Package: bfbin2arm (via r-universe)

June 1, 2026

Type Package

Title Bayes Factor Design for Two-Arm Binomial Trials

Version 0.1.3

Date 2026-05-06

Description Design and analysis of one- and two-stage binomial clinical phase II trials using Bayes factors. Implements Bayes factors for point-null and directional hypotheses, predictive densities under different hypotheses, and power and sample size calibration. Both one-arm trials with only a single treatment arm and two-arm trials with treatment and control arm are implemented for the one- and two-stage designs.

Depends R (>= 4.0.0)

Imports stats, VGAM, dplyr, parallel, utils, ggplot2, patchwork, rlang

Suggests knitr, rmarkdown, kableExtra, testthat (>= 3.0.0)

License GPL-3

Encoding UTF-8

RoxygenNote 7.3.3

VignetteBuilder knitr

URL https://arxiv.org/abs/2511.23144, https://arxiv.org/abs/2603.01715, https://arxiv.org/abs/2502.02914

NeedsCompilation no

Author Riko Kelter [aut, cre]

Maintainer Riko Kelter <rkelter@uni-koeln.de>

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

Date/Publication 2026-06-01 19:40:02 UTC

RemoteUrl https://github.com/cran/bfbin2arm

RemoteRef HEAD

RemoteSha 20d706194ec0b9fa82b38b4b1aab6f93e3515ba5

Contents

`.sustained_singlearm_feasibility`
Check sustained feasibility over future n

Description

Given vectors of operating characteristics over n, check whether power, type-I-error, and CE(H0) satisfy their thresholds at n and for at least sustain_n subsequent sample sizes.

Usage

```
.sustained_singlearm_feasibility(
  n_vec,
  power_vec,
  type1_vec,
  ce_vec,
  target_power,
  target_type1,
  target_ce,
  sustain_n
)
```

Arguments

`n_vec`	Integer vector of sample sizes.
`power_vec`	Numeric vector of power values (same length as n_vec).
`type1_vec`	Numeric vector of type-I-error values.
`ce_vec`	Numeric vector of CE(H0) values (may contain NA).
`target_power`	Numeric target power.
`target_type1`	Numeric target type-I-error.
`target_ce`	Numeric target CE(H0); if <= 0, CE constraint is ignored.
`sustain_n`	Integer, number of subsequent sample sizes that must also satisfy the constraints.

Value

Logical vector of length length(n_vec): TRUE if n_i and the next sustain_n sample sizes satisfy all active constraints.

as.data.frame.singlearm_onestage_bf_design

Convert a one-stage single-arm BF design to a data frame

Description

Convert a one-stage single-arm BF design to a data frame

Usage

```
## S3 method for class 'singlearm_onestage_bf_design'
as.data.frame(x, row.names = NULL, optional = FALSE, ...)
```

Arguments

x	An object of class "singlearm_onestage_bf_design".
row.names	Ignored.
optional	Ignored.
...	Currently unused.

Value

A data frame with the search results.

BFminus0

Bayes factor BF-0: H- vs H0

Description

Bayes factor BF-0: H- vs H0

Usage

```
BFminus0(BFminus1, BF01)
```

Arguments

BFminus1	Value of BF_{-1}.
BF01	Value of BF_{01}.

Value

Numeric scalar, BF_{-0}.

BFminus1 *Bayes factor BF-1: H- vs H1*

Description

Bayes factor BF-1: H- vs H1

Usage

```
BFminus1(y1, y2, n1, n2, a_1_a = 1, b_1_a = 1, a_2_a = 1, b_2_a = 1)
```

Arguments

y1	Number of successes in arm 1 (control).
y2	Number of successes in arm 2 (treatment).
n1	Sample size in arm 1.
n2	Sample size in arm 2.
a_1_a, b_1_a	Shape parameters of the Beta prior for p_1 under the alternative (analysis prior).
a_2_a, b_2_a	Shape parameters of the Beta prior for p_2 under the alternative (analysis prior).

Value

Numeric scalar, BF_{-1}.

BFplus0 *Bayes factor BF+0: H+ vs H0*

Description

Bayes factor BF+0: H+ vs H0

Usage

```
BFplus0(BFplus1, BF01)
```

Arguments

BFplus1	Value of BF_{+1}.
BF01	Value of BF_{01}.

Value

Numeric scalar, BF_{+0}.

`BFplus1` *Bayes factor BF+1: H+ vs H1*

Description

Bayes factor BF+1: H+ vs H1

Usage

```
BFplus1(y1, y2, n1, n2, a_1_a = 1, b_1_a = 1, a_2_a = 1, b_2_a = 1)
```

Arguments

`y1`	Number of successes in arm 1 (control).
`y2`	Number of successes in arm 2 (treatment).
`n1`	Sample size in arm 1.
`n2`	Sample size in arm 2.
`a_1_a, b_1_a`	Shape parameters of the Beta prior for p_1 under the alternative (analysis prior).
`a_2_a, b_2_a`	Shape parameters of the Beta prior for p_2 under the alternative (analysis prior).

Value

Numeric scalar, BF_{+1} = posterior odds / prior odds for H+ vs H1.

`BFplusMinus` *Bayes factor BF+-: H+ vs H-*

Description

Bayes factor BF+-: H+ vs H-

Usage

```
BFplusMinus(BFplus1, BFminus1)
```

Arguments

`BFplus1`	Value of BF_{+1}.
`BFminus1`	Value of BF_{-1}.

Value

Numeric scalar, BF_{+-}.

design_singlearm_bf *Design or evaluate a single-arm two-stage Bayes factor trial*

Description

Calibrates or evaluates a single-arm two-stage Bayes factor design for a binary endpoint with one interim analysis for futility.

Usage

```
design_singlearm_bf(
  n1_min,
  n2_max,
  k,
  k_f,
  p0,
  a0 = 1,
  b0 = 1,
  a1 = 1,
  b1 = 1,
  dp = NA_real_,
  da0 = 1,
  db0 = 1,
  da1 = 1,
  db1 = 1,
  type = c("point", "direction"),
  calibration = c("Bayesian", "frequentist", "hybrid", "full"),
  target_power = 0.8,
  target_type1 = 0.05,
  target_ce_h0 = 0,
  target_freq_power = 0.8,
  target_freq_type1 = 0.05,
  algorithm = c("optimal", "manual"),
  interim = NULL,
  final = NULL,
  power_cushion = 0,
  ...
)
```

Arguments

n1_min	Integer. Minimum admissible interim sample size.
n2_max	Integer. Maximum admissible final sample size.
k	Numeric scalar greater than 0. Efficacy threshold on the BF_{01} scale.
k_f	Numeric scalar greater than 1. Futility threshold on the BF_{01} scale.
p0	Numeric scalar in $(0, 1)$. Null response probability.

a0, b0	Positive numeric scalars. Beta analysis-prior parameters under H_0.
a1, b1	Positive numeric scalars. Beta analysis-prior parameters under H_1.
dp	Optional numeric scalar in $(0, 1)$. Fixed point alternative used for frequentist power calculations under H_1.
da0, db0	Positive numeric scalars. Beta design-prior parameters under H_0.
da1, db1	Positive numeric scalars. Beta design-prior parameters under H_1.
type	Character string specifying the Bayes-factor test. One of "point" or "direction".
calibration	Character string specifying the calibration mode. One of "Bayesian", "frequentist", "hybrid", or "full".
target_power	Numeric scalar in $(0, 1)$. Target corrected Bayesian power.
target_type1	Numeric scalar in $(0, 1)$. Target corrected Bayesian type-I error.
target_ce_h0	Numeric scalar in $[0, 1)$. Optional lower bound on the corrected Bayesian probability of compelling evidence in favour of H_0.
target_freq_power	Numeric scalar in $(0, 1)$. Target corrected frequentist power at dp.
target_freq_type1	Numeric scalar in $(0, 1)$. Target corrected frequentist type-I error at $p = p_0$.
algorithm	Character string specifying whether the design should be optimized or only evaluated.
interim	Optional integer interim sample size used when algorithm = "manual".
final	Optional integer final sample size used when algorithm = "manual".
power_cushion	Optional additive cushion applied only in the fixed-sample anchor search of the first optimization step. This can be useful because introducing an interim futility analysis typically reduces corrected power relative to the fixed-sample anchor.
...	Reserved for future extensions.

Details

The design uses the Bayes factor BF_{01}. Small values of BF_{01} indicate evidence against H_0, so final efficacy is concluded when BF_{01}
lek. Large values indicate evidence in favour of H_0, so interim futility is concluded when BF_{01}
gek_f.

Analysis priors are specified separately under H_0 and H_1 via a0, b0, a1, b1. Design priors are specified separately under H_0 and H_1 via da0, db0, da1, db1.

Value

An object of class "singlearm_bf_design".

design_singlearm_onestage_bf
Design or evaluate a one-stage single-arm Bayes factor trial

Description

Calibrates or evaluates a one-stage single-arm Bayes factor design for a binary endpoint.

Usage

```
design_singlearm_onestage_bf(
  n_min,
  n_max,
  k,
  k_ce = NULL,
  p0,
  a0 = 1,
  b0 = 1,
  a1 = 1,
  b1 = 1,
  dp = NA_real_,
  da0 = 1,
  db0 = 1,
  da1 = 1,
  db1 = 1,
  type = c("point", "direction"),
  calibration = c("Bayesian", "frequentist", "hybrid", "full"),
  target_power = 0.8,
  target_type1 = 0.05,
  target_ce_h0 = 0,
  target_freq_power = 0.8,
  target_freq_type1 = 0.05,
  algorithm = c("optimal", "manual"),
  n = NULL,
  power_cushion = 0,
  sustain_n = 10L,
  ...
)
```

Arguments

n_min	Integer. Minimum admissible sample size.
n_max	Integer. Maximum admissible sample size.
k	Numeric scalar greater than 0. Evidence threshold on the BF_{01} scale for efficacy, used for power and type-I error.

k_ce	Optional numeric scalar greater than 1. Threshold on the BF_{01} scale used for CE(H0) / PCE(H0). Must be supplied when target_ce_h0 > 0.
p0	Numeric scalar in $(0, 1)$. Null response probability.
a0, b0	Positive numeric scalars. Beta analysis-prior parameters under H_0.
a1, b1	Positive numeric scalars. Beta analysis-prior parameters under H_1.
dp	Optional numeric scalar in $(0, 1)$. Fixed point alternative used for frequentist power calculations under H_1.
da0, db0	Positive numeric scalars. Beta design-prior parameters under H_0.
da1, db1	Positive numeric scalars. Beta design-prior parameters under H_1.
type	Character string specifying the Bayes-factor test. One of "point" or "direction".
calibration	Character string specifying the calibration mode. One of "Bayesian", "frequentist", "hybrid", or "full".
target_power	Numeric scalar in $(0, 1)$. Target corrected Bayesian power.
target_type1	Numeric scalar in $(0, 1)$. Target corrected Bayesian type-I error.
target_ce_h0	Numeric scalar in $[0, 1)$. Optional lower bound on the corrected Bayesian probability of compelling evidence in favour of H_0.
target_freq_power	Numeric scalar in $(0, 1)$. Target corrected frequentist power at dp.
target_freq_type1	Numeric scalar in $(0, 1)$. Target corrected frequentist type-I error at $p = p_0$.
algorithm	Character string specifying whether the design should be optimized or only evaluated.
n	Optional integer sample size used when algorithm = "manual".
power_cushion	Optional additive cushion applied to the power targets in the optimizer.
sustain_n	Non-negative integer. A candidate design is considered feasible only if the relevant operating characteristics satisfy their target constraints at the candidate sample size and for the next sustain_n larger sample sizes, subject to the search range. This also applies to the CE(H0) constraint when target_ce_h0 > 0.
...	Reserved for future extensions.

Details

The design uses the Bayes factor BF_{01}. Small values of BF_{01} indicate evidence against H_0, so efficacy is concluded when $BF_{01} \leq k$. Large values indicate evidence in favour of H_0, and the optional CE(H0) / PCE(H0) constraint is evaluated using the separate threshold k_ce.

Analysis priors are specified separately under H_0 and H_1 via a0, b0, a1, b1. Design priors are specified separately under H_0 and H_1 via da0, db0, da1, db1.

Value

An object of class "singlearm_onestage_bf_design".

design_twoarm_onestage_bf
Design or evaluate a one-stage two-arm Bayes factor trial

Description

Calibrates or evaluates a one-stage two-arm Bayes factor design for a binary endpoint with fixed randomisation between the two arms.

Usage

```
design_twoarm_onestage_bf(
  n_min,
  n_max,
  k = 1/3,
  k_f = 3,
  test = c("BF01", "BF+0", "BF-0", "BF+-"),
  a_0_d = 1,
  b_0_d = 1,
  a_0_a = 1,
  b_0_a = 1,
  a_1_d = 1,
  b_1_d = 1,
  a_2_d = 1,
  b_2_d = 1,
  a_1_a = 1,
  b_1_a = 1,
  a_2_a = 1,
  b_2_a = 1,
  a_1_d_Hminus = 1,
  b_1_d_Hminus = 1,
  a_2_d_Hminus = 1,
  b_2_d_Hminus = 1,
  a_1_a_Hminus = 1,
  b_1_a_Hminus = 1,
  a_2_a_Hminus = 1,
  b_2_a_Hminus = 1,
  alloc1 = 0.5,
  alloc2 = 0.5,
  calibration = c("Bayesian", "frequentist", "hybrid", "full"),
  target_power = 0.8,
  target_type1 = 0.05,
  target_ce_h0 = 0,
  target_freq_power = 0.8,
  target_freq_type1 = 0.05,
  p1_grid = seq(0.01, 0.99, 0.02),
  p2_grid = seq(0.01, 0.99, 0.02),
```

```
  p1_power = NULL,
  p2_power = NULL,
  power_cushion = 0,
  sustain_n = 10L,
  report_freq_type1 = FALSE,
  algorithm = c("optimal", "manual"),
  n_total = NULL,
  progress = FALSE,
  ...
)
```

Arguments

`n_min` Integer. Minimum admissible total sample size.

`n_max` Integer. Maximum admissible total sample size.

`k` Numeric scalar greater than 0. Evidence threshold used for power and type-I error.

`k_f` Numeric scalar greater than 1. Threshold used for CE(H0) / PCE(H0).

`test` Character string, one of "BF01", "BF+0", "BF-0", or "BF+-".

`a_0_d, b_0_d, a_0_a, b_0_a`
Shape parameters for design and analysis priors under H_0.

`a_1_d, b_1_d, a_2_d, b_2_d`
Shape parameters for design priors under H_1 or H_+.

`a_1_a, b_1_a, a_2_a, b_2_a`
Shape parameters for analysis priors under H_1 or H_+.

`a_1_d_Hminus, b_1_d_Hminus, a_2_d_Hminus, b_2_d_Hminus`
Optional design priors under H_- for directional tests.

`a_1_a_Hminus, b_1_a_Hminus, a_2_a_Hminus, b_2_a_Hminus`
Optional analysis priors under H_- for directional tests.

`alloc1, alloc2` Fixed randomisation probabilities for arm 1 and arm 2. Must be positive and sum to 1.

`calibration` Character string specifying the calibration mode. One of "Bayesian", "frequentist", "hybrid", or "full".

`target_power` Numeric scalar in $(0, 1)$. Target corrected Bayesian power.

`target_type1` Numeric scalar in $(0, 1)$. Target corrected Bayesian type-I error.

`target_ce_h0` Numeric scalar in $[0, 1)$. Optional lower bound on the corrected Bayesian probability of compelling evidence in favour of H_0 (or H_- for test = "BF+-").

`target_freq_power`
Numeric scalar in $(0, 1)$. Target frequentist power under p1_power, p2_power.

`target_freq_type1`
Numeric scalar in $(0, 1)$. Target frequentist type-I error.

`p1_grid, p2_grid`
Grids of true proportions used to compute supremum frequentist type-I error.

p1_power, p2_power
: Optional true proportions used for frequentist power.

power_cushion
: Non-negative numeric scalar. Optional additive cushion applied to the power targets during calibration.

sustain_n
: Non-negative integer. A candidate total sample size is considered feasible only if the relevant target constraints hold at that total sample size and for the next sustain_n larger total sample sizes in the search range.

report_freq_type1
: Logical. If TRUE, compute and report the frequentist type-I error for the final selected design even when the chosen calibration mode does not use frequentist criteria. This additional computation has no effect on the calibration itself. Defaults to FALSE.

algorithm
: Character string specifying whether the design should be optimized or only evaluated.

n_total
: Optional integer total sample size used when algorithm = "manual".

progress
: Logical; if TRUE, print simple progress information during optimization.

...
: Reserved for future extensions.

Details

The design uses one of the Bayes factor tests implemented in powertwoarmbinbf01(). Small values of the relevant inverted Bayes factor indicate evidence against the null, so efficacy is concluded when the Bayes factor is below k. Large values indicate evidence in favour of the null (or H_- for test = "BF+-"), and the optional CE(H0) / PCE(H0) constraint is evaluated using k_f.

Value

An object of class "twoarm_onestage_bf_design".

---

design_twoarm_twostage_bf
Design an optimal two-stage two-arm Bayes factor trial

Description

Calibrates a two-stage two-arm Bayes factor design for a binary endpoint by calling optimal_twostage_2arm_bf() and packaging the result in a user-friendly object of class "twoarm_twostage_bf_design".

Usage

```
design_twoarm_twostage_bf(
  n1_min,
  n2_max,
  alloc1 = 0.5,
  alloc2 = 0.5,
```

```
  power_cushion = 0,
  interim_fraction = c(0.25, 0.75),
  grid_step = 1L,
  coarse_step = 4L,
  max_iter = 40L,
  ncores = getOption("bfbin2arm.ncores", 1L),
  k = 1/3,
  k_f = 3,
  test = c("BF01", "BF+0", "BF-0", "BF+-"),
  a_0_d = 1,
  b_0_d = 1,
  a_0_a = 1,
  b_0_a = 1,
  a_1_d = 1,
  b_1_d = 1,
  a_2_d = 1,
  b_2_d = 1,
  a_1_a = 1,
  b_1_a = 1,
  a_2_a = 1,
  b_2_a = 1,
  a_1_d_Hminus = 1,
  b_1_d_Hminus = 1,
  a_2_d_Hminus = 1,
  b_2_d_Hminus = 1,
  a_1_a_Hminus = 1,
  b_1_a_Hminus = 1,
  a_2_a_Hminus = 1,
  b_2_a_Hminus = 1,
  calibration = c("Bayesian", "frequentist", "hybrid"),
  calibration_en = c("Bayesian", "frequentist"),
  target_power = 0.8,
  target_type1 = 0.05,
  target_ce_h0 = 0,
  target_freq_power = 0.8,
  target_freq_type1 = 0.05,
  p1_power = NULL,
  p2_power = NULL,
  p1_en_h0 = NULL,
  p2_en_h0 = NULL,
  p_null_grid = NULL,
  progress = FALSE,
  ...
)
```

Arguments

n1_min	Numeric vector of length 2, minimum interim sample sizes for arms 1 and 2.

`n2_max` Numeric vector of length 2, maximum final sample sizes for arms 1 and 2.

`alloc1`, `alloc2` Positive numbers, allocation probabilities to arms 1 and 2.

`power_cushion` Numeric scalar, optional extra power cushion used in the fixed-sample search of step 1.

`interim_fraction`
: Numeric vector of length 2 giving lower and upper bounds for the interim sample size in each arm as a fraction of the fixed sample size.

`grid_step` Positive integer giving the spacing of the interim design grid.

`coarse_step` Positive integer giving the spacing of the coarse fixed-sample search grid in step 1.

`max_iter` Integer, maximum number of total fixed-sample sizes searched in step 1.

`ncores` Integer; number of parallel worker processes to use in the calibration. Defaults to `getOption("bfbin2arm.ncores", 1L)`. In vignettes and examples, a conservative value (e.g. 1 or 2) is recommended for CRAN checks, whereas users can increase this to exploit all available cores on their own machines.

`k` Numeric scalar, efficacy threshold; evidence against the null hypothesis is declared when the corresponding Bayes factor is smaller than `k`.

`k_f` Numeric scalar, futility threshold; compelling evidence for the null hypothesis is declared when the corresponding Bayes factor is at least `k_f`.

`test` Character string, one of `"BF01"`, `"BF+0"`, `"BF-0"`, `"BF+-"`.

`a_0_d`, `b_0_d`, `a_0_a`, `b_0_a`
: Shape parameters for design and analysis priors under H_0.

`a_1_d`, `b_1_d`, `a_2_d`, `b_2_d`
: Shape parameters for design priors under H_1 or H_+.

`a_1_a`, `b_1_a`, `a_2_a`, `b_2_a`
: Shape parameters for analysis priors under H_1 or H_+.

`a_1_d_Hminus`, `b_1_d_Hminus`, `a_2_d_Hminus`, `b_2_d_Hminus`
: Optional design priors under H_- for directional tests.

`a_1_a_Hminus`, `b_1_a_Hminus`
: Shape parameters of the analysis prior under the directional null hypothesis H0- for arm 1.

`a_2_a_Hminus`, `b_2_a_Hminus`
: Shape parameters of the analysis prior under the directional null hypothesis H0- for arm 2.

`calibration` Character string specifying the calibration mode at the wrapper level. One of `"Bayesian"`, `"frequentist"`, or `"hybrid"`. This is passed to `optimal_twostage_2arm_bf()` as `calibration_mode`.

`calibration_en` Character string or `NULL` specifying whether the design is ranked by Bayesian or frequentist expected sample size under the null hypothesis. This is passed to `optimal_twostage_2arm_bf()` as `calibration_EN`.

`target_power`, `target_type1`, `target_ce_h0`, `target_freq_power`, `target_freq_type1`
: Numeric targets for Bayesian and frequentist operating characteristics. These are translated to the `alpha`, `beta`, `alpha_freq`, and `beta_freq` arguments of `optimal_twostage_2arm_bf()`.

p1_power, p2_power
: Optional true response probabilities used for frequentist power. Passed through to `optimal_twostage_2arm_bf()`.

p1_en_h0, p2_en_h0
: Optional null response probabilities used when `calibration_en = "frequentist"` to compute expected sample size under the null.

p_null_grid
: Optional grid of null response probabilities used for frequentist type-I-error maximisation.

progress
: Logical; if `TRUE`, print simple progress information during the calibration.

...
: Reserved for future extensions; currently ignored.

Details

The design uses one of the Bayes factor tests implemented in `powertwoarmbinbf01()`. Small values of the relevant inverted Bayes factor indicate evidence against the null, so efficacy is concluded when the Bayes factor is below `k`. Large values indicate evidence in favour of the null (or H_- for `test = "BF+-"`), and the optional CE(H0) / PCE(H0) constraint is evaluated using `k_f`.

Value

An object of class `"twoarm_twostage_bf_design"`.

`ntwoarmbinbf01` *Sample size calibration for two-arm binomial Bayes factor designs*

Description

Searches over a grid of total sample sizes n to find the smallest n such that Bayesian power, Bayesian type-I error, and probability of compelling evidence under H0 meet specified design criteria. Optionally, frequentist type-I error and power constraints are also evaluated. Unequal fixed randomisation between the two arms is allowed via alloc1 and alloc2.

Backward-compatible wrapper around `design_twoarm_onestage_bf()`.

Usage

```
ntwoarmbinbf01(
  k = 1/3,
  k_f = 3,
  power = 0.8,
  alpha = 0.05,
  pce_H0 = 0.9,
  test = c("BF01", "BF+0", "BF-0", "BF+-"),
  nrange = c(10, 150),
  n_step = 1,
  progress = TRUE,
  compute_freq_t1e = FALSE,
```

```
  p1_grid = seq(0.01, 0.99, 0.02),
  p2_grid = seq(0.01, 0.99, 0.02),
  p1_power = NULL,
  p2_power = NULL,
  a_0_d = 1,
  b_0_d = 1,
  a_0_a = 1,
  b_0_a = 1,
  a_1_d = 1,
  b_1_d = 1,
  a_2_d = 1,
  b_2_d = 1,
  a_1_a = 1,
  b_1_a = 1,
  a_2_a = 1,
  b_2_a = 1,
  output = c("plot", "numeric"),
  a_1_d_Hminus = 1,
  b_1_d_Hminus = 1,
  a_2_d_Hminus = 1,
  b_2_d_Hminus = 1,
  a_1_a_Hminus = 1,
  b_1_a_Hminus = 1,
  a_2_a_Hminus = 1,
  b_2_a_Hminus = 1,
  alloc1 = 0.5,
  alloc2 = 0.5,
  sustain_n = 10L
)

ntwoarmbinbf01(
  k = 1/3,
  k_f = 3,
  power = 0.8,
  alpha = 0.05,
  pce_H0 = 0.9,
  test = c("BF01", "BF+0", "BF-0", "BF+-"),
  nrange = c(10, 150),
  n_step = 1,
  progress = TRUE,
  compute_freq_t1e = FALSE,
  p1_grid = seq(0.01, 0.99, 0.02),
  p2_grid = seq(0.01, 0.99, 0.02),
  p1_power = NULL,
  p2_power = NULL,
  a_0_d = 1,
  b_0_d = 1,
  a_0_a = 1,
```

```
  b_0_a = 1,
  a_1_d = 1,
  b_1_d = 1,
  a_2_d = 1,
  b_2_d = 1,
  a_1_a = 1,
  b_1_a = 1,
  a_2_a = 1,
  b_2_a = 1,
  output = c("plot", "numeric"),
  a_1_d_Hminus = 1,
  b_1_d_Hminus = 1,
  a_2_d_Hminus = 1,
  b_2_d_Hminus = 1,
  a_1_a_Hminus = 1,
  b_1_a_Hminus = 1,
  a_2_a_Hminus = 1,
  b_2_a_Hminus = 1,
  alloc1 = 0.5,
  alloc2 = 0.5,
  sustain_n = 10L
)
```

Arguments

k	Evidence threshold for rejecting the null (inverted BF).
k_f	Evidence threshold for "compelling evidence" in favour of the null.
power	Target Bayesian power.
alpha	Target Bayesian type-I error.
pce_H0	Target probability of compelling evidence under H_0.
test	Character string, one of "BF01", "BF+0", "BF-0", "BF+-".
nrange	Integer vector of length 2 giving the search range for total n.
n_step	Step size for n. Currently only n_step = 1 is supported in the object-based calibration workflow.
progress	Logical; if TRUE, print progress to the console.
compute_freq_t1e	Logical; if TRUE, compute frequentist type-I error over a grid.
p1_grid, p2_grid	Grids of true proportions for frequentist T1E.
p1_power, p2_power	Optional true proportions for frequentist power.
a_0_d, b_0_d, a_0_a, b_0_a	Shape parameters for design and analysis priors under H_0.
a_1_d, b_1_d, a_2_d, b_2_d	Shape parameters for design priors under H_1 or H_+.

a_1_a, b_1_a, a_2_a, b_2_a
: Shape parameters for analysis priors under H_1 or H_+.

output
: "plot" or "numeric".

a_1_d_Hminus, b_1_d_Hminus, a_2_d_Hminus, b_2_d_Hminus
: Optional design priors under H_- for directional tests.

a_1_a_Hminus, b_1_a_Hminus, a_2_a_Hminus, b_2_a_Hminus
: Shape parameters for analysis priors under H-.

alloc1, alloc2
: Fixed randomisation probabilities for arm 1 and arm 2; must be positive and sum to 1.

sustain_n
: Non-negative integer. A candidate total sample size is considered feasible only if the relevant target constraints hold at that total sample size and for the next sustain_n larger total sample sizes in the search range.

Value

If output = "plot", returns invisibly a list with recommended sample sizes and a ggplot object printed to the device. If output = "numeric", returns a list with recommended n and summary.

If output = "numeric", returns a "twoarm_onestage_bf_design" object. If output = "plot", the plot is printed and the design object is returned invisibly.

Examples

```
# Standard calibration with equal allocation: power 80%, type-I 5%, CE(H0) 80%

ntwoarmbinbf01(power = 0.8, alpha = 0.05, pce_H0 = 0.8, output = "numeric")


# 1:2 allocation (control:treatment) via alloc1 = 1/3, alloc2 = 2/3

ntwoarmbinbf01(power = 0.8, alpha = 0.05, pce_H0 = 0.8,
               alloc1 = 1/3, alloc2 = 2/3, output = "numeric")


# BF+0 directional test with plot

ntwoarmbinbf01(power = 0.8, alpha = 0.05, pce_H0 = 0.9,
               test = "BF+0", output = "plot")
```

---

optimal_onestage_singlearm_bf

Internal calibration routine for one-stage single-arm BF designs

---

Description

Internal calibration routine for one-stage single-arm BF designs

Usage

```
optimal_onestage_singlearm_bf(
  n_min,
  n_max,
  k,
  p0,
  a0 = 1,
  b0 = 1,
  a1 = 1,
  b1 = 1,
  dp = NA_real_,
  da0 = 1,
  db0 = 1,
  da1 = 1,
  db1 = 1,
  type = c("point", "direction"),
  calibration = c("Bayesian", "frequentist", "hybrid", "full"),
  target_power = 0.8,
  target_type1 = 0.05,
  target_ce_h0 = 0,
  target_freq_power = 0.8,
  target_freq_type1 = 0.05,
  power_cushion = 0,
  k_ce = NULL,
  sustain_n = 10L
)
```

Arguments

`n_min`	Integer. Minimum admissible sample size in the search grid.
`n_max`	Integer. Maximum admissible sample size in the search grid.
`k`	Numeric scalar greater than 0. Evidence threshold on the BF_{01} scale used for efficacy.
`p0`	Numeric scalar in $(0, 1)$. Null response probability.
`a0`, `b0`	Positive numeric scalars. Beta analysis-prior parameters under H_0.
`a1`, `b1`	Positive numeric scalars. Beta analysis-prior parameters under H_1.
`dp`	Optional numeric scalar in $(0, 1)$. Fixed point alternative used for frequentist power calculations under H_1.
`da0`, `db0`	Positive numeric scalars. Beta design-prior parameters under H_0.
`da1`, `db1`	Positive numeric scalars. Beta design-prior parameters under H_1.
`type`	Character string specifying the Bayes-factor test. One of `"point"` or `"direction"`.
`calibration`	Character string specifying the calibration mode. One of `"Bayesian"`, `"frequentist"`, `"hybrid"`, or `"full"`.
`target_power`	Numeric scalar in $(0, 1)$. Target corrected Bayesian power.
`target_type1`	Numeric scalar in $(0, 1)$. Target corrected Bayesian type-I error.

`target_ce_h0` Numeric scalar in $[0, 1)$. Optional lower bound on the corrected Bayesian probability of compelling evidence in favour of H_0.

`target_freq_power` Numeric scalar in $(0, 1)$. Target corrected frequentist power at `dp`.

`target_freq_type1` Numeric scalar in $(0, 1)$. Target corrected frequentist type-I error at $p = p_0$.

`power_cushion` Non-negative numeric scalar. Optional additive cushion applied to the power targets during calibration.

`k_ce` Optional numeric scalar greater than 1. Threshold on the BF_{01} scale used for CE(H0) / PCE(H0) calculations.

`sustain_n` Non-negative integer. A candidate sample size is declared feasible only if the relevant constraints are satisfied at that sample size and for the next `sustain_n` larger sample sizes, subject to the search range.

Value

A list with feasibility, selected design, operating characteristics, and full search results.

`optimal_onestage_twoarm_bf` *Internal calibration routine for one-stage two-arm BF designs*

Description

Internal calibration routine for one-stage two-arm BF designs

Usage

```
optimal_onestage_twoarm_bf(
  n_min,
  n_max,
  k = 1/3,
  k_f = 3,
  test = c("BF01", "BF+0", "BF-0", "BF+-"),
  a_0_d = 1,
  b_0_d = 1,
  a_0_a = 1,
  b_0_a = 1,
  a_1_d = 1,
  b_1_d = 1,
  a_2_d = 1,
  b_2_d = 1,
  a_1_a = 1,
  b_1_a = 1,
  a_2_a = 1,
  b_2_a = 1,
```

```
  a_1_d_Hminus = 1,
  b_1_d_Hminus = 1,
  a_2_d_Hminus = 1,
  b_2_d_Hminus = 1,
  a_1_a_Hminus = 1,
  b_1_a_Hminus = 1,
  a_2_a_Hminus = 1,
  b_2_a_Hminus = 1,
  alloc1 = 0.5,
  alloc2 = 0.5,
  calibration = c("Bayesian", "frequentist", "hybrid", "full"),
  target_power = 0.8,
  target_type1 = 0.05,
  target_ce_h0 = 0,
  target_freq_power = 0.8,
  target_freq_type1 = 0.05,
  p1_grid = seq(0.01, 0.99, 0.02),
  p2_grid = seq(0.01, 0.99, 0.02),
  p1_power = NULL,
  p2_power = NULL,
  power_cushion = 0,
  sustain_n = 10L,
  progress = FALSE
)
```

Arguments

- `n_min` Integer. Minimum admissible total sample size.
- `n_max` Integer. Maximum admissible total sample size.
- `k` Numeric scalar greater than 0. Evidence threshold used for power and type-I error.
- `k_f` Numeric scalar greater than 1. Threshold used for CE(H0).
- `test` Character string, one of "BF01", "BF+0", "BF-0", or "BF+-".
- `a_0_d, b_0_d, a_0_a, b_0_a` Shape parameters for design and analysis priors under H_0.
- `a_1_d, b_1_d, a_2_d, b_2_d` Shape parameters for design priors under H_1 or H_+.
- `a_1_a, b_1_a, a_2_a, b_2_a` Shape parameters for analysis priors under H_1 or H_+.
- `a_1_d_Hminus, b_1_d_Hminus, a_2_d_Hminus, b_2_d_Hminus` Optional design priors under H_-.
- `a_1_a_Hminus, b_1_a_Hminus, a_2_a_Hminus, b_2_a_Hminus` Optional analysis priors under H_-.
- `alloc1, alloc2` Fixed randomisation probabilities for arm 1 and arm 2.
- `calibration` Character string specifying the calibration mode.

`target_power`, `target_type1`, `target_ce_h0`, `target_freq_power`, `target_freq_type1`
: Target operating characteristics.

`p1_grid`, `p2_grid`
: Grids for supremum frequentist type-I error.

`p1_power`, `p2_power`
: Optional true proportions for frequentist power.

`power_cushion` Non-negative numeric scalar applied to power targets.

`sustain_n` Non-negative integer. Rolling feasibility window size.

`progress` Logical; if `TRUE`, emit progress information.

Value

A list with feasibility, selected design, operating characteristics, and full search results.

`optimal_twostage_2arm_bf`
Optimal two-stage two-arm Bayes-factor design for binary endpoints

---

Description

Computes an optimal two-stage two-arm Bayes-factor design for binary endpoints, minimizing the expected sample size under the null hypothesis while correcting the operating characteristics for the possibility of early stopping for futility.

Usage

```
optimal_twostage_2arm_bf(
  alpha = 0.05,
  beta = 0.2,
  k = 1/3,
  k_f = 3,
  n1_min = c(5, 5),
  n2_max = c(50, 50),
  alloc1 = 0.5,
  alloc2 = 0.5,
  power_cushion = 0,
  pceH0 = NULL,
  interim_fraction = c(0, 1),
  grid_step = 1L,
  coarse_step = 10L,
  progress = TRUE,
  max_iter = 10000L,
  ncores = getOption("bfbin2arm.ncores", 1L),
  compute_freq_oc = NULL,
  calibration_mode = c("Bayesian", "frequentist", "hybrid"),
```

```
  calibration_EN = NULL,
  p1_EN_H0 = NULL,
  p2_EN_H0 = NULL,
  alpha_freq = alpha,
  beta_freq = beta,
  p1_power = NULL,
  p2_power = NULL,
  p_null_grid = NULL,
  test = "BF01",
  a_0_d = 1,
  b_0_d = 1,
  a_0_a = 1,
  b_0_a = 1,
  a_1_d = 1,
  b_1_d = 1,
  a_2_d = 1,
  b_2_d = 1,
  a_1_a = 1,
  b_1_a = 1,
  a_2_a = 1,
  b_2_a = 1,
  a_1_d_Hminus = 1,
  b_1_d_Hminus = 1,
  a_2_d_Hminus = 1,
  b_2_d_Hminus = 1,
  a_1_a_Hminus = 1,
  b_1_a_Hminus = 1,
  a_2_a_Hminus = 1,
  b_2_a_Hminus = 1
)
```

Arguments

alpha	Numeric scalar, Bayesian type-I-error target.
beta	Numeric scalar, 1 minus the minimal Bayesian power target.
k	Numeric scalar, efficacy threshold; evidence against the null hypothesis is declared when the corresponding Bayes factor is smaller than k.
k_f	Numeric scalar, futility threshold; compelling evidence for the null hypothesis is declared when the corresponding Bayes factor is at least k_f.
n1_min	Numeric vector of length 2, minimum interim sample sizes for arms 1 and 2.
n2_max	Numeric vector of length 2, maximum final sample sizes for arms 1 and 2.
alloc1, alloc2	Positive numbers, allocation probabilities to arms 1 and 2.
power_cushion	Numeric scalar, optional extra power cushion used in the fixed-sample search of step 1.
pceH0	Optional numeric scalar in [0,1]. If specified, candidate two-stage designs must satisfy corrected CE_H0 >= pceH0.

interim_fraction
: Numeric vector of length 2 giving lower and upper bounds for the interim sample size in each arm as a fraction of the fixed sample size.

grid_step
: Positive integer giving the spacing of the interim design grid.

coarse_step
: Positive integer giving the spacing of the coarse fixed-sample search grid in step 1.

progress
: Logical; if TRUE, prints progress information.

max_iter
: Integer, maximum number of total fixed-sample sizes searched in step 1.

ncores
: Integer; number of parallel worker processes to use in the calibration. Defaults to getOption("bfbin2arm.ncores", 1L). In vignettes and examples, a conservative value (e.g. 1 or 2) is recommended for CRAN checks, whereas users can increase this to exploit all available cores on their own machines.

compute_freq_oc
: Logical or NULL. Controls whether frequentist operating characteristics are computed for candidate two-stage designs during the search.

calibration_mode
: Character string specifying the calibration mode. Must be one of "Bayesian", "frequentist", or "hybrid".

calibration_EN
: Character string or NULL specifying whether the design is ranked by Bayesian or frequentist expected sample size under the null hypothesis.

p1_EN_H0, p2_EN_H0
: Numeric scalars specifying the null response probabilities in control and treatment arm used when calibration_EN = "frequentist".

alpha_freq
: Numeric scalar, frequentist type-I error target.

beta_freq
: Numeric scalar, 1 minus the frequentist power target.

p1_power, p2_power
: Numeric scalars specifying the success probabilities in control and treatment arm used for the frequentist power calculation.

p_null_grid
: Optional numeric vector giving the grid of null response probabilities used for frequentist type-I-error maximization. If NULL, a default grid is used.

test
: Character string, one of "BF01", "BF+0", "BF-0", "BF+-".

a_0_d, b_0_d, a_0_a, b_0_a
: Shape parameters for design and analysis priors under H_0.

a_1_d, b_1_d, a_2_d, b_2_d
: Shape parameters for design priors under H_1 or H_+.

a_1_a, b_1_a, a_2_a, b_2_a
: Shape parameters for analysis priors under H_1 or H_+.

a_1_d_Hminus, b_1_d_Hminus, a_2_d_Hminus, b_2_d_Hminus
: Optional design priors under H_- for directional tests.

a_1_a_Hminus, b_1_a_Hminus
: Shape parameters of the analysis prior under the directional null hypothesis H0- for arm 1.

a_2_a_Hminus, b_2_a_Hminus
: Shape parameters of the analysis prior under the directional null hypothesis H0- for arm 2.

Value

A list with the following components:

design — Four-element integer vector containing the selected two-stage design: interim sample sizes in arms 1 and 2 followed by final sample sizes in arms 1 and 2.

naive_oc — Named list of uncorrected fixed-sample operating characteristics and fixed-sample sizes found in step 1.

occ — Named numeric vector of corrected Bayesian operating characteristics for the selected two-stage design.

priors — List storing design hyperparameters and search settings.

freq_occ — Named numeric vector with fixed-sample and two-stage frequentist operating characteristics for the final design when frequentist calibration or reporting is active; otherwise NULL.

conv — Character string describing the search outcome. Typical values include "converged", "no_feasible_fixed", "no_interim_grid", and "no_feasible_design". In frequentist or hybrid calibration modes, additional informative status values may be returned when the best available design is returned although all requested constraints were not fully satisfied.

Examples

```
## Fast Bayesian example with small search space
res <- optimal_twostage_2arm_bf(
  alpha = 0.10,
  beta = 0.20,
  k = 1 / 3,
  k_f = 3,
  n1_min = c(3, 3),
  n2_max = c(12, 12),
  alloc1 = 0.5,
  alloc2 = 0.5,
  power_cushion = 0,
  pceH0 = NULL,
  interim_fraction = c(0.25, 0.75),
  grid_step = 2L,
  coarse_step = 4L,
  progress = FALSE,
  max_iter = 24L,
  calibration_mode = "Bayesian",
  test = "BF01",
  a_0_d = 1, b_0_d = 1,
  a_0_a = 1, b_0_a = 1,
  a_1_d = 1, b_1_d = 1,
  a_2_d = 1, b_2_d = 1,
  a_1_a = 1, b_1_a = 1,
  a_2_a = 1, b_2_a = 1
)
res$design
res$occ
```

```
res2 <- optimal_twostage_2arm_bf(
  alpha = 0.05,
  beta = 0.20,
  k = 1 / 3,
  k_f = 3,
  n1_min = c(5, 5),
  n2_max = c(20, 20),
  alloc1 = 0.5,
  alloc2 = 0.5,
  power_cushion = 0.02,
  pceH0 = 0.50,
  interim_fraction = c(0.25, 0.75),
  grid_step = 1L,
  coarse_step = 4L,
  progress = FALSE,
  max_iter = 40L,
  calibration_mode = "Bayesian",
  test = "BF+0",
  a_0_d = 1, b_0_d = 1,
  a_0_a = 1, b_0_a = 1,
  a_1_d = 1, b_1_d = 2,
  a_2_d = 2, b_2_d = 1,
  a_1_a = 1, b_1_a = 1,
  a_2_a = 1, b_2_a = 1,
  a_1_d_Hminus = 1, b_1_d_Hminus = 1,
  a_2_d_Hminus = 1, b_2_d_Hminus = 1,
  a_1_a_Hminus = 1, b_1_a_Hminus = 1,
  a_2_a_Hminus = 1, b_2_a_Hminus = 1
)
res2$design
res2$occ
```

`optimal_twostage_singlearm_bf`

Optimal two-stage single-arm Bayes factor design

Description

Searches over admissible two-stage single-arm designs with a binary endpoint and returns the feasible design with smallest expected sample size under `H0`.

Usage

```
optimal_twostage_singlearm_bf(
  n1_min,
  n2_max,
```

```
  k,
  k_f,
  p0,
  a0 = 1,
  b0 = 1,
  a1 = 1,
  b1 = 1,
  dp = NA_real_,
  da0 = 1,
  db0 = 1,
  da1 = 1,
  db1 = 1,
  type = c("point", "direction"),
  calibration = c("Bayesian", "frequentist", "hybrid", "full"),
  target_power = 0.8,
  target_type1 = 0.05,
  target_ce_h0 = 0,
  target_freq_power = 0.8,
  target_freq_type1 = 0.05,
  power_cushion = 0
)
```

Arguments

n1_min	Minimum admissible interim sample size.
n2_max	Maximum admissible final sample size.
k	Efficacy threshold on the BF01 scale.
k_f	Futility threshold on the BF01 scale.
p0	Null response probability.
a0, b0	Beta analysis-prior parameters under H0.
a1, b1	Beta analysis-prior parameters under H1.
dp	Optional fixed point alternative used for frequentist power.
da0, db0	Beta design-prior parameters under H0.
da1, db1	Beta design-prior parameters under H1.
type	Character string; one of "point" or "direction".
calibration	Character string; one of "Bayesian", "frequentist", "hybrid", or "full".
target_power	Target corrected Bayesian power.
target_type1	Target corrected Bayesian type-I error.
target_ce_h0	Optional lower bound on corrected Bayesian compelling evidence in favour of H0.
target_freq_power	Target corrected frequentist power at dp.
target_freq_type1	Target corrected frequentist type-I error at p0.
power_cushion	Optional additive cushion for the fixed-sample power target in the first step of the search.

Details

Analysis priors are specified separately under `H0` and `H1` via `a0, b0, a1, b1`. Design priors are specified separately under `H0` and `H1` via `da0, db0, da1, db1`.

Value

A list describing the optimal design and search results.

`plot.singlearm_bf_design` *Plot a single-arm Bayes factor design*

Description

Produces a diagnostic plot for a fitted single-arm two-stage Bayes factor design. Depending on the available information in the object, the plot shows the interim-search results, selected operating characteristics, and the design and analysis priors under H_0 and H_1.

Usage

```
## S3 method for class 'singlearm_bf_design'
plot(x, ...)
```

Arguments

`x`	An object of class "`singlearm_bf_design`".
`...`	Currently unused.

Value

The input object `x`, invisibly.

See Also

`summary.singlearm_bf_design()`, `design_singlearm_bf()`, `optimal_twostage_singlearm_bf()`

plot.singlearm_onestage_bf_design

Plot a one-stage single-arm BF design

Description

Plot a one-stage single-arm BF design

Usage

```
## S3 method for class 'singlearm_onestage_bf_design'
plot(x, what = c("all", "oc"), legend_pos = "right", legend_inset = 0, ...)
```

Arguments

x	An object of class "singlearm_onestage_bf_design".
what	Character string; currently one of "all" or "oc".
legend_pos	Position passed to legend. Either a keyword such as "topright" or a numeric vector c(x, y).
legend_inset	Numeric inset passed to legend() when legend_pos is a keyword.
...	Currently unused.

Value

Invisibly returns x.

plot_twostage_2arm_bf

Plot an optimal two-stage two-arm Bayes factor design

Description

Given the result from optimal_twostage_2arm_bf(), this function produces a six-panel base R plot showing the design schematic, operating characteristics, and the design and analysis priors under H_0 and H_1.

Usage

```
plot_twostage_2arm_bf(
  res,
  main = "Optimal two-stage two-arm Bayes factor design"
)
```

Arguments

- `res` A list returned by `optimal_twostage_2arm_bf()`, containing components `$design`, `$naive_oc`, `$occ` and `$priors`.
- `main` Character string with the main title of the plot.

Value

Invisibly returns `NULL`; called for its side effect of producing a plot.

Examples

```
res <- optimal_twostage_2arm_bf(
  alpha = 0.10, beta = 0.20, k = 1/3, k_f = 3,
  n1_min = c(3, 3), n2_max = c(8, 8),
  alloc1 = 0.5, alloc2 = 0.5,
  power_cushion = 0,
  interim_fraction = c(0.5, 0.5),
  grid_step = 1,
  progress = FALSE,
  max_iter = 16,
  test = "BF01",
  a_0_d = 1, b_0_d = 1,
  a_0_a = 1, b_0_a = 1,
  a_1_d = 1, b_1_d = 1,
  a_2_d = 1, b_2_d = 1,
  a_1_a = 1, b_1_a = 1,
  a_2_a = 1, b_2_a = 1
)
if (is.numeric(res$design) && length(res$design) == 4 && !anyNA(res$design)) {
  plot_twostage_2arm_bf(res)
}
```

`postProbHminus` *Posterior probability P(p2 <= p1 | data)*

Description

Posterior probability P(p2 <= p1 | data)

Usage

```
postProbHminus(y1, y2, n1, n2, a_1_a = 1, b_1_a = 1, a_2_a = 1, b_2_a = 1)
```

Arguments

- `y1` Number of successes in arm 1 (control).
- `y2` Number of successes in arm 2 (treatment).
- `n1` Sample size in arm 1.

n2	Sample size in arm 2.
a_1_a, b_1_a	Shape parameters of the Beta prior for p_1 under the alternative (analysis prior).
a_2_a, b_2_a	Shape parameters of the Beta prior for p_2 under the alternative (analysis prior).

Value

Numeric scalar, posterior probability $P(p_2 <= p_1|y_1, y_2)$.

---

postProbHplus *Posterior probability P(p2 > p1 | data) under independent Beta priors*

Description

Uses Beta posteriors induced by the analysis priors to compute $P(p_2 > p_1 \mid y_1, y_2)$.

Usage

```
postProbHplus(y1, y2, n1, n2, a_1_a = 1, b_1_a = 1, a_2_a = 1, b_2_a = 1)
```

Arguments

y1	Number of successes in arm 1 (control).
y2	Number of successes in arm 2 (treatment).
n1	Sample size in arm 1.
n2	Sample size in arm 2.
a_1_a, b_1_a	Shape parameters of the Beta prior for p_1 under the alternative (analysis prior).
a_2_a, b_2_a	Shape parameters of the Beta prior for p_2 under the alternative (analysis prior).

Value

Numeric scalar, posterior probability $P(p_2 > p_1|y_1, y_2)$.

powerbinbf01_fixed *Bayesian and frequentist operating characteristics for a fixed-sample single-arm BF design*

Description

Computes operating characteristics for a genuine fixed-sample single-arm binomial design with final efficacy decision based on BF01 <= k.

Usage

```
powerbinbf01_fixed(
  n,
  k,
  p0,
  a0 = 1,
  b0 = 1,
  a1 = 1,
  b1 = 1,
  da0 = 1,
  db0 = 1,
  da1 = 1,
  db1 = 1,
  dp = NA_real_,
  type = c("point", "direction"),
  k_ce = NULL,
  grid_size = 801L
)
```

Arguments

n	Integer scalar. Total sample size.
k	Numeric scalar. Efficacy threshold on the BF01 scale.
p0	Numeric scalar in (0, 1). Null response probability.
a0, b0	Numeric scalars. Beta analysis-prior parameters under H0.
a1, b1	Numeric scalars. Beta analysis-prior parameters under H1.
da0, db0	Numeric scalars. Beta design-prior parameters under H0.
da1, db1	Numeric scalars. Beta design-prior parameters under H1.
dp	Optional numeric scalar in (0,1). If supplied, frequentist power under H1 is computed at p = dp.
type	Character string. One of \"point\" or \"direction\".
k_ce	Optional numeric scalar greater than 1. Threshold for compelling evidence in favour of H0 on the BF01 scale.
grid_size	Integer number of grid points used for numerical averaging.

Details

Bayesian operating characteristics are computed under separate design priors:

- for type = "direction", Bayesian power averages over p > p0 under the H1 design prior truncated to (p0, 1], Bayesian type-I error averages over p <= p0 under the H0 design prior truncated to [0, p0], and CE(H0) is averaged over the same truncated H0 design prior;
- for type = "point", Bayesian power averages under the H1 design prior on (0, 1), Bayesian type-I error is evaluated at the point null p = p0, and CE(H0) is also evaluated at p = p0.

Value

A list with Bayesian and frequentist operating characteristics for the fixed-sample design.

---

powerbinbf01seq — *Bayesian and frequentist operating characteristics for a single-arm two-stage BF design*

Description

Computes naive fixed-sample and corrected two-stage operating characteristics for a single-arm binomial design with one interim analysis for futility. The Bayes factor is oriented as BF01, so efficacy corresponds to small values (BF01 <= k) and futility corresponds to large values (BF01 >= kf).

Usage

```
powerbinbf01seq(
  n1,
  n2,
  k,
  kf,
  p0,
  a0 = 1,
  b0 = 1,
  a1 = 1,
  b1 = 1,
  da0 = 1,
  db0 = 1,
  da1 = 1,
  db1 = 1,
  dp = NA_real_,
  type = c("point", "direction"),
  k_ce = NULL,
  grid_size = 801L
)
```

Arguments

n1	Integer scalar. Interim sample size.
n2	Integer scalar. Final sample size, with n1 < n2.
k	Numeric scalar. Efficacy threshold on the BF01 scale.
kf	Numeric scalar. Futility threshold on the BF01 scale.
p0	Numeric scalar in (0, 1). Null response probability.
a0, b0	Numeric scalars. Beta analysis-prior parameters under H0.
a1, b1	Numeric scalars. Beta analysis-prior parameters under H1.
da0, db0	Numeric scalars. Beta design-prior parameters under H0.
da1, db1	Numeric scalars. Beta design-prior parameters under H1.
dp	Optional numeric scalar in (0,1). If supplied, frequentist power under H1 is computed at p = dp.
type	Character string. One of \"point\" or \"direction\".
k_ce	Optional numeric scalar greater than 1. Threshold for compelling evidence in favour of H0 on the BF01 scale.
grid_size	Integer number of grid points used for numerical averaging.

Details

Bayesian operating characteristics are computed under separate design priors:

- for type = "direction", Bayesian power averages over p > p0 under the H1 design prior truncated to (p0, 1], and Bayesian type-I error averages over p <= p0 under the H0 design prior truncated to [0, p0];
- for type = "point", Bayesian power averages under the H1 design prior on (0, 1), and Bayesian type-I error is evaluated at the point null p = p0.

If dp is supplied, additional frequentist power under H1 is computed at the fixed point alternative p = dp. Frequentist type-I error is computed at p = p0.

Value

A list with Bayesian and frequentist operating characteristics.

---

powertwoarmbinbf01 — *Bayesian power, type-I error, and PCE(H0) for two-arm binomial Bayes factors*

Description

Computes Bayesian power, Bayesian type-I error, and the probability of compelling evidence under H_0 (or H_- for BF+-), for a given sample size and Bayes factor test. Optionally, frequentist type-I error and frequentist power are computed by summing over the rejection region.

Usage

```
powertwoarmbinbf01(
  n1,
  n2,
  k = 1/3,
  k_f = 1/3,
  test = c("BF01", "BF+0", "BF-0", "BF+-"),
  a_0_d = 1,
  b_0_d = 1,
  a_0_a = 1,
  b_0_a = 1,
  a_1_d = 1,
  b_1_d = 1,
  a_2_d = 1,
  b_2_d = 1,
  a_1_a = 1,
  b_1_a = 1,
  a_2_a = 1,
  b_2_a = 1,
  output = c("numeric", "predDensmatrix", "t1ematrix", "ceH0matrix", "frequentist_t1e"),
  a_1_d_Hminus = 1,
  b_1_d_Hminus = 1,
  a_2_d_Hminus = 1,
  b_2_d_Hminus = 1,
  compute_freq_t1e = FALSE,
  p1_grid = seq(0.01, 0.99, 0.02),
  p2_grid = seq(0.01, 0.99, 0.02),
  p1_power = NULL,
  p2_power = NULL,
  a_1_a_Hminus = 1,
  b_1_a_Hminus = 1,
  a_2_a_Hminus = 1,
  b_2_a_Hminus = 1
)
```

Arguments

n1, n2 — Sample sizes in arms 1 and 2.

k — Evidence threshold for rejecting the null (inverted BF).

k_f — Evidence threshold for "compelling evidence" in favour of the null.

test — Character string, one of "BF01", "BF+0", "BF-0", "BF+-".

a_0_d, b_0_d, a_0_a, b_0_a — Shape parameters for design and analysis priors under H_0.

a_1_d, b_1_d, a_2_d, b_2_d — Shape parameters for design priors under H_1 or H_+.

a_1_a, b_1_a, a_2_a, b_2_a — Shape parameters for analysis priors under H_1 or H_+.

output One of "numeric", "predDensmatrix", "t1ematrix", "ceH0matrix", "frequentist_t1e".

a_1_d_Hminus, b_1_d_Hminus, a_2_d_Hminus, b_2_d_Hminus
: Optional design priors under H_- for directional tests.

compute_freq_t1e
: Logical; if TRUE, compute frequentist type-I error over a grid.

p1_grid, p2_grid
: Grids of true proportions for frequentist T1E.

p1_power, p2_power
: Optional true proportions for frequentist power.

a_1_a_Hminus, b_1_a_Hminus, a_2_a_Hminus, b_2_a_Hminus
: Shape parameters for analysis priors under H_- (directional tests).

Value

Depending on output, either a named numeric vector with components Power, Type1_Error, CE_H0 (and optionally frequentist metrics) or matrices of predictive densities.

Examples

```
# Basic Bayesian power for BF01 test
powertwoarmbinbf01(n1 = 30, n2 = 30, k = 1/3, test = "BF01")

# Directional test BF+0 with frequentist type-I error
powertwoarmbinbf01(n1 = 40, n2 = 40, k = 1/3, k_f = 3,
                   test = "BF+0", compute_freq_t1e = TRUE)

# Predictive density matrices (advanced)
powertwoarmbinbf01(n1 = 25, n2 = 25, output = "predDensmatrix")
```

---

predictiveDensityH0 *Predictive density under H0: p1 = p2 = p*

Description

Beta-binomial predictive density for data (y1,y2) under H0.

Usage

```
predictiveDensityH0(y1, y2, n1, n2, a_0_d = 1, b_0_d = 1)
```

Arguments

y1	Number of successes in arm 1 (control).
y2	Number of successes in arm 2 (treatment).
n1	Sample size in arm 1.
n2	Sample size in arm 2.
a_0_d, b_0_d	Design-prior parameters for common p under H0.

Value

Numeric scalar, predictive density.

---

predictiveDensityH1 *Predictive density under H1: p1 != p2*

Description

Product of two independent Beta-binomial predictive densities.

Usage

```
predictiveDensityH1(y1, y2, n1, n2, a_1_d = 1, b_1_d = 1, a_2_d = 1, b_2_d = 1)
```

Arguments

y1	Number of successes in arm 1 (control).
y2	Number of successes in arm 2 (treatment).
n1	Sample size in arm 1.
n2	Sample size in arm 2.
a_1_d, b_1_d	Design-prior parameters for p1.
a_2_d, b_2_d	Design-prior parameters for p2.

Value

Numeric scalar, predictive density.

---

predictiveDensityHminus_trunc *Predictive density under H-: p2 <= p1 (truncated prior)*

Description

Predictive density under H-: p2 <= p1 (truncated prior)

Usage

```
predictiveDensityHminus_trunc(
  y1,
  y2,
  n1,
  n2,
  a_1_d = 1,
  b_1_d = 1,
  a_2_d = 1,
  b_2_d = 1
)
```

Arguments

y1	Number of successes in arm 1 (control).
y2	Number of successes in arm 2 (treatment).
n1	Sample size in arm 1.
n2	Sample size in arm 2.
a_1_d, b_1_d	Design-prior parameters for p1.
a_2_d, b_2_d	Design-prior parameters for p2.

Value

Numeric scalar, predictive density under H-.

---

predictiveDensityHplus_trunc

Predictive density under H+: p2 > p1 (truncated prior)

Description

Predictive density under H+: p2 > p1 (truncated prior)

Usage

```
predictiveDensityHplus_trunc(
  y1,
  y2,
  n1,
  n2,
  a_1_d = 1,
  b_1_d = 1,
  a_2_d = 1,
  b_2_d = 1
)
```

Arguments

y1	Number of successes in arm 1 (control).
y2	Number of successes in arm 2 (treatment).
n1	Sample size in arm 1.
n2	Sample size in arm 2.
a_1_d, b_1_d	Design-prior parameters for p1.
a_2_d, b_2_d	Design-prior parameters for p2.

Value

Numeric scalar, predictive density under H+.

print.singlearm_onestage_bf_design

Print method for one-stage single-arm BF designs

Description

Print method for one-stage single-arm BF designs

Usage

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

Arguments

`x`	An object of class "`singlearm_onestage_bf_design`".
`...`	Currently unused.

Value

The input object `x`, invisibly.

print.summary.singlearm_bf_design

Print method for summary.singlearm_bf_design

Description

Print method for summary.singlearm_bf_design

Usage

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

Arguments

`x`	An object of class "summary.singlearm_bf_design".
`...`	Additional arguments passed to or from other methods.

Value

The input object `x`, invisibly.

print.summary.singlearm_onestage_bf_design

Print method for summaries of one-stage single-arm BF designs

Description

Print method for summaries of one-stage single-arm BF designs

Usage

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

Arguments

`x`	An object of class "`summary.singlearm_onestage_bf_design`".
`digits`	Number of digits to print.
`...`	Currently unused.

Value

The input object `x`, invisibly.

priorProbHminus

Prior probability P(p2 <= p1) under independent Beta priors

Description

Prior probability P(p2 <= p1) under independent Beta priors

Usage

```
priorProbHminus(a_1_a, b_1_a, a_2_a, b_2_a)
```

Arguments

`a_1_a`, `b_1_a`	Shape parameters of the Beta prior for p_1.
`a_2_a`, `b_2_a`	Shape parameters of the Beta prior for p_2.

Value

Numeric scalar, prior probability P(p2 <= p1).

priorProbHplus *Prior probability P(p2 > p1) under independent Beta priors*

Description

Prior probability P(p2 > p1) under independent Beta priors

Usage

```
priorProbHplus(a_1_a, b_1_a, a_2_a, b_2_a)
```

Arguments

a_1_a, b_1_a	Shape parameters of the Beta prior for p_1.
a_2_a, b_2_a	Shape parameters of the Beta prior for p_2.

Value

Numeric scalar, prior probability P(p2 > p1).

summary.singlearm_bf_design *Summary for single-arm BF designs*

Description

Summary for single-arm BF designs

Usage

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

Arguments

object	An object of class "singlearm_bf_design".
...	Additional arguments (currently unused).

Value

An object of class "summary.singlearm_bf_design".

`summary.singlearm_onestage_bf_design` *Summarize a one-stage single-arm BF design*

Description

Summarize a one-stage single-arm BF design

Usage

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

Arguments

`object`	An object of class "`singlearm_onestage_bf_design`".
`...`	Currently unused.

Value

An object of class "`summary.singlearm_onestage_bf_design`".

`twoarmbinbf_plus0_direct` *Bayes factor BF+0 for the directional alternative vs point-null*

Description

Computes the Bayes factor BF_{+0} comparing the directional alternative hypothesis H_+ (p_2 > p_1) against the point-null H_0 (p_1 = p_2).

Usage

```
twoarmbinbf_plus0_direct(
  y1,
  y2,
  n1,
  n2,
  a_0_a = 1,
  b_0_a = 1,
  a_1_a = 1,
  b_1_a = 1,
  a_2_a = 1,
  b_2_a = 1
)
```

Arguments

y1, y2	Integer counts of successes in arms 1 and 2.
n1, n2	Integer sample sizes in arms 1 and 2.
a_0_a, b_0_a	Shape parameters of the **analysis** prior for the common response probability under H_0.
a_1_a, b_1_a	Shape parameters of the **analysis** prior for the response probability in arm 1 under H_+.
a_2_a, b_2_a	Shape parameters of the **analysis** prior for the response probability in arm 2 under H_+.

Details

Both marginal likelihoods are formed using the **analysis** priors, which represent inferential beliefs at the time the data are evaluated. The design priors are used only for computing Bayesian operating characteristics (predictive power / type-I error) and play no role here.

Value

Numeric scalar; the Bayes factor $BF_{+0} = m_+(y_1, y_2)/m_0(y_1, y_2)$.

---

twoarmbinbf01 — *Two-arm binomial Bayes factor BF01*

Description

Computes the Bayes factor BF_{01} comparing the point-null $H_0 : p_1 = p_2$ to the alternative $H_1 : p_1 \neq p_2$ in a two-arm binomial setting with Beta priors.

Usage

```
twoarmbinbf01(
  y1,
  y2,
  n1,
  n2,
  a_0_a = 1,
  b_0_a = 1,
  a_1_a = 1,
  b_1_a = 1,
  a_2_a = 1,
  b_2_a = 1
)
```

Arguments

`y1`	Number of successes in arm 1 (control).
`y2`	Number of successes in arm 2 (treatment).
`n1`	Sample size in arm 1.
`n2`	Sample size in arm 2.
`a_0_a`, `b_0_a`	Shape parameters of the Beta prior for the common-p under the null model (analysis prior).
`a_1_a`, `b_1_a`	Shape parameters of the Beta prior for p_1 under the alternative (analysis prior).
`a_2_a`, `b_2_a`	Shape parameters of the Beta prior for p_2 under the alternative (analysis prior).

Value

Numeric scalar, the Bayes factor BF_{01}.

Examples

```
twoarmbinbf01(10, 20, 30, 30)
```

Index