

Package ‘bayesCT’

March 13, 2019

Type Package

Title Simulation and Analysis of Adaptive Bayesian Clinical Trials

Version 0.99.1

Description Simulation and analysis of Bayesian adaptive clinical trial, incorporates historical data and allows early stopping for futility or early success.

LazyLoad yes

License GPL-3

NeedsCompilation no

URL <https://github.com/thevaachandereng/bayesCT/>

BugReports <https://github.com/thevaachandereng/bayesCT/issues/>

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

Suggests testthat, rmarkdown, pkgdown, devtools, knitr

VignetteBuilder knitr

Imports bayesDP, dplyr, purrr, magrittr

Author Thevaa Chandereng [aut, cre, cph]
(<<https://orcid.org/0000-0003-4078-9176>>),
Donald Musgrove [aut, cph],
Tarek Haddad [aut, cph],
Graeme Hickey [aut, cph],
Timothy Hanson [aut, cph],
Theodore Lystig [aut, cph]

Maintainer Thevaa Chandereng <chandereng@wisc.edu>

Repository CRAN

Date/Publication 2019-03-13 18:33:29 UTC

R topics documented:

analysis	2
beta_prior	3
binomialBACT	4
binomialdata	6
binomial_analysis	7
binomial_outcome	9
data_binomial	10
data_normal	10
enrollment	11
enrollment_rate	12
historical_binomial	12
historical_normal	14
hypothesis	15
impute	16
normalBACT	16
normaldata	19
normal_analysis	20
normal_outcome	22
randomization	22
randomize	23
simulate	24
study_details	25
Index	26

analysis	<i>Analysis wrapper function</i>
----------	----------------------------------

Description

Wrapper function to analyze bayesian trials.

Usage

```
analysis(input, type = "binomial", .data = NULL)
```

Arguments

input	list. Input function for all the analysis.
type	character. Type of analysis to be ran (binomial (default), normal. etc.).
.data	NULL. stores the all the details, please do not fill it in.

Value

a list with results of the analysis of bayesian trial.

prob_of_accepting_alternative scalar. The input parameter of probability of accepting the alternative.

margin scalar. The margin input value of difference between mean estimate of treatment and mean estimate of the control.

alternative character. The input parameter of alternative hypothesis.

N_treatment scalar. The number of patients enrolled in the experimental group for each simulation.

N_control scalar. The number of patients enrolled in the control group for each simulation.

N_enrolled vector. The number of patients enrolled in the trial (sum of control and experimental group for each simulation.)

N_complete scalar. The number of patients who completed the trial and had no loss to follow-up.

post_prob_accept_alternative vector. The final probability of accepting the alternative hypothesis after the analysis is done.

est_final scalar. The final estimate of the difference in posterior estimate of treatment and posterior estimate of the control group.

stop_futility scalar. Did the trial stop for futility during imputation of patient who had loss to follow up? 1 for yes and 0 for no.

stop_expected_success scalar. Did the trial stop for early success during imputation of patient who had loss to follow up? 1 for yes and 0 for no.

beta_prior	<i>Beta prior for for control and treatment group</i>
------------	---

Description

Wrapper function for beta prior $\text{beta}(a_0, b_0)$.

Usage

```
beta_prior(a0 = 1, b0 = 1, .data = NULL)
```

Arguments

a0	numeric. The first shape paramater in the beta distribution ($\text{beta}(a_0, b_0)$).
b0	numeric. The second shape paramater in the beta distribution ($\text{beta}(a_0, b_0)$).
.data	NULL. stores the beta prior rate, please do not fill it in.

Value

a list with vector of beta rate for the beta prior for treatment and control group.

Examples

```
beta_prior(a0 = 1, b0 = 1)
```

binomialBACT

Binomial counts for Bayesian Adaptive Trials

Description

Simulation for binomial counts for Bayesian Adaptive trial with different inputs to control for power, sample size, type 1 error rate, etc.

Usage

```
binomialBACT(p_treatment, p_control = NULL, y0_treatment = NULL,
             N0_treatment = NULL, y0_control = NULL, N0_control = NULL,
             discount_function = "identity", N_total, lambda = 0.3,
             lambda_time = NULL, interim_look = NULL, EndofStudy, prior = c(1,
             1), block = 2, rand_ratio = c(1, 1), prop_loss_to_followup = 0.1,
             alternative = "greater", h0 = 0, futility_prob = 0.05,
             expected_success_prob = 0.9, prob_ha = 0.95, N_impute = 10,
             number_mcmc = 10000, alpha_max = 1, fix_alpha = FALSE,
             weibull_scale = 0.135, weibull_shape = 3)
```

Arguments

<code>p_treatment</code>	scalar. Proportion of events under the treatment arm.
<code>p_control</code>	scalar. Proportion of events under the control arm.
<code>y0_treatment</code>	scalar. Number of events for the historical treatment arm.
<code>N0_treatment</code>	scalar. Sample size of the historical treatment arm.
<code>y0_control</code>	scalar. Number of events for the historical control arm.
<code>N0_control</code>	scalar. Sample size of the historical control arm.
<code>discount_function</code>	character. If incorporating historical data, specify the discount function. Currently supports the Weibull function (<code>discount_function="weibull"</code>), the scaled-Weibull function (<code>discount_function="scaledweibull"</code>), and the identity function (<code>discount_function="identity"</code>). The scaled-Weibull discount function scales the output of the Weibull CDF to have a max value of 1. The identity discount function uses the posterior probability directly as the discount weight. Default value is "identity". See bdpnormal for more details.
<code>N_total</code>	scalar. Total sample size.
<code>lambda</code>	vector. Enrollment rates across simulated enrollment times. See enrollment for more details.
<code>lambda_time</code>	vector. Enrollment time(s) at which the enrollment rates change. Must be same length as lambda. See enrollment for more details.

interim_look	vector. Sample size for each interim look. Note: the maximum sample size should not be included.
EndofStudy	scalar. Length of the study.
prior	vector. Prior value of beta rate, beta(a0, b0). The default is set to beta(1, 1).
block	scalar. Block size for generating the randomization schedule.
rand_ratio	vector. Randomization allocation for the ratio of control to treatment. Integer values mapping the size of the block. See randomization for more details.
prop_loss_to_followup	scalar. Overall proportion of subjects lost to follow-up.
alternative	character. The string specifying the alternative hypothesis, must be one of "greater" (default), "less" or "two.sided".
h0	scalar. Threshold for comparing two mean values. Default is $h_0=0$.
futility_prob	scalar. Probability of stopping early for futility.
expected_success_prob	scalar. Probability of stopping early for success.
prob_ha	scalar. Probability of alternative hypothesis.
N_impute	scalar. Number of imputations for Monte Carlo simulation of missing data.
number_mcmc	scalar. Number of Monte Carlo Markov Chain draws in sampling posterior.
alpha_max	scalar. Maximum weight the discount function can apply. Default is 1. For a two-arm trial, users may specify a vector of two values where the first value is used to weight the historical treatment group and the second value is used to weight the historical control group.
fix_alpha	logical. Fix alpha at alpha_max? Default value is FALSE.
weibull_scale	scalar. Scale parameter of the Weibull discount function used to compute alpha, the weight parameter of the historical data. Default value is 0.135. For a two-arm trial, users may specify a vector of two values where the first value is used to estimate the weight of the historical treatment group and the second value is used to estimate the weight of the historical control group. Not used when discount_function = "identity".
weibull_shape	scalar. Shape parameter of the Weibull discount function used to compute alpha, the weight parameter of the historical data. Default value is 3. For a two-arm trial, users may specify a vector of two values where the first value is used to estimate the weight of the historical treatment group and the second value is used to estimate the weight of the historical control group. Not used when discount_function = "identity".

Value

a list of output for a single trial simulation.

p_treatment scalar. The input parameter of proportion of events in the treatment group.

p_control scalar. The input parameter of proportion of events in the control group.

prob_of_accepting_alternative scalar. The input parameter of probability threshold of accepting the alternative.

`margin` scalar. The margin input value of difference between mean estimate of treatment and mean estimate of the control.

`alternative` character. The input parameter of alternative hypothesis.

`interim_look` vector. The sample size for each interim look.

`N_treatment` scalar. The number of patients enrolled in the experimental group for each simulation.

`N_control` scalar. The number of patients enrolled in the control group for each simulation.

`N_enrolled` vector. The number of patients enrolled in the trial (sum of control and experimental group for each simulation.)

`N_complete` scalar. The number of patients who completed the trial and had no loss to follow-up.

`post_prob_accept_alternative` vector. The final probability of accepting the alternative hypothesis after the analysis is done.

`est_final` scalar. The final estimate of the difference in posterior estimate of treatment and posterior estimate of the control group.

`stop_futility` scalar. Did the trial stop for futility during imputation of patient who had loss to follow up? 1 for yes and 0 for no.

`stop_expected_success` scalar. Did the trial stop for early success during imputation of patient who had loss to follow up? 1 for yes and 0 for no.

`est_interim` scalar. The interim estimate of the difference in posterior estimate of treatment and posterior estimate of the control group.

Examples

```
binomialBACT(p_control = 0.12, p_treatment = 0.10,
             y0_treatment = 8, N0_treatment = 90,
             y0_control = 13, N0_control = 95,
             N_total = 300, N_impute = 100,
             lambda = c(0.3, 1), lambda_time = c(25),
             interim_look = c(210, 240, 270),
             EndofStudy = 50)
```

binomialdata

Binomial dataset for analyzing adaptive Bayesian trials

Description

A dataset containing the results of 300 patients with binomial outcome, the dataset is filled with loss to follow up.

Usage

```
data(binomialdata)
```

Format

A data frame with 300 rows and 4 variables:

id Patient ID in the trial

treatment treatment assignment for patients, 1 for treatment group 0 for control group

outcome binomial outcome of the trial, 1 for response (success or failure), 0 for no response

complete 1 for complete outcome, 0 for loss to follow up

Examples

```
data(binomialdata)
```

binomial_analysis *Analyzing bayesian trial for binomial counts*

Description

Function to analyze bayesian trial for binomial count data which allows early stopping and incorporation of historical data using the discount function approach

Usage

```
binomial_analysis(treatment, outcome, complete = NULL,
  y0_treatment = NULL, N0_treatment = NULL, y0_control = NULL,
  N0_control = NULL, alternative = "greater", N_impute = 10,
  h0 = 0, number_mcmc = 10000, prob_ha = 0.95, futility_prob = 0.1,
  expected_success_prob = 0.9, prior = c(1, 1),
  discount_function = "identity", fix_alpha = FALSE, alpha_max = 1,
  weibull_scale = 0.135, weibull_shape = 3)
```

Arguments

treatment	vector. treatment assignment for patients, 1 for treatment group and 0 for control group
outcome	vector. binomial outcome of the trial, 1 for response (success or failure), 0 for no response.
complete	vector. similar length as treatment and outcome variable, 1 for complete outcome, 0 for loss to follow up. If complete is not provided, the dataset is assumed to be complete.
y0_treatment	scalar. Number of events for the historical treatment arm.
N0_treatment	scalar. Number of observations of the historical treatment group.
y0_control	scalar. Number of events for the historical control arm.
N0_control	scalar. Number of observations of the historical control group.

alternative	character. The string specifying the alternative hypothesis, must be one of "greater" (default), "less" or "two.sided".
N_impute	scalar. Number of imputations for Monte Carlo simulation of missing data.
h0	scalar. Threshold for comparing two mean values. Default is $h_0=0$.
number_mcmc	scalar. Number of Monte Carlo Markov Chain draws in sampling posterior.
prob_ha	scalar. Probability of alternative hypothesis.
futility_prob	scalar. Probability of stopping early for futility.
expected_success_prob	scalar. Probability of stopping early for success.
prior	vector. Prior value of beta rate, beta(a0, b0). The default is set to beta(1, 1).
discount_function	character. If incorporating historical data, specify the discount function. Currently supports the Weibull function (discount_function="weibull"), the scaled-Weibull function (discount_function="scaledweibull"), and the identity function (discount_function="identity"). The scaled-Weibull discount function scales the output of the Weibull CDF to have a max value of 1. The identity discount function uses the posterior probability directly as the discount weight. Default value is "identity". See bdpnormal for more details.
fix_alpha	logical. Fix alpha at alpha_max? Default value is FALSE.
alpha_max	scalar. Maximum weight the discount function can apply. Default is 1. For a two-arm trial, users may specify a vector of two values where the first value is used to weight the historical treatment group and the second value is used to weight the historical control group.
weibull_scale	scalar. Scale parameter of the Weibull discount function used to compute alpha, the weight parameter of the historical data. Default value is 0.135. For a two-arm trial, users may specify a vector of two values where the first value is used to estimate the weight of the historical treatment group and the second value is used to estimate the weight of the historical control group. Not used when discount_function = "identity".
weibull_shape	scalar. Shape parameter of the Weibull discount function used to compute alpha, the weight parameter of the historical data. Default value is 3. For a two-arm trial, users may specify a vector of two values where the first value is used to estimate the weight of the historical treatment group and the second value is used to estimate the weight of the historical control group. Not used when discount_function = "identity".

Value

	a list of output for the bayesian trial for binomial count.
prob_of_accepting_alternative	scalar. The input parameter of probability of accepting the alternative.
margin	scalar. The margin input value of difference between mean estimate of treatment and mean estimate of the control.
alternative	character. The input parameter of alternative hypothesis.

`N_treatment` scalar. The number of patients enrolled in the experimental group for each simulation.

`N_control` scalar. The number of patients enrolled in the control group for each simulation.

`N_enrolled` vector. The number of patients enrolled in the trial (sum of control and experimental group for each simulation.)

`N_complete` scalar. The number of patients who completed the trial and had no loss to follow-up.

`post_prob_accept_alternative` vector. The final probability of accepting the alternative hypothesis after the analysis is done.

`est_final` scalar. The final estimate of the difference in posterior estimate of treatment and posterior estimate of the control group.

`stop_futility` scalar. Did the trial stop for futility during imputation of patient who had loss to follow up? 1 for yes and 0 for no.

`stop_expected_success` scalar. Did the trial stop for early success during imputation of patient who had loss to follow up? 1 for yes and 0 for no.

binomial_outcome	<i>Proportion of an event in control and treatment</i>
------------------	--

Description

Wrapper function for proportion of an event in control and treatment group with binomial outcome.

Usage

```
binomial_outcome(p_treatment = NULL, p_control = NULL, .data = NULL)
```

Arguments

`p_treatment` numeric. The proportion of an event in the treatment group, $0 < p_{treatment} < 1$.

`p_control` numeric. The proportion of an event in the control group, $0 < p_{control} < 1$.

`.data` NULL. stores the proportion of control and treatment, please do not fill it in.

Value

a list with proportion of control and treatment group.

Examples

```
binomial_outcome(p_control = 0.12, p_treatment = 0.08)
```

data_binomial	<i>Data file for binomial analysis</i>
---------------	--

Description

Wrapper function for data file in binomial analysis.

Usage

```
data_binomial(treatment, outcome, complete, .data = NULL)
```

Arguments

treatment	vector. treatment assignment for patients, 1 for treatment group and 0 for control group
outcome	vector. binomial outcome of the trial, 1 for response (success or failure), 0 for no response.
complete	vector. similar length as treatment and outcome variable, 1 for complete outcome, 0 for loss to follow up. If complete is not provided, the dataset is assumed to be complete.
.data	NULL. stores the binomial data for analysis, please do not fill it in.

Value

a list with treatment, outcome and loss to follow up vector with binomial outcome.

Examples

```
data_binomial(treatment = c(0, 1), outcome = c(1, 1), complete = c(1, 1))
```

data_normal	<i>Data file for normal analysis</i>
-------------	--------------------------------------

Description

Wrapper function for data file in normal analysis.

Usage

```
data_normal(treatment, outcome, complete, .data = NULL)
```

Arguments

treatment	vector. treatment assignment for patients, 1 for treatment group and 0 for control group
outcome	vector. normal outcome of the trial.
complete	vector. similar length as treatment and outcome variable, 1 for complete outcome, 0 for loss to follow up. If complete is not provided, the dataset is assumed to be complete.
.data	NULL. stores the normal data for analysis, please do not fill it in.

Value

a list with treatment, outcome and loss to follow up vector with normal outcome.

enrollment	<i>Simulating enrollment dates</i>
------------	------------------------------------

Description

This function simulates enrollment dates using either poisson distribution

Usage

```
enrollment(param, N_total, time = NULL)
```

Arguments

param	a vector of lambda in poisson
N_total	a numeric value of total sample size
time	a vector of the length(param) - 1 indicating end of time when a specific lambda is used

Value

a vector of enrollment dates

Examples

```
enrollment(param = c(0.003, 0.7), 100, time = 10)
enrollment(param = c(0.3, 0.5, 0.9, 1.2, 2.1), 200, c(20, 30, 40, 60))
```

enrollment_rate	<i>Enrollment rate wrapper</i>
-----------------	--------------------------------

Description

Wrapper function for enrollment rate.

Usage

```
enrollment_rate(lambda = 0.3, time = NULL, .data = NULL)
```

Arguments

lambda	vector. Vector with different enrollment rate parameters.
time	vector. Vector with different cut-off times for lambda.
.data	NULL. This should not be changed by the user.

Value

a list with enrollment rate information

Examples

```
enrollment_rate(lambda = c(0.3, 1), time = 25)
```

historical_binomial	<i>Historical data for binomial distribution</i>
---------------------	--

Description

Wrapper function for historical data from binomial outcome.

Usage

```
historical_binomial(y0_treatment = NULL, N0_treatment = NULL,  
  discount_function = "identity", y0_control = NULL,  
  N0_control = NULL, alpha_max = 1, fix_alpha = FALSE,  
  weibull_scale = 0.135, weibull_shape = 3, .data = NULL)
```

Arguments

<code>y0_treatment</code>	numeric. The proportion of event in the historical treatment group.
<code>N0_treatment</code>	numeric. The sample size for the historical treatment group.
<code>discount_function</code>	character. Specify the discount function to use. Currently supports <code>weibull</code> , <code>scaledweibull</code> , and <code>identity</code> . The discount function <code>scaledweibull</code> scales the output of the Weibull CDF to have a max value of 1. The <code>identity</code> discount function uses the posterior probability directly as the discount weight. Default value is <code>"identity"</code> .
<code>y0_control</code>	numeric. The proportion of event in the historical control group.
<code>N0_control</code>	numeric. The sample size for the historical control group.
<code>alpha_max</code>	scalar. Maximum weight the discount function can apply. Default is 1. For a two-arm trial, users may specify a vector of two values where the first value is used to weight the historical treatment group and the second value is used to weight the historical control group.
<code>fix_alpha</code>	logical. Fix alpha at <code>alpha_max</code> ? Default value is <code>FALSE</code> .
<code>weibull_scale</code>	scalar. Scale parameter of the Weibull discount function used to compute alpha, the weight parameter of the historical data. Default value is 0.135. For a two-arm trial, users may specify a vector of two values where the first value is used to estimate the weight of the historical treatment group and the second value is used to estimate the weight of the historical control group. Not used when <code>discount_function = "identity"</code> .
<code>weibull_shape</code>	scalar. Shape parameter of the Weibull discount function used to compute alpha, the weight parameter of the historical data. Default value is 3. For a two-arm trial, users may specify a vector of two values where the first value is used to estimate the weight of the historical treatment group and the second value is used to estimate the weight of the historical control group. Not used when <code>discount_function = "identity"</code> .
<code>.data</code>	<code>NULL</code> . stores the proportion of control and treatment, please do not fill it in.

Value

a list with historical data for control and treatment group with the discount function.

Examples

```
historical_binomial(y0_treatment = 5, N0_treatment = 10, y0_control = 15, N0_control = 23)
historical_binomial(y0_treatment = 5, N0_treatment = 10, y0_control = 15, N0_control = 23,
  discount_function = "weibull", alpha_max = 1, fix_alpha = FALSE,
  weibull_scale = 0.135, weibull_shape = 3)
```

historical_normal *Historical data for normal distribution*

Description

Wrapper function for historical data from normal outcome.

Usage

```
historical_normal(mu0_treatment = NULL, sd0_treatment = NULL,
  N0_treatment = NULL, mu0_control = NULL, sd0_control = NULL,
  N0_control = NULL, discount_function = "identity", alpha_max = 1,
  fix_alpha = FALSE, weibull_scale = 0.135, weibull_shape = 3,
  .data = NULL)
```

Arguments

mu0_treatment	numeric. Mean of the historical treatment group.
sd0_treatment	numeric. The Standard deviation of the historical treatment group.
N0_treatment	numeric. scalar. Number of observations of the historical treatment group.
mu0_control	numeric. Mean of the historical control group.
sd0_control	numeric. The Standard deviation of the historical control group.
N0_control	numeric. umber of observations of the historical control group.
discount_function	character. Specify the discount function to use. Currently supports weibull, scaledweibull, and identity. The discount function scaledweibull scales the output of the Weibull CDF to have a max value of 1. The identity discount function uses the posterior probability directly as the discount weight. Default value is "identity".
alpha_max	scalar. Maximum weight the discount function can apply. Default is 1. For a two-arm trial, users may specify a vector of two values where the first value is used to weight the historical treatment group and the second value is used to weight the historical control group.
fix_alpha	logical. Fix alpha at alpha_max? Default value is FALSE.
weibull_scale	scalar. Scale parameter of the Weibull discount function used to compute alpha, the weight parameter of the historical data. Default value is 0.135. For a two-arm trial, users may specify a vector of two values where the first value is used to estimate the weight of the historical treatment group and the second value is used to estimate the weight of the historical control group. Not used when discount_function = "identity".
weibull_shape	scalar. Shape parameter of the Weibull discount function used to compute alpha, the weight parameter of the historical data. Default value is 3. For a two-arm trial, users may specify a vector of two values where the first value is used to estimate the weight of the historical treatment group and the second value

is used to estimate the weight of the historical control group. Not used when `discount_function = "identity"`.

`.data` NULL. stores the proportion of control and treatment, please do not fill it in.

Value

a list with historical data for control and treatment group with the discount function.

Examples

```
historical_normal(mu0_treatment = 15, sd0_treatment = 2, N0_treatment = 10,
                 mu0_control = 17, sd0_control = 3, N0_control = 20)
```

hypothesis

Hypothesis wrapper

Description

Wrapper function for the hypothesis in the trial.

Usage

```
hypothesis(delta = 0, futility_prob = 0.05, prob_accept_ha = 0.95,
           expected_success_prob = 0.9, alternative = "greater", .data = NULL)
```

Arguments

`delta` numeric. Threshold set for margin in null hypothesis. The default is set to 0.

`futility_prob` numeric. Probability of futility. The default is 0.05.

`prob_accept_ha` numeric. Posterior probability of accepting alternative hypothesis. The default is 0.95.

`expected_success_prob` numeric. Probability of expected success.

`alternative` character. The string specifying the alternative hypothesis, must be one of "greater" (default), "less" or "two.sided".

`.data` NULL. This should not be changed by the user.

Value

a list with information of hypothesis testing (threshold, futility probability, probability of accepting the alternative hypothesis, and probability of expected success).

Examples

```
hypothesis(delta = 0, futility_prob = 0.05, prob_accept_ha = 0.95,
           expected_success_prob = 0.90, alternative = "greater")
hypothesis(delta = 0.2, futility_prob = 0.1, prob_accept_ha = 0.975,
           expected_success_prob = 0.80, alternative = "less")
```

impute	<i>Imputation wrapper</i>
--------	---------------------------

Description

Wrapper function for no_of_impute.

Usage

```
impute(no_of_impute = 10000, number_mcmc = 10000, .data = NULL)
```

Arguments

no_of_impute	integer. Number of Monte Carlo imputation for missing data.
number_mcmc	scalar. Number of Monte Carlo Markov Chain draws from posterior distribution.
.data	NULL. This should not be changed by the user.

Value

a list with number of imputation

Examples

```
impute(no_of_impute = 100, number_mcmc = 1000)
```

normalBACT	<i>Normal distribution for Bayesian Adaptive Trials</i>
------------	---

Description

Simulation of normally distributed data for Bayesian adaptive trials with various inputs to control for power, sample size, type I error rate, etc.

Usage

```
normalBACT(mu_treatment, sd_treatment, mu_control = NULL,
  sd_control = NULL, mu0_treatment = NULL, sd0_treatment = NULL,
  N0_treatment = NULL, mu0_control = NULL, sd0_control = NULL,
  N0_control = NULL, N_total, lambda = 0.3, lambda_time = NULL,
  interim_look = NULL, EndofStudy, block = 2, rand_ratio = c(1, 1),
  discount_function = "identity", alternative = "greater",
  prop_loss_to_followup = 0.15, h0 = 0, futility_prob = 0.05,
  expected_success_prob = 0.9, prob_ha = 0.95, N_impute = 10,
  number_mcmc = 10000, alpha_max = 1, fix_alpha = FALSE,
  weibull_scale = 0.135, weibull_shape = 3)
```


Arguments

mu_treatment	scalar. Mean outcome in the treatment arm.
sd_treatment	scalar. Standard deviation of outcome in the treatment
mu_control	scalar. Mean outcome in the control arm.
sd_control	scalar. Standard deviation of outcome in the control arm. arm.
mu0_treatment	scalar. Mean of the historical treatment group.
sd0_treatment	scalar. Standard deviation of the historical treatment group.
N0_treatment	scalar. Number of observations of the historical treatment group.
mu0_control	scalar. Mean of the historical control group.
sd0_control	scalar. Standard deviation of the historical control group.
N0_control	scalar. Number of observations of the historical control group.
N_total	scalar. Total sample size.
lambda	vector. Enrollment rates across simulated enrollment times. See enrollment for more details.
lambda_time	vector. Enrollment time(s) at which the enrollment rates change. Must be same length as lambda. See enrollment for more details.
interim_look	vector. Sample size for each interim look. Note: the maximum sample size should not be included.
EndofStudy	scalar. Length of the study.
block	scalar. Block size for generating the randomization schedule.
rand_ratio	vector. Randomization allocation for the ratio of control to treatment. Integer values mapping the size of the block. See randomization for more details.
discount_function	character. If incorporating historical data, specify the discount function. Currently supports the Weibull function (discount_function="weibull"), the scaled-Weibull function (discount_function="scaledweibull"), and the identity function (discount_function="identity"). The scaled-Weibull discount function scales the output of the Weibull CDF to have a max value of 1. The identity discount function uses the posterior probability directly as the discount weight. Default value is "identity". See bdpnormal for more details.
alternative	character. The string specifying the alternative hypothesis, must be one of "greater" (default), "less" or "two.sided".
prop_loss_to_followup	scalar. Overall proportion of subjects lost to follow-up.
h0	scalar. Threshold for comparing two mean values. Default is h0=0.
futility_prob	scalar. Probability of stopping early for futility.
expected_success_prob	scalar. Probability of stopping early for success.
prob_ha	scalar. Probability of alternative hypothesis.
N_impute	scalar. Number of imputations for Monte Carlo simulation of missing data.

number_mcmc	scalar. Number of Monte Carlo Markov Chain draws in sampling posterior.
alpha_max	scalar. Maximum weight the discount function can apply. Default is 1. For a two-arm trial, users may specify a vector of two values where the first value is used to weight the historical treatment group and the second value is used to weight the historical control group.
fix_alpha	logical. Fix alpha at alpha_max? Default value is FALSE.
weibull_scale	scalar. Scale parameter of the Weibull discount function used to compute alpha, the weight parameter of the historical data. Default value is 0.135. For a two-arm trial, users may specify a vector of two values where the first value is used to estimate the weight of the historical treatment group and the second value is used to estimate the weight of the historical control group. Not used when discount_function = "identity".
weibull_shape	scalar. Shape parameter of the Weibull discount function used to compute alpha, the weight parameter of the historical data. Default value is 3. For a two-arm trial, users may specify a vector of two values where the first value is used to estimate the weight of the historical treatment group and the second value is used to estimate the weight of the historical control group. Not used when discount_function = "identity".

Value

a list of output for a single trial simulation.

mu_treatment	scalar. The input parameter of mean value of the outcome in the treatment group.
p_control	scalar. The input parameter of mean value of the outcome in the control group.
sd_treatment	scalar. The input parameter of standard deviation of the outcome in the control group.
sd_control	scalar. The input parameter of standard deviation of the outcome in the control group.
prob_of_accepting_alternative	scalar. The input parameter of probability threshold of accepting the alternative.
margin	scalar. The margin input value of difference between mean estimate of treatment and mean estimate of the control.
alternative	character. The input parameter of alternative hypothesis.
interim_look	vector. The sample size for each interim look.
N_treatment	scalar. The number of patients enrolled in the experimental group for each simulation.
N_control	scalar. The number of patients enrolled in the control group for each simulation.
N_enrolled	vector. The number of patients enrolled in the trial (sum of control and experimental group for each simulation.)
N_complete	scalar. The number of patients who completed the trial and had no loss to follow-up.
post_prob_accept_alternative	vector. The final probability of accepting the alternative hypothesis after the analysis is done.
est_final	scalar. The final estimate of the difference in posterior estimate of treatment and posterior estimate of the control group.

`stop_futility` scalar. Did the trial stop for futility during imputation of patient who had loss to follow up? 1 for yes and 0 for no.

`stop_expected_success` scalar. Did the trial stop for early success during imputation of patient who had loss to follow up? 1 for yes and 0 for no.

`est_interim` scalar. The interim estimate of the difference in posterior estimate of treatment and posterior estimate of the control group.

Examples

```
normalBACT(mu_treatment = 8,  
           sd_treatment = 1.2, N_total = 300,  
           lambda = c(0.3, 1), lambda_time = c(25),  
           interim_look = c(110, 140, 220, 270),  
           EndofStudy = 50)
```

normaldata

Gaussian dataset for analyzing adaptive Bayesian trials

Description

A dataset containing the results of 300 patients with continuous (normal) outcome, the dataset is filled with loss to follow up.

Usage

```
data(normaldata)
```

Format

A data frame with 300 rows and 4 variables:

id Patient ID in the trial

treatment treatment assignment for patients, 1 for treatment group 0 for control group

outcome continuous outcome of the trial (gaussian distribution)

complete 1 for complete outcome, 0 for loss to follow up

Examples

```
data(normaldata)
```

normal_analysis	<i>Analyzing bayesian trial for normal mean data</i>
-----------------	--

Description

Function to analyze bayesian trial for normal mean data which allows early stopping and incorporation of historical data using the discount function approach

Usage

```
normal_analysis(treatment, outcome, complete = NULL,
  mu0_treatment = NULL, sd0_treatment = NULL, N0_treatment = NULL,
  mu0_control = NULL, sd0_control = NULL, N0_control = NULL,
  alternative = "greater", N_impute = 100, h0 = 0,
  number_mcmc = 10000, prob_ha = 0.95, futility_prob = 0.1,
  expected_success_prob = 0.9, discount_function = "identity",
  fix_alpha = FALSE, alpha_max = 1, weibull_scale = 0.135,
  weibull_shape = 3)
```

Arguments

treatment	vector. treatment assignment for patients, 1 for treatment group and 0 for control group
outcome	vector. normal outcome of the trial.
complete	vector. similar length as treatment and outcome variable, 1 for complete outcome, 0 for loss to follow up. If complete is not provided, the dataset is assumed to be complete.
mu0_treatment	scalar. Mean of the historical treatment group.
sd0_treatment	scalar. Standard deviation of the historical treatment group.
N0_treatment	scalar. Number of observations of the historical treatment group.
mu0_control	scalar. Mean of the historical control group.
sd0_control	scalar. Standard deviation of the historical control group.
N0_control	scalar. Number of observations of the historical control group.
alternative	character. The string specifying the alternative hypothesis, must be one of "greater" (default), "less" or "two.sided".
N_impute	scalar. Number of imputations for Monte Carlo simulation of missing data.
h0	scalar. Threshold for comparing two mean values. Default is $h_0=0$.
number_mcmc	scalar. Number of Monte Carlo Markov Chain draws in sampling posterior.
prob_ha	scalar. Probability of alternative hypothesis.
futility_prob	scalar. Probability of stopping early for futility.
expected_success_prob	scalar. Probability of stopping early for success.

discount_function	character. If incorporating historical data, specify the discount function. Currently supports the Weibull function (discount_function="weibull"), the scaled-Weibull function (discount_function="scaledweibull"), and the identity function (discount_function="identity"). The scaled-Weibull discount function scales the output of the Weibull CDF to have a max value of 1. The identity discount function uses the posterior probability directly as the discount weight. Default value is "identity". See bdpnormal for more details.
fix_alpha	logical. Fix alpha at alpha_max? Default value is FALSE.
alpha_max	scalar. Maximum weight the discount function can apply. Default is 1. For a two-arm trial, users may specify a vector of two values where the first value is used to weight the historical treatment group and the second value is used to weight the historical control group.
weibull_scale	scalar. Scale parameter of the Weibull discount function used to compute alpha, the weight parameter of the historical data. Default value is 0.135. For a two-arm trial, users may specify a vector of two values where the first value is used to estimate the weight of the historical treatment group and the second value is used to estimate the weight of the historical control group. Not used when discount_function = "identity".
weibull_shape	scalar. Shape parameter of the Weibull discount function used to compute alpha, the weight parameter of the historical data. Default value is 3. For a two-arm trial, users may specify a vector of two values where the first value is used to estimate the weight of the historical treatment group and the second value is used to estimate the weight of the historical control group. Not used when discount_function = "identity".

Value

	a list of output for the analysis of bayesian trial for normal mean.
prob_of_accepting_alternative	scalar. The input parameter of probability of accepting the alternative.
margin	scalar. The margin input value of difference between mean estimate of treatment and mean estimate of the control.
alternative	character. The input parameter of alternative hypothesis.
N_treatment	scalar. The number of patients enrolled in the experimental group for each simulation.
N_control	scalar. The number of patients enrolled in the control group for each simulation.
N_enrolled	vector. The number of patients enrolled in the trial (sum of control and experimental group for each simulation.)
N_complete	scalar. The number of patients who completed the trial and had no loss to follow-up.
post_prob_accept_alternative	vector. The final probability of accepting the alternative hypothesis after the analysis is done.
est_final	scalar. The final estimate of the difference in posterior estimate of treatment and posterior estimate of the control group.

`stop_futility` scalar. Did the trial stop for futility during imputation of patient who had loss to follow up? 1 for yes and 0 for no.

`stop_expected_success` scalar. Did the trial stop for early success during imputation of patient who had loss to follow up? 1 for yes and 0 for no.

`normal_outcome` *Parameters for treatment and control in normal case*

Description

Wrapper function for mean and standard deviation with normal outcome.

Usage

```
normal_outcome(mu_control = NULL, sd_control = NULL,
              mu_treatment = NULL, sd_treatment = NULL, data = NULL)
```

Arguments

<code>mu_control</code>	numeric. The mean for the control group.
<code>sd_control</code>	numeric. The standard deviation for the control group.
<code>mu_treatment</code>	numeric. The mean for the treatment group.
<code>sd_treatment</code>	numeric. The standard deviation for the treatment group.
<code>data</code>	NULL. stores the proportion of control and treatment, please do not fill it in.

Value

a list with proportion of control and treatment group.

Examples

```
normal_outcome(mu_control = 12, mu_treatment = 8, sd_treatment = 2.2, sd_control = 1.6)
```

`randomization` *Randomization allocation*

Description

Implements a randomization allocation for control and treatment arms with different randomization ratios and block sizes.

Usage

```
randomization(N_total, block = 2, allocation = c(1, 1))
```

Arguments

`N_total` an integer value of total sample size for randomization allocation.
`block` a vector value of the block size for randomization. Note that it needs to be a multiple of the sum of allocation.
`allocation` a numeric vector of the randomization allocation in the order `c(control, treatment)`.

Value

the randomization allocation with 0, 1 for control and treatment

Examples

```
# Implementing treatment allocation for control to treatment with 1:1.5 randomization ratio
randomization(N_total = 100, block = 5, allocation = c(2, 3))

# Treatment allocation with 2:1 for control to treatment
randomization(N_total = 70, block = 9, allocation = c(2, 1))

# Treatment allocation for control to treatment with 1:2 for control to treatment with
# multiple block sizes c(3, 9, 6)
randomization(N_total = 100, block = c(3, 9, 6), allocation = c(1, 2))

# For complete randomization set the N_total to block size
randomization(N_total = 100, block = 100, allocation = c(1, 1))
```

randomize	<i>Randomization scheme wrapper</i>
-----------	-------------------------------------

Description

Wrapper function for the randomization scheme in the trial.

Usage

```
randomize(block_size = 2, randomization_ratio = c(1, 1),
          .data = NULL)
```

Arguments

`block_size` integer. Block size for the complete randomization in a block.
`randomization_ratio` vector. The randomization allocation for control to treatment.
`.data` NULL. This should not be changed by the user.

Value

a list with randomization details (block size and ratio).

Examples

```
randomize(block_size = 100, randomization_ratio = c(2, 3))
randomize(block_size = 10, randomization_ratio = c(1, 4))
```

 simulate

Simulation wrapper for binomial and normal.

Description

Wrapper function for complete binomial and normal function to compute power and type 1 error.

Usage

```
simulate(input, no_of_sim = 10000, .data = NULL)
```

Arguments

input	list. Input function for all inputs in binomial and normal .
no_of_sim	numeric. Number of simulations to run
.data	NULL. stores the proportion of control and treatment, please do not fill it in.

Value

a list with results of the simulation (power and type I error) and the input.

input A list of input values used in the trial simulation.

power_data_frame. A data frame with the interim look and power at each look.

type1_error scalar. The type 1 error or the number of times the trial rejects the null when the parameters are simulated under the null hypothesis.

est_final vector. The final estimate of the difference in posterior estimate of treatment and posterior estimate of the control group for all the simulation.

post_prob_accept_alternative vector. The final probability of accepting the alternative for the simulations.

N_enrolled vector. The number of patients enrolled in the trial (sum of control and experimental group for each simulation.)

stop_futility vector. Did the trial stop for futility during imputation of patient who had loss to follow up? 1 for yes and 0 for no.

stop_expected_success vector. Did the trial stop for early success during imputation of patient who had loss to follow up? 1 for yes and 0 for no.

study_details	<i>Details of the clinical study</i>
---------------	--------------------------------------

Description

Wrapper function for details of the clinical trial simulation.

Usage

```
study_details(total_sample_size, study_period, interim_look = NULL,  
              prop_loss_to_followup = 0.1, .data = NULL)
```

Arguments

total_sample_size	integer. The number of sample size needed.
study_period	integer. The length of the study.
interim_look	vector. Vector with interim looks.
prop_loss_to_followup	integer. The proportion of loss to follow up.
.data	NULL. This should not be changed by the user.

Value

a list with sample size, length of the study, interim looks and proportion loss to follow up

Examples

```
study_details(total_sample_size = 300, study_period = 50, interim_look = c(210, 240, 270))
```

Index

*Topic **dataset**

binomialdata, 6
normaldata, 19

analysis, 2

bdpnormal, 4, 8, 17, 21
beta_prior, 3
binomial_analysis, 7
binomial_outcome, 9
binomialBACT, 4
binomialdata, 6

data_binomial, 10
data_normal, 10

enrollment, 4, 11, 17
enrollment_rate, 12

historical_binomial, 12
historical_normal, 14
hypothesis, 15

impute, 16

normal_analysis, 20
normal_outcome, 22
normalBACT, 16
normaldata, 19

randomization, 5, 17, 22
randomize, 23

simulate, 24
study_details, 25