# Package 'ppseq'

September 9, 2021

**Title** Design Clinical Trials using Sequential Predictive Probability Monitoring

**Version** 0.1.1

**Description** Functions are available to calibrate designs over a range of posterior and predictive thresholds, to plot the various design options, and to obtain the operating characteristics of optimal accuracy and optimal efficiency designs.

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.1

**Depends** R (>= 3.5.0)

**Suggests** covr, gt, knitr, rmarkdown, spelling, testthat (>= 3.0.0), vdiffr

**VignetteBuilder** knitr

**Imports** dplyr, furrr, future, ggplot2, plotly, purrr, tibble, patchwork, tidyr

**Config/testthat/edition** 3

**Language** en-US

**NeedsCompilation** no

**Author** Emily C. Zabor [aut, cre] (<https://orcid.org/0000-0002-1402-4498>), Brian P. Hobbs [aut], Michael J. Kane [aut] (<https://orcid.org/0000-0003-1899-6662>)

**Maintainer** Emily C. Zabor <zabore2@ccf.org>

**Repository** CRAN

**Date/Publication** 2021-09-09 09:00:02 UTC

## R topics documented:

---

`calc_decision_rules` *Calculate a decision rule table for interim monitoring of a pre-specified design*

---

## Description

This function will take the posterior and predictive thresholds for a pre-specified design with a given null response rate and fixed interim looks and total sample size, and return the decision rules at each interim analysis and the end of the trial. Intended for use after selecting an optimal design using the functions `calibrate_thresholds` and `optimize_design`.

## Usage

```
calc_decision_rules(
  n,
  N,
  theta,
  ppp,
  p0,
  direction = "greater",
  delta = NULL,
  prior = c(0.5, 0.5),
  S = 5000
)
```

## Arguments

| | |
|---|---|
| `n` | matrix containing the total number of patients accrued so far at each interim look in the standard of care (column 1) and experimental (column 2) arms for two-sample case; vector of sample size accrued so far at each interim look for one-sample case. The last value should be equal to the total sample size at the end of the trial. If only a single look will be done at the end of the trial, this can |

| | |
|---|---|
| | be a vector specifying the total sample size c(N0, N1) for the two-sample case or an integer specifying the total sample size N for the one-sample case |
| N | the total planned sample size at the end of the trial, c(N0, N1) for two-sample case; integer of total planned sample size at end of trial N for one-sample case |
| theta | The target posterior probability. e.g. Efficacy decision if P(p1 > p0) > theta for the two-sample case with greater direction. |
| ppp | The target predictive probability. e.g. Stop the trial if the predictive probability falls below this target. |
| p0 | The target value to compare to in the one-sample case. Set to NULL for the two-sample case. |
| direction | "greater" (default) if interest is in P(p1 > p0) and "less" if interest is in P(p1 < p0) for two-sample case. For one-sample case, "greater" if interest is in P(p > p0) and "less" if interest is in P(p < p0). |
| delta | clinically meaningful difference between groups. Typically 0 for two-sample case. NULL for one-sample case (default). |
| prior | hyperparameters of prior beta distribution. Beta(0.5, 0.5) is default |
| S | number of samples, default is 5000 |

## Value

In the one-sample case, returns a tibble with n at each look, r at each look, and ppp, the associated posterior predictive probability. Stop the trial at that look if the number of observed responses is <=r. At the end of the trial, the treatment is considered promising if the number of observed responses is >r. In the two-sample case, returns a tibble with n0 and n1, the number enrolled subjects in the control and experimental arms at each look, respectively, r0 and r1, the number of possible responses in the control and experimental arms at each look, respectively, and ppp, the associated posterior predictive probability. Any NA value in either table represents an interim look where there is no number of responses that would lead to stopping the trial.

## Examples

```
set.seed(123)

# One-sample case
calc_decision_rules(n = seq(5, 25, 5), N = 25, theta = 0.86, ppp = 0.2, p0 = 0.1)

# Two-sample case
calc_decision_rules(n = cbind(seq(5, 25, 5), seq(5, 25, 5)), N = c(25, 25),
theta = 0.86, ppp = 0.2, p0 = NULL, direction = "greater", delta = 0,
prior = c(0.5, 0.5), S = 5000)
```

`calc_posterior` *Calculate a single posterior probability*

## Description

This function is meant to be used in the context of a clinical trial with a binary endpoint. For the two-sample case, the total number of events in the standard-of-care arm is y0 and the total number of events in the experimental arm is y1. The function samples from the posterior beta distribution based on the data and the prior beta hyperparameters, and returns the posterior probability that p1 is greater than (or less than) p0 given the data. The one-sample case is also available, in which a target p0 must be specified and the function returns the posterior probability that p is greater than (or less than) p0 given the data.

## Usage

```
calc_posterior(
  y,
  n,
  p0,
  direction = "greater",
  delta = NULL,
  prior = c(0.5, 0.5),
  S = 5000
)
```

## Arguments

| | |
|---|---|
| `y` | vector of length two containing total responses c(y0, y1) for two-sample case; integer of total responses y for one-sample case |
| `n` | vector of length two containing the sample size so far c(n0, n1) for two-sample case; integer of sample size so far n for one-sample case |
| `p0` | The target value to compare to in the one-sample case. Set to NULL in two-sample case. |
| `direction` | "greater" (default) if interest is in p(p1 > p0) and "less" if interest is in p(p1 < p0) for two-sample case. For one-sample case, "greater" if interest is in p(p > p0) and "less" if interest is in p(p < p0). |
| `delta` | clinically meaningful difference between groups. Typically 0 for two-sample case. NULL for one-sample case (default). |
| `prior` | hyperparameters of prior beta distribution. Beta(0.5, 0.5) is default |
| `S` | number of samples, default is 5000 |

## Value

Returns the posterior probability of interest

## Examples

```
set.seed(123)

# Two-sample case
calc_posterior(y = c(14, 23), n = c(100, 100), p0 = NULL, delta = 0)

# One-sample case
calc_posterior(y = 27, n = 100, p0 = 0.2)
```

---

`calc_predictive` *Calculate a single posterior predictive value*

---

## Description

This function is meant to be used in the context of a clinical trial with a binary endpoint. The goal is to calculate the posterior predictive probability of success at the end of a trial, given the data available at an interim analysis. For the two-arm case the number of events observed at interim analysis, the sample size at interim analysis, and the total planned sample size are denoted y0, n0, and N0 in the standard-of-care arm and y1, n1, and N1 in the experimental arm.

## Usage

```
calc_predictive(
  y,
  n,
  p0,
  N,
  direction = "greater",
  delta = NULL,
  prior = c(0.5, 0.5),
  S = 5000,
  theta = 0.95
)
```

## Arguments

| | |
|---|---|
| `y` | vector of length two containing number of events observed so far c(y0, y1) for two-sample case; integer of number of events y observed so far for one-sample case |
| `n` | vector of length two containing the sample size so far c(n0, n1) for two-sample case; integer of sample size so far for one-sample case |
| `p0` | The target value to compare to in the one-sample case. Set to NULL for the two-sample case. |
| `N` | the total planned sample size at the end of the trial, c(N0, N1) for two-sample case; integer of total planned sample size at end of trial N for one-sample case |

| | |
|---|---|
| direction | "greater" (default) if interest is in P(p1 > p0) and "less" if interest is in P(p1 < p0) for two-sample case. For one-sample case, "greater" if interest is in P(p > p0) and "less" if interest is in P(p < p0). |
| delta | clinically meaningful difference between groups. Typically 0 for the two-sample case. NULL for one-sample case (default). |
| prior | hyperparameters of prior beta distribution. Beta(0.5, 0.5) is default |
| S | number of samples, default is 5000 |
| theta | The target posterior probability. e.g. Efficacy decision if P(p1 > p0) > theta for the two-sample case with greater direction. Default is 0.95. |

## Value

Returns the posterior predictive probability of interest

## Examples

```
set.seed(123)

# One-sample case
calc_predictive(y = 14, n = 50, p0 = 0.2, N = 100)

# Two-sample case (not run)
calc_predictive(
  y = c(7, 12),
  n = c(50, 50),
  p0 = NULL,
  N = c(100, 100),
  delta = 0)
```

---

# calibrate_posterior_threshold

*Calibrate the posterior probability threshold*

---

## Description

This function is meant to be used in the context of a clinical trial with a binary endpoint. For a vector of possible posterior decision thresholds, the function simulates many trials and then calculates the average number of times the posterior probability exceeds a given threshold. In a null case, this will result in the type I error at a given threshold. In an alternative case, this will result in the power at a given threshold.

## Usage

```
calibrate_posterior_threshold(
  p,
  N,
  p0,
  direction = "greater",
  delta = NULL,
  prior = c(0.5, 0.5),
  S = 5000,
  theta
)
```

## Arguments

| | |
|---|---|
| p | vector of length two containing the probability of event in the standard of care and experimental arm c(p0, p1) for the two-sample case; integer of event probability for one-sample case |
| N | vector of length two containing the total sample size c(N0, N1) for two-sample case; integer of sample size so far N for one-sample case |
| p0 | The target value to compare to in the one-sample case. Set to NULL for the two-sample case. |
| direction | "greater" (default) if interest is in p(p1 > p0) and "less" if interest is in p(p1 < p0) for two-sample case. For one-sample case, "greater" if interest is in p(p > p0) and "less" if interest is in p(p < p0). |
| delta | clinically meaningful difference between groups. Typically 0 for the two-sample case. NULL for the one-sample case (default). |
| prior | hyperparameters of prior beta distribution. Beta(0.5, 0.5) is default |
| S | number of samples drawn from the posterior, and number of simulated trials. Default is 5000 |
| theta | The target posterior probability thresholds to consider. Integer or vector. |

## Value

Returns a tibble with the posterior probability threshold(s) and associated proportion of positive trials.

## Examples

```
set.seed(123)

# One-sample case
calibrate_posterior_threshold(
  p = 0.1,
  N = 50,
  direction = "greater",
  p0 = 0.1,
```

```
  delta = NULL,
  prior = c(0.5, 0.5),
  S = 5000,
  theta = c(0.9, 0.95))

# Two-sample case
calibrate_posterior_threshold(
  p = c(0.1, 0.1),
  N = c(50, 50),
  direction = "greater",
  p0 = NULL,
  delta = 0,
  prior = c(0.5, 0.5),
  S = 5000,
  theta = c(0.9, 0.95))
```

---

`calibrate_thresholds` *Calibrate according to posterior probability threshold and predictive probability threshold with interim futility monitoring*

---

## Description

This function is meant to be used in the context of a clinical trial with a binary endpoint. For a vector of possible posterior decision thresholds, the function simulates many trials and then calculates the average number of times the posterior probability exceeds a given threshold. In a null case, this will result in the type I error at a given threshold. In an alternative case, this will result in the power at a given threshold.

## Usage

```
calibrate_thresholds(
  p_null,
  p_alt,
  n,
  N,
  pp_threshold,
  ppp_threshold,
  direction = "greater",
  delta = NULL,
  prior = c(0.5, 0.5),
  S = 5000,
  nsim = 1000
)
```

## Arguments

| | |
|---|---|
| `p_null` | vector of length two containing the probability of event in the standard of care and experimental arm c(p0, p1) for the two-sample case for the null scenario; integer of event probability for one-sample case |

| | |
|---|---|
| p_alt | vector of length two containing the probability of event in the standard of care and experimental arm c(p0, p1) for the two-sample case for the alternative scenario; integer of event probability for one-sample case |
| n | matrix containing the total number of patients accrued so far at each interim look in the standard of care (column 1) and experimental (column 2) arms for two-sample case; vector of sample size accrued so far at each interim look for one-sample case. The last value should be equal to the total sample size at the end of the trial. If only a single look will be done at the end of the trial, this can be a vector specifying the total sample size c(N0, N1) for the two-sample case or an integer specifying the total sample size N for the one-sample case |
| N | the total planned sample size at the end of the trial, c(N0, N1) for two-sample case; integer of total planned sample size at end of trial N for one-sample case |
| pp_threshold | the posterior probability threshold of interest |
| ppp_threshold | the posterior probability threshold of interest for futility monitoring |
| direction | "greater" (default) if interest is in p(p1 > p0) and "less" if interest is in p(p1 < p0) for two-sample case. For one-sample case, "greater" if interest is in p(p > p0) and "less" if interest is in p(p < p0). |
| delta | clinically meaningful difference between groups. Typically 0 for the two-sample case. NULL for the one-sample case (default). |
| prior | hyperparameters of prior beta distribution. Beta(0.5, 0.5) is default |
| S | number of samples drawn from the posterior. Default is 5000 |
| nsim | Number of simulated trial datasets. |

## Value

A list containing a

1. a tibble 'res_summary' containing the posterior probability threshold (pp_threshold), the predictive probability threshold (ppp_threshold), the mean sample size under the null (mean_n0_null and mean_n1_null for two-sample case; mean_n1_null for one-sample case), the proportion of positive trials under the null (prop_pos_null), the proportion of trials stopped early under the null (prop_stopped_null), the mean sample size under the alternative (mean_n0_alt and mean_n1_alt for two-sample case; mean_n1_alt for one-sample case), the proportion of positive trials under the alternative (prop_pos_alt), the proportion of trials stopped early under the alternative (prop_stopped_alt)
2. 'call_list' containing the original function call
3. 'calibrate_thresholds_inputs' a list containing the inputs to the original function call

The proportion of positive trials will be a measure of the type I error for a null setting, and a measure of the power in the alternative setting.

## Examples

```
# One-sample case

set.seed(123)
```

```
calibrate_thresholds(
  p_null = 0.1, p_alt = 0.3,
  n = seq(5, 25, 5), N = 25,
  pp_threshold = c(0.9, 0.95, 0.96, 0.98),
  ppp_threshold = seq(0.05, 0.2, 0.05),
  direction = "greater", delta = NULL,
  prior = c(0.5, 0.5), S = 5000, nsim = 1000
)


# Two-sample case

set.seed(123)

calibrate_thresholds(
  p_null = c(0.1, 0.1), p_alt = c(0.1, 0.3),
  n = cbind(seq(5, 25, 5), seq(5, 25, 5)), N = c(25, 25),
  pp_threshold = c(0.9, 0.95, 0.96, 0.98),
  ppp_threshold = seq(0.05, 0.2, 0.05),
  direction = "greater", delta = 0,
  prior = c(0.5, 0.5), S = 5000, nsim = 1000
)
```

---

`eval_thresh` *Evaluate a single dataset for a single pp_threshold and ppp_threshold combination*

---

## Description

Helper function for calibrate_thresholds() function that evaluates a single combination of a pp_threshold and a ppp_threshold for a single dataset

## Usage

```
eval_thresh(
  data,
  pp_threshold,
  ppp_threshold,
  p0,
  N,
  direction = "greater",
  delta = NULL,
  prior = c(0.5, 0.5),
  S = 5000
)
```

### Arguments

| | |
|---|---|
| data | the name of the dataset |
| pp_threshold | the posterior probability threshold of interest |
| ppp_threshold | the posterior probability threshold of interest for futility monitoring |
| p0 | The target value to compare to in the one-sample case. Set to NULL for the two-sample case. |
| N | the total planned sample size at the end of the trial, c(N0, N1) for two-sample case; integer of total planned sample size at end of trial N for one-sample case |
| direction | "greater" (default) if interest is in P(p1 > p0) and "less" if interest is in P(p1 < p0) for two-sample case. For one-sample case, "greater" if interest is in P(p > p0) and "less" if interest is in P(p < p0). |
| delta | clinically meaningful difference between groups. Typically 0 for the two-sample case. NULL for the one-sample case (default). |
| prior | hyperparameters of prior beta distribution. Beta(0.5, 0.5) is default |
| S | number of samples, default is 5000 |

### Value

Returns a tibble with the total sample size at the end of the trial, the number of responses observed at the end of the trial, the pp_threshold considered, the ppp_threshold considered, the observed predictive probability generated from calc_predictive(), and an indicator for whether the trial was positive or not at the end

---

## one_sample_cal_tbl *Output from a one-sample call to* calibrate_thresholds

---

### Description

This .rda file contains output from a one-sample call to `calibrate_thresholds()`. See the vignette titled "One-sample expansion cohort" for a description of the input parameters used, or run `one_sample_cal_tbl$calibrate_thresholds_inputs` to see a list of the original function inputs. For use in testing functions and in vignettes.

### Usage

```
data(one_sample_cal_tbl)
```

### Format

A list containing a

1. a tibble 'res_summary' containing the posterior probability threshold (pp_threshold), the predictive probability threshold (ppp_threshold), the mean sample size (mean_n0 and mean_n1 for two-sample case; mean_n1 for one-sample case), and the proportion of positive trials under the null and alternative response rates.

2. 'call_list' containing the original function call
3. 'calibrate_thresholds_inputs' a list containing the inputs to the original function call
4. 'protocol_res' gives results from the protocol-specified design of the atezolizumab case study, as detailed in the vignette titled "One-sample expansion cohort"

---

`one_sample_decision_tbl` *Output from a one-sample call to* `calc_decision_rules`

---

## Description

This .rda file contains output from a one-sample call to `calc_decision_rules()`. See the vignette titled "One-sample expansion cohort" for a description of the input parameters used.

## Usage

```
data(one_sample_decision_tbl)
```

## Format

A tibble containing n, the number of patients enrolled at each futility monitoring point, and r, the number of responses at which we would stop the trial at a given look if the number of observed responses is <=r, or at the end of the trial the treatment is considered promising if the number of observed responses is >r.

---

`optimize_design` *Function to setup usage of* `optimize_design.calibrate_thresholds`

---

## Description

Function to setup usage of `optimize_design.calibrate_thresholds`

## Usage

```
optimize_design(x, type1_range = c(0, 1), minimum_power = 0, ...)
```

## Arguments

| | |
|---|---|
| `x` | an object of class 'calibrate_thresholds', usually returned by the `calibrate_thresholds` function |
| `type1_range` | a vector specifying the minimum and maximum acceptable type I error. Specify NULL to return the full range of resulting type I error. Defaults to c(0, 1) to return all results. |
| `minimum_power` | a numeric between 0 and 1 specifying the minimum acceptable power. Specify NULL to return the full range of resulting power. Defaults to 0 to return all results. |
| `...` | ignored |

### Value

No return value, called for side effects

---

`optimize_design.calibrate_thresholds`
*Custom optimization method for* `calibrate_thresholds` *objects*

---

### Description

Determines the optimal designs based on a variety of criteria. The optimal efficiency design is the one with the shortest Euclidean distance to the upper left point on a plot of the average sample size under the null by the average sample size under the alternative. The optimal accuracy design is the one with the shortest Euclidean distance to the upper left point on a plot of the type I error by the power.

### Usage

```
## S3 method for class 'calibrate_thresholds'
optimize_design(x, type1_range = c(0.05, 0.1), minimum_power = 0.8, ...)
```

### Arguments

| | |
|---|---|
| `x` | an object of class 'calibrate_thresholds', usually returned by the `calibrate_thresholds` function |
| `type1_range` | a vector specifying the minimum and maximum acceptable type I error. Specify NULL to return the full range of resulting type I error. Defaults to c(0.05, 0.1) |
| `minimum_power` | a numeric between 0 and 1 specifying the minimum acceptable power. Specify NULL to return the full range of resulting power. Defaults to 0.8. |
| `...` | ignored |

### Value

A list of length two containing details of the optimal efficiency and optimal accuracy designs

### Examples

```
# One-sample case
set.seed(123)

cal_tbl <- calibrate_thresholds(
  p_null = 0.1, p_alt = 0.3,
  n = seq(5, 25, 5), N = 25,
  pp_threshold = c(0.9, 0.95, 0.96, 0.98),
  ppp_threshold = seq(0.05, 0.2, 0.05),
  direction = "greater", delta = NULL,
```

```
  prior = c(0.5, 0.5), S = 5000, nsim = 1000
)

optimize_design(cal_tbl)


# Two-sample case
set.seed(123)

cal_tbl2 <-
calibrate_thresholds(
  p_null = c(0.1, 0.1), p_alt = c(0.1, 0.3),
  n = cbind(seq(5, 25, 5), seq(5, 25, 5)), N = c(25, 25),
  pp_threshold = c(0.9, 0.95, 0.96, 0.98),
  ppp_threshold = seq(0.05, 0.2, 0.05),
  direction = "greater", delta = 0,
  prior = c(0.5, 0.5), S = 5000, nsim = 1000
)

optimize_design(cal_tbl2)
```

---

`plot.calc_decision_rules` *Plot method for* `calc_decision_rules` *objects*

---

## Description

Returns a plot of decision rules from the results of `calc_decision_rules` that can interactively show when to stop and when to proceed at the various interim analyses

## Usage

```
## S3 method for class 'calc_decision_rules'
plot(x, plotly = TRUE, ...)
```

## Arguments

| | |
|---|---|
| `x` | an object of class 'calc_decision_rules', usually returned by the `calc_decision_rules` function |
| `plotly` | should the plot be rendered in plotly? (Default is TRUE) |
| `...` | unused |

## Value

In the one-sample case, a heatmap plot with number enrolled on the x-axis and number of responses on the y-axis. In the two-sample case, a grid of heatmap plots. Each plot is a combination of the number enrolled so far in the experimental and control arms. The x-axis is the number of responses in the control arm and the y-axis is the number of responses in the experimental arm. Green indicates combinations where the trial would proceed and red indicates combinations where the trial would stop.

## Examples

```
set.seed(123)

# Two-sample case
dec_tbl <- calc_decision_rules(
n = cbind(seq(5, 25, 5), seq(5, 25, 5)),
N = c(25, 25),
theta = 0.86,
ppp = 0.2,
p0 = NULL,
direction = "greater",
delta = 0,
prior = c(0.5, 0.5),
S = 5000)

plot(dec_tbl)
```

---

# plot.calibrate_thresholds

*Plot method for* `calibrate_thresholds` *objects*

---

## Description

Returns two interactive `plotly` plots (if plotly=TRUE) or two static `ggplot2` plots (if plotly=FALSE) to compare results from various designs generated from a call to `calibrate_thresholds` based on various criteria, and to assist in selecting an optimal design.

## Usage

```
## S3 method for class 'calibrate_thresholds'
plot(x, type1_range = c(0, 1), minimum_power = 0, plotly = FALSE, ...)
```

## Arguments

| | |
|---|---|
| `x` | an object of class 'calibrate_thresholds', usually returned by the `calibrate_thresholds` function |

| | |
|---|---|
| type1_range | a vector specifying the minimum and maximum acceptable type I error. Specify c(0, 1) to return the full range of resulting type I error. Defaults to c(0, 1) |
| minimum_power | a numeric between 0 and 1 specifying the minimum acceptable power. Specify 0 to return the full range of resulting power. Defaults to 0. |
| plotly | a logical indicator of whether you want the plots returned as interactive plotly plots or non-interactive ggplots. Defaults to FALSE. |
| ... | unused |

## Value

Plots of the average sample size under the null by the average sample size under the alternative, and the type I error by the power for designs meeting the specified type1_range and minimum_power

## Examples

```
set.seed(123)

cal_tbl <- calibrate_thresholds(
  p_null = 0.1, p_alt = 0.3,
  n = seq(5, 25, 5), N = 25,
  pp_threshold = c(0.9, 0.95, 0.96, 0.98),
  ppp_threshold = seq(0.05, 0.2, 0.05),
  direction = "greater", delta = NULL,
  prior = c(0.5, 0.5), S = 5000, nsim = 1000
)

plot(cal_tbl, type1_range = c(0.01, 0.2), minimum_power = 0.7)
```

---

# print.calibrate_thresholds

*Print method for* calibrate_thresholds *objects*

---

## Description

By default prints only the res_summary table from an object of class 'calibrate_thresholds'. The table can be limited to a range of type 1 error and a minimum value of power using the arguments 'type1_range' and 'minimum_power' respectively.

## Usage

```
## S3 method for class 'calibrate_thresholds'
print(x, type1_range = c(0, 1), minimum_power = 0, ...)
```

### Arguments

| | |
|---|---|
| x | an object of class 'calibrate_thresholds', usually returned by the calibrate_thresholds function |
| type1_range | a vector specifying the minimum and maximum acceptable type I error. Specify c(0, 1) to return the full range of resulting type I error. Defaults to c(0, 1) |
| minimum_power | a numeric between 0 and 1 specifying the minimum acceptable power. Specify 0 to return the full range of resulting power. Defaults to 0. |
| ... | ignored |

### Value

Returns a tibble

### Examples

```
set.seed(123)

cal_tbl <- calibrate_thresholds(
  p_null = 0.1, p_alt = 0.3,
  n = seq(5, 25, 5), N = 25,
  pp_threshold = c(0.9, 0.95, 0.96, 0.98),
  ppp_threshold = seq(0.05, 0.2, 0.05),
  direction = "greater", delta = NULL,
  prior = c(0.5, 0.5), S = 5000, nsim = 1000
)

print(cal_tbl)
print(cal_tbl, type1_range = NULL, minimum_power = NULL)
```

---

## sim_dat1 *Simulate a single dataset based on the response probability(ies), the total sample size(s), and the interim look schedule(s)*

---

### Description

Helper function for calibrate_thresholds() function that generates a single dataset of n and response count at each look based on the response probability(ies)

### Usage

```
sim_dat1(p, n)
```

### Arguments

| | |
|---|---|
| p | vector of length two containing the probability of event in the standard of care and experimental arm c(p0, p1) for the two-sample case; integer of event probability for one-sample case |
| n | matrix containing the total number of patients accrued so far at each interim look in the standard of care (column 1) and experimental (column 2) arms for two-sample case; vector of sample size accrued so far at each interim look for one-sample case. The last value should be equal to the total sample size at the end of the trial. If only a single look will be done at the end of the trial, this can be a vector specifying the total sample size c(N0, N1) for the two-sample case or an integer specifying the total sample size N for the one-sample case |

### Value

Returns a tibble with n0, n1, y0, y1 for the two-sample case and a tibble with n1 and y1 for the one-sample case

---

## sim_single_trial *Simulate a single trial with posterior probability monitoring*

---

### Description

This function is meant to be used in the context of a clinical trial with a binary endpoint. The goal is to simulate event counts from the binomial distribution based on the number of patients accrued at each interim look, and calculate the posterior predictive probability of success (or futility) at the end of a trial, given the data available at each interim analysis.

### Usage

```
sim_single_trial(
  p,
  n,
  p0,
  N,
  direction = "greater",
  delta = NULL,
  prior = c(0.5, 0.5),
  S = 5000,
  theta = 0.95
)
```

### Arguments

| | |
|---|---|
| p | vector of length two containing the probability of event in the standard of care and experimental arm c(p0, p1) for the two-sample case; integer of event probability for one-sample case |

| | |
|---|---|
| n | matrix containing the total number of patients accrued so far at each interim look in the standard of care (column 1) and experimental (column 2) arms for two-sample case; vector of sample size accrued so far at each interim look for one-sample case. The last value should be equal to the total sample size at the end of the trial. If only a single look will be done at the end of the trial, this can be a vector specifying the total sample size c(N0, N1) for the two-sample case or an integer specifying the total sample size N for the one-sample case. |
| p0 | The target value to compare to in the one-sample case |
| N | the total planned sample size at the end of the trial, c(N0, N1) for two-sample case; integer of total planned sample size at end of trial N for one-sample case |
| direction | "greater" (default) if interest is in P(p1 > p0) and "less" if interest is in P(p1 < p0) for two-sample case. For one-sample case, "greater" if interest is in P(p > p0) and "less" if interest is in P(p < p0). |
| delta | clinically meaningful difference between groups. Typically 0 for the two-sample case. NULL for the one-sample case (default). |
| prior | hyperparameters of prior beta distribution. Beta(0.5, 0.5) is default |
| S | number of samples, default is 5000 |
| theta | The target posterior probability. e.g. Efficacy decision if P(p1 > p0) > theta for the two-sample case with greater direction. Default is 0.95. Can be a vector if interest is in selecting from among a variety of thresholds. |

## Value

Returns a tibble with pp_threshold (i.e. theta, the target posterior probability), number of responses, sample size, posterior probability, and posterior predictive probability at each look

## Examples

```
# One-sample case
set.seed(123)
sim_single_trial(
  p = 0.3, n = c(5, 10), direction = "greater",
  p0 = 0.1, delta = NULL, prior = c(0.5, 0.5), S = 50, N = 25, theta = 0.95
)

# # Two-sample case (not run)
# set.seed(123)
#  sim_single_trial(
#    p = c(0.1, 0.3), n = cbind(c(5, 10), c(5, 10)),
#    direction = "greater", p0 = NULL, delta = 0, prior = c(0.5, 0.5),
#    S = 5000, N = c(50, 50), theta = 0.95
#  )
```

# Index