

Package ‘DTEBOP2’

July 21, 2025

Title Bayesian Optimal Phase II Randomized Clinical Trial Design with Delayed Outcomes

Version 1.0.3

Description Implements a Bayesian Optimal Phase II design (DTE-BOP2) for trials with delayed treatment effects, particularly relevant to immunotherapy studies where treatment benefits may emerge after a delay. The method builds upon the BOP2 framework and incorporates uncertainty in the delay timepoint through a truncated gamma prior, informed by expert knowledge or default settings. Supports two-arm trial designs with functionality for sample size determination, interim and final analyses, and comprehensive simulation under various delay and design scenarios. Ensures rigorous type I and II error control while improving trial efficiency and power when the delay effect is present. A manuscript describing the methodology is under development and will be formally referenced upon publication.

License GPL-3

Encoding UTF-8

RoxygenNote 7.3.2

Imports truncdist, doParallel, foreach, parallel, invgamma

Suggests knitr, rmarkdown, survRM2adapt, ggplot2, reconstructKM, survival, survminer

LazyData true

VignetteBuilder knitr

NeedsCompilation no

Author Zhongheng Cai [aut, cre],
Haitao Pan [aut]

Maintainer Zhongheng Cai <zhonghengcai123@gmail.com>

Depends R (>= 3.5.0)

Repository CRAN

Date/Publication 2025-05-09 10:40:06 UTC

Contents

conduct 2

event_fun	4
Experimental	5
generate_pe	5
get.optimal_2arm_piecewise	6
getoc_2arm_piecewise	9
rtrunc_gamma	11
SOC	12
trunc_gamma_para	12
TTOT_fun	13
Two_stage_sample_size	14
Index	17

conduct	<i>Computes the hazard ratio between post- and pre-separation periods, provides the 95% Bayesian credible interval, and makes a Go/No-Go decision based on the trial data. This function is intended for interim and final analyses in two-arm survival trials with delayed treatment effects.</i>
---------	--

Description

Computes the hazard ratio between post- and pre-separation hazard rates, along with the 95% Bayesian credible interval. Based on the user-supplied survival data, the function also provides a Go/No-Go decision for interim or final analysis in two-arm survival trials with delayed treatment effects

Usage

```
conduct(  
  data_E,  
  data_C,  
  median.1,  
  median.2,  
  gprior.E_1 = NULL,  
  gprior.E_2 = NULL,  
  L,  
  U,  
  trunc.param = c(1, 1),  
  lambda,  
  gamma,  
  nmax,  
  S_likely = (L + U)/2,  
  value = FALSE  
)
```

Arguments

data_E	A data frame containing the survival data for the experimental (treatment) arm. The first column should be the observed survival times, and the second column should be the event indicator (1 = event, 0 = censored).
data_C	A data frame containing the survival data for the control arm. The format is the same as data_E: first column for observed time, second column for status (1 = event, 0 = censored).
median.1	Numeric. The overall median survival time for the standard-of-care (SOC) arm.
median.2	Numeric. The overall median survival time for the experimental arm.
gprior.E_1	Optional. A numeric vector of length two specifying the shape and scale parameters of the inverse-gamma prior for the pre-separation mean survival time (i.e., 1/hazard rate). If NULL, the default is $c(4, 3/\log(2)*median.1)$
gprior.E_2	Optional. A numeric vector of length two specifying the shape and scale parameters of the inverse-gamma prior for the post-separation mean survival time (i.e., 1/hazard rate). If NULL, the default is $c(4, 6/\log(2)*median.1)$
L	Numeric. The lower bound of the delayed treatment effect (DTE) separation timepoint.
U	Numeric. The upper bound of the delayed treatment effect (DTE) separation timepoint.
trunc.para	Numeric vector of length two. Specifies the shape and scale parameters of the truncated Gamma prior for the DTE timepoint.
lambda	Numeric. Weighting parameter used in the posterior decision function: $1 - \lambda(n/N)^\gamma$
gamma	Numeric. Exponent parameter used in the posterior decision function: $1 - \lambda(n/N)^\gamma$
nmax	Integer. Maximum sample size per arm.
S_likely	Numeric. The most likely value of the DTE separation timepoint. Defaults to the midpoint of L and U
value	If TRUE, returns posterior means, 95% credible intervals, and event rates. Default is FALSE.

Value

A list with the following components:

posterior_mean	Posterior mean of the hazard ratio (post- / pre-separation).
credible_interval	95% Bayesian credible interval for the hazard ratio.
event_rates	Event rates for the experimental arm, control arm, and overall.
decision	One of the following trial decisions based on observed data: Go Trial proceeds to next stage. No-Go Trial stops early for futility. Reject Null Hypothesis Final analysis leads to rejecting H0. Not Reject Null Hypothesis Final analysis fails to reject H0.

Examples

```
conduct(data_E, data_C, median.1 = 2.8, median.2 = 3.5, L = 2, U = 2.5,
        trunc.para = c(1, 1), lambda = 0.95, gamma = 1, nmax = 63, S_likely = 2.3)
```

event_fun

Return expected event number at interim analysis and final analysis.

Description

Based on the overall median times (median.1, median.2), the most likely separation timepoint, accrual rate and follow-up duration, this function will return the expected number of events in total at each interim analysis and final analysis based on the simulations.

Usage

```
event_fun(
  median.1,
  median.2,
  S_likely = (L + U)/2,
  n.interim,
  rate,
  FUP,
  n.sim = 1000,
  seed = NULL
)
```

Arguments

median.1	Numeric. The overall median survival time for SOC.
median.2	Numeric. The overall median survival time for the experimental arm.
S_likely	Numeric. The most likely separation time. Defaults to the midpoint of L and U.
n.interim	A vector of sample sizes per arm at each interim analysis. <ul style="list-style-type: none"> Each element except the last represents an interim sample size per arm. The final element is the total sample size N per arm.
rate	Numeric value. Patient accrual rate (e.g., patients per month).
FUP	Numeric value. Duration of follow-up. Default is 6 month/year in the context.
n.sim	Integer. Number of simulations to generate. Default is 1000.
seed	Optional integer. If provided, sets the seed for reproducibility.

Value

A data frame with two columns: the first column indicates the stage label, and the second column contains the expected number of events for each stage.

Examples

```
result <- event_fun(
  median.1 = 2.8,
  median.2 = 3.5,
  S_likely = 2.1,
  n.interim = c(28,40),
  rate = 6,
  FUP = 6
)
```

Experimental

*Nivolumab KM data used for reconstructing PFS KM curve***Description**

Kaplan-Meier (KM) survival data for the experimental arm (nivolumab) extracted from a published figure using PlotDigitizer. This dataset is used to reconstruct the progression-free survival (PFS) KM curve in the CheckMate 017 study.

Usage

```
data(Experimental)
```

Format

A data frame with two variables:

time Numeric. Time points (in months).

survival Numeric. Estimated survival probabilities at corresponding time points.

Examples

```
data(Experimental)
head(Experimental)
```

generate_pe

*Generate event times from a piecewise exponential distribution.***Description**

Generates random event times under a piecewise exponential distribution, where different constant hazard rates apply before and after a specified separation timepoint. This is commonly used for simulating survival data with delayed treatment effects.

Usage

```
generate_pe(n, t, lambda1, lambda2)
```

Arguments

n	Integer. Number of event times to generate (i.e., sample size).
t	Numeric. Separation timepoint at which the hazard rate changes.
lambda1	Numeric. Constant hazard rate before the separation timepoint.
lambda2	Numeric. Constant hazard rate after the separation timepoint.

Value

A numeric vector of length n, representing simulated survival times drawn from a piecewise exponential distribution with hazard rate lambda1 before time t, and lambda2 after time t. Times are expressed in the same time unit as specified in the hazard rate (typically months)

Examples

```
set.seed(42)
generate_pe(1000, 1, 1, 0.5)
```

```
get.optimal_2arm_piecewise
```

Obtain Optimal Design Parameters for DTE-BOP2 Design

Description

Computes the optimal tuning parameters λ and γ for use in posterior decision rules of two-arm, multi-stage survival trials under a piecewise exponential model with delayed treatment effect. These parameters are selected to control the type I error and maximize power, given user-specified interim sample sizes, prior distribution for the delayed separation timepoint, and hypothesized median survival times.

Usage

```
get.optimal_2arm_piecewise(
  median.1,
  median.2,
  gprior.E_1 = NULL,
  gprior.E_2 = NULL,
  L,
  U,
  S_likely = (L + U)/2,
  lambda.seq = seq(0.5, 0.975, by = 0.025),
  gamma.seq = seq(0, 1, by = 0.1),
  trunc.para,
```

```

Uniform = FALSE,
err1,
nsim,
n.interim,
rate,
FUP,
track = TRUE,
control = FALSE,
control.point = NULL
)

```

Arguments

median.1	Numeric. The overall median survival time for the standard-of-care (SOC) arm under the null hypothesis.
median.2	Numeric. The overall median survival time for the experimental arm under the alternative hypothesis.
gprior.E_1	Optional. A numeric vector of length two specifying the shape and scale parameters of the inverse-gamma prior for the pre-separation mean survival time (i.e., $1/\text{hazard rate}$). If NULL, the default is $c(4, 3/\log(2)*\text{median.1})$
gprior.E_2	Optional. A numeric vector of length two specifying the shape and scale parameters of the inverse-gamma prior for the post-separation mean survival time (i.e., $1/\text{hazard rate}$). If NULL, the default is $c(4, 6/\log(2)*\text{median.1})$
L	Numeric. The lower bound of the delayed treatment effect (DTE) separation timepoint.
U	Numeric. The upper bound of the delayed treatment effect (DTE) separation timepoint.
S_likely	Numeric. The most likely value of the DTE separation timepoint. Defaults to the midpoint of L and U.
lambda.seq	A vector. The sequence of λ in $1 - \lambda(n/N)^\gamma$
gamma.seq	A vector. The sequence of γ in $1 - \lambda(n/N)^\gamma$
trunc.para	Vector value with two elements. The first element is the shape parameter for the truncated gamma prior and the second one is the scale parameter.
Uniform	Logical value. <ul style="list-style-type: none"> • Default FALSE. The truncated gamma distribution for the separation time will be utilized. • If TRUE the average type I error and power are calculated based on 20 evenly divided points in the interval $[L, U]$.
err1	Type I error rate.
nsim	Integer. Number of simulations to generate. Default is 10000.
n.interim	A vector of sample sizes per arm at each interim analysis. <ul style="list-style-type: none"> • Each element except the last represents an interim sample size per arm. • The final element is the total sample size N per arm.
rate	Numeric value. Patient accrual rate (e.g., patients per month).

FUP	Numeric value. Duration of follow-up. Default is 6 month/year in the context.
track	Logical value. If true, it will report the current lambda value in the running simulations.
control	Logical value. If TRUE, the type I error will be additionally controlled at control point. Default is FALSE.
control.point	Vector. It specifies the points where the type I error is controlled.

Value

A list with the following components:

optimal A numeric vector of length four. Contains the optimal values of λ and γ , along with the average type I error and average power achieved under these tuning parameters.

oc.mat A numeric matrix containing all evaluated combinations of λ and γ that satisfy the type I error constraint. Each row includes values for λ , γ , average type I error, and average power.

Examples

```
# Define design and simulation parameters
median.1 <- 6
median.2 <- 10
L <- 2.6
U <- 3.4
S_likely <- 2.7
trunc.para <- c(1, 1)
rate <- 3
FUP <- 9
err1 <- 0.15
n.interim <- c(30, 50) # Each arm has 30 patients at interim and 50 at final

get.optimal_2arm_piecewise(
  median.1 = median.1,
  median.2 = median.2,
  L = L,
  U = U,
  Uniform = FALSE,
  S_likely = S_likely,
  trunc.para = trunc.para,
  err1 = err1,
  n.interim = n.interim,
  rate = rate,
  FUP = FUP,
  track = TRUE,
  nsim = 10000
)
```

getoc_2arm_piecewise	<i>Compute Operating Characteristics for Two-Arm Piecewise Exponential Designs</i>
----------------------	--

Description

Computes the operating characteristics (e.g., type I error, power, and other performance metrics) of a two-arm survival trial design under the DTE-BOP2 framework with delayed treatment effect, based on a piecewise exponential model. This function is typically used after selecting design parameters (e.g., lambda, gamma) to assess the statistical properties of the proposed design.

Usage

```
getoc_2arm_piecewise(
  median.true,
  gprior.E_1 = NULL,
  gprior.E_2 = NULL,
  lambda,
  gamma,
  n.interim,
  L,
  U,
  S_likely = (L + U)/2,
  Uniform = FALSE,
  trunc.para,
  rate,
  FUP = 6,
  nsim = 1000,
  track = FALSE,
  seed = NULL
)
```

Arguments

median.true	<p>A numeric vector of length two.</p> <ul style="list-style-type: none"> • The first element is the overall median survival time (in months) for the standard-of-care (SOC) arm under the null hypothesis. • The second element is the overall median survival time for the experimental arm.
gprior.E_1	<p>Optional. A numeric vector of length two specifying the shape and scale parameters of the inverse-gamma prior for the pre-separation mean survival time (i.e., 1/hazard rate). If NULL, the default is <code>c(4, 3/log(2)*median.true[1])</code></p>
gprior.E_2	<p>Optional. A numeric vector of length two specifying the shape and scale parameters of the inverse-gamma prior for the post-separation mean survival time (i.e., 1/hazard rate). If NULL, the default is <code>c(4, 6/log(2)*median.true[1])</code></p>

lambda	Numeric. The tuning parameter λ in the decision boundary function $1 - \lambda \cdot (n/N)^\gamma$.
gamma	Numeric. The tuning parameter γ in the decision boundary function $1 - \lambda \cdot (n/N)^\gamma$.
n.interim	A numeric vector. Specifies the sample sizes per arm at each analysis. The last element represents the final sample size per arm.
L	Numeric. Lower bound of the delayed treatment effect (DTE) separation time.
U	Numeric. Upper bound of the DTE separation time.
S_likely	Numeric. The most likely value of the separation time. Defaults to the midpoint of L and U. Default is the midpoint of L and U.
Uniform	Logical value. <ul style="list-style-type: none"> • Default FALSE. The truncated gamma distribution for the separation time will be utilized. • If TRUE the average type I error and power are calculated based on 20 evenly divided points in the interval $[L, U]$.
trunc.para	Vector value with two elements. The first element is the shape parameter for the truncated gamma prior and the second one is the scale parameter.
rate	Numeric. Patient accrual rate (e.g., patients per month).
FUP	Numeric value. Duration of follow-up. Default is 6 month/year in the context.
nsim	Integer. Number of simulations to generate. Default is 10000.
track	Logical value. <ul style="list-style-type: none"> • Default FALSE • If TRUE, prints progress updates every 1000 simulations.
seed	Optional integer. If provided, sets the seed for reproducibility.

Value

A list with the following components:

earlystop Numeric. The proportion of simulated trials that stopped early due to futility or predefined stopping rules.

reject Numeric. The proportion of trials in which the null hypothesis (H_0) was rejected.

average.patients Numeric. The average number of patients enrolled across all simulated trials.

trial.duration Numeric. The average total duration of the trial (in months), including follow-up.

Examples

```
# Define trial parameters
median.1 <- 6
median.2 <- 9
trunc.para <- c(1, 1)
rate <- 3
FUP <- 9
n.interim <- c(30, 50) # Each arm: 30 pts at interim, 50 pts at final
```

```
# Run operating characteristics computation
getoc_2arm_piecewise(median.true = c(median.1, median.2),
  Uniform = FALSE, lambda = 0.9, gamma = 1, n.interim = n.interim,
  L = 3, U = 3, S_likely = 3, FUP = FUP, trunc.para = trunc.para,
  rate = rate, nsim = 1)
```

rtrunc_gamma

Generate Random Variables from a Truncated Gamma Distribution.

Description

Simulates random variables from a truncated gamma distribution, where the gamma distribution is bounded between user-specified lower and upper limits. This function is typically used to define priors (e.g., for the separation time in delayed treatment effect models).

Usage

```
rtrunc_gamma(n, L, U, shape, scale)
```

Arguments

n	Integer. Number of random variables to generate.
L	Numeric. Lower bound of the truncation interval. Must satisfy $L \leq U$.
U	Numeric. Upper bound of the truncation interval. Must satisfy $U \geq L$.
shape	Numeric. Shape parameter of the underlying gamma distribution.
scale	Numeric. Scale parameter of the underlying gamma distribution.

Value

A numeric vector of length n, containing random variables drawn from a gamma distribution truncated to the interval $[L, U]$.

Examples

```
rtrunc_gamma(10, 1, 1, 1, 1)
```

SOC	<i>Docetaxel KM data used for reconstructing PFS KM curve</i>
-----	---

Description

Kaplan-Meier (KM) survival data for the standard-of-care (SOC) arm (docetaxel), extracted using PlotDigitizer from a published KM curve. This dataset is used to reconstruct the progression-free survival (PFS) curve for trial simulation and validation.

Usage

```
data(SOC)
```

Format

A data frame with two variables:

time Numeric. Time points (e.g., in months) at which survival probabilities are estimated.

survival Numeric. Estimated survival probability corresponding to each time point.

Examples

```
data(SOC)
head(SOC)
```

trunc_gamma_para	<i>Estimate Shape and Scale Parameters for Truncated Gamma Distribution</i>
------------------	---

Description

This function estimates the shape and scale parameters of a truncated Gamma distribution based on expert-provided summary statistics, including the mean, median, standard deviation, and selected quantiles. Parameter estimation is performed using a grid search combined with weighted least squares optimization. Parallel computing is employed to accelerate the estimation process.

Usage

```
trunc_gamma_para(
  L,
  U,
  expert_data,
  weights = c(10, 10, 2, 1, 1),
  num_cores = 4,
  seed = NULL
)
```

Arguments

L	Numeric. Lower bound of the truncated Gamma distribution.
U	Numeric. Upper bound of the truncated Gamma distribution.
expert_data	A list of named lists, where each inner list represents one expert's input. Each expert can provide any subset of the following named elements: mean Numeric. The expected mean of the distribution. median Numeric. The expected median of the distribution. sd Numeric. The expected standard deviation of the distribution. q25 Numeric. The 2.5th percentile. q975 Numeric. The 97.5th percentile.
weights	Numeric vector of length 5. Specifies the relative importance of each summary statistic in the optimization procedure. The order corresponds to: c(mean, median, sd, q25, q975). Default is c(10, 10, 2, 1, 1).
num_cores	Integer. Number of CPU cores to use for parallel computation. Default is 4.
seed	Optional integer. If provided, sets the seed for reproducibility.

Value

A list with the following components:

shape Numeric. Estimated shape parameter of the Gamma distribution.

scale Numeric. Estimated scale parameter of the Gamma distribution.

Examples

```
# Define expert-provided summary data
expert_data_correct <- list(
  list(mean = 2.2, median = 2.27, sd = NULL, q25 = NULL, q975 = NULL), # Expert A
  list(mean = 2.1, median = 2.3, sd = NULL, q25 = NULL, q975 = NULL), # Expert B
  list(mean = NULL, median = 2.31, sd = NULL, q25 = NULL, q975 = NULL) # Expert C
)

# Estimate parameters using truncated gamma prior
trunc_gamma_para(L = 2, U = 2.5, expert_data = expert_data_correct, num_cores = 4)
```

TTOT_fun

Compute the Total Time-on-Test (TTOT) Statistic

Description

Calculates the Total Time-on-Test (TTOT) statistic for a given vector of event times and censoring indicators. The TTOT function is commonly used in survival analysis for visualizing and assessing the cumulative hazard or comparing survival patterns between groups.

Usage

```
TTOT_fun(data, tA = 0, tB = 10000, t)
```

Arguments

<code>data</code>	A data frame containing the input survival data. The first column should be named "Time" and contain the observed survival times (e.g., in months or years). The second column should be named "Status", where 0 indicates a censored observation and 1 indicates an event.
<code>tA</code>	Numeric. The starting time point for computing the Total Time-on-Test (TTOT) function.
<code>tB</code>	Numeric. The ending time point for computing the TTOT function.
<code>t</code>	Numeric. The time point(s) at which the TTOT function is evaluated.

Value

This function calculates the total time-on-test (TTOT) for a given dataset within the time interval (tA,tB).

Examples

```
# Generate sample survival data
set.seed(123)
sim_time <- generate_pe(n = 20, t = 1, lambda1 = 1, lambda2 = 0.5)
sim_status <- sample(c(0, 1), 20, replace = TRUE)
Data <- data.frame(Time = sim_time, Status = sim_status)
# Compute TTOT when time = 1
TTOT_fun(Data, t = 1)
```

Two_stage_sample_size *Calculate Sample Sizes for a Two-Stage Trial Design.*

Description

Computes the required sample sizes for the interim and final analyses in a two-stage trial design, based on pre-specified operating characteristics such as desired power, type I error rate assumptions.

Usage

```
Two_stage_sample_size(
  median.1,
  median.2,
  gprior.E_1 = NULL,
  gprior.E_2 = NULL,
  L,
  U,
  S_likely = (L + U)/2,
```

```

trunc.para = c(1, 1),
err1,
err2,
FUP,
rate,
weight = 0.5,
method = "optimal",
earlystop_prob = NULL,
seed = NULL
)

```

Arguments

median.1	Numeric. The overall median survival time for SOC.
median.2	Numeric. The overall median survival time for the experimental arm.
gprior.E_1	Optional. A numeric vector of length two, representing the shape and scale parameters of the inverse-gamma prior for μ_0 , the mean survival time before the separation time. If NULL, the default is <code>c(4, 3 / (log(2) * median.1))</code> .
gprior.E_2	Optional. A numeric vector of length two, representing the inverse-gamma prior for μ_1 , the mean survival time after the separation time. If NULL, the default is <code>c(4, 6 / (log(2) * median.1))</code> .
L	Numeric. The lower bound of the separation time window.
U	Numeric. The upper bound of the separation time window.
S_likely	Numeric. The most likely separation time. Defaults to the midpoint of L and U.
trunc.para	Numeric vector of length two. Specifies the shape and scale parameters of the truncated gamma prior for the separation time.
err1	Numeric. The pre-specified type I error rate (e.g., 0.1).
err2	Numeric. The pre-specified type II error rate (i.e., β , e.g., 0.2).
FUP	Numeric. The duration of follow-up per patient (in months or years). Default is 6.
rate	Numeric. The recruitment rate (e.g., patients per month or year).
weight	Numeric. Weight given to the expected sample size under H_0 . Default is 0.5.
method	Character. Specifies which method to use: "optimal" (default) or "suboptimal". Determines how the sample size is calculated.
earlystop_prob	Optional. Numeric. If specified, ensures that the probability of early stopping under H_0 is at least this value, while also maintaining the power at or above $1 - \beta$ when the separation time is no more than U.
seed	Optional integer. If provided, sets the seed for reproducibility.

Value

A numeric vector of length 2. The first element is the required sample size for interim analysis, and the second element is the total sample size for final analysis in a two-stage design.

Examples

```
# Define design parameters
median.1 <- 4      # Median survival for standard-of-care
median.2 <- 6      # Median survival for experimental arm
L <- 2            # Lower bound of separation time
U <- 2.5          # Upper bound of separation time
S_likely <- 2.28   # Most likely separation time
rate <- 6         # Accrual rate (patients/month)
FUP <- 6          # Follow-up duration (months)
err1 <- 0.1       # Type I error
err2 <- 0.15      # Type II error
```

```
Two_stage_sample_size(
  median.1 = median.1,
  median.2 = median.2,
  L = L,
  U = U,
  S_likely = S_likely,
  err1 = err1,
  err2 = err2,
  FUP = FUP,
  rate = rate,
  weight = 0.5,
  method = "optimal"
)
```


Index

* **datasets**

Experimental, [5](#)

SOC, [12](#)

conduct, [2](#)

event_fun, [4](#)

Experimental, [5](#)

generate_pe, [5](#)

get.optimal_2arm_piecewise, [6](#)

getoc_2arm_piecewise, [9](#)

rtrunc_gamma, [11](#)

SOC, [12](#)

trunc_gamma_para, [12](#)

TTOT_fun, [13](#)

Two_stage_sample_size, [14](#)