

Package ‘caradpt’

May 8, 2026

Type Package

Title Covariate-Adjusted Response-Adaptive Designs for Clinical Trials

Version 0.1.0

Description Tools for implementing covariate-adjusted response-adaptive procedures for binary, continuous and survival responses. Users can flexibly choose between two functions based on their specific needs for each procedure: use real patient data from clinical trials to compute allocation probabilities directly, or use built-in simulation functions to generate synthetic patient data. Detailed methodologies and algorithms used in this package are described in the following references:

Zhang, L. X., Hu, F., Cheung, S. H., & Chan, W. S. (2007) <doi:10.1214/009053606000001424>

Zhang, L. X. & Hu, F. (2009) <doi:10.1007/s11766-009-0001-6>

Hu, J., Zhu, H., & Hu, F. (2015) <doi:10.1080/01621459.2014.903846>

Zhao, W., Ma, W., Wang, F., & Hu, F. (2022) <doi:10.1002/pst.2160>

Mukherjee, A., Jana, S., & Coad, S. (2024) <doi:10.1177/09622802241287704>.

License GPL-3

Imports survival

Depends R (>= 3.6.0)

Encoding UTF-8

RoxygenNote 7.3.2

Maintainer Renjie Luo <jerry.luorj@gmail.com>

NeedsCompilation no

Author Renjie Luo [aut, cre],
Feifang Hu [aut, ctb]

Repository CRAN

Date/Publication 2025-08-28 07:30:02 UTC

Contents

CADBCD_Alloc	2
CADBCD_Alloc_Surv	4
CADBCD_Sim	6

CADBCD_Sim_Surv	7
CARAEE_Alloc	9
CARAEE_Alloc_Surv	10
CARAEE_Sim	12
CARAEE_Sim_Surv	13
WBR_Alloc	14
WBR_Alloc_Surv	16
WBR_Sim	18
WBR_Sim_Surv	20
ZhaoNew_Alloc	22
ZhaoNew_Alloc_Surv	24
ZhaoNew_Sim	26
ZhaoNew_Sim_Surv	28

Index	31
--------------	-----------

CADBCD_Alloc	<i>Allocation Function of Covariate Adjusted Doubly Biased Coin Design for Binary and Continuous Response</i>
--------------	---

Description

Calculating the probability of assigning the upcoming patient to treatment A based on the patient's covariates and the previous patients' covariates and responses for Covariate Adjusted Doubly Biased Coin procedure proposed by Zhang and Hu.

Usage

```
CADBCD_Alloc(ptsb.cov, ptsb.t, ptsb.Y, ptnow.cov, v = 2, response, target)
```

Arguments

ptsb.cov	a $n \times k$ covariate matrix of previous patients.
ptsb.t	a treatment vector of previous patients with length n .
ptsb.Y	a response vector of previous patients with length n .
ptnow.cov	a covariate vector of the incoming patient with length k .
v	a non-negative integer that controls the randomness of CADBCD design. The default value is 2.
response	the type of the response. Options are "Binary" or "Cont".
target	the type of optimal allocation target. Options are "Neyman" or "RSIHR".

Details

To start, we let θ_0 be an initial estimate of θ , and assign m_0 subjects to each treatment using a restricted randomization. Assume that m (with $m \geq 2m_0$) subjects have been assigned to treatments. Their responses $\{Y_j, j = 1, \dots, m\}$ and the corresponding covariates $\{\xi_j, j = 1, \dots, m\}$ are observed. We let $\hat{\theta}_m = (\hat{\theta}_{m,1}, \hat{\theta}_{m,2})$ be an estimate of $\theta = (\theta_1, \theta_2)$. For each $k = 1, 2$, the estimator $\hat{\theta}_{m,k} = \hat{\theta}_{m,k}(Y_{j,k}, \xi_j : X_{j,k} = 1, j = 1, \dots, m)$ is based on the observed sample of size $N_{m,k}$, that is, $\{(Y_{j,k}, \xi_j) : X_{j,k} = 1, j = 1, \dots, m\}$.

Define $\hat{\rho}_m = \frac{1}{m} \sum_{i=1}^m \pi_1(\hat{\theta}_m, \xi_i)$ and $\hat{\pi}_m = \pi_1(\hat{\theta}_m, \xi_{m+1})$. When the $(m+1)$ -th subject is ready for randomization and the corresponding covariate ξ_{m+1} is recorded, we assign the patient to treatment 1 with the probability:

$$\psi_{m+1,1} = \frac{\hat{\pi}_m \left(\frac{\hat{\rho}_m}{N_{m,1}/m} \right)^v}{\hat{\pi}_m \left(\frac{\hat{\rho}_m}{N_{m,1}/m} \right)^v + (1 - \hat{\pi}_m) \left(\frac{1 - \hat{\rho}_m}{1 - N_{m,1}/m} \right)^v}$$

and to treatment 2 with probability $\psi_{m+1,2} = 1 - \psi_{m+1,1}$, where $v \geq 0$ is a constant controlling the degree of randomness—from the most random when $v = 0$ to the most deterministic when $v \rightarrow \infty$. See Zhang and Hu(2009) for more details.

Value

prob Probability of assigning the upcoming patient to treatment A.

References

- Zhang, L. X., Hu, F., Cheung, S. H., & Chan, W. S. (2007). Asymptotic properties of covariate-adjusted response-adaptive designs. *Annals of Statistics*, 35(3), 1166–1182.
- Zhang, L. X., & Hu, F. F. (2009). A new family of covariate-adjusted response adaptive designs and their properties. *Applied Mathematics—A Journal of Chinese Universities*, 24(1), 1–13.

Examples

```
set.seed(123)

n_prev = 40
covariates = cbind(Z1 = rnorm(n_prev), Z2 = rnorm(n_prev))
treatment = sample(c(0, 1), n_prev, replace = TRUE)
response = rbinom(n_prev, size = 1, prob = 0.6)

# Simulate new incoming patient
new_patient_cov = c(Z1 = rnorm(1), Z2 = rnorm(1))

# Run allocation function
result = CADBCD_Alloc(
  ptsb.cov = covariates,
  ptsb.t = treatment,
  ptsb.Y = response,
  ptnow.cov = new_patient_cov,
  response = "Binary",
```

```

    target = "Neyman"
  )
  print(result$prob)

```

CADBCD_Alloc_Surv *Allocation Function of Covariate Adjusted Doubly Biased Coin Design for Survival Response*

Description

Calculating the probability of assigning the upcoming patient to treatment A based on the patient's covariates and the previous patients' covariates and responses for Covariate Adjusted Doubly Biased Coin procedure for survival trial.

Usage

```
CADBCD_Alloc_Surv(ptsb.cov, ptsb.t, ptsb.Y, ptsb.E, ptnow.cov, v = 2, target)
```

Arguments

ptsb.cov	a n x k covariate matrix of previous patients.
ptsb.t	a treatment vector of previous patients with length n.
ptsb.Y	a response vector of previous patients with length n.
ptsb.E	a censoring indicator vector (1 = event observed, 0 = censored)with length n.
ptnow.cov	a covariate vector of the incoming patient with length k.
v	a non-negative integer that controls the randomness of CADBCD design. The default value is 2.
target	the type of optimal allocation target. Options are "Neyman" or "RSIHR".

Details

For the first $m = 2m_0$ patients, a restricted randomization procedure is used to allocate them equally to treatments A and B.

After the initial enrollment of $2m_0$ patients, adaptive treatment assignment begins from patient $2m_0 + 1$ onward. Importantly, this adaptive assignment does *not* require complete observation of outcomes from earlier patients. Instead, at each new patient arrival, the treatment effect parameters $(\tilde{\beta}_{A,m}, \tilde{\beta}_{B,m})$ are re-estimated via partial likelihood using the accrued survival data subject to *dynamic administrative censoring*. Specifically, each previously enrolled patient is censored at the current arrival time if their event has not yet occurred. These updated estimates are then used to guide covariate-adjusted allocation.

When the $(m+1)$ -th patient enters the trial with covariate vector \mathbf{Z}_{m+1} , the probability of assigning this patient to treatment A is computed as:

$$\phi_{m+1} = g(\tilde{\beta}_{A,m}, \tilde{\beta}_{B,m}, \mathbf{Z}_{m+1}),$$

where $0 \leq g(\cdot) \leq 1$ is an appropriately chosen allocation function that favors the better-performing treatment arm.

Following Zhang and Hu's CADBCD procedure, let $N_A(m)$ and $N_B(m) = m - N_A(m)$ denote the numbers of patients allocated to treatments A and B after m assignments. Let $\tilde{\pi}_m = \pi_A(\tilde{\beta}_{A,m}, \tilde{\beta}_{B,m}, \mathbf{Z}_{m+1})$ be the target allocation probability for the incoming patient. The average target allocation proportion among the first m patients is defined as:

$$\tilde{\rho}_m = \frac{1}{m} \sum_{i=1}^m \pi_A(\tilde{\beta}_{A,m}, \tilde{\beta}_{B,m}, \mathbf{Z}_i).$$

Under the CADBCD scheme, the probability of assigning treatment A to the $(m+1)$ -th patient is given by:

$$\phi_{m+1} = \frac{\tilde{\pi}_m \left(\frac{\tilde{\rho}_m}{N_A(m)/m} \right)^v}{\tilde{\pi}_m \left(\frac{\tilde{\rho}_m}{N_A(m)/m} \right)^v + (1 - \tilde{\pi}_m) \left(\frac{1 - \tilde{\rho}_m}{N_B(m)/m} \right)^v},$$

and to treatment 2 with probability $\psi_{m+1,2} = 1 - \psi_{m+1,1}$, where $v \geq 0$ is a constant controlling the degree of randomness—from the most random when $v = 0$ to the most deterministic when $v \rightarrow \infty$. Similar procedures for survival responses are used in all other designs.

Value

prob Probability of assigning the upcoming patient to treatment A.

References

Mukherjee, A., Jana, S., & Coad, S. (2024). Covariate-adjusted response-adaptive designs for semiparametric survival models. *Statistical Methods in Medical Research*, 09622802241287704.

Examples

```
set.seed(123)
n = 40
covariates = cbind(rexp(40), rexp(40))
treatment = sample(c(0, 1), n, replace = TRUE)
survival_time = rexp(n, rate = 1)
censoring = runif(n)
event = as.numeric(survival_time < censoring)
new_patient_cov = c(Z1 = 1, Z2 = 0.5)
result = CADBCD_Alloc_Surv(
  ptsb.cov = covariates,
  ptsb.t = treatment,
  ptsb.Y = survival_time,
  ptsb.E = event,
  ptnow.cov = new_patient_cov,
  v = 2,
  target = "Neyman"
)
print(result$prob)
```

CADBCD_Sim	<i>Simulation Function of Covariate Adjusted Doubly Biased Coin Design for Binary and Continuous Response</i>
------------	---

Description

This function simulates a clinical trial using the Covariate Adjusted Doubly Biased Coin Design (CADBCD) with Binary or Continuous Responses.

Usage

```
CADBCD_Sim(n, thetaA, thetaB, m0 = 40, pts.cov, v = 2, response, target)
```

Arguments

n	a positive integer. The value specifies the total number of participants involved in each round of the simulation.
thetaA	a vector of length k+1. The true coefficient parameter value for treatment A.
thetaB	a vector of length k+1. The true coefficient parameter value for treatment B.
m0	a positive integer. The number of first 2m0 patients will be allocated equally for estimation. The default value is 40.
pts.cov	a n x k matrix. The simulated covariate matrix for patients.
v	a non-negative integer that controls the randomness of CADBCD design. The default value is 2.
response	the type of the response. Options are "Binary" or "Cont".
target	the type of optimal allocation target. Options are "Neyman" or "RSIHR".

Value

A list with the following elements:

method	The name of the procedure.
sampleSize	Total number of patients.
parameter	Estimated parameter values.
assignment	Treatment assignment vector.
proportion	Proportion of patients allocated to treatment A.
responses	Simulated response values.
failureRate	Proportion of treatment failures (if response = "Binary").
meanResponse	Mean response value (if response = "Cont").
rejectNull	Logical. Indicates whether the treatment effect is statistically significant based on a Wald test.

Examples

```
set.seed(123)
results = CADBCD_Sim(n = 400,
                    pts.cov = cbind(rnorm(400), rnorm(400)),
                    thetaA = c(-1, 1, 1),
                    thetaB = c(3, 1, 1),
                    response = "Binary",
                    target = "Neyman")

results

## view the settings
results$method
results$sampleSize

## view the simulation results
results$parameter
results$assignments
results$proportion
results$responses
results$failureRate
```

CADBCD_Sim_Surv

Simulation For Covariate Adjusted Doubly Biased Coin Design for Survival Response

Description

This function simulates a clinical trial with time-to-event (survival) outcomes using the Covariate Adjusted Doubly Biased Coin Design (CADBCD). Patient responses are generated under the Cox proportional hazards model, assuming the proportional hazards assumption holds.

Usage

```
CADBCD_Sim_Surv(
  n,
  thetaA,
  thetaB,
  m0 = 40,
  pts.cov,
  v = 2,
  target,
  censor.time,
  arrival.rate
)
```

Arguments

n	a positive integer. The value specifies the total number of participants involved in each round of the simulation.
thetaA	a vector of length k. The true coefficient parameter value for treatment A.
thetaB	a vector of length k. The true coefficient parameter value for treatment B.
m0	a positive integer. The number of first 2m0 patients will be allocated equally for estimation. The default value is 40.
pts.cov	a n x k matrix. The simulated covariate matrix for patients.
v	a non-negative integer that controls the randomness of CADBCD design. The default value is 2.
target	the type of optimal allocation target. Options are "Neyman" or "RSIHR".
sensor.time	a positive value. The upper bound to the simulated uniform censor time.
arrival.rate	a positive value. The rate of simulated exponential arrival time.

Value

A list with the following elements:

method	The name of procedure.
sampleSize	Sample size of the trial.
parameter	Estimated parameters used to do the simulations.
N.events	Total number of events of the trial.
assignment	The randomization sequence.
proportion	Average allocation proportion for treatment A.
responses	The simulated observed survival responses of patients.
events	Whether events are observed for patients(1=event,0=censored).
rejectNull	Whether the study to detect a significant difference of treatment effect using Wald test.

Examples

```
set.seed(123)

## Run CADBCD simulation with survival response
results = CADBCD_Sim_Surv(
  thetaA = c(0.1, 0.1),
  thetaB = c(-1, 0.1),
  n = 400,
  pts.cov = cbind(sample(c(1, 0), 400, replace = TRUE), rnorm(400)),
  target = "RSIHR",
  sensor.time = 2,
  arrival.rate = 150
)
```

CARAEE_Alloc	<i>Allocation Function of CARA Designs Based on Efficiency and Ethics for Binary and Continuous Response.</i>
--------------	---

Description

Calculating the probability of assigning the upcoming patient to treatment A based on the patient's covariates and the previous patients' covariates and responses for CARAEE procedure.

Usage

CARAEE_Alloc(ptsb.cov, ptsb.t, ptsb.Y, ptnow.cov, gamma, response)

Arguments

ptsb.cov	a n x k covariate matrix of previous patients.
ptsb.t	a treatment vector of previous patients with length n.
ptsb.Y	a response vector of previous patients with length n.
ptnow.cov	a covariate vector of the incoming patient with length k.
gamma	a non-negative number. A tuning parameter that reflects the importance of the efficiency component compared to the ethics component.
response	the type of the response. Options are "Binary" or "Cont".

Details

Covariate-Adjusted Response-Adaptive with Ethics and Efficiency (CARAEE) Design: The CARAEE procedure balances both ethical considerations and statistical efficiency when assigning subjects to treatments.

As a start-up rule, m_0 subjects are assigned to each treatment using a balanced randomization scheme.

Assume that $m \geq 2m_0$ subjects have been assigned, and their responses $\{\mathbf{X}_i, i = 1, \dots, m\}$ and covariates $\{\mathbf{Z}_i, i = 1, \dots, m\}$ are observed. Let $\hat{\boldsymbol{\theta}}(m) = (\hat{\theta}_1(m), \hat{\theta}_2(m))$, where $\hat{\theta}_k(m)$ is the maximum likelihood estimate of the treatment-specific parameter θ_k based on the data for treatment group k .

For the incoming subject $(m + 1)$ with covariates \mathbf{Z}_{m+1} , we define the efficiency and ethics measures for each treatment as: $\mathbf{d}(\mathbf{Z}, \boldsymbol{\theta}) = (d_1(\mathbf{Z}, \boldsymbol{\theta}), d_2(\mathbf{Z}, \boldsymbol{\theta}))$, $\mathbf{e}(\mathbf{Z}, \boldsymbol{\theta}) = (e_1(\mathbf{Z}, \boldsymbol{\theta}), e_2(\mathbf{Z}, \boldsymbol{\theta}))$.

The allocation probability of assigning subject $(m + 1)$ to treatment 1 is given by:

$$\phi_{m+1}(\mathbf{Z}_{m+1}, \hat{\boldsymbol{\theta}}(m)) = \frac{e_1(\mathbf{Z}_{m+1}, \hat{\boldsymbol{\theta}}(m)) \cdot d_1^\gamma(\mathbf{Z}_{m+1}, \hat{\boldsymbol{\theta}}(m))}{e_1(\mathbf{Z}_{m+1}, \hat{\boldsymbol{\theta}}(m)) \cdot d_1^\gamma(\mathbf{Z}_{m+1}, \hat{\boldsymbol{\theta}}(m)) + e_2(\mathbf{Z}_{m+1}, \hat{\boldsymbol{\theta}}(m)) \cdot d_2^\gamma(\mathbf{Z}_{m+1}, \hat{\boldsymbol{\theta}}(m))}.$$

This allocation rule is scale-invariant in both efficiency and ethics components due to its ratio-based form. The tuning parameter $\gamma \geq 0$ controls the trade-off between the two: when $\gamma = 0$, the assignment is based purely on ethical considerations; larger values of γ increase the emphasis on statistical efficiency. More details can be found in Hu, Zhu & Zhang(2015).

Value

prob Probability of assigning the upcoming patient to treatment A for binary and continuous response.

References

Hu, J., Zhu, H., & Hu, F. (2015). A unified family of covariate-adjusted response-adaptive designs based on efficiency and ethics. *Journal of the American Statistical Association*, 110(509), 357–367.

Examples

```
set.seed(123)

n_prev = 40
covariates = cbind(Z1 = rnorm(n_prev), Z2 = rnorm(n_prev))
treatment = sample(c(0, 1), n_prev, replace = TRUE)
response = rbinom(n_prev, size = 1, prob = 0.6)

# Simulate new incoming patient
new_patient_cov = c(Z1 = rnorm(1), Z2 = rnorm(1))

# Run allocation function
result = CARAEE_Alloc(
  ptsb.cov = covariates,
  ptsb.t = treatment,
  ptsb.Y = response,
  ptnow.cov = new_patient_cov,
  response = "Binary",
  gamma=1
)
print(result$prob)
```

CARAEE_Alloc_Surv

*Allocation Function of CARA Designs Based on Efficiency and Ethics
for Survival Response*

Description

Calculating the probability of assigning the upcoming patient to treatment A based on the patient's covariates and the previous patients' covariates and responses using CARA Designs Based on Efficiency and Ethics for survival trial.

Usage

```
CARAEE_Alloc_Surv(
  ptsb.cov,
  ptsb.t,
  ptsb.Y,
```

```

    ptsb.E,
    ptnow.cov,
    gamma,
    event.prob
  )

```

Arguments

ptsb.cov	a $n \times k$ covariate matrix of previous patients.
ptsb.t	a treatment vector of previous patients with length n .
ptsb.Y	a response vector of previous patients with length n .
ptsb.E	a censoring indicator vector (1 = event observed, 0 = censored) with length n .
ptnow.cov	a covariate vector of the incoming patient with length k .
gamma	a non-negative number. A tuning parameter that reflects the importance of the efficiency component compared to the ethics component.
event.prob	a vector with length 2. The probability of events of upcoming patient for two treatments.

Value

prob	Probability of assigning the upcoming patient to treatment A.
------	---

Examples

```

set.seed(123)
n = 40
covariates = cbind(rexp(40), rexp(40))
treatment = sample(c(0, 1), n, replace = TRUE)
survival_time = rexp(n, rate = 1)
censoring = runif(n)
event = as.numeric(survival_time < censoring)

new_patient_cov = c(Z1 = 1, Z2 = 0.5)

result = CARAEE_Alloc_Surv(
  ptsb.cov = covariates,
  ptsb.t = treatment,
  ptsb.Y = survival_time,
  ptsb.E = event,
  ptnow.cov = new_patient_cov,
  gamma=1,
  event.prob = c(0.5, 0.7)
)
print(result$prob)

```

CARAEE_Sim	<i>Simulation Function of of CARA Designs Based on Efficiency and Ethics for Binary and Continuous Response.</i>
------------	--

Description

This function simulates a clinical trial using CARA Designs Based on Efficiency and Ethics (CARAEE) with Binary or Continuous Responses.

Usage

```
CARAEE_Sim(n, thetaA, thetaB, m0 = 40, pts.cov, response, gamma)
```

Arguments

n	a positive integer. The value specifies the total number of participants involved in each round of the simulation.
thetaA	a vector of length $k+1$. The true coefficient parameter value for treatment A.
thetaB	a vector of length $k+1$. The true coefficient parameter value for treatment B.
m0	a positive integer. The number of first $2m_0$ patients will be allocated equally for estimation. The default value is 40.
pts.cov	a $n \times k$ matrix. The simulated covariate matrix for patients.
response	the type of the response. Options are "Binary" or "Cont".
gamma	a non-negative number. A tuning parameter that reflects the importance of the efficiency component compared to the ethics component.

Value

A list with the following elements:

method	The name of the procedure.
sampleSize	Total number of patients.
parameter	Estimated parameter values.
assignment	Treatment assignment vector.
proportion	Proportion of patients allocated to treatment A.
responses	Simulated response values.
failureRate	Proportion of treatment failures (if response = "Binary").
meanResponse	Mean response value (if response = "Cont").
rejectNull	Logical. Indicates whether the treatment effect is statistically significant based on a Wald test.

Examples

```

set.seed(123)
results = CARAEE_Sim(n = 400,
                    pts.cov = cbind(rnorm(400), rnorm(400)),
                    thetaA = c(-1, 1, 1),
                    thetaB = c(3, 1, 1),
                    response = "Binary",
                    gamma=1)

```

CARAEE_Sim_Surv	<i>Simulation Function of of CARA Designs Based on Efficiency and Ethics for Survival Response.</i>
-----------------	---

Description

This function simulates a clinical trial with time-to-event (survival) outcomes using the CARA Designs Based on Efficiency and Ethics for Survival Response(CARAEE). Patient responses are generated under the Cox proportional hazards model, assuming the proportional hazards assumption holds.

Usage

```

CARAEE_Sim_Surv(
  n,
  thetaA,
  thetaB,
  m0 = 40,
  pts.cov,
  gamma,
  censor.time,
  arrival.rate
)

```

Arguments

n	a positive integer. The value specifies the total number of participants involved in each round of the simulation.
thetaA	a vector of length k. The true coefficient parameter value for treatment A.
thetaB	a vector of length k. The true coefficient parameter value for treatment B.
m0	a positive integer. The number of first 2m0 patients will be allocated equally for estimation. The default value is 40.
pts.cov	a n x k matrix. The simulated covariate matrix for patients.
gamma	a non-negative number. A tuning parameter that reflects the importance of the efficiency component compared to the ethics component.
censor.time	a positive value. The upper bound to the simulated uniform censor time.
arrival.rate	a positive value. The rate of simulated exponential arrival time.

Value

A list with the following elements:

method	The name of procedure.
sampleSize	Sample size of the trial.
parameter	Estimated parameters used to do the simulations.
N.events	Total number of events of the trial.
assignment	The randomization sequence.
proportion	Average allocation proportion for treatment A.
responses	The simulated observed survival responses of patients.
events	Whether events are observed for patients(1=event,0=censored).
rejectNull	Logical. Indicates whether the treatment effect is statistically significant based on a Wald test.

Examples

```
set.seed(123)

results = CARAEE_Sim_Surv(
  thetaA = c(0.1, 0.1),
  thetaB = c(-1, 0.1),
  n = 400,
  pts.cov = cbind(sample(c(1, 0), 400, replace = TRUE), rnorm(400)),
  gamma=1,
  censor.time = 2,
  arrival.rate = 150
)
```

WBR_Alloc

Allocation Function of Weighted Balance Ratio Design for Binary and Continuous Response

Description

Calculating the probability of assigning the upcoming patient to treatment A based on the patient's covariates and the previous patients' covariates and responses for Weighted Balance Ratio procedure for binary and continuous response.

Usage

```
WBR_Alloc(
  ptsb.X,
  ptsb.Z,
  ptsb.Y,
  ptsb.t,
```

```

    ptnow.X,
    ptnow.Z,
    weight,
    v = 2,
    response
)

```

Arguments

ptsb.X	a vector of length n of the predictive covariates of previous patients. Must be binary.
ptsb.Z	a n x k of the prognostic covariates of previous patients. Must be binary.
ptsb.Y	a vector of length n of the responses of previous patients.
ptsb.t	a vector of length n of the treatment allocation of previous patients.
ptnow.X	a binary value of the predictive covariate of the present patient.
ptnow.Z	a vector of length k of the binary prognostic covariate of the present patient.
weight	a vector of length 2+k. The weight of balance ratio in overall,margin and stratum levels.
v	a positive value that controls the randomness of allocation probability function.
response	the type of the response. Options are "Binary" or "Cont".

Details

This function implements a covariate-adjusted response-adaptive design using a weighted balancing ratio rule (WBR) combined with a DBCD-type allocation function.

The first two steps follow Zhao et al. (2022): the first $2K$ patients are randomized using restricted randomization with K patients in each treatment group. For patient $n > 2K$, suppose the prognostic covariates \mathbf{Z}_n fall into stratum (k_1^*, \dots, k_J^*) .

Define the weighted imbalance ratio $r_{n-1}(\mathbf{X}_n)$ for patient n as:

$$r_{n-1}(\mathbf{X}_n) = w_o \frac{N_{n-1}(\mathbf{X}_n)}{N_{n-1}(\mathbf{X}_n)} + \sum_{j=1}^J w_{m,j} \frac{N_{(j;k_j^*),n-1}(\mathbf{X}_n)}{N_{(j;k_j^*),n-1}(\mathbf{X}_n)} + w_s \frac{N_{(k_1^*, \dots, k_J^*),n-1}(\mathbf{X}_n)}{N_{(k_1^*, \dots, k_J^*),n-1}(\mathbf{X}_n)},$$

where $w_o, w_{m,1}, \dots, w_{m,J}, w_s$ are non-negative weights that sum to 1.

Based on the patient's covariates, the target allocation probability $\hat{\rho}_{n-1}(\mathbf{X}_n)$ is estimated from previous data.

The final probability of assigning patient n to treatment k is computed using a CARA-type allocation function from Hu and Zhang (2009):

$$\phi_{n,k}(\mathbf{X}_n) = g_k(r_{n-1}(\mathbf{X}_n), \hat{\rho}_{n-1}(\mathbf{X}_n)) = \frac{\hat{\rho}_{n-1} \left(\frac{\hat{\rho}_{n-1}}{r_{n-1}} \right)^v}{\hat{\rho}_{n-1} \left(\frac{\hat{\rho}_{n-1}}{r_{n-1}} \right)^v + (1 - \hat{\rho}_{n-1}) \left(\frac{1 - \hat{\rho}_{n-1}}{1 - r_{n-1}} \right)^v},$$

where $v \geq 0$ is a tuning parameter controlling the degree of randomness. A value of $v = 2$ is commonly recommended.

This approach combines stratified covariate balancing with covariate-adjusted optimal targeting. More details can be found in Yu(2025).

Value

prob Probability of assigning the upcoming patient to treatment A.

References

Yu, J. (2025). A New Family of Covariate-Adjusted Response-Adaptive Randomization Procedures for Precision Medicine (Doctoral dissertation, The George Washington University).

Examples

```
set.seed(123)

# Generate historical data for 400 patients
ptsb.X = sample(c(1, -1), 400, replace = TRUE) # predictive covariate
ptsb.Z = cbind(
  sample(c(1, -1), 400, replace = TRUE),      # prognostic covariate 1
  sample(c(1, -1), 400, replace = TRUE)      # prognostic covariate 2
)
ptsb.Y = sample(c(1, 0), 400, replace = TRUE) # binary responses
ptsb.t = sample(c(1, 0), 400, replace = TRUE) # treatment assignments

# Incoming patient covariates
ptnow.X = 1
ptnow.Z = c(1, -1)

# Calculate allocation probability using WBR method
result = WBR_Alloc(
  ptsb.X = ptsb.X,
  ptsb.Z = ptsb.Z,
  ptsb.Y = ptsb.Y,
  ptsb.t = ptsb.t,
  ptnow.X = ptnow.X,
  ptnow.Z = ptnow.Z,
  weight = rep(0.25, 4),
  response = "Binary"
)

# View probability of assigning to treatment A
result$prob
```

WBR_Alloc_Surv

Allocation Function of Weighted Balance Ratio Design for Survival Response

Description

Calculating the probability of assigning the upcoming patient to treatment A based on the patient's covariates and the previous patients' covariates and responses for Weighted Balance Ratio procedure for survival response.

Usage

```
WBR_Alloc_Surv(
  ptsb.X,
  ptsb.Z,
  ptsb.Y,
  ptsb.t,
  ptsb.E,
  ptnow.X,
  ptnow.Z,
  v = 2,
  weight
)
```

Arguments

ptsb.X	a vector of length n of the predictive covariates of previous patients. Must be binary.
ptsb.Z	a n x k of the prognostic covariates of previous patients. Must be binary.
ptsb.Y	a vector of length n of the responses of previous patients.
ptsb.t	a vector of length n of the treatment allocation of previous patients.
ptsb.E	a vector of length n with value 1 or 0 of the status of event and censoring.
ptnow.X	a binary value of the predictive covariate of the present patient.
ptnow.Z	a vector of length k of the binary prognostic covariate of the present patient.
v	a positive value that controls the randomness of allocation probability function.
weight	a vector of length 2+k. The weight of balance ratio in overall,margin and stratum levels.

Value

prob	Probability of assigning the upcoming patient to treatment A.
------	---

Examples

```
set.seed(123)

# Generate historical data for 400 patients
ptsb.X = sample(c(1, -1), 400, replace = TRUE) # predictive covariate
ptsb.Z = cbind(
  sample(c(1, -1), 400, replace = TRUE),      # prognostic covariate 1
  sample(c(1, -1), 400, replace = TRUE)      # prognostic covariate 2
)
ptsb.Y = rexp(400, rate = 1)                 # observed time
ptsb.E = sample(c(1, 0), 400, replace = TRUE) # event indicator (1=event, 0=censored)
ptsb.t = sample(c(1, 0), 400, replace = TRUE) # treatment assignments

# Incoming patient covariates
ptnow.X = 1
```

```

ptnow.Z = c(1, -1)

# Calculate allocation probability using WBR for survival response
result = WBR_Alloc_Surv(
  ptsb.X = ptsb.X,
  ptsb.Z = ptsb.Z,
  ptsb.Y = ptsb.Y,
  ptsb.E = ptsb.E,
  ptsb.t = ptsb.t,
  ptnow.X = ptnow.X,
  ptnow.Z = ptnow.Z,
  weight = rep(0.25, 4)
)

# View probability of assigning to treatment A
result$prob

```

WBR_Sim

Simulation Function of Weighted Balance Ratio Design for Binary and Continuous Response

Description

This function simulates a trial using Weighted Balance Ratio design for binary and continuous responses.

Usage

```
WBR_Sim(n, mu, beta, gamma, m0 = 40, pts.X, pts.Z, response, weight, v = 2)
```

Arguments

n	a positive integer. The sample size of the simulated data.
mu	a vector of length 2. The true parameters of treatment.
beta	a vector of length 2. The true parameters of predictive covariate and interaction with treatment.
gamma	a vector of length k. The true parameters of prognostic covariates.
m0	a positive integer. The number of first 2m0 patients will be allocated equally to both treatments.
pts.X	a vector of length n. The vector of patients' binary predictive covariates.
pts.Z	a matrix of n x k. The matrix of patients' binary prognostic covariates.
response	the type of the response. Options are "Binary" or "Cont".
weight	a vector of length 2+k. The weight of balance ratio in overall, margin and stratum levels.
v	a positive value that controls the randomness of allocation probability function.

Value

method	The name of procedure.
sampleSize	The sample size of the trial.
assignment	The randomization sequence.
X1proportion	Average allocation proportion for treatment A when predictive covariate equals the smaller value.
X2proportion	Average allocation proportion for treatment A when predictive covariate equals the larger value.
proportion	Average allocation proportion for treatment A.
failureRate	Proportion of treatment failures (if response = "Binary").
meanResponse	Mean response value (if response = "Cont").
#'	
rejectNull	Logical. Indicates whether the treatment effect is statistically significant based on a Wald test.

Examples

```

set.seed(123)

# Simulation settings
n = 400                                # total number of patients
mu = c(0.8, 0.8)                       # treatment effects (muA, muB)
beta = c(0.8, -0.8)                    # predictive effect and interaction
gamma = c(0.8, 0.8)                    # prognostic covariate effects
weight = rep(0.25, 4)                  # weights for imbalance components

# Generate patient covariates
pts.X = sample(c(1, -1), n, replace = TRUE) # predictive covariate
pts.Z = cbind(
  sample(c(1, -1), n, replace = TRUE),    # prognostic Z1
  sample(c(1, -1), n, replace = TRUE)    # prognostic Z2
)

# Run simulation for continuous response
result = WBR_Sim(
  n = n,
  mu = mu,
  beta = beta,
  gamma = gamma,
  pts.X = pts.X,
  pts.Z = pts.Z,
  response = "Cont",
  weight = weight
)

```

WBR_Sim_Surv

Simulation Function of Weighted Balance Ratio Design for Survival Response

Description

This function simulates a trial using Weighted Balance Ratio design for survival responses.

Usage

```
WBR_Sim_Surv(
  n,
  mu,
  beta,
  gamma,
  m0 = 40,
  pts.X,
  pts.Z,
  censor.time,
  arrival.rate,
  weight,
  v = 2
)
```

Arguments

n	a number. The sample size of the simulated data.
mu	a number. The true parameters of treatment effect.
beta	a vector of length 2. The true parameters of predictive covariate and interaction with treatment.
gamma	a vector of length k. The true parameters of prognostic covariates.
m0	a positive integer. The number of first 2m0 patients will be allocated equally to both treatments.
pts.X	a vector of length n. The vector of patients' binary predictive covariates.
pts.Z	a matrix of n x k. The matrix of patients' binary prognostic covariates.
censor.time	a positive number. The upper bound of the uniform censor time in year.
arrival.rate	a positive integer. The arrival rate of patients each year.
weight	a vector of length 2+k. The weight of balance ratio in overall, margin and stratum levels.
v	a positive value that controls the randomness of allocation probability function.

Value

A list with the following elements:

method	The name of procedure.
sampleSize	The sample size of the trial.
assignment	The randomization sequence.
X1proportion	Average allocation proportion for treatment A when predictive covariate equals the smaller value.
X2proportion	Average allocation proportion for treatment A when predictive covariate equals the larger value.
proportion	Average allocation proportion for treatment A.
N.events	Total number of events occurred of the trial.
responses	Observed survival responses of patients.
events	Survival status vector of patients(1=event,0=censored)
rejectNull	Logical. Indicates whether the treatment effect is statistically significant based on a Wald test.

Examples

```

set.seed(123)

# Simulation settings
n = 400                                # total number of patients
mu = 0.5                                # treatment effect (log hazard ratio)
beta = c(0.5, -0.5)                     # predictive effect and interaction
gamma = c(0.5, 0.5)                     # prognostic covariate effects
censor.time = 2                          # maximum censoring time (years)
arrival.rate = 1.5                       # arrival rate per year
weight = rep(0.25, 4)                   # imbalance weights for overall, margins, and stratum

# Generate patient covariates
pts.X = sample(c(1, -1), n, replace = TRUE) # predictive covariate
pts.Z = cbind(
  sample(c(1, -1), n, replace = TRUE),    # prognostic Z1
  sample(c(1, -1), n, replace = TRUE)    # prognostic Z2
)

# Run simulation for survival outcome
result = WBR_Sim_Surv(
  n = n,
  mu = mu,
  beta = beta,
  gamma = gamma,
  pts.X = pts.X,
  pts.Z = pts.Z,
  censor.time = censor.time,
  arrival.rate = arrival.rate,
  weight = weight

```

)

ZhaoNew_Alloc

Allocation Function of Zhao's New Design for Binary and Continuous Response

Description

Calculating the probability of assigning the upcoming patient to treatment A based on the patient's predictive covariates and the previous patients' predictive covariates and responses for Zhao's New procedure.

Usage

```
ZhaoNew_Alloc(
  ptsb.X,
  ptsb.Z,
  ptsb.t,
  ptsb.Y,
  ptnow.X,
  ptnow.Z,
  response,
  omega,
  p = 0.8
)
```

Arguments

ptsb.X	a vector of length n of the predictive covariates of previous patients. Must be binary.
ptsb.Z	a n x k of the prognostic covariates of previous patients. Must be binary.
ptsb.t	a vector of length n of the treatment allocation of previous patients.
ptsb.Y	a vector of length n of the responses of previous patients.
ptnow.X	a binary value of the predictive covariate of the present patient.
ptnow.Z	a vector of length k of the prognostic covariate of the present patient.
response	the type of the response. Options are "Binary" or "Cont".
omega	a vector of length 2+k. The weight of imbalance.
p	a positive value between 0.75 and 0.95. The probability parameter of Efron's biased coin design.

Details

This function implements a stratified covariate-adjusted response-adaptive design that balances treatment allocation within and across strata defined by prognostic covariates.

The first $2K$ patients are randomized using a restricted randomization procedure, with K patients assigned to each treatment. For patient $n > 2K$, let \mathbf{X}_n denote predictive covariates, and \mathbf{Z}_n denote stratification covariates, placing the patient into stratum (k_1^*, \dots, k_I^*) .

Based on the covariate profiles and responses of the first $n - 1$ patients, we estimate the target allocation probability $\hat{\rho}(\mathbf{X}_n)$ for the current patient.

If assigned to treatment 1, we compute imbalance measures between actual and target allocation at three levels:

- Overall imbalance: $D_n^{(1)}(\mathbf{X}_n)$,
- Marginal imbalance: $D_n^{(1)}(i; k_i^*; \mathbf{X}_n)$,
- Within-stratum imbalance: $D_n^{(1)}(k_1^*, \dots, k_I^*; \mathbf{X}_n)$.

These are combined into a weighted imbalance function:

$$\text{Imb}_n^{(1)}(\mathbf{X}_n) = w_o[D_n^{(1)}(\mathbf{X}_n)]^2 + \sum_{i=1}^I w_{m,i}[D_n^{(1)}(i; k_i^*; \mathbf{X}_n)]^2 + w_s[D_n^{(1)}(k_1^*, \dots, k_I^*; \mathbf{X}_n)]^2.$$

A similar imbalance $\text{Imb}_n^{(2)}(\mathbf{X}_n)$ is defined for treatment 2. The patient is then assigned to treatment 1 with probability:

$$\phi_n = g\left(\text{Imb}_n^{(1)}(\mathbf{X}_n) - \text{Imb}_n^{(2)}(\mathbf{X}_n)\right),$$

where $g(x)$ is a biasing function satisfying $g(-x) = 1 - g(x)$ and $g(x) < 0.5$ for $x \geq 0$. One common choice is Efron's biased coin function:

$$g(x) = \begin{cases} q, & \text{if } x > 0 \\ 0.5, & \text{if } x = 0 \\ p, & \text{if } x < 0 \end{cases}$$

with $p > 0.5$ and $q = 1 - p$.

This design unifies covariate-adjusted response-adaptive randomization and marginal/stratified balance. It reduces to Hu & Hu's design when \mathbf{X}_n is excluded, and to CARA designs when \mathbf{Z}_n is ignored. More detail can be found in Zhao et al.(2022).

Value

prob Probability of assigning the upcoming patient to treatment A.

References

Zhao, W., Ma, W., Wang, F., & Hu, F. (2022). Incorporating covariates information in adaptive clinical trials for precision medicine. *Pharmaceutical Statistics*, 21(1), 176–195.

Examples

```

set.seed(123)
ptsb.X = sample(c(1, -1), 400, replace = TRUE)
ptsb.Z = cbind(
  sample(c(1, -1), 400, replace = TRUE),
  sample(c(1, -1), 400, replace = TRUE)
)
ptsb.Y = sample(c(1, 0), 400, replace = TRUE)
ptsb.t = sample(c(1, 0), 400, replace = TRUE)

## Incoming patient
ptnow.X = 1
ptnow.Z = c(1, -1)

## Run allocation probability calculation
prob = ZhaoNew_Alloc(
  ptsb.X = ptsb.X,
  ptsb.Z = ptsb.Z,
  ptsb.Y = ptsb.Y,
  ptsb.t = ptsb.t,
  ptnow.X = ptnow.X,
  ptnow.Z = ptnow.Z,
  response = "Binary",
  omega = rep(0.25, 4),
  p = 0.8
)

## View allocation probability for treatment A
prob

```

ZhaoNew_Alloc_Surv *Allocation Function of Zhao's Design for Survival Response*

Description

Calculating the probability of assigning the upcoming patient to treatment A based on the patient's covariates and the previous patients' covariates and responses for Zhao's New procedure for survival trials.

Usage

```

ZhaoNew_Alloc_Surv(
  ptsb.X,
  ptsb.Z,
  ptsb.t,
  ptsb.Y,
  ptsb.E,
  ptnow.X,
  ptnow.Z,

```

```

    omega,
    p = 0.8
)

```

Arguments

ptsb.X	a vector of length n of the predictive covariates of previous patients. Must be binary.
ptsb.Z	a n x k of the prognostic covariates of previous patients. Must be binary.
ptsb.t	a vector of length n of the treatment allocation of previous patients.
ptsb.Y	a vector of length n of the responses of previous patients.
ptsb.E	a vector of length n with value 1 or 0 of the status of event and censoring.
ptnow.X	a binary value of the predictive covariate of the present patient.
ptnow.Z	a vector of length k of the binary prognostic covariate of the present patient.
omega	a vector of length 2+k. The weight of imbalance.
p	a positive value between 0.75 and 0.95. The probability parameter of Efron's biased coin design.

Value

prob	Probability of assigning the upcoming patient to treatment A.
------	---

Examples

```

set.seed(123)

# Generate historical data for 400 patients
ptsb.X = sample(c(1, -1), 400, replace = TRUE) # predictive covariate
ptsb.Z = cbind(
  sample(c(1, -1), 400, replace = TRUE),      # prognostic covariate 1
  sample(c(1, -1), 400, replace = TRUE)      # prognostic covariate 2
)
ptsb.Y = rexp(400, rate = 1)                  # survival time (response)
ptsb.E = sample(c(1, 0), 400, replace = TRUE) # event indicator (1 = event, 0 = censored)
ptsb.t = sample(c(1, 0), 400, replace = TRUE) # treatment assignment

# Incoming patient covariates
ptnow.X = 1
ptnow.Z = c(1, -1)

# Allocation probability calculation
prob = ZhaoNew_Alloc_Surv(
  ptsb.X = ptsb.X,
  ptsb.Z = ptsb.Z,
  ptsb.Y = ptsb.Y,
  ptsb.E = ptsb.E,
  ptsb.t = ptsb.t,
  ptnow.X = ptnow.X,
  ptnow.Z = ptnow.Z,

```

```

    omega = rep(0.25, 4),
    p = 0.8
  )

  # View the allocation probability for treatment A
  prob

```

 ZhaoNew_Sim

Simulation Function of Zhao's New Design for Binary and Continuous Response

Description

This function simulates a trial using Zhao's new design for binary and continuous responses.

Usage

```

ZhaoNew_Sim(
  n,
  mu,
  beta,
  gamma,
  m0 = 40,
  pts.X,
  pts.Z,
  response,
  omega,
  p = 0.8
)

```

Arguments

n	a positive integer. The sample size of the simulated data.
mu	a vector of length 2. The true parameters of treatment.
beta	a vector of length 2. The true parameters of predictive covariate and interaction with treatment.
gamma	a vector of length k. The true parameters of prognostic covariates.
m0	a positive integer. The number of first 2m0 patients will be allocated equally to both treatments.
pts.X	a vector of length n. The vector of patients' binary predictive covariates.
pts.Z	a matrix of n x k. The matrix of patients' binary prognostic covariates.
response	the type of the response. Options are "Binary" or "Cont".
omega	a vector of length 2+k. The weight of imbalance.
p	a positive value between 0.75 and 0.95. The probability parameter of Efron's biased coin design.

Value

method	The name of procedure.
sampleSize	The sample size of the trial.
assignment	The randomization sequence.
X1proportion	Average allocation proportion for treatment A when predictive covariate equals the smaller value.
X2proportion	Average allocation proportion for treatment A when predictive covariate equals the larger value.
proportion	Average allocation proportion for treatment A.
failureRate	Proportion of treatment failures (if response = "Binary").
meanResponse	Mean response value (if response = "Cont").
rejectNull	Logical. Indicates whether the treatment effect is statistically significant based on a Wald test.

Examples

```

set.seed(123)

# Simulation settings
n = 400                # total number of patients
m0 = 40               # initial burn-in sample size
mu = c(0.5, 0.8)      # potential means (for continuous or logistic link)
beta = c(1, 1)        # treatment effect and predictive covariate effect
gamma = c(0.1, 0.5)   # prognostic covariate effects
omega = rep(0.25, 4)  # imbalance weights
p = 0.8               # biased coin probability

# Generate patient covariates
pts.X = sample(c(1, -1), n, replace = TRUE) # predictive covariate
pts.Z = cbind(
  sample(c(1, -1), n, replace = TRUE),      # prognostic Z1
  sample(c(1, -1), n, replace = TRUE)      # prognostic Z2
)

# Run the simulation (binary response setting)
result = ZhaoNew_Sim(
  n = n,
  mu = mu,
  beta = beta,
  gamma = gamma,
  m0 = m0,
  pts.X = pts.X,
  pts.Z = pts.Z,
  response = "Binary",
  omega = omega,
  p = p
)

```

Description

This function simulates a clinical trial using Zhao's New design for survival responses.

Usage

```
ZhaoNew_Sim_Surv(
  n,
  mu,
  beta,
  gamma,
  m0 = 40,
  pts.X,
  pts.Z,
  censor.time,
  arrival.rate,
  omega,
  p = 0.8
)
```

Arguments

n	a positive integer. The sample size of the simulated data.
mu	a number. The true parameters of treatment.
beta	a vector of length 2. The true parameters of predictive covariate and interaction with treatment.
gamma	a vector of length k. The true parameters of prognostic covariates.
m0	a positive integer. The number of first 2m0 patients will be allocated equally to both treatments.
pts.X	a vector of length n. The vector of patients' binary predictive covariates. Must be binary.
pts.Z	a matrix of n x k. The matrix of patients' binary prognostic covariates. Must be binary.
censor.time	a positive number. The upper bound of the uniform censor time in year.
arrival.rate	a positive integer. The arrival rate of patients each year.
omega	a vector of length 2+k. The weight of imbalance.
p	a positive value between 0.75 and 0.95. The probability parameter of Efron's biased coin design.

Value

A list with the following elements:

method	The name of procedure.
sampleSize	The sample size of the trial.
assignment	The randomization sequence.
X1proportion	Average allocation proportion for treatment A when predictive covariate equals the smaller value.
X2proportion	Average allocation proportion for treatment A when predictive covariate equals the larger value.
proportion	Average allocation proportion for treatment A.
N.events	Total number of events occurred of the trial.
responses	Observed survival responses of patients.
events	Survival status vector of patients(1=event,0=censored)
rejectNull	Logical. Indicates whether the treatment effect is statistically significant based on a Wald test.

Examples

```

set.seed(123)

# Simulation settings
n = 400                # total number of patients
m0 = 40                # initial burn-in sample size
mu = 0.5              # potential means (for continuous or logistic link)
beta = c(1, 1)         # treatment effect and predictive covariate effect
gamma = c(0.1, 0.5)   # prognostic covariate effects
omega = rep(0.25, 4)  # imbalance weights
p = 0.8                # biased coin probability
censor.time = 2       # maximum censoring time
arrival.rate = 150     # arrival rate of patients

# Generate patient covariates
pts.X = sample(c(1, -1), n, replace = TRUE) # predictive covariate
pts.Z = cbind(
  sample(c(1, -1), n, replace = TRUE),      # prognostic Z1
  sample(c(1, -1), n, replace = TRUE)      # prognostic Z2
)

# Run the simulation (binary response setting)
result = ZhaoNew_Sim_Surv(
  n = n,
  mu = mu,
  beta = beta,
  gamma = gamma,
  m0 = m0,
  pts.X = pts.X,
  pts.Z = pts.Z,
  omega = omega,

```

```
p = p,  
  censor.time = censor.time,  
  arrival.rate = arrival.rate  
)
```

Index

- * **CADBCD Design**
 - CADBCD_Alloc, 2
 - CADBCD_Alloc_Surv, 4
 - CADBCD_Sim, 6
 - CADBCD_Sim_Surv, 7
- * **CARAEE Design**
 - CARAEE_Alloc, 9
 - CARAEE_Alloc_Surv, 10
 - CARAEE_Sim, 12
 - CARAEE_Sim_Surv, 13
- * **Weighted Balance Ratio Design**
 - WBR_Alloc, 14
 - WBR_Alloc_Surv, 16
 - WBR_Sim, 18
 - WBR_Sim_Surv, 20
- * **Zhao's New Design**
 - ZhaoNew_Alloc, 22
 - ZhaoNew_Alloc_Surv, 24
 - ZhaoNew_Sim, 26
 - ZhaoNew_Sim_Surv, 28

CADBCD_Alloc, 2
CADBCD_Alloc_Surv, 4
CADBCD_Sim, 6
CADBCD_Sim_Surv, 7
CARAEE_Alloc, 9
CARAEE_Alloc_Surv, 10
CARAEE_Sim, 12
CARAEE_Sim_Surv, 13

WBR_Alloc, 14
WBR_Alloc_Surv, 16
WBR_Sim, 18
WBR_Sim_Surv, 20

ZhaoNew_Alloc, 22
ZhaoNew_Alloc_Surv, 24
ZhaoNew_Sim, 26
ZhaoNew_Sim_Surv, 28