and 4 and 1 on treatment 2.
theta0	A vector of length k used to smooth the success-rate estimates, $\hat{p}_k = (S_k + \theta_{0k}) / (N_k + 1)$, where S_k and N_k are the number of observed successes and observed responses on treatment k . If NULL (default), all values are 0.5.

target.alloc	Desired allocation proportion. One of "Neyman", "RSIHR", "RPW", "WeisUrn", "OptimalNeyman" or "OptimalRSIHR". The default is "RPW". See Details.
r	A non-negative number. The tuning parameter of Hu and Zhang's allocation function, used when allocation = "DBCD". Values between 2 and 4 are common. The default value is 2.
nsim	A positive integer. The number of simulated trials, with a default value of 2000.
eTime	A positive number. The time between the enrollment of consecutive groups. Allocation probabilities are updated once per group, using the responses observed by then. The default is 7.
mRate	A number between 0 and 1 giving the probability that a response is missing. Missingness is simulated completely at random (MCAR), and missing responses are excluded from estimation and testing. The default NULL means no missing responses. As in Zhai et al. (2024), missing responses are excluded from the estimates, the failure rate and the test, while the allocation proportions (both those used by the allocation function and the reported ones) count all enrolled patients.
alpha	A number between 0 and 1. The significance level of the two-sided test, with a default value of 0.05.
allocation	The allocation function. "DBCD" (default) uses the allocation function of Hu and Zhang (2004) with parameter r. "ERADE" uses the efficient randomized adaptive design of Hu, Zhang and He (2009), in its multi-arm version (Alkhnefr, Hu and Zhai, 2025), with parameter erade.alpha. See Details.
erade.alpha	A number between 0 and 1. The degree of randomization of ERADE, used when allocation = "ERADE". Smaller values push the allocation more strongly towards the target, and values of 1/2 or 2/3 are recommended. The default is 0.5.
lower.bound	A number between 0 and $1/k$. The smallest allowed target proportion of each arm for the targets "OptimalNeyman" and "OptimalRSIHR". The default is 0.
test.fun	An optional function function(outcome, assignment) that returns the p-value of a test for one simulated trial, where outcome holds the observed responses and assignment the treatment labels (integers 1 to k); missing responses are removed first. The null hypothesis is rejected when the p-value is at most alpha. The default NULL uses the built-in two-sided tests: the t-test for two arms and the Wald chi-squared test of equal means for more than two arms (for binary responses, if some arms have no variability, the variances are computed from $(S_k + 1)/(n_k + 2)$ as in Agresti and Caffo, 2000).
typeI	Logical. If TRUE, the simulation is repeated under the null hypothesis, with every arm's success rate set to the average of p, and the rejection rate is reported as type I error. The default is FALSE.
seed	An optional integer passed to set.seed before simulating, so that the results can be reproduced. The default NULL leaves the random number generator unchanged.

Details

Hu and Zhang's doubly biased coin design (DBCD) adjusts the probability of assigning each patient to a specific treatment group in a clinical trial, based on the responses of all previous patients. The

group DBCD is a more practical version of this approach. It updates the allocation probabilities for the patients in each group based on the available responses of all preceding groups, either when the data become available or at fixed time intervals (for example weekly or biweekly). This function implements the group doubly biased coin design for delayed binary responses.

The process begins by assigning the first n_0 patients (possibly the first few groups) to treatment groups using restricted randomization and collecting their responses. Initial parameter estimates for the response variable are then obtained for each treatment group. Based on these estimates, the estimated desired allocation proportion is calculated. Then Hu and Zhang's allocation function is applied to determine the probabilities for the next group of patients to be assigned to each treatment group, which drive the allocation proportion towards the desired one. This process is repeated sequentially for each group until the predetermined number of patients has been allocated.

This methodology was introduced by Zhai, Li, Zhang and Hu (2024) in their paper 'Group response-adaptive randomization with delayed and missing responses'.

Target allocations. With $q_k = 1 - p_k$, "Neyman" and "RSIHR" are proportional to $\sqrt{p_k q_k}$ and $\sqrt{p_k}$. For two arms these are the Neyman allocation and the optimal allocation of Rosenberger et al. (2001), and for more arms they are simple generalizations. "RPW" and "WeisUrn" are proportional to $1/q_k$, the limiting allocation of the randomized play-the-winner rule and of Wei's urn. "OptimalNeyman" and "OptimalRSIHR" are the k-arm optimal allocations of Tymofeyev, Rosenberger and Hu (2007). They minimize the total sample size and the expected number of failures, respectively, for a fixed noncentrality parameter of the chi-squared test of equal success rates, with every proportion at least lower.bound. For two arms and lower.bound = 0 they equal "Neyman" and "RSIHR". For three or more arms the optimum without a lower bound can assign no patients to the middle arms, so a positive lower.bound such as 0.1 is recommended.

Allocation functions. With allocation = "ERADE" and $\alpha = \text{erade.alpha}$, arm k receives the next patient (in group designs, each patient of the next group) with probability $\alpha \hat{p}_k$ if it is over-allocated ($N_k/m > \hat{p}_k$), $\hat{p}_k$ if it is on target, and $\hat{p}_k + (1 - \alpha) \sum_{j \in S} \hat{p}_j / |T|$ if it is under-allocated, where S and T are the sets of over- and under-allocated arms, N_k/m is the current allocation proportion and $\hat{p}$ is the estimated target (Alkhnefr, Hu and Zhai, 2025, eq. (1)). For two arms this is the ERADE of Hu, Zhang and He (2009). ERADE attains the lower bound of the asymptotic variance of the allocation proportions, so its allocation is less variable than that of the DBCD.

Value

An object of class "grouprar", a list that is printed as a short summary (see `print.grouprar`), with the following elements.

method	The name of the procedure.
sample size	The total sample size (Total) and the expected number of patients with observed responses (Effective Size).
parameter	The true success rates used in the simulations, named pA, pB, ...
propotion	The mean allocation proportion of each arm over the simulations, named treatment A, treatment B, ...
sd of propotion	The standard deviation of the allocation proportion of each arm over the simulations.
failure rate	The mean failure rate over the simulations.

sd of failure rate	The standard deviation of the failure rate (or mean response) over the simulations.
power	The proportion of simulated trials that reject the null hypothesis of equal success rates. Simulations in which the test cannot be computed are dropped.
data: failureRate	The failure rate (or mean response) of each simulated trial.
data: test	The test decision of each simulated trial (1 = reject).
data: assignment	The treatment assignments of the last simulated trial.
data: propotion	A data frame with the allocation proportions of each simulated trial.
data: allocation	An nsim by ssn matrix with the treatment assignments of every simulated trial.
type I error	Only if typeI = TRUE. The rejection rate under the null hypothesis.
duration	The mean time from the first enrollment to the last observed response. Trials without any observed response have duration NA and are left out of the mean.
sd of duration	The standard deviation of the duration over the simulations.
enrollment duration	The mean time from the first to the last enrollment.
data: duration	The duration of each simulated trial.
data: enrollment	The enrollment duration of each simulated trial.

References

- Alkhnefr, N., Hu, F. and Zhai, G. (2025). *Efficient randomized adaptive designs for multi-arm clinical trials*. Statistical Methods in Medical Research, 34(9), 1886-1898. doi:10.1177/09622802251362644
- Hu, F. and Zhang, L. X. (2004). *Asymptotic properties of doubly adaptive biased coin designs for multitreatment clinical trials*. The Annals of Statistics, 32(1), 268-301. doi:10.1214/aos/1079120137
- Hu, F., Zhang, L. X. and He, X. (2009). *Efficient randomized-adaptive designs*. The Annals of Statistics, 37(5A), 2543-2560. doi:10.1214/08AOS655
- Rosenberger, W. F., Stallard, N., Ivanova, A., Harper, C. N. and Ricks, M. L. (2001). *Optimal adaptive designs for binary response trials*. Biometrics, 57(3), 909-913. doi:10.1111/j.0006-341X.2001.00909.x
- Tymofeyev, Y., Rosenberger, W. F. and Hu, F. (2007). *Implementing optimal allocation in sequential binary response experiments*. Journal of the American Statistical Association, 102(477), 224-234. doi:10.1198/016214506000000906
- Zhai, G., Li, Y., Zhang, L. and Hu, F. (2024). *Group response-adaptive randomization with delayed and missing responses*. Statistics in Medicine, 43(27), 5047-5059. doi:10.1002/sim.10220

Examples

```
# a simple use
## Arguments for generating the simulated data
### For response simulation
p = c(0.7, 0.5)
k = 2
ssn = 200

### for entry time and response time simulation
eTime = 7
rspT.param = rep(10, 4)
rspT.dist = "exponential"
gsize.param = 5

## Arguments for the design
n0 = 10
target.alloc = "RPW"

res <- Group.dyldDBCD_Bin(n0 = n0, p = p, k = k, ssn = ssn, gsize.param = gsize.param,
  rspT.dist = rspT.dist, rspT.param = rspT.param, theta0 = NULL,
  target.alloc = target.alloc, r = 2,
  nsim = 100, eTime = eTime, mRate = NULL, alpha = 0.05)

# View the output (a list of all results)
res

## three arms, ERADE and the optimal RSIHR target, with the trial duration
res3 <- Group.dyldDBCD_Bin(n0 = 21, p = c(0.6, 0.7, 0.8), k = 3, ssn = 120,
  gsize.param = 5, rspT.dist = "exponential",
  rspT.param = rep(5, 6), target.alloc = "OptimalRSIHR",
  lower.bound = 0.1, nsim = 30, allocation = "ERADE", seed = 1)

res3[["duration"]]
```

Group.dyldDBCD_Cont	<i>Group Doubly Biased Coin Design with Delayed Continuous Response</i>
---------------------	---

Description

Simulating the group doubly biased coin design with delayed continuous response (number of arms ≥ 2) in a clinical trial context. (Inference: two-sided t-test for two arms or chi-square test for more than two arms.)

Usage

```
Group.dyldDBCD_Cont(n0 = 20, theta, k, ssn, gsize.param, rspT.dist,
  rspT.param, target.alloc = "Neyman", r = 2, nsim = 2000,
  eTime = 7, mRate = NULL, alpha = 0.05,
  allocation = "DBCD", erade.alpha = 0.5, lower.bound = 0,
  test.fun = NULL, typeI = FALSE, seed = NULL)
```

Arguments

<code>n0</code>	A positive integer and a multiple of k . Whole groups are assigned by restricted randomization until at least <code>n0</code> patients are enrolled, for initial parameter estimation.
<code>theta</code>	A numerical vector of length $2k$ giving the true mean and variance of each treatment, used to generate data for the simulations. For example, with $k = 2$ use <code>theta = c(13, 4.0^2, 15, 2.5^2)</code> .
<code>k</code>	A positive integer. The number of treatment groups in the trial ($k \geq 2$).
<code>ssn</code>	A positive integer. The total number of participants in each simulated trial.
<code>gsize.param</code>	A positive number. Group sizes are drawn from a zero-truncated Poisson distribution with rate <code>gsize.param</code> , so every group has at least one patient.
<code>rspT.dist</code>	The distribution of the time from enrollment until the response is observed. One of "exponential", "normal" or "uniform".
<code>rspT.param</code>	A numeric vector of parameters for the response-time distribution of each treatment. For "exponential" give the k means. For "normal" give k (mean, sd) pairs and for "uniform" give k (min, max) pairs, that is $2k$ values in total. Normal times below 0 are set to 0. For example, with 3 treatments and normal times with (mean, sd) pairs (3, 2), (2, 1) and (4, 1), use <code>rspT.param = c(3, 2, 2, 1, 4, 1)</code> .
<code>target.alloc</code>	Desired allocation proportion. One of "Neyman", "ZR", "OptimalNeyman" or "DaOptimal". The default is "Neyman". See Details.
<code>r</code>	A non-negative number. The tuning parameter of Hu and Zhang's allocation function, used when <code>allocation = "DBCD"</code> . Values between 2 and 4 are common. The default value is 2.
<code>nsim</code>	A positive integer. The number of simulated trials, with a default value of 2000.
<code>eTime</code>	A positive number. The time between the enrollment of consecutive groups. Allocation probabilities are updated once per group, using the responses observed by then. The default is 7.
<code>mRate</code>	A number between 0 and 1 giving the probability that a response is missing. Missingness is simulated completely at random (MCAR), and missing responses are excluded from estimation and testing. The default NULL means no missing responses. As in Zhai et al. (2024), missing responses are excluded from the estimates, the failure rate and the test, while the allocation proportions (both those used by the allocation function and the reported ones) count all enrolled patients.
<code>alpha</code>	A number between 0 and 1. The significance level of the two-sided test, with a default value of 0.05.
<code>allocation</code>	The allocation function. "DBCD" (default) uses the allocation function of Hu and Zhang (2004) with parameter <code>r</code> . "ERADE" uses the efficient randomized adaptive design of Hu, Zhang and He (2009), in its multi-arm version (Alkhnefr, Hu and Zhai, 2025), with parameter <code>erade.alpha</code> . See Details.
<code>erade.alpha</code>	A number between 0 and 1. The degree of randomization of ERADE, used when <code>allocation = "ERADE"</code> . Smaller values push the allocation more strongly towards the target, and values of 1/2 or 2/3 are recommended. The default is 0.5.

lower.bound	A number between 0 and $1/k$. The smallest allowed target proportion of each arm for the targets "ZR" and "OptimalNeyman". The default is 0.
test.fun	An optional function <code>function(outcome, assignment)</code> that returns the p-value of a test for one simulated trial, where <code>outcome</code> holds the observed responses and <code>assignment</code> the treatment labels (integers 1 to k); missing responses are removed first. The null hypothesis is rejected when the p-value is at most α . The default NULL uses the built-in two-sided tests: the t-test for two arms and the Wald chi-squared test of equal means for more than two arms (for binary responses, if some arms have no variability, the variances are computed from $(S_k + 1)/(n_k + 2)$ as in Agresti and Caffo, 2000).
typeI	Logical. If TRUE, the simulation is repeated under the null hypothesis, with the mean of every arm set to the average of the means in <code>theta</code> and the variance of each arm kept, and the rejection rate is reported as type I error. The default is FALSE.
seed	An optional integer passed to <code>set.seed</code> before simulating, so that the results can be reproduced. The default NULL leaves the random number generator unchanged.

Details

Hu and Zhang's doubly biased coin design (DBCD) adjusts the probability of assigning each patient to a specific treatment group in a clinical trial, based on the responses of all previous patients. The group DBCD is a more practical version of this approach. It updates the allocation probabilities for the patients in each group based on the available responses of all preceding groups, either when the data become available or at fixed time intervals (for example weekly or biweekly). This function implements the group doubly biased coin design for delayed continuous responses.

The process begins by assigning the first n_0 patients (possibly the first few groups) to treatment groups using restricted randomization and collecting their responses. Initial parameter estimates for the response variable are then obtained for each treatment group. Based on these estimates, the estimated desired allocation proportion is calculated. Then Hu and Zhang's allocation function is applied to determine the probabilities for the next group of patients to be assigned to each treatment group, which drive the allocation proportion towards the desired one. This process is repeated sequentially for each group until the predetermined number of patients has been allocated.

This methodology was introduced by Zhai, Li, Zhang and Hu (2024) in their paper 'Group response-adaptive randomization with delayed and missing responses'.

Target allocations. "Neyman" is proportional to σ_k . "ZR" is the allocation of Zhang and Rosenberger (2006), which minimizes the total expected response (smaller responses are better, and all means must be positive). For two arms it is their rule (7): $\rho_1 = \sigma_1 \sqrt{\mu_2} / (\sigma_1 \sqrt{\mu_2} + \sigma_2 \sqrt{\mu_1})$ when this assigns more patients to the arm with the smaller mean, and $1/2$ otherwise. For three or more arms it minimizes the total expected response for a fixed noncentrality parameter of the chi-squared test of equal means, with every proportion at least `lower.bound`, the continuous analogue of Tymofeyev, Rosenberger and Hu (2007). "OptimalNeyman" minimizes the total sample size under the same constraints and equals "Neyman" for two arms with `lower.bound` = 0. For three or more arms a positive `lower.bound` such as 0.1 is recommended with "ZR" and "OptimalNeyman". "DaOptimal" is proportional to $\sigma_k^{4/3}$. While the target cannot be estimated (for example with fewer than two responses in an arm) equal allocation is used.

Allocation functions. With `allocation = "ERADE"` and `alpha = erade.alpha`, arm k receives the next patient (in group designs, each patient of the next group) with probability $\alpha \hat{\rho}_k$ if it is over-allocated ($N_k/m > \hat{\rho}_k$), $\hat{\rho}_k$ if it is on target, and $\hat{\rho}_k + (1 - \alpha) \sum_{j \in S} \hat{\rho}_j / |T|$ if it is under-allocated, where S and T are the sets of over- and under-allocated arms, N_k/m is the current allocation proportion and $\hat{\rho}$ is the estimated target (Alkhnefr, Hu and Zhai, 2025, eq. (1)). For two arms this is the ERADE of Hu, Zhang and He (2009). ERADE attains the lower bound of the asymptotic variance of the allocation proportions, so its allocation is less variable than that of the DBCD.

Value

An object of class "grouprar", a list that is printed as a short summary (see `print.grouprar`), with the following elements.

<code>method</code>	The name of the procedure.
<code>sample size</code>	The total sample size (Total) and the expected number of patients with observed responses (Effective Size).
<code>parameter</code>	The true means and variances used in the simulations, named <code>muA</code> , <code>sigma2A</code> , <code>muB</code> , ...
<code>propotion</code>	The mean allocation proportion of each arm over the simulations, named <code>treatment A</code> , <code>treatment B</code> , ...
<code>sd of propotion</code>	The standard deviation of the allocation proportion of each arm over the simulations.
<code>failure rate</code>	The mean response over the simulations (for continuous responses this element holds the mean response, not a failure rate).
<code>sd of failure rate</code>	The standard deviation of the failure rate (or mean response) over the simulations.
<code>power</code>	The proportion of simulated trials that reject the null hypothesis of equal means. Simulations in which the test cannot be computed are dropped.
<code>data: failureRate</code>	The failure rate (or mean response) of each simulated trial.
<code>data: test</code>	The test decision of each simulated trial (1 = reject).
<code>data: assignment</code>	The treatment assignments of the last simulated trial.
<code>data: propotion</code>	A data frame with the allocation proportions of each simulated trial.
<code>data: allocation</code>	An <code>nsim</code> by <code>ssn</code> matrix with the treatment assignments of every simulated trial.
<code>type I error</code>	Only if <code>typeI = TRUE</code> . The rejection rate under the null hypothesis.
<code>duration</code>	The mean time from the first enrollment to the last observed response. Trials without any observed response have duration NA and are left out of the mean.
<code>sd of duration</code>	The standard deviation of the duration over the simulations.
<code>enrollment duration</code>	The mean time from the first to the last enrollment.
<code>data: duration</code>	The duration of each simulated trial.
<code>data: enrollment</code>	The enrollment duration of each simulated trial.

References

- Alkhnefr, N., Hu, F. and Zhai, G. (2025). *Efficient randomized adaptive designs for multi-arm clinical trials*. Statistical Methods in Medical Research, 34(9), 1886-1898. doi:[10.1177/09622802251362644](https://doi.org/10.1177/09622802251362644)
- Hu, F. and Zhang, L. X. (2004). *Asymptotic properties of doubly adaptive biased coin designs for multitreatment clinical trials*. The Annals of Statistics, 32(1), 268-301. doi:[10.1214/aos/1079120137](https://doi.org/10.1214/aos/1079120137)
- Hu, F., Zhang, L. X. and He, X. (2009). *Efficient randomized-adaptive designs*. The Annals of Statistics, 37(5A), 2543-2560. doi:[10.1214/08AOS655](https://doi.org/10.1214/08AOS655)
- Tymofyeyev, Y., Rosenberger, W. F. and Hu, F. (2007). *Implementing optimal allocation in sequential binary response experiments*. Journal of the American Statistical Association, 102(477), 224-234. doi:[10.1198/016214506000000906](https://doi.org/10.1198/016214506000000906)
- Zhai, G., Li, Y., Zhang, L. and Hu, F. (2024). *Group response-adaptive randomization with delayed and missing responses*. Statistics in Medicine, 43(27), 5047-5059. doi:[10.1002/sim.10220](https://doi.org/10.1002/sim.10220)
- Zhang, L. and Rosenberger, W. F. (2006). *Response-adaptive randomization for clinical trials with continuous outcomes*. Biometrics, 62(2), 562-569. doi:[10.1111/j.15410420.2005.00496.x](https://doi.org/10.1111/j.15410420.2005.00496.x)

Examples

```
# a simple use
## Arguments for generating the simulated data
### For response simulation
theta = c(13, 4.0^2, 15, 2.5^2)
k = 2
ssn = 120

### for entry time and response time simulation
eTime = 7
gsize.param = 5
rspT.param = rep(10, 2)
rspT.dist = "exponential"

## Arguments for the design
n0 = 10
target.alloc = "Neyman"

res <- Group.dyldBCD_Cont(n0 = n0, theta = theta, k = k, ssn = ssn,
                        gsize.param = gsize.param, rspT.dist = rspT.dist,
                        rspT.param = rspT.param, target.alloc = target.alloc,
                        r = 2, nsim = 100, eTime = eTime, mRate = 0.2, alpha = 0.05)

# View the output (a list of all results)
res
```

Description

Implements group response-adaptive randomization procedures, which include standard (non-group) response-adaptive randomization methods as special cases. The package also handles delayed and missing responses, which broadens its use in real-world trials. It offers functions for simulating a variety of response-adaptive randomization procedures, to help guide the choice of design for a clinical trial.

Urn designs: `RPWRule`, `WeiUrn`, `PolyaUrn`, `DLRule`, `GDLRule`, `Bai.Hu.Shen.Urn` and `BirthDeathUrn`, with `CRDesign` (complete randomization) as a non-adaptive benchmark.

Doubly adaptive biased coin designs (DBCD) for binary and continuous responses, with immediate or delayed responses, for individual patients or groups of patients: `DBCD_Bin`, `DBCD_Cont`, `dyldDBCD_Bin`, `dyldDBCD_Cont`, `Group.DBCD_Bin`, `Group.DBCD_Cont`, `Group.dyldDBCD_Bin` and `Group.dyldDBCD_Cont`. These designs support any number of arms, the allocation function of Hu and Zhang (2004) or the multi-arm efficient randomized adaptive design (ERADE), several target allocations including the k-arm optimal allocations of Tymofyeyev, Rosenberger and Hu (2007) with a lower bound, and missing responses.

Further tools: `nextAlloc` computes the allocation probabilities of the next patient or group in an ongoing trial, and `sqMonitor` and `sqBoundary` add group sequential monitoring with alpha spending to two-arm designs (Zhu and Hu, 2010). Every design accepts a user-supplied test (`test.fun`), can estimate the type I error (`typeI`) and takes a seed. The results are objects of class "grouprar" with print and summary methods (`print.grouprar`), and contain the allocation sequences of all simulated trials and, for delayed designs, the trial duration.

For details of the methods and algorithms, see the following references: Wei, L. J. (1979) doi:10.1214/aos/1176344614; Wei, L. J. and Durham, S. (1978) doi:10.1080/01621459.1978.10480109; Durham, S. D., Flournoy, N. and Li, W. (1998) doi:10.2307/3315771; Ivanova, A., Rosenberger, W. F., Durham, S. D. and Flournoy, N. (2000) <https://www.jstor.org/stable/25053121>; Bai, Z. D., Hu, F. and Shen, L. (2002) doi:10.1006/jmva.2001.1987; Ivanova, A. (2003) doi:10.1007/s001840200220; Hu, F. and Zhang, L. X. (2004) doi:10.1214/aos/1079120137; Hu, F. and Rosenberger, W. F. (2006, ISBN:978-0-471-65396-7); Zhang, L. X., Chan, W. S., Cheung, S. H. and Hu, F. (2007) <https://www.jstor.org/stable/26432528>; Zhang, L. and Rosenberger, W. F. (2006) doi:10.1111/j.15410420.2005.00496.x; Tymofyeyev, Y., Rosenberger, W. F. and Hu, F. (2007) doi:10.1198/016214506000000906; Hu, F., Zhang, L. X., Cheung, S. H. and Chan, W. S. (2008) doi:10.1002/cjs.5550360404; Hu, F., Zhang, L. X. and He, X. (2009) doi:10.1214/08AOS655; Zhu, H. and Hu, F. (2010) doi:10.1214/10AOS796; Zhai, G., Li, Y., Zhang, L. and Hu, F. (2024) doi:10.1002/sim.10220; Alkhnefr, N., Hu, F. and Zhai, G. (2025) doi:10.1177/09622802251362644.

Author(s)

Guannan Zhai <guannanzhai1996@gmail.com>; Feifang Hu <feifang@gwu.edu>.

References

- Alkhnefr, N., Hu, F. and Zhai, G. (2025). *Efficient randomized adaptive designs for multi-arm clinical trials*. Statistical Methods in Medical Research, 34(9), 1886-1898.
- Bai, Z. D., Hu, F. and Shen, L. (2002). *An adaptive design for multi-arm clinical trials*. Journal of Multivariate Analysis, 81(1), 1-18.
- Durham, S. D., Flournoy, N. and Li, W. (1998). *A sequential design for maximizing the probability of a favourable response*. Canadian Journal of Statistics, 26(3), 479-495.

- Hu, F. and Rosenberger, W. F. (2003). *Optimality, variability, power: evaluating response-adaptive randomization procedures for treatment comparisons*. Journal of the American Statistical Association, 98(463), 671-678.
- Hu, F. and Rosenberger, W. F. (2006). *The Theory of Response-Adaptive Randomization in Clinical Trials*. John Wiley & Sons.
- Hu, F. and Zhang, L. X. (2004). *Asymptotic properties of doubly adaptive biased coin designs for multitreatment clinical trials*. The Annals of Statistics, 32(1), 268-301.
- Hu, F., Zhang, L. X. and He, X. (2009). *Efficient randomized-adaptive designs*. The Annals of Statistics, 37(5A), 2543-2560.
- Hu, F., Zhang, L. X., Cheung, S. H. and Chan, W. S. (2008). *Doubly adaptive biased coin designs with delayed responses*. Canadian Journal of Statistics, 36(4), 541-559.
- Ivanova, A. (2003). *A play-the-winner-type urn design with reduced variability*. Metrika, 58, 1-13.
- Ivanova, A., Rosenberger, W. F., Durham, S. D. and Flournoy, N. (2000). *A birth and death urn for randomized clinical trials: asymptotic methods*. Sankhya, Series B, 62(1), 104-118.
- Tymofeyev, Y., Rosenberger, W. F. and Hu, F. (2007). *Implementing optimal allocation in sequential binary response experiments*. Journal of the American Statistical Association, 102(477), 224-234.
- Wei, L. J. (1979). *The generalized Polya's urn design for sequential medical trials*. The Annals of Statistics, 7(2), 291-296.
- Wei, L. J. and Durham, S. (1978). *The randomized play-the-winner rule in medical trials*. Journal of the American Statistical Association, 73(364), 840-843.
- Zelen, M. (1969). *Play the winner rule and the controlled clinical trial*. Journal of the American Statistical Association, 64(325), 131-146.
- Zhai, G., Li, Y., Zhang, L. and Hu, F. (2024). *Group response-adaptive randomization with delayed and missing responses*. Statistics in Medicine, 43(27), 5047-5059.
- Zhang, L. and Rosenberger, W. F. (2006). *Response-adaptive randomization for clinical trials with continuous outcomes*. Biometrics, 62(2), 562-569.
- Zhang, L. X., Chan, W. S., Cheung, S. H. and Hu, F. (2007). *A generalized drop-the-loser urn for clinical trials with delayed responses*. Statistica Sinica, 17(1), 387-409.
- Zhu, H. and Hu, F. (2010). *Sequential monitoring of response-adaptive randomized clinical trials*. The Annals of Statistics, 38(4), 2218-2241.

nextAlloc

Allocation Probabilities for an Ongoing Trial

Description

Computes the allocation probabilities of the next patient, or of the next group of patients, in an ongoing response-adaptive randomized trial from the assignments and responses observed so far, and optionally draws the assignments. The same estimates, targets and allocation functions as in the simulation functions (for example `DBCD_Bin` and `Group.DBCD_Bin`) are used.

Usage

```
nextAlloc(alloc, outcome, k, response = "binary", target.alloc = NULL,
          allocation = "DBCD", r = 2, erade.alpha = 0.5, lower.bound = 0,
          theta0 = NULL, size = 1)
```

Arguments

alloc	An integer vector with the treatment (1 to k) of every patient enrolled so far.
outcome	A numeric vector of the same length as alloc with the response of each patient (0 or 1 for binary responses). Use NA for responses that are not observed yet or are missing.
k	A positive integer. The number of treatment groups in the trial ($k \geq 2$).
response	The response type, "binary" (default) or "continuous".
target.alloc	Desired allocation proportion. For binary responses one of "Neyman", "RSIHR", "RPW", "WeisUrn", "OptimalNeyman" or "OptimalRSIHR", with default "RPW". For continuous responses one of "Neyman", "ZR", "OptimalNeyman" or "DaOptimal", with default "Neyman". See DBCD_Bin and DBCD_Cont for their definitions.
allocation	The allocation function, "DBCD" (default, Hu and Zhang, 2004) or "ERADE" (Hu, Zhang and He, 2009; Alkhnefr, Hu and Zhai, 2025). See DBCD_Bin .
r	A non-negative number. The tuning parameter of Hu and Zhang's allocation function, used when allocation = "DBCD". The default value is 2.
erade.alpha	A number between 0 and 1. The degree of randomization of ERADE, used when allocation = "ERADE". The default is 0.5.
lower.bound	A number between 0 and $1/k$. The smallest allowed target proportion of each arm for the optimal targets and "ZR". The default is 0.
theta0	A vector of length k used to smooth the success-rate estimates of binary responses, $\hat{p}_k = (S_k + \theta_{0k}) / (N_k + 1)$, where S_k and N_k are the number of observed successes and observed responses on treatment k . If NULL (default), all values are 0.5. Not used for continuous responses.
size	A non-negative integer. The number of assignments to draw with the returned probabilities, for example the size of the next group. The default is 1, and 0 draws none.

Details

The parameters of each arm are estimated from the observed responses only (NA values are ignored): smoothed success rates for binary responses, and the mean and variance (which needs at least two responses) for continuous responses. The current allocation proportions use all enrolled patients, including those whose responses are not available yet. If the target cannot be estimated, equal allocation is used as the target, and with no enrolled patients every arm has probability $1/k$. All patients of a group are assigned independently with the same probabilities, as in the group designs of Zhai, Li, Zhang and Hu (2024).

Value

A list with the following elements.

prob	The allocation probabilities of the next patient for each arm, named treatment A, treatment B, ...
target	The estimated target allocation proportions.
estimate	The parameter estimates, named pA, pB, ... or muA, sigma2A, ...
assignment	An integer vector of size assignments drawn with probabilities prob.

References

- Alkhnefr, N., Hu, F. and Zhai, G. (2025). *Efficient randomized adaptive designs for multi-arm clinical trials*. Statistical Methods in Medical Research, 34(9), 1886-1898. doi:[10.1177/09622802251362644](https://doi.org/10.1177/09622802251362644)
- Hu, F. and Zhang, L. X. (2004). *Asymptotic properties of doubly adaptive biased coin designs for multitreatment clinical trials*. The Annals of Statistics, 32(1), 268-301. doi:[10.1214/aos/1079120137](https://doi.org/10.1214/aos/1079120137)
- Hu, F., Zhang, L. X. and He, X. (2009). *Efficient randomized-adaptive designs*. The Annals of Statistics, 37(5A), 2543-2560. doi:[10.1214/08AOS655](https://doi.org/10.1214/08AOS655)
- Zhai, G., Li, Y., Zhang, L. and Hu, F. (2024). *Group response-adaptive randomization with delayed and missing responses*. Statistics in Medicine, 43(27), 5047-5059. doi:[10.1002/sim.10220](https://doi.org/10.1002/sim.10220)

See Also

[DBCD_Bin](#), [DBCD_Cont](#), [Group.DBCD_Bin](#) for simulating the designs.

Examples

```
## 30 patients enrolled in a three-arm trial, the last three responses are pending
set.seed(1)
alloc <- sample(1:3, 30, replace = TRUE)
outcome <- rbinom(30, 1, c(0.6, 0.7, 0.8)[alloc])
outcome[28:30] <- NA

## probabilities for the next patient with the DBCD and the RSIHR target
nextAlloc(alloc, outcome, k = 3, target.alloc = "RSIHR")

## assignments for the next group of 5 patients with ERADE
nextAlloc(alloc, outcome, k = 3, target.alloc = "RSIHR", allocation = "ERADE",
          size = 5)$assignment

## continuous responses
y <- rnorm(30, mean = c(13, 15, 14)[alloc], sd = 3)
nextAlloc(alloc, y, k = 3, response = "continuous", target.alloc = "Neyman")$prob
```

PolyaUrn

*Randomized Pólya urn procedure***Description**

Simulating the randomized Pólya urn procedure (number of arms ≥ 2) with two-sided hypothesis testing in a clinical trial context.

Usage

```
PolyaUrn(k, p, ssn, Y0 = NULL, nsim = 2000, alpha = 0.05, test.fun = NULL,
         typeI = FALSE, seed = NULL)
```

Arguments

k	A positive integer. The number of treatment groups in the trial ($k \geq 2$).
p	A vector of length k with values between 0 and 1. The true success rates of the treatments, used to generate data for the simulations.
ssn	A positive integer. The total number of participants in each simulated trial.
Y0	A vector of length k giving the initial urn composition (number of balls of each treatment type). For instance, if $Y0 = c(1, 1, 1)$, the first patient is assigned to each treatment with probability $Y0 / \text{sum}(Y0)$. If Y0 is NULL (default), it is set to a vector of k ones.
nsim	A positive integer. The number of simulated trials, with a default value of 2000.
alpha	A number between 0 and 1. The significance level of the two-sided test, with a default value of 0.05.
test.fun	An optional function function(outcome, assignment) that returns the p-value of a test for one simulated trial, where outcome holds the observed responses and assignment the treatment labels (integers 1 to k); missing responses are removed first. The null hypothesis is rejected when the p-value is at most alpha. The default NULL uses the built-in two-sided tests: the t-test for two arms and the Wald chi-squared test of equal means for more than two arms (for binary responses, if some arms have no variability, the variances are computed from $(S_k + 1)/(n_k + 2)$ as in Agresti and Caffo, 2000).
typeI	Logical. If TRUE, the simulation is repeated under the null hypothesis, with every arm's success rate set to the average of p, and the rejection rate is reported as type I error. The default is FALSE.
seed	An optional integer passed to set.seed before simulating, so that the results can be reproduced. The default NULL leaves the random number generator unchanged.

Details

The randomized Pólya urn (RPU) procedure can be described as follows. An urn initially contains at least one ball of each of the K treatment types. A ball is drawn from the urn with replacement. If a type i ball is drawn, $i = 1, \dots, K$, then treatment i is assigned to the next patient. If the response is a success, a ball of type i is added to the urn. Otherwise the urn remains unchanged.

Value

An object of class "grouprar", a list that is printed as a short summary (see `print.grouprar`), with the following elements.

method	The name of the procedure.
sample size	The total sample size.
parameter	The true success rates used in the simulations, named pA, pB, ...
propotion	The mean allocation proportion of each arm over the simulations, named treatment A, treatment B, ...
sd of propotion	The standard deviation of the allocation proportion of each arm over the simulations.
failure rate	The mean failure rate over the simulations.
sd of failure rate	The standard deviation of the failure rate (or mean response) over the simulations.
power	The proportion of simulated trials that reject the null hypothesis of equal success rates. Simulations in which the test cannot be computed are dropped.
data: failureRate	The failure rate (or mean response) of each simulated trial.
data: test	The test decision of each simulated trial (1 = reject).
data: assignment	The treatment assignments of the last simulated trial.
data: propotion	A data frame with the allocation proportions of each simulated trial.
data: allocation	An nsim by ssn matrix with the treatment assignments of every simulated trial.
type I error	Only if typeI = TRUE. The rejection rate under the null hypothesis.

The randomized Pólya urn tends to concentrate on one arm, so some arms can end with very few patients. The asymptotic tests are then far from their nominal level (the rejection rate under the null hypothesis can be well above alpha, especially for $K \geq 3$), and trials in which an arm has no patients give no test. Use typeI = TRUE to check the type I error.

References

Durham, S. D., Flournoy, N. and Li, W. (1998). *A sequential design for maximizing the probability of a favourable response*. Canadian Journal of Statistics, 26(3), 479-495. doi:10.2307/3315771

Examples

```
## a simple use
Polya.res <- PolyaUrn(k = 3, p = c(0.6, 0.7, 0.6), ssn = 200, Y0 = NULL,
                     nsim = 100, alpha = 0.05)

## view the output
Polya.res

## view all simulation settings
Polya.res[["method"]]
Polya.res[["parameter"]]

## view the simulation results
Polya.res[["propotion"]]
Polya.res[["failure rate"]]
Polya.res[["power"]]
Polya.res[["data: assignment"]]
```

print.grouprar

Print and Summarize Simulation Results

Description

Methods for the objects of class "grouprar" returned by the simulation functions of the package, such as [DBCD_Bin](#) or [WeiUrn](#).

Usage

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

## S3 method for class 'grouprar'
summary(object, ...)

## S3 method for class 'summary.grouprar'
print(x, digits = 3, ...)
```

Arguments

x	An object of class "grouprar" (for print) or "summary.grouprar" (for print.summary.grouprar).
object	An object of class "grouprar".
digits	The number of decimal places to print. The default is 3.
...	Further arguments passed to or from other methods.

Details

print shows the summary of the object. The full simulation results, including the per-simulation data, remain available as list elements, for example x[["data: allocation"]].

Value

summary returns an object of class "summary.grouprar", a list with the elements

method	The name of the procedure.
sample size	The sample size.
nsim	The number of simulated trials.
arms	A data frame with one row per arm: the true parameters, the mean allocation proportion and its standard deviation over the simulations.
stats	A named vector with the failure rate (or mean response for continuous designs) and its standard deviation, the power, and, when available, the type I error, the trial and enrollment durations, and the expected sample size.
stopping probability	For monitored designs, the proportion of simulated trials that stop at each look.

The print methods return their argument invisibly.

See Also

[DBCD_Bin](#), [CRDesign](#).

Examples

```
res <- CRDesign(k = 3, p = c(0.6, 0.7, 0.8), ssn = 100, nsim = 50, seed = 1)
res
s <- summary(res)
s$arms
print(s, digits = 2)
```

RPWRule

Randomized Play-the-winner Rule

Description

Simulating the randomized play-the-winner rule (two arms) with two-sided hypothesis testing in a clinical trial context.

Usage

```
RPWRule(k, p, ssn, Y0 = NULL, nsim = 2000, alpha = 0.05, test.fun = NULL,
        typeI = FALSE, seed = NULL)
```

Arguments

<code>k</code>	A positive integer. The number of treatment groups in the trial. Only $k = 2$ is supported, and other values give an error; see WeiUrn for more than two arms.
<code>p</code>	A vector of length k with values between 0 and 1. The true success rates of the treatments, used to generate data for the simulations.
<code>ssn</code>	A positive integer. The total number of participants in each simulated trial.
<code>Y0</code>	A vector of length k giving the initial urn composition (number of balls of each treatment type). For instance, if $Y0 = c(1, 1)$, the first patient is assigned to each treatment with probability $Y0 / \text{sum}(Y0)$. If $Y0$ is NULL (default), it is set to a vector of k ones.
<code>nsim</code>	A positive integer. The number of simulated trials, with a default value of 2000.
<code>alpha</code>	A number between 0 and 1. The significance level of the two-sided test, with a default value of 0.05.
<code>test.fun</code>	An optional function <code>function(outcome, assignment)</code> that returns the p-value of a test for one simulated trial, where <code>outcome</code> holds the observed responses and <code>assignment</code> the treatment labels (integers 1 to k); missing responses are removed first. The null hypothesis is rejected when the p-value is at most α . The default NULL uses the built-in two-sided tests: the t-test for two arms and the Wald chi-squared test of equal means for more than two arms (for binary responses, if some arms have no variability, the variances are computed from $(S_k + 1)/(n_k + 2)$ as in Agresti and Caffo, 2000).
<code>typeI</code>	Logical. If TRUE, the simulation is repeated under the null hypothesis, with every arm's success rate set to the average of p , and the rejection rate is reported as type I error. The default is FALSE.
<code>seed</code>	An optional integer passed to set.seed before simulating, so that the results can be reproduced. The default NULL leaves the random number generator unchanged.

Details

The randomized play-the-winner rule allocates future subjects in a clinical trial to treatment groups based on the performance of previously treated subjects. A ball is drawn from the urn to assign the next subject. A success on a treatment adds a ball of the same type to the urn, and a failure adds a ball of the other type. This increases the chance that future patients are assigned to the better-performing treatment.

Value

An object of class "grouprar", a list that is printed as a short summary (see [print.grouprar](#)), with the following elements.

<code>method</code>	The name of the procedure.
<code>sample size</code>	The total sample size.
<code>parameter</code>	The true success rates used in the simulations, named <code>pA</code> , <code>pB</code> , ...
<code>propotion</code>	The mean allocation proportion of each arm over the simulations, named <code>treatment A</code> , <code>treatment B</code> , ...

sd of propotion	The standard deviation of the allocation proportion of each arm over the simulations.
failure rate	The mean failure rate over the simulations.
sd of failure rate	The standard deviation of the failure rate (or mean response) over the simulations.
power	The proportion of simulated trials that reject the null hypothesis of equal success rates. Simulations in which the test cannot be computed are dropped.
data: failureRate	The failure rate (or mean response) of each simulated trial.
data: test	The test decision of each simulated trial (1 = reject).
data: assignment	The treatment assignments of the last simulated trial.
data: propotion	A data frame with the allocation proportions of each simulated trial.
data: allocation	An nsim by ssn matrix with the treatment assignments of every simulated trial.
type I error	Only if typeI = TRUE. The rejection rate under the null hypothesis.

References

Wei, L. J. and Durham, S. (1978). *The randomized play-the-winner rule in medical trials*. Journal of the American Statistical Association, 73(364), 840-843. doi:[10.1080/01621459.1978.10480109](https://doi.org/10.1080/01621459.1978.10480109)

Examples

```
## a simple use
RPW.res <- RPWRule(k = 2, p = c(0.7, 0.8), ssn = 200, Y0 = NULL, nsim = 100, alpha = 0.05)
## view the output
RPW.res

## view all simulation settings
RPW.res[["method"]]
RPW.res[["parameter"]]

## view the simulation results
RPW.res[["propotion"]]
RPW.res[["failure rate"]]
RPW.res[["power"]]
RPW.res[["data: assignment"]]
```

sqMonitor

*Group Sequential Monitoring of Response-Adaptive Randomized Trials***Description**

sqMonitor specifies the interim looks and the alpha spending function of a group sequential design, to be passed as the monitor argument of `DBCD_Bin`, `DBCD_Cont`, `Group.DBCD_Bin` and `Group.DBCD_Cont`. sqBoundary computes the corresponding two-sided boundaries for the standardized test statistic.

Usage

```
sqMonitor(t, spend = "OBF")
```

```
sqBoundary(t, alpha = 0.05, spend = "OBF")
```

Arguments

t	A numeric vector of information times in (0, 1], the fraction of the total sample size at which the looks are taken. The final analysis at $t = 1$ is added if it is missing.
spend	The alpha spending function. One of "OBF" (O'Brien-Fleming-type, default), "Pocock" (Pocock-type) or "Linear".
alpha	A number between 0 and 1. The overall two-sided significance level, with a default value of 0.05.

Details

The spending functions of Lan and DeMets (1983) are used, with each tail spending the one-sided function at level $\alpha/2$ (Proschan, Lan and Wittes, 2006), as in Zhu and Hu (2010). For one tail at level $\alpha = \alpha/2$ they are $2\{1 - \Phi(z_{\alpha/2}/\sqrt{t})\}$ for "OBF", $a \log\{1 + (e - 1)t\}$ for "Pocock" and at for "Linear". The boundary c_j of look j is chosen so that, under the null hypothesis, the probability that $|Z|$ first reaches the boundary at look j equals the alpha spent between looks $j - 1$ and j , where Z at the information times $t_1 < t_2 < \dots$ is a standard normal process with correlation $\sqrt{t_i/t_j}$. Zhu and Hu (2010) showed that the sequential test statistics of a two-arm trial randomized by the doubly adaptive biased coin design have asymptotically this structure, so these boundaries keep the type I error. The boundaries are computed by recursive numerical integration of the density of $Z\sqrt{t}$ over the continuation region (Armitage, McPherson and Rowe, 1969), which is fast for any number of looks.

Sequential monitoring is available for two arms only, because the theory of Zhu and Hu (2010) covers the comparison of two treatments.

Value

sqMonitor returns an object of class "sqMonitor", a list with the sorted information times t and the spending function spend.

sqBoundary returns a numeric vector with the boundary for $|Z|$ at each look, named look 1, look 2, ... A boundary is Inf if no alpha is spent at that look.

References

- Lan, K. K. G. and DeMets, D. L. (1983). *Discrete sequential boundaries for clinical trials*. Biometrika, 70(3), 659-663. doi:10.1093/biomet/70.3.659
- Armitage, P., McPherson, C. K. and Rowe, B. C. (1969). *Repeated significance tests on accumulating data*. Journal of the Royal Statistical Society, Series A, 132(2), 235-244. doi:10.2307/2343787
- Proschan, M. A., Lan, K. K. G. and Wittes, J. T. (2006). *Statistical Monitoring of Clinical Trials: A Unified Approach*. Springer.
- Zhu, H. and Hu, F. (2010). *Sequential monitoring of response-adaptive randomized clinical trials*. The Annals of Statistics, 38(4), 2218-2241. doi:10.1214/10AOS796

See Also

[DBCD_Bin](#), [DBCD_Cont](#), [Group.DBCD_Bin](#), [Group.DBCD_Cont](#).

Examples

```
## the boundaries of Zhu and Hu (2010): 4.877, 2.963 and 1.969
sqBoundary(c(0.2, 0.5, 1), alpha = 0.05, spend = "OBF")
sqBoundary(c(0.2, 0.5, 1), alpha = 0.05, spend = "Pocock")

## two interim looks after 1/3 and 2/3 of the patients
m <- sqMonitor(c(1/3, 2/3), spend = "OBF")
res <- DBCD_Bin(n0 = 20, p = c(0.6, 0.8), k = 2, ssn = 150, target.alloc = "RSIHR",
               nsim = 30, monitor = m, seed = 1)
res[["stopping probability"]]
```

WeiUrn

Wei's Urn: Randomized Play-the-winner Rule with Multiple Arms

Description

Simulating Wei's urn, the randomized play-the-winner rule for multiple arms (number of arms ≥ 2), with two-sided hypothesis testing in a clinical trial context.

Usage

```
WeiUrn(k, p, ssn, Y0 = NULL, nsim = 2000, alpha = 0.05, test.fun = NULL,
       typeI = FALSE, seed = NULL)
```

Arguments

<code>k</code>	A positive integer. The number of treatment groups in the trial ($k \geq 2$).
<code>p</code>	A vector of length k with values between 0 and 1. The true success rates of the treatments, used to generate data for the simulations.
<code>ssn</code>	A positive integer. The total number of participants in each simulated trial.
<code>Y0</code>	A vector of length k giving the initial urn composition (number of balls of each treatment type). For instance, if $Y0 = c(1, 1, 1)$, the first patient is assigned to each treatment with probability $Y0 / \text{sum}(Y0)$. If $Y0$ is NULL (default), it is set to a vector of k ones.
<code>nsim</code>	A positive integer. The number of simulated trials, with a default value of 2000.
<code>alpha</code>	A number between 0 and 1. The significance level of the two-sided test, with a default value of 0.05.
<code>test.fun</code>	An optional function <code>function(outcome, assignment)</code> that returns the p-value of a test for one simulated trial, where <code>outcome</code> holds the observed responses and <code>assignment</code> the treatment labels (integers 1 to k); missing responses are removed first. The null hypothesis is rejected when the p-value is at most <code>alpha</code> . The default NULL uses the built-in two-sided tests: the t-test for two arms and the Wald chi-squared test of equal means for more than two arms (for binary responses, if some arms have no variability, the variances are computed from $(S_k + 1)/(n_k + 2)$ as in Agresti and Caffo, 2000).
<code>typeI</code>	Logical. If TRUE, the simulation is repeated under the null hypothesis, with every arm's success rate set to the average of <code>p</code> , and the rejection rate is reported as type I error. The default is FALSE.
<code>seed</code>	An optional integer passed to set.seed before simulating, so that the results can be reproduced. The default NULL leaves the random number generator unchanged.

Details

Wei's urn procedure extends the randomized play-the-winner rule (Wei and Durham, 1978) from $k = 2$ to $k > 2$ treatments, which makes it usable in multi-arm clinical trials. A success on treatment k adds one ball of type k to the urn, and a failure on treatment k adds $1/(K - 1)$ balls of each of the other $K - 1$ types. With $k = 2$ it reduces to the randomized play-the-winner rule.

Value

An object of class "grouprar", a list that is printed as a short summary (see [print.grouprar](#)), with the following elements.

<code>method</code>	The name of the procedure.
<code>sample size</code>	The total sample size.
<code>parameter</code>	The true success rates used in the simulations, named <code>pA</code> , <code>pB</code> , ...
<code>propotion</code>	The mean allocation proportion of each arm over the simulations, named <code>treatment A</code> , <code>treatment B</code> , ...

sd of propotion	The standard deviation of the allocation proportion of each arm over the simulations.
failure rate	The mean failure rate over the simulations.
sd of failure rate	The standard deviation of the failure rate (or mean response) over the simulations.
power	The proportion of simulated trials that reject the null hypothesis of equal success rates. Simulations in which the test cannot be computed are dropped.
data: failureRate	The failure rate (or mean response) of each simulated trial.
data: test	The test decision of each simulated trial (1 = reject).
data: assignment	The treatment assignments of the last simulated trial.
data: propotion	A data frame with the allocation proportions of each simulated trial.
data: allocation	An nsim by ssn matrix with the treatment assignments of every simulated trial.
type I error	Only if typeI = TRUE. The rejection rate under the null hypothesis.

References

- Wei, L. J. (1979). *The generalized Polya's urn design for sequential medical trials*. The Annals of Statistics, 7(2), 291-296. doi:[10.1214/aos/11176344614](https://doi.org/10.1214/aos/11176344614)
- Wei, L. J. and Durham, S. (1978). *The randomized play-the-winner rule in medical trials*. Journal of the American Statistical Association, 73(364), 840-843.

Examples

```
## a simple use
wei.res <- WeiUrn(k = 3, p = c(0.7, 0.8, 0.7), ss = 200, Y0 = NULL, nsim = 100, alpha = 0.05)

## view the output
wei.res

## view all simulation settings
wei.res[["method"]]
wei.res[["parameter"]]

## view the simulation results
wei.res[["propotion"]]
wei.res[["failure rate"]]
wei.res[["power"]]
wei.res[["data: assignment"]]
```

Index

- * **Bai Hu Shen urn**
 - Bai.Hu.Shen.Urn, [3](#)
- * **Wei's urn**
 - WeiUrn, [60](#)
- * **alpha spending function**
 - sqMonitor, [59](#)
- * **birth and death urn**
 - BirthDeathUrn, [5](#)
- * **complete randomization**
 - CRDesign, [7](#)
- * **design**
 - Bai.Hu.Shen.Urn, [3](#)
 - BirthDeathUrn, [5](#)
 - CRDesign, [7](#)
 - DBCD_Bin, [9](#)
 - DBCD_Cont, [14](#)
 - DLRule, [18](#)
 - dyldDBCD_Bin, [20](#)
 - dyldDBCD_Cont, [24](#)
 - GDRule, [29](#)
 - Group.DBCD_Bin, [31](#)
 - Group.DBCD_Cont, [35](#)
 - Group.dyldDBCD_Bin, [40](#)
 - Group.dyldDBCD_Cont, [44](#)
 - nextAlloc, [50](#)
 - PolyaUrn, [53](#)
 - RPWRule, [56](#)
 - sqMonitor, [59](#)
 - WeiUrn, [60](#)
- * **doubly adaptive biased coin design with continuous response**
 - DBCD_Cont, [14](#)
- * **doubly adaptive biased coin designs with delayed responses**
 - dyldDBCD_Bin, [20](#)
 - dyldDBCD_Cont, [24](#)
- * **doubly adaptive biased coin design**
 - DBCD_Bin, [9](#)
 - DBCD_Cont, [14](#)
 - dyldDBCD_Bin, [20](#)
 - dyldDBCD_Cont, [24](#)
 - nextAlloc, [50](#)
- * **drop-the-loser rule**
 - DLRule, [18](#)
- * **efficient randomized adaptive design**
 - nextAlloc, [50](#)
- * **generalized drop-the-loser rule**
 - GDRule, [29](#)
- * **group doubly adaptive biased coin design with continuous response**
 - Group.DBCD_Cont, [35](#)
- * **group doubly adaptive biased coin design with delayed binary response**
 - Group.dyldDBCD_Bin, [40](#)
- * **group doubly adaptive biased coin design with delayed continuous response**
 - Group.dyldDBCD_Cont, [44](#)
- * **group doubly adaptive biased coin design**
 - Group.DBCD_Bin, [31](#)
 - Group.DBCD_Cont, [35](#)
 - Group.dyldDBCD_Bin, [40](#)
 - Group.dyldDBCD_Cont, [44](#)
- * **group response-adaptive randomization**
 - Group.DBCD_Bin, [31](#)
 - Group.DBCD_Cont, [35](#)
 - Group.dyldDBCD_Bin, [40](#)
 - Group.dyldDBCD_Cont, [44](#)
- * **group sequential design**
 - sqMonitor, [59](#)
- * **package**
 - grouprar, [48](#)
- * **print**
 - print.grouprar, [55](#)
- * **randomized Polya urn**
 - PolyaUrn, [53](#)
- * **randomized play-the-winner rule**
 - RPWRule, [56](#)
 - WeiUrn, [60](#)

* **response-adaptive randomization**

Bai.Hu.Shen.Urn, [3](#)
 BirthDeathUrn, [5](#)
 DBCD_Bin, [9](#)
 DBCD_Cont, [14](#)
 DLRule, [18](#)
 dyldDBCD_Bin, [20](#)
 dyldDBCD_Cont, [24](#)
 GDLRule, [29](#)
 Group.DBCD_Bin, [31](#)
 nextAlloc, [50](#)
 PolyaUrn, [53](#)
 RPWRule, [56](#)
 sqMonitor, [59](#)
 WeiUrn, [60](#)

Bai.Hu.Shen.Urn, [3](#), [49](#)

BirthDeathUrn, [5](#), [49](#)

CRDesign, [7](#), [49](#), [56](#)

DBCD_Bin, [9](#), [17](#), [49–52](#), [55](#), [56](#), [59](#), [60](#)

DBCD_Cont, [14](#), [49](#), [51](#), [52](#), [59](#), [60](#)

DLRule, [18](#), [49](#)

dyldDBCD_Bin, [20](#), [49](#)

dyldDBCD_Cont, [17](#), [24](#), [49](#)

GDLRule, [29](#), [49](#)

Group.DBCD_Bin, [31](#), [49](#), [50](#), [52](#), [59](#), [60](#)

Group.DBCD_Cont, [35](#), [49](#), [59](#), [60](#)

Group.dyldDBCD_Bin, [40](#), [49](#)

Group.dyldDBCD_Cont, [44](#), [49](#)

grouprar, [48](#)

grouprar-package (grouprar), [48](#)

nextAlloc, [49](#), [50](#)

PolyaUrn, [49](#), [53](#)

print.grouprar, [4](#), [6](#), [8](#), [12](#), [16](#), [19](#), [23](#), [27](#),
[30](#), [34](#), [38](#), [42](#), [47](#), [49](#), [54](#), [55](#), [57](#), [61](#)

print.summary.grouprar
 (print.grouprar), [55](#)

RPWRule, [49](#), [56](#)

set.seed, [3](#), [6](#), [8](#), [11](#), [15](#), [19](#), [22](#), [26](#), [29](#), [33](#),
[37](#), [41](#), [46](#), [53](#), [57](#), [61](#)

sqBoundary, [11](#), [12](#), [15](#), [16](#), [32](#), [33](#), [37](#), [38](#), [49](#)

sqBoundary (sqMonitor), [59](#)

sqMonitor, [10](#), [15](#), [32](#), [37](#), [49](#), [59](#)

summary.grouprar (print.grouprar), [55](#)

WeiUrn, [49](#), [55](#), [57](#), [60](#)