Package: BayesianQDM (via r-universe)

October 31, 2026

Type Package

Title Bayesian Quantitative Decision-Making Framework for Binary, Continuous and Time-to-Event Endpoints

Version 1.0.1

Description Bayesian quantitative decision-making for proof-of-concept clinical trials with Go, NoGo and Gray decisions. A study is specified by its sample sizes, conjugate prior distributions, external data borrowed through power priors, a concurrent, external, hypothetical or fixed control group, and a decision criterion based on the posterior or posterior predictive probability of the treatment effect. The package computes the decision for observed data, the operating characteristics of a design under scenarios for the true parameters, the decision thresholds that control the probabilities of false Go and false NoGo decisions, and the observed data that lead to each decision. One or two binary endpoints (Beta-binomial and Dirichlet-multinomial models), one or two continuous endpoints (Normal-Inverse-chi-squared and Normal-Inverse-Wishart models) and one time-to-event endpoint evaluated by the milestone survival probability or the hazard ratio (exponential model with Gamma priors) are supported, with larger or smaller effects as favorable. The posterior distributions have closed forms, so Markov chain Monte Carlo (MCMC) sampling is not needed; the probabilities are computed from closed-form distribution functions, by exact summation, numerical integration, a moment-matching approximation or Monte Carlo simulation, with the main computations in C++. See Kang, Yamaguchi, and Han (2026) <doi:10.1080/10543406.2026.2655410> for the methodological framework.

License GPL (>= 2)

Encoding UTF-8

Language en-US

Imports ggplot2 (>= 3.4.0), grDevices, Rcpp, stats

LinkingTo Rcpp

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

VignetteBuilder knitr

Config/testthat/edition 3

URL https://gosukehommaEX.github.io/BayesianQDM/, https://github.com/gosukehommaEX/BayesianQDM

BugReports https://github.com/gosukehommaEX/BayesianQDM/issues

Config/roxygen2/version 8.0.0

NeedsCompilation yes

Author Gosuke Homma [aut, cre], Yusuke Yamaguchi [aut]

Maintainer Gosuke Homma <my.name.is.gosuke@gmail.com>

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

Date/Publication 2026-10-31 12:48:23 UTC

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

RemoteRef HEAD

RemoteSha 5426452883d0685a805f700a2f2fa2b152ca2d9d

Contents

borrow . . . 3
cellprob_binary2 . . . 4
conduct_tte . . . 5
criterion_posterior . . . 6
criterion_predictive . . . 9
data_binary . . . 11
data_binary2 . . . 11
data_continuous . . . 12
data_continuous2 . . . 13
data_tte . . . 13
design_binary . . . 14
design_binary2 . . . 16
design_continuous . . . 19
design_continuous2 . . . 22
design_tte . . . 24
format.qdm_borrow . . . 28
format.qdm_boundary . . . 28
format.qdm_calibration . . . 29
format.qdm_conduct . . . 30
format.qdm_criterion . . . 31
format.qdm_data . . . 31
format.qdm_decision . . . 32
format.qdm_design . . . 33
format.qdm_hypothetical . . . 33
format.qdm_prior . . . 34

hypothetical . . . 35
pbetabinomdiff . . . 36
pbetadiff . . . 37
phazardratio . . . 38
plot.qdm_boundary . . . 39
plot.qdm_calibration . . . 40
plot.qdm_oc . . . 42
print.qdm_borrow . . . 44
print.qdm_boundary . . . 45
print.qdm_calibration . . . 46
print.qdm_conduct . . . 46
print.qdm_criterion . . . 47
print.qdm_data . . . 48
print.qdm_decision . . . 48
print.qdm_design . . . 49
print.qdm_hypothetical . . . 50
print.qdm_oc . . . 50
print.qdm_prior . . . 51
prior_beta . . . 52
prior_dirichlet . . . 52
prior_gamma . . . 53
prior_niw . . . 54
prior_nix . . . 55
prior_vague . . . 55
psurvdiff . . . 56
ptdiff . . . 57
qdm_boundary . . . 59
qdm_calibrate . . . 62
qdm_decide . . . 64
qdm_oc . . . 66
qdm_regions . . . 68
rdirichlet . . . 69

Index **71**

`borrow` *External data borrowed through a power prior*

Description

Specifies external (historical) data for one group together with the power prior weight a_0. The likelihood of the external data is raised to the power a_0, so that $a_0 = 0$ ignores the external data and $a_0 = 1$ pools it with the current data. For the conjugate models used in this package the resulting posterior distribution has a closed form, so no Markov chain Monte Carlo sampling is needed.

Usage

```
borrow(data, a0)
```

Arguments

`data`	External data created by `data_binary`, `data_continuous`, `data_binary2`, `data_continuous2` or `data_tte`.
`a0`	Power prior weight, a number from 0 to 1.

Value

An object of class `"qdm_borrow"` with elements `data` and `a0`.

Examples

```
borrow(data_binary(n = 20, y = 8), a0 = 0.5)
```

`cellprob_binary2` *Joint cell probabilities of two binary endpoints*

Description

Converts the marginal response rates of two binary endpoints and their correlation into the probabilities of the four joint response patterns.

Usage

```
cellprob_binary2(pi1, pi2, rho, tol = 1e-10)
```

Arguments

`pi1`	Response rate of endpoint 1, strictly between 0 and 1.
`pi2`	Response rate of endpoint 2, strictly between 0 and 1.
`rho`	Correlation between the two endpoints.
`tol`	Tolerance for the feasibility check.

Details

For two Bernoulli variables with response rates π_1 and π_2, the Pearson correlation ρ determines

$$p_{11} = \pi_1\pi_2 + \rho\sqrt{\pi_1(1-\pi_1)\pi_2(1-\pi_2)},$$

and $p_{10} = \pi_1 - p_{11}$, $p_{01} = \pi_2 - p_{11}$, $p_{00} = 1 - p_{10} - p_{01} - p_{11}$. All four probabilities are non-negative only if $\max(0, \pi_1 + \pi_2 - 1) \le p_{11} \le \min(\pi_1, \pi_2)$, which bounds the feasible range of ρ. An error is raised if `rho` is outside this range by more than `tol`.

Value

A named numeric vector of the probabilities of the cells `"00"`, `"01"`, `"10"` and `"11"` (first digit: endpoint 1; second digit: endpoint 2), summing to 1.

See Also

Other distributions: pbetabinomdiff(), pbetadiff(), phazardratio(), psurvdiff(), ptdiff(), rdirichlet()

Examples

```
cellprob_binary2(pi1 = 0.3, pi2 = 0.4, rho = 0.3)
```

---

conduct_tte *Conduct of a study with a time-to-event endpoint*

Description

Specifies how subjects are enrolled, how they drop out and when the analysis takes place. The operating characteristics of a design with a time-to-event endpoint (design_tte) are computed by simulating studies conducted in this way, and the posterior predictive probabilities by simulating future studies.

Usage

```
conduct_tte(
  accrual_time,
  accrual_rate = rep(1, length(accrual_time) - 1L),
  dropout_time = 0,
  dropout_hazard = 0,
  cut_time = NULL,
  cut_events = NULL
)
```

Arguments

accrual_time	Boundaries of the enrollment intervals: an increasing numeric vector that starts at 0, of length at least 2.
accrual_rate	Non-negative relative enrollment rates of the intervals, one fewer than accrual_time, with a positive sum. The default is a constant rate (uniform enrollment).
dropout_time	Start times of the pieces of the dropout hazard: an increasing numeric vector that starts at 0.
dropout_hazard	Non-negative dropout hazards of the pieces, of the same length as dropout_time. The default 0 means no dropout.
cut_time	Calendar time of the analysis after the start of enrollment (positive), or NULL.
cut_events	Number of events that triggers the analysis (a whole number of at least 1), or NULL. Exactly one of cut_time and cut_events must be given.

Details

The subjects of all groups are enrolled at independent times with a piecewise uniform distribution: the enrollment period is divided by `accrual_time`, and the probability of an interval is proportional to its rate in `accrual_rate` times its length (only the relative rates matter). The dropout time of a subject, measured from enrollment, has a piecewise exponential distribution with hazard `dropout_hazard[j]` from time `dropout_time[j]` (the last hazard applies until infinity), independently of the event time. The analysis takes place at the calendar time `cut_time` after the start of enrollment, or at the calendar time of the `cut_events`-th event among all subjects of the study (if fewer events occur, all subjects are followed until their event or dropout). A subject enrolled at time e with event time T and dropout time C contributes the observed time $\min(T, C, \text{cut} - e)$ (0 if enrolled after the analysis) and an event if $T \leq C$ and $e + T \leq$ cut.

Value

An object of class `"qdm_conduct"` with the arguments and the element `accrual_cum`, the cumulative probabilities of enrollment at the ends of the intervals.

See Also

`design_tte`

Examples

```
# Enrollment over 15 months with increasing rates, dropout hazard 0.00226
# per month, analysis 15.44 months after the start of enrollment
conduct_tte(accrual_time = c(0, 5, 10, 15), accrual_rate = c(0.4, 2, 4.5),
            dropout_hazard = 0.00226, cut_time = 15.44)

# Uniform enrollment over 12 months, analysis at the 20th event
conduct_tte(accrual_time = c(0, 12), cut_events = 20)
```

`criterion_posterior` *Decision criterion based on posterior probability*

Description

Specifies a Go/NoGo decision criterion based on the posterior probability of the treatment effect θ (the difference between the treatment and control groups, or their hazard ratio for a time-to-event endpoint) given the proof-of-concept data D.

Usage

```
criterion_posterior(
  theta_tv,
  theta_mav,
  gamma_go,
  gamma_nogo,
  go_regions = 1L,
```

```
  nogo_regions = 9L,
  rule = c("joint", "and", "or"),
  direction = "greater"
)
```

Arguments

`theta_tv`	Target value (TV) of the treatment effect: a number for one endpoint or a numeric vector of length 2 for two endpoints.
`theta_mav`	Minimum acceptable value (MAV) of the treatment effect, of the same length as `theta_tv`. Each element must be smaller than the corresponding element of `theta_tv` (larger for `direction = "less"`).
`gamma_go`	Probability threshold for the Go criterion, strictly between 0 and 1. With `rule = "and"` or `"or"`, one threshold per endpoint (a single number is used for both).
`gamma_nogo`	Probability threshold for the NoGo criterion, strictly between 0 and 1, given in the same way as `gamma_go`.
`go_regions`	Region numbers (from 1 to 9) forming the Go region for two endpoints with `rule = "joint"`. Ignored for one endpoint. The default, region 1, requires both effects to be beyond their target values in the favorable direction.
`nogo_regions`	Region numbers (from 1 to 9) forming the NoGo region for two endpoints with `rule = "joint"`. Ignored for one endpoint. The default, region 9, requires both effects to be at most their minimum acceptable values (at least, for `direction = "less"`).
`rule`	For two endpoints, `"joint"` (the default; probabilities of unions of regions), `"and"` or `"or"` (marginal probabilities of each endpoint, combined with AND or OR). Ignored for one endpoint.
`direction`	`"greater"` (the default) if larger treatment effects are favorable, or `"less"` if smaller effects are favorable. For two endpoints, a single value for both or one value per endpoint.

Details

For one endpoint, the Go region is $\theta > \theta_{TV}$ and the NoGo region is $\theta \leq \theta_{MAV}$. The Go criterion is $\Pr(\theta > \theta_{TV} \mid D) \geq \gamma_{go}$ and the NoGo criterion is $\Pr(\theta \leq \theta_{MAV} \mid D) \geq \gamma_{nogo}$. In the notation of Kang et al. (2026), where the NoGo criterion is $\Pr(\theta > \theta_{MAV} \mid D) \leq \gamma_2$, this corresponds to $\gamma_{nogo} = 1 - \gamma_2$.

By default, larger effects are favorable (`direction = "greater"`, with $\theta_{TV} > \theta_{MAV}$), as for most efficacy endpoints. For an endpoint on which smaller effects are favorable, such as the rate of an adverse event or a hazard ratio, `direction = "less"` reverses the inequalities (with $\theta_{TV} < \theta_{MAV}$): the Go region is $\theta < \theta_{TV}$, the NoGo region is $\theta \geq \theta_{MAV}$, and the criteria are $\Pr(\theta < \theta_{TV} \mid D) \geq \gamma_{go}$ and $\Pr(\theta \geq \theta_{MAV} \mid D) \geq \gamma_{nogo}$.

For two endpoints, the target values and minimum acceptable values divide the plane of (θ_1, θ_2) into nine regions (see `qdm_regions`). The Go criterion is that the posterior probability of the union of `go_regions` is at least γ_{go}, and the NoGo criterion is that the posterior probability of the union of `nogo_regions` is at least γ_{nogo} (Yamaguchi et al., submitted, equations 2 and 3). With `rule = "and"` or `"or"`, the criteria are instead based on the marginal posterior probabilities of each endpoint

(equations 4 and 5): the Go criterion is $\Pr(\theta_1 > \theta_{TV1} \mid D) \geq \gamma_{go1}$ and (or) $\Pr(\theta_2 > \theta_{TV2} \mid D) \geq \gamma_{go2}$, and the NoGo criterion is $\Pr(\theta_1 \leq \theta_{MAV1} \mid D) \geq \gamma_{nogo1}$ and (or) $\Pr(\theta_2 \leq \theta_{MAV2} \mid D) \geq \gamma_{nogo2}$.

With two endpoints, `direction` can differ between the endpoints, for example an efficacy endpoint (`"greater"`) and a safety endpoint (`"less"`). The regions are then defined on the favorable side of each endpoint: region 1 has both effects beyond their target values in the favorable direction (see `qdm_regions`).

The decision is Go if only the Go criterion is met, NoGo if only the NoGo criterion is met, Gray if neither is met, and Miss if both are met. For one endpoint, Miss can occur only when $\gamma_{go} + \gamma_{nogo} \leq 1$.

Value

An object of class `c("qdm_criterion_posterior", "qdm_criterion")`.

References

Kang, R., Yamaguchi, Y. and Han, C. (2026). Bayesian quantitative decision-making framework in cell and gene therapy development for rare diseases. *Journal of Biopharmaceutical Statistics*. doi:10.1080/10543406.2026.2655410

Yamaguchi, Y., Hu, H., Huang, Y., Han, C., Homma, G. and Wang, A. (submitted). Bivariate Bayesian quantitative decision-making for two endpoints in rare disease drug development.

See Also

Other criteria: `criterion_predictive()`

Examples

```
# One endpoint
criterion_posterior(theta_tv = 0.2, theta_mav = 0.1,
                    gamma_go = 0.5, gamma_nogo = 0.8)

# Two endpoints
criterion_posterior(theta_tv = c(0.2, 0.15), theta_mav = c(0.1, 0.05),
                    gamma_go = 0.3, gamma_nogo = 0.4)

# Two endpoints, marginal criteria combined with AND
criterion_posterior(theta_tv = c(0.2, 0.15), theta_mav = c(0.1, 0.05),
                    gamma_go = c(0.6, 0.5), gamma_nogo = 0.7, rule = "and")

# Rate of an adverse event: Go if it is likely to be reduced by more than
# 0.1, NoGo if it is likely not to be reduced
criterion_posterior(theta_tv = -0.1, theta_mav = 0, gamma_go = 0.6,
                    gamma_nogo = 0.7, direction = "less")

# Efficacy (larger is better) and safety (smaller is better) endpoints
criterion_posterior(theta_tv = c(0.2, -0.1), theta_mav = c(0.1, 0),
                    gamma_go = 0.5, gamma_nogo = 0.5,
                    direction = c("greater", "less"))
```

`criterion_predictive` *Decision criterion based on posterior predictive probability*

Description

Specifies a Go/NoGo decision criterion based on the posterior predictive probability of the treatment effect $\tilde{\theta}$ that will be observed in a future study with `m_t` and `m_c` subjects in the treatment and control groups, given the proof-of-concept data D. The future effect is the difference between the future sample proportions (binary endpoints) or sample means (continuous endpoints), or, for a time-to-event endpoint, the difference of the estimated milestone survival probabilities or the ratio of the estimated hazards (see `design_tte`).

Usage

```
criterion_predictive(
  theta_null,
  m_t,
  m_c = m_t,
  gamma_go,
  gamma_nogo,
  go_regions = 1L,
  nogo_regions = 4L,
  rule = c("joint", "and", "or"),
  direction = "greater"
)
```

Arguments

`theta_null`	Null value of the treatment effect in the future study: a number for one endpoint or a numeric vector of length 2 for two endpoints.
`m_t`, `m_c`	Numbers of subjects in the treatment and control groups of the future study (whole numbers of at least 1).
`gamma_go`	Probability threshold for the Go criterion, strictly between 0 and 1. With `rule = "and"` or `"or"`, one threshold per endpoint (a single number is used for both).
`gamma_nogo`	Probability threshold for the NoGo criterion, strictly between 0 and 1, given in the same way as `gamma_go`.
`go_regions`	Region numbers (from 1 to 4) forming the Go region for two endpoints with `rule = "joint"`. Ignored for one endpoint. The default, region 1, requires both future effects to exceed their null values.
`nogo_regions`	Region numbers (from 1 to 4) forming the NoGo region for two endpoints with `rule = "joint"`. Ignored for one endpoint. The default, region 4, requires both future effects to be at most their null values.
`rule`	For two endpoints, `"joint"` (the default; probabilities of unions of regions), `"and"` or `"or"` (marginal probabilities of each endpoint, combined with AND or OR). Ignored for one endpoint.

`direction` `"greater"` (the default) if larger treatment effects are favorable, or `"less"` if smaller effects are favorable. For two endpoints, a single value for both or one value per endpoint.

Details

For one endpoint, the Go region is $\tilde{\theta} > \theta_{NULL}$ and the NoGo region is $\tilde{\theta} \leq \theta_{NULL}$. The Go criterion is $\Pr(\tilde{\theta} > \theta_{NULL} \mid D) \geq \gamma_{go}$ and the NoGo criterion is $\Pr(\tilde{\theta} \leq \theta_{NULL} \mid D) \geq \gamma_{nogo}$. Because the two probabilities add up to 1, a Gray decision is possible only when $\gamma_{go} + \gamma_{nogo} > 1$, and Miss decisions can occur only when $\gamma_{go} + \gamma_{nogo} \leq 1$.

With `direction = "less"`, for an endpoint on which smaller effects are favorable, the Go region is $\tilde{\theta} < \theta_{NULL}$ and the NoGo region is $\tilde{\theta} \geq \theta_{NULL}$.

For two endpoints, the null values divide the plane of $(\tilde{\theta}_1, \tilde{\theta}_2)$ into four regions (see `qdm_regions`), and the criteria are based on the predictive probabilities of the unions of `go_regions` and `nogo_regions` (Yamaguchi et al., submitted, equations 7 and 8). With `rule = "and"` or `"or"`, they are based on the marginal predictive probabilities of each endpoint, $\Pr(\tilde{\theta}_k > \theta_{NULLk} \mid D)$ for Go and $\Pr(\tilde{\theta}_k \leq \theta_{NULLk} \mid D)$ for NoGo, combined with AND or OR (equations 9 and 10). With two endpoints, `direction` can differ between the endpoints, and the regions are defined on the favorable side of each endpoint.

Value

An object of class `c("qdm_criterion_predictive", "qdm_criterion")`.

References

Yamaguchi, Y., Hu, H., Huang, Y., Han, C., Homma, G. and Wang, A. (submitted). Bivariate Bayesian quantitative decision-making for two endpoints in rare disease drug development.

See Also

Other criteria: `criterion_posterior()`

Examples

```
criterion_predictive(theta_null = 0, m_t = 50, gamma_go = 0.9,
                     gamma_nogo = 0.7)

# Smaller effects are favorable
criterion_predictive(theta_null = 0, m_t = 50, gamma_go = 0.9,
                     gamma_nogo = 0.7, direction = "less")
```

data_binary *Binary endpoint data*

Description

Summarizes the data of one group for a single binary endpoint as the number of subjects and the number of responders. Used for observed proof-of-concept data, external data and hypothetical control data.

Usage

```
data_binary(n, y)
```

Arguments

n	Number of subjects (a whole number of at least 1).
y	Number of responders (a whole number from 0 to n).

Value

An object of class `c("qdm_data_binary", "qdm_data")` with integer elements `n` and `y`.

See Also

Other data: `data_binary2()`, `data_continuous()`, `data_continuous2()`, `data_tte()`

Examples

```
data_binary(n = 12, y = 7)
```

data_binary2 *Data for two binary endpoints*

Description

Summarizes the data of one group for two binary endpoints by the counts of the four joint response patterns.

Usage

```
data_binary2(counts)
```

Arguments

counts	A numeric vector of four non-negative whole numbers in the cell order `"00"`, `"01"`, `"10"`, `"11"`, with a total of at least 1.

Details

The cells are ordered as `"00"`, `"01"`, `"10"` and `"11"`, where the first digit is the response on endpoint 1 and the second digit is the response on endpoint 2 (1 = response, 0 = no response).

Value

An object of class `c("qdm_data_binary2", "qdm_data")` with elements `n` (the total count) and `counts` (a named integer vector of length 4).

See Also

Other data: `data_binary()`, `data_continuous()`, `data_continuous2()`, `data_tte()`

Examples

```
data_binary2(c(4, 2, 3, 6))
```

---

data_continuous *Continuous endpoint data*

Description

Summarizes the data of one group for a single continuous endpoint by its sufficient statistics: the number of subjects, the sample mean and the sample standard deviation (with divisor `n - 1`).

Usage

```
data_continuous(n, mean, sd)
```

Arguments

`n`	Number of subjects (a whole number of at least 2).
`mean`	Sample mean.
`sd`	Sample standard deviation (positive).

Value

An object of class `c("qdm_data_continuous", "qdm_data")` with elements `n`, `mean` and `sd`.

See Also

Other data: `data_binary()`, `data_binary2()`, `data_continuous2()`, `data_tte()`

Examples

```
data_continuous(n = 12, mean = 3.2, sd = 2.0)
```

data_continuous2 *Data for two continuous endpoints*

Description

Summarizes the data of one group for two continuous endpoints by their sufficient statistics: the number of subjects, the vector of sample means and the sample covariance matrix (with divisor `n - 1`).

Usage

```
data_continuous2(n, mean, cov)
```

Arguments

`n`	Number of subjects (a whole number of at least 3).
`mean`	Numeric vector of length 2, the sample means of endpoints 1 and 2.
`cov`	A 2 x 2 symmetric positive definite sample covariance matrix.

Value

An object of class `c("qdm_data_continuous2", "qdm_data")` with elements `n`, `mean` and `cov`.

See Also

Other data: `data_binary()`, `data_binary2()`, `data_continuous()`, `data_tte()`

Examples

```
data_continuous2(n = 12, mean = c(3.2, 1.5),
                 cov = matrix(c(4, 1.2, 1.2, 2), nrow = 2))
```

data_tte *Time-to-event endpoint data*

Description

Summarizes the data of one group for a time-to-event endpoint by the sufficient statistics of the exponential model: the number of events and the total observed time (the sum over subjects of the time from entry to the event or censoring). Used for observed proof-of-concept data, external data and hypothetical control data.

Usage

```
data_tte(events, exposure)
```

Arguments

events	Number of events (a whole number of at least 0).
exposure	Total observed time (positive), in the same unit as the milestone time of the design.

Details

With exponential event times and independent censoring, the likelihood of the hazard λ is proportional to $\lambda^d \exp(-\lambda E)$ for d events and total observed time E, so these two numbers are all that the analysis needs.

Value

An object of class `c("qdm_data_tte", "qdm_data")` with elements `events` and `exposure`.

See Also

Other data: `data_binary()`, `data_binary2()`, `data_continuous()`, `data_continuous2()`

Examples

```
data_tte(events = 5, exposure = 40)
```

---

design_binary *Design with one binary endpoint*

Description

Specifies a proof-of-concept study with one binary endpoint: the sample sizes, the prior distributions, the external data, the form of the control group and the decision criterion. The design is used by `qdm_decide`, `qdm_oc` and `qdm_calibrate`.

Usage

```
design_binary(
  n_t,
  n_c,
  prior_t,
  prior_c = NULL,
  criterion,
  external_t = NULL,
  external_c = NULL,
  hypothetical_c = NULL,
  fixed_c = NULL
)
```

Arguments

`n_t`	Number of subjects in the treatment group (a whole number of at least 1).
`n_c`	Number of subjects in the concurrent control group (a whole number of at least 0).
`prior_t`, `prior_c`	Beta prior distributions of the response rates of the treatment and control groups, created by `prior_beta`. `prior_c` must be `NULL` (the default) for a fixed control and is required otherwise.
`criterion`	Decision criterion for one endpoint, created by `criterion_posterior` or `criterion_predictive`. Its argument `direction` sets whether larger or smaller effects are favorable.
`external_t`, `external_c`	External data of the treatment and control groups, created by `borrow` with `data_binary`, or `NULL` (no external data).
`hypothetical_c`	Hypothetical control, created by `hypothetical` with `data_binary`, or `NULL`. It can be used only when `n_c = 0`.
`fixed_c`	Fixed control response rate, a number from 0 to 1, or `NULL`. It can be used only when `n_c = 0`.

Details

The response rates π_t and π_c of the treatment and control groups have Beta prior distributions, and the treatment effect is $\theta = \pi_t - \pi_c$. The control group takes one of four forms.

Concurrent control `n_c` is at least 1. External control data can be added through `external_c`.

External control only `n_c = 0` and `external_c` is given. The distribution of π_c is the posterior distribution given the external control data.

Hypothetical control `n_c = 0` and `hypothetical_c` is given. The distribution of π_c is the posterior distribution given the hypothetical control data. The hypothetical control $\pi_c \sim \text{Beta}(\alpha_t + z, \beta_t + n_t - z)$ of Kang et al. (2026, Section 4.1.4) is obtained with `prior_c` equal to `prior_t` and `hypothetical(data_binary(n_t, z))`.

Fixed control `n_c = 0` and `fixed_c` is given: the control response rate is a known value π_c, as in a single-arm study compared with a benchmark. The posterior probabilities are $\Pr(\pi_t > \theta_{TV} + \pi_c \mid D)$ and $\Pr(\pi_t \le \theta_{MAV} + \pi_c \mid D)$, computed from the Beta posterior distribution of π_t (with the inequalities reversed for `direction = "less"`). For the posterior predictive criterion the future effect is $Y_t/m_t - \pi_c$, where Y_t follows the Beta-binomial posterior predictive distribution (a future effect that differs from the null value by less than 10^{-10} is treated as equal to it); `m_c` is not used. `prior_c` is not used and must be `NULL`.

External treatment data (`external_t`) can be used with any form of the control group. External data enter through a power prior: with external data of n_e subjects and y_e responders and weight a_0, the posterior distribution of the response rate is $\text{Beta}(\alpha + a_0 y_e + y, \beta + a_0(n_e - y_e) + n - y)$.

Value

An object of class `c("qdm_design_binary", "qdm_design")`, a list with the arguments and the element `control` (`"concurrent"`, `"external"`, `"hypothetical"` or `"fixed"`).

References

Kang, R., Yamaguchi, Y. and Han, C. (2026). Bayesian quantitative decision-making framework in cell and gene therapy development for rare diseases. *Journal of Biopharmaceutical Statistics*. doi:10.1080/10543406.2026.2655410

See Also

Other designs: `design_binary2()`, `design_continuous()`, `design_continuous2()`, `design_tte()`

Examples

```
# Concurrent control with external control data
design_binary(
  n_t = 12, n_c = 12,
  prior_t = prior_beta(0.5, 0.5), prior_c = prior_beta(0.5, 0.5),
  criterion = criterion_posterior(theta_tv = 0.2, theta_mav = 0.1,
                                  gamma_go = 0.5, gamma_nogo = 0.8),
  external_c = borrow(data_binary(n = 20, y = 8), a0 = 0.5)
)

# Uncontrolled study with a hypothetical control
design_binary(
  n_t = 12, n_c = 0,
  prior_t = prior_beta(0.5, 0.5), prior_c = prior_beta(0.5, 0.5),
  criterion = criterion_predictive(theta_null = 0, m_t = 50,
                                   gamma_go = 0.9, gamma_nogo = 0.7),
  hypothetical_c = hypothetical(data = data_binary(n = 12, y = 4))
)

# Single-arm study compared with a fixed control response rate of 0.2
design_binary(
  n_t = 20, n_c = 0, prior_t = prior_beta(1, 1),
  criterion = criterion_posterior(theta_tv = 0.2, theta_mav = 0.05,
                                  gamma_go = 0.8, gamma_nogo = 0.8),
  fixed_c = 0.2
)
```

`design_binary2` *Design with two binary endpoints*

Description

Specifies a proof-of-concept study with two binary endpoints: the sample sizes, the Dirichlet prior distributions, the external data, the form of the control group, the decision criterion for two endpoints and the computation settings. The design is used by `qdm_decide`, `qdm_oc` and `qdm_calibrate`.

Usage

```
design_binary2(
  n_t,
  n_c,
  prior_t,
  prior_c = NULL,
  criterion,
  external_t = NULL,
  external_c = NULL,
  hypothetical_c = NULL,
  fixed_c = NULL,
  nMC = 10000L,
  enumerate = TRUE
)
```

Arguments

`n_t`	Number of subjects in the treatment group (a whole number of at least 1).
`n_c`	Number of subjects in the concurrent control group (a whole number of at least 0).
`prior_t, prior_c`	Dirichlet prior distributions of the four cell probabilities of the treatment and control groups, created by `prior_dirichlet`. `prior_c` must be `NULL` (the default) for a fixed control and is required otherwise.
`criterion`	Decision criterion for two endpoints, created by `criterion_posterior` or `criterion_predictive`. Its argument `direction` sets whether larger or smaller effects are favorable on each endpoint.
`external_t, external_c`	External data of the treatment and control groups, created by `borrow` with `data_binary2`, or `NULL` (no external data).
`hypothetical_c`	Hypothetical control, created by `hypothetical` with `data_binary2`, or `NULL`. It can be used only when `n_c = 0`.
`fixed_c`	Fixed control response rates of endpoints 1 and 2, a numeric vector of length 2 with values from 0 to 1, or `NULL`. It can be used only when `n_c = 0`.
`nMC`	Number of Monte Carlo draws for posterior probabilities.
`enumerate`	Logical; if `TRUE` (the default), the operating characteristics are computed by enumerating all outcomes, otherwise by simulation.

Details

The responses of a subject on the two endpoints fall into one of the four cells `"00"`, `"01"`, `"10"` and `"11"` (first digit: endpoint 1), whose probabilities have Dirichlet prior distributions in each group (Yamaguchi et al., submitted, Appendix A.1). The treatment effects are the differences of the marginal response rates, $\theta_k = \pi_{tk} - \pi_{ck}$ with $\pi_1 = p_{10} + p_{11}$ and $\pi_2 = p_{01} + p_{11}$. External data with counts x_e enter through a power prior with weight a_0, which adds $a_0 x_e$ to the Dirichlet parameters. The control group takes one of four forms, as in `design_binary`: a concurrent control group

(n_c of at least 1), external control data only (n_c = 0 and external_c), a hypothetical control (n_c = 0 and hypothetical_c created from data_binary2, whose counts are added to prior_c; the hypothetical control distribution Dir(beta) of Yamaguchi et al., Appendix A.1.6, is obtained with prior_c = prior_dirichlet(beta - counts)), or a fixed control (n_c = 0 and fixed_c, the known response rates of the two endpoints in the control group; prior_c must then be NULL). With a fixed control, the posterior probabilities are estimated from draws of the treatment distribution only, and the future effects of the predictive criterion are $S_{tk}/m_t - \pi_{ck}$ (a future effect that differs from the null value by less than 10^{-10} is treated as equal to it; m_c is not used).

The posterior probabilities of the regions (posterior criteria) are estimated by Monte Carlo simulation with nMC draws from the Dirichlet posterior distributions. The posterior predictive probabilities (predictive criteria) are computed exactly: the future effects depend only on the numbers of responders on each endpoint, whose joint distribution in each group is obtained from the Dirichlet-multinomial distribution, and a future effect equal to a null value is not counted as larger.

For qdm_oc and qdm_calibrate, with enumerate = TRUE all outcomes of the study are enumerated: there are choose(n + 3, 3) outcomes in a group of n subjects (286 for n = 10), and the probabilities of the regions are computed for every pair of treatment and control outcomes, with the same Monte Carlo draws for all pairs. For a posterior criterion the computing time is proportional to the number of pairs times nMC, and enumeration is limited to 1e7 pairs (for example, 24 subjects in each group). With enumerate = FALSE, nsim studies are simulated under each scenario, which is faster for large sample sizes.

Value

An object of class c("qdm_design_binary2", "qdm_design"), a list with the arguments and the element control ("concurrent", "external", "hypothetical" or "fixed").

References

Yamaguchi, Y., Hu, H., Huang, Y., Han, C., Homma, G. and Wang, A. (submitted). Bivariate Bayesian quantitative decision-making for two endpoints in rare disease drug development.

See Also

Other designs: design_binary(), design_continuous(), design_continuous2(), design_tte()

Examples

```
# Case study 1 of Yamaguchi et al. (submitted), with a small nMC for speed
des <- design_binary2(
  n_t = 10, n_c = 10,
  prior_t = prior_dirichlet(rep(0.25, 4)),
  prior_c = prior_dirichlet(rep(0.25, 4)),
  criterion = criterion_posterior(theta_tv = c(0.15, 0.15),
                                  theta_mav = c(0.10, 0.10),
                                  gamma_go = 0.263, gamma_nogo = 0.347),
  nMC = 2000
)
des
qdm_decide(des, data_t = data_binary2(c(3, 2, 2, 3)),
           data_c = data_binary2(c(6, 2, 1, 1)), seed = 1)
```

`design_continuous` *Design with one continuous endpoint*

Description

Specifies a proof-of-concept study with one normally distributed endpoint: the sample sizes, the prior distributions, the external data, the form of the control group, the decision criterion and the method used to compute the probabilities. The design is used by `qdm_decide`, `qdm_oc` and `qdm_calibrate`.

Usage

```
design_continuous(
  n_t,
  n_c,
  prior_t,
  prior_c = NULL,
  criterion,
  external_t = NULL,
  external_c = NULL,
  hypothetical_c = NULL,
  fixed_c = NULL,
  method = c("NI", "MM", "MC"),
  nMC = 10000L
)
```

Arguments

`n_t`	Number of subjects in the treatment group (a whole number of at least 2).
`n_c`	Number of subjects in the concurrent control group: 0 (no concurrent control group) or a whole number of at least 2.
`prior_t, prior_c`	Prior distributions of the mean and variance of the treatment and control groups, created by `prior_nix` or `prior_vague`. `prior_c` must be `NULL` (the default) for a hypothetical or fixed control and is required otherwise.
`criterion`	Decision criterion for one endpoint, created by `criterion_posterior` or `criterion_predictive`. Its argument `direction` sets whether larger or smaller effects are favorable.
`external_t, external_c`	External data of the treatment and control groups, created by `borrow` with `data_continuous`, or `NULL` (no external data).
`hypothetical_c`	Hypothetical control, created by `hypothetical` with `mean` (of length 1) and `r`, or `NULL`. It can be used only when `n_c = 0`.
`fixed_c`	Fixed control mean, a number, or `NULL`. It can be used only when `n_c = 0`.

`method` Method used to compute the probabilities: `"NI"` (numerical integration, the default), `"MM"` (moment-matching approximation, which requires the distributions of both group means to have more than 4 degrees of freedom) or `"MC"` (Monte Carlo simulation). See `ptdiff`. Numerical integration is exact up to a small numerical error but is the slowest of the deterministic methods: for operating characteristics, which need the probabilities for many simulated studies, `"MM"` is faster by about two orders of magnitude and is suited to exploring designs, and `"NI"` can be used to confirm the chosen design. The computing time of `"MC"` is proportional to `nMC`.

`nMC` Number of Monte Carlo draws for `method = "MC"`.

Details

The outcomes of group k ($k = t, c$) follow $N(\mu_k, \sigma_k^2)$, and the treatment effect is $\theta = \mu_t - \mu_c$. With the Normal-Inverse-chi-squared prior (`prior_nix`), the posterior distribution of (μ_k, σ_k^2) is Normal-Inverse-chi-squared with parameters $(\mu_{nk}, \kappa_{nk}, \nu_{nk}, \sigma_{nk}^2)$, and the posterior distribution of the mean is $t_{\nu_{nk}}(\mu_{nk}, \sigma_{nk}^2/\kappa_{nk})$. With the vague prior (`prior_vague`) and n_k subjects with sample mean $\bar{y}_k$ and standard deviation s_k, it is $t_{n_k-1}(\bar{y}_k, s_k^2/n_k)$.

External data enter through a power prior with weight a_0: data with n_e subjects, mean $\bar{y}_e$ and standard deviation s_e are combined with the prior as if they were $a_0 n_e$ subjects with the same mean and the sum of squares $a_0(n_e - 1)s_e^2$. The posterior distribution then has a closed form (Huang et al., submitted). With the vague prior and external data, the posterior distribution of the mean has $a_0 n_e + n_k - 1$ degrees of freedom.

The control group takes one of four forms.

Concurrent control `n_c` is at least 2. External control data can be added through `external_c`.

External control only `n_c = 0` and `external_c` is given. The distribution of μ_c is the posterior distribution given the external control data. With the vague prior, $a_0 n_e$ must be greater than 1.

Hypothetical control `n_c = 0` and `hypothetical_c` is given by `hypothetical(mean = mu0c, r = r)`. Following Kang et al. (2026, Section 4.2.4), the control mean has the same t distribution as the posterior distribution of the treatment mean, with location μ_{0c} and squared scale multiplied by r: $\mu_c \sim t_{\nu_{nt}}(\mu_{0c}, r\sigma_{nt}^2/\kappa_{nt})$. `prior_c` is not used and must be `NULL`.

Fixed control `n_c = 0` and `fixed_c` is given: the control mean is a known value μ_c, as in a single-arm study compared with a benchmark. The probabilities of the Go and NoGo regions are computed exactly from the t distribution of the treatment mean (posterior) or of the mean of m_t future subjects (posterior predictive), shifted by μ_c; `method`, `nMC` and `m_c` are not used. `prior_c` is not used and must be `NULL`.

For the posterior predictive criterion (`criterion_predictive`), the future effect is the difference between the sample means of m_t and m_c future subjects. The posterior predictive distribution of the mean of m future subjects of a group is $t_{\nu_n}(\mu_n, \sigma_n^2(1/\kappa_n + 1/m))$; for a hypothetical control it is $t_{\nu_{nt}}(\mu_{0c}, r\sigma_{nt}^2(1/\kappa_{nt} + 1/m_c))$.

The probabilities of the Go and NoGo regions are probabilities of the difference of two t variables, computed by `ptdiff` with the chosen `method`. With `method = "MC"`, the probabilities of the two regions are estimated from the same `nMC` draws.

Value

An object of class `c("qdm_design_continuous", "qdm_design")`, a list with the arguments and the element `control` (`"concurrent"`, `"external"`, `"hypothetical"` or `"fixed"`).

References

Kang, R., Yamaguchi, Y. and Han, C. (2026). Bayesian quantitative decision-making framework in cell and gene therapy development for rare diseases. *Journal of Biopharmaceutical Statistics*. doi:10.1080/10543406.2026.2655410

Huang, Y., Yamaguchi, Y., Homma, G., Takeda, K. and Maruo, K. (submitted). Conjugate representation of power priors for efficient Bayesian analysis of normal data.

See Also

Other designs: `design_binary()`, `design_binary2()`, `design_continuous2()`, `design_tte()`

Examples

```
# Concurrent control with external treatment and control data
design_continuous(
  n_t = 12, n_c = 12,
  prior_t = prior_vague(), prior_c = prior_vague(),
  criterion = criterion_posterior(theta_tv = 30, theta_mav = 15,
                                  gamma_go = 0.6, gamma_nogo = 0.7),
  external_t = borrow(data_continuous(n = 6, mean = 20, sd = 30), a0 = 0.5),
  external_c = borrow(data_continuous(n = 30, mean = -12, sd = 20),
                      a0 = 0.4)
)

# Uncontrolled study with a hypothetical control
design_continuous(
  n_t = 12, n_c = 0, prior_t = prior_vague(),
  criterion = criterion_predictive(theta_null = 0, m_t = 50,
                                   gamma_go = 0.9, gamma_nogo = 0.7),
  hypothetical_c = hypothetical(mean = -10, r = 1),
  method = "MM"
)

# Single-arm study compared with a fixed control mean of -10
design_continuous(
  n_t = 12, n_c = 0, prior_t = prior_vague(),
  criterion = criterion_posterior(theta_tv = 30, theta_mav = 15,
                                  gamma_go = 0.6, gamma_nogo = 0.7),
  fixed_c = -10
)
```

`design_continuous2` *Design with two continuous endpoints*

Description

Specifies a proof-of-concept study with two continuous endpoints: the sample sizes, the prior distributions, the external data, the form of the control group, the decision criterion for two endpoints and the method used to compute the probabilities. The design is used by `qdm_decide`, `qdm_oc` and `qdm_calibrate`.

Usage

```
design_continuous2(
  n_t,
  n_c,
  prior_t,
  prior_c = NULL,
  criterion,
  external_t = NULL,
  external_c = NULL,
  hypothetical_c = NULL,
  fixed_c = NULL,
  method = c("MM", "MC"),
  nMC = 10000L
)
```

Arguments

`n_t`	Number of subjects in the treatment group (a whole number of at least 3).
`n_c`	Number of subjects in the concurrent control group: 0 or a whole number of at least 3.
`prior_t, prior_c`	Prior distributions of the treatment and control groups, created by `prior_niw` or `prior_vague`. For a hypothetical or fixed control, `prior_c` must be `NULL` (the default).
`criterion`	Decision criterion for two endpoints, created by `criterion_posterior` or `criterion_predictive`. Its argument `direction` sets whether larger or smaller effects are favorable on each endpoint.
`external_t, external_c`	External data of the treatment and control groups, created by `borrow` with `data_continuous2`, or `NULL` (no external data).
`hypothetical_c`	Hypothetical control, created by `hypothetical` with `mean` of length 2 and `r`, or `NULL`. It can be used only when `n_c = 0`.
`fixed_c`	Fixed control means of endpoints 1 and 2, a numeric vector of length 2, or `NULL`. It can be used only when `n_c = 0`.
`method`	Method used to compute the probabilities: `"MM"` (default) or `"MC"`.
`nMC`	Number of Monte Carlo draws for `method = "MC"`.

Details

The responses of a subject on the two endpoints have a bivariate normal distribution with mean vector μ_k and covariance matrix Σ_k in group k (Yamaguchi et al., submitted, Appendix A.2). The prior distribution of (μ_k, Σ_k) is Normal-Inverse-Wishart (`prior_niw`) or vague (`prior_vague`, the density $|\Sigma_k|^{-3/2}$). External data enter through a power prior with weight a_0, which has the conjugate form, and the posterior parameters are updated with the external data and then with the current data. The vague prior corresponds to $\kappa_0 = 0$, $\nu_0 = -1$ and $\Lambda_0 = 0$, so the posterior distribution of the mean is $t_{n-2}(\bar{y}, S/(n(n-2)))$ without external data, where S is the matrix of sums of squares and cross-products, $n-1$ times the sample covariance matrix (Yamaguchi et al., submitted, equation A.9).

The treatment effects are the differences of the means, $\theta_k = \mu_{tk} - \mu_{ck}$. The mean of each group has a bivariate t distribution with $\nu_n - 1$ degrees of freedom, location μ_n and scale matrix $\Lambda_n/(\kappa_n(\nu_n - 1))$. For a predictive criterion, the mean of m future subjects has the same distribution with the scale matrix $(1/\kappa_n + 1/m)\Lambda_n/(\nu_n - 1)$, because the future subjects share the same mean and covariance matrix (version 0.1.0 and Yamaguchi et al. used the scale matrix $(1+\kappa_n)\Lambda_n/(\kappa_n(\nu_n - 1)m)$, which treats them as independent draws from the predictive distribution of one subject). A hypothetical control (`hypothetical(mean = c(m1, m2), r = r)`) gives the control mean the distribution of the treatment mean with location `mean` and the scale matrix multiplied by `r` (Yamaguchi et al., Section A.2.6, with the hyperparameters of the treatment group). With external control data only (`n_c = 0` and `external_c`) and the vague prior, $a_0 n_e$ must be greater than 2. A fixed control (`n_c = 0` and `fixed_c`, the known control means of the two endpoints) shifts the distribution of the treatment mean, so the effect has a bivariate t distribution and the probabilities of the regions are computed exactly from its distribution function; `method`, `nMC` and `m_c` are then not used, and `prior_c` must be `NULL`.

The probabilities of the regions are computed by one of two methods:

`"MM"` moment-matching approximation: the effect is approximated by a bivariate t distribution with the same mean, covariance matrix and radial fourth moment (Yamaguchi, Homma, Takeda and Maruo, submitted, Theorem 3), whose rectangle probabilities are computed by numerical integration for real degrees of freedom. It requires more than 4 degrees of freedom for both groups and is the fastest method.

`"MC"` Monte Carlo simulation with `nMC` draws from each distribution; the probabilities of all regions are estimated from the same draws. The computing time is proportional to `nMC`, so `"MM"` is much faster for operating characteristics, which need the probabilities for many simulated studies.

The operating characteristics are computed by simulating the sample means and the sums of squares and cross-products of `nsim` studies (see `qdm_oc`).

Value

An object of class `c("qdm_design_continuous2", "qdm_design")`, a list with the arguments and the element `control` (`"concurrent"`, `"external"`, `"hypothetical"` or `"fixed"`).

References

Yamaguchi, Y., Hu, H., Huang, Y., Han, C., Homma, G. and Wang, A. (submitted). Bivariate Bayesian quantitative decision-making for two endpoints in rare disease drug development.

Yamaguchi, Y., Homma, G., Takeda, K. and Maruo, K. (submitted). Moment-matching approximation for t-differences in clinical trials with continuous endpoints.

See Also

Other designs: design_binary(), design_binary2(), design_continuous(), design_tte()

Examples

```
# Case study 2 of Yamaguchi et al. (submitted)
L0 <- matrix(c(80^2, 15, 15, 6^2), 2, 2)
des <- design_continuous2(
  n_t = 30, n_c = 30,
  prior_t = prior_niw(mu0 = c(0, 1), kappa0 = 0.1, nu0 = 4, Lambda0 = L0),
  prior_c = prior_niw(mu0 = c(-10, -1), kappa0 = 0.1, nu0 = 4,
                      Lambda0 = L0),
  criterion = criterion_posterior(theta_tv = c(10, 2),
                                  theta_mav = c(5, 1),
                                  gamma_go = 0.296, gamma_nogo = 0.244)
)
des
qdm_decide(des,
           data_t = data_continuous2(30, c(5, 2),
                                     matrix(c(6400, 190, 190, 36), 2, 2)),
           data_c = data_continuous2(30, c(-10, -1),
                                     matrix(c(5600, 150, 150, 42), 2, 2)))
```

---

design_tte *Design with one time-to-event endpoint*

Description

Specifies a proof-of-concept study with one time-to-event endpoint evaluated by the milestone survival probability or by the hazard ratio: the sample sizes, the prior distributions, the external data, the form of the control group, the decision criterion, the effect measure and the conduct of the study. The design is used by qdm_decide, qdm_oc, qdm_calibrate and qdm_boundary.

Usage

```
design_tte(
  n_t,
  n_c,
  prior_t,
  prior_c = NULL,
  criterion,
  milestone = NULL,
  effect = c("milestone", "hazard_ratio"),
  conduct = NULL,
  external_t = NULL,
```

```
  external_c = NULL,
  hypothetical_c = NULL,
  fixed_c = NULL,
  conduct_future = conduct,
  nMC = 10000L
)
```

Arguments

`n_t`	Number of subjects in the treatment group (a whole number of at least 1).
`n_c`	Number of subjects in the concurrent control group (a whole number of at least 0).
`prior_t`, `prior_c`	Gamma prior distributions of the hazards of the treatment and control groups, created by `prior_gamma`. `prior_c` must be `NULL` (the default) for a fixed control and is required otherwise.
`criterion`	Decision criterion for one endpoint, created by `criterion_posterior` or `criterion_predictive`, with thresholds on the scale of the effect: the milestone survival difference (or the milestone survival itself with `fixed_c = 0`), or the hazard ratio (positive thresholds).
`milestone`	Milestone time t_0 (positive), in the same unit as the total observed time of the data. Required for `effect = "milestone"`; must be `NULL` (the default) for the hazard ratio.
`effect`	Effect measure: `"milestone"` (the default; difference of the milestone survival probabilities) or `"hazard_ratio"`.
`conduct`	Conduct of the study, created by `conduct_tte`, used to simulate studies for the operating characteristics; `NULL` (the default) if they are not needed. With an analysis at an event (`cut_events`), the number of events must not exceed the number of subjects (`n_t` plus `n_c`).
`external_t`, `external_c`	External data of the treatment and control groups, created by `borrow` with `data_tte`, or `NULL` (no external data).
`hypothetical_c`	Hypothetical control, created by `hypothetical` with `data_tte`, or `NULL`. It can be used only when `n_c = 0`.
`fixed_c`	Fixed control milestone survival probability, a number from 0 to 1 (`effect = "milestone"`), or fixed control median survival time, a positive number (`effect = "hazard_ratio"`); or `NULL`. It can be used only when `n_c = 0`.
`conduct_future`	Conduct of the future study for a predictive criterion, created by `conduct_tte`; by default the same as `conduct`. Required for a predictive criterion. With an analysis at an event, the number of events must not exceed the number of future subjects.
`nMC`	Number of simulated future studies for posterior predictive probabilities.

Details

The event times of group k ($k = t, c$) are exponential with hazard λ_k, which has a Gamma prior distribution (`prior_gamma`). The data of a group are summarized by the number of events d and

the total observed time E (data_tte), and the posterior distribution of the hazard is Gamma with shape $a + d$ and rate $b + E$. External data enter through a power prior with weight a_0, which adds $a_0 d_e$ to the shape and $a_0 E_e$ to the rate. The treatment effect is either

`effect = "milestone"` the difference of the milestone survival probabilities at time $t_0 =$ `milestone`, $\theta = S_t(t_0) - S_c(t_0)$ with $S_k(t_0) = \exp(-\lambda_k t_0)$, or

`effect = "hazard_ratio"` the hazard ratio $\theta = \lambda_t/\lambda_c$, as in Huang et al. (2019), whose decision rules are based on the normal approximation of the estimated log hazard ratio. Smaller hazard ratios are favorable, so the criterion is usually specified with `direction = "less"`, for example `theta_tv = 0.67` and `theta_mav = 0.8`. Under the exponential model the hazard ratio is also the ratio of the median survival times of the control and treatment groups.

The control group takes one of four forms.

Concurrent control `n_c` is at least 1. External control data can be added through `external_c`.

External control only `n_c = 0` and `external_c` is given. The distribution of λ_c is the posterior distribution given the external control data.

Hypothetical control `n_c = 0` and `hypothetical_c` is given by `hypothetical(data_tte(events, exposure))`, whose events and total observed time are added to `prior_c`.

Fixed control `n_c = 0` and `fixed_c` is given: the control milestone survival probability (`effect = "milestone"`) or the control median survival time (`effect = "hazard_ratio"`, control hazard $\log(2)$ / `fixed_c`) is a known value, as in a single-arm study compared with a benchmark (with the milestone survival, `fixed_c = 0` makes the effect the treatment milestone survival itself). `prior_c` is not used and must be `NULL`.

Posterior probabilities of the Go and NoGo regions are computed exactly for a fixed control (from the Gamma distribution of λ_t). With another control, they are computed by numerical integration for the milestone survival (see psurvdiff) and exactly from the F distribution for the hazard ratio (see phazardratio).

For the posterior predictive criterion (criterion_predictive), a future study with `m_t` treatment and `m_c` control subjects (no control group for a fixed control) is conducted as specified by `conduct_future`. The hazard of each group in the future study is estimated by $(a+d)/(b+E)$, the posterior mean of the hazard given the future data alone with the prior (a, b) of the group (`prior_t` or `prior_c`); with a weak prior this is close to the maximum likelihood estimate d/E and, unlike it, is positive when a group has no events, so that the hazard ratio is defined. The milestone survival is estimated by $\exp(-t_0(a + d)/(b + E))$. The future effect is the difference of the two milestone survival estimates (or the treatment estimate minus the fixed value), or the ratio of the two hazard estimates (or the treatment estimate divided by the fixed control hazard). The predictive probabilities are estimated by simulating `nMC` future studies, each with hazards drawn from the posterior distributions given the proof-of-concept data (see conduct_tte for the simulation of a study). They are therefore Monte Carlo estimates, and their computing time is proportional to `nMC` times the number of future subjects.

The operating characteristics (qdm_oc, qdm_calibrate) are computed by simulating `nsim` studies conducted as specified by `conduct`, with exponential event times whose hazards are given by the median survival times of the scenarios. With a predictive criterion, every simulated study requires `nMC` simulated future studies, so small values of `nsim` and `nMC` are advisable for exploration.

Value

An object of class `c("qdm_design_tte", "qdm_design")`, a list with the arguments and the element `control` (`"concurrent"`, `"external"`, `"hypothetical"` or `"fixed"`).

References

Huang, B., Talukder, E., Han, L. and Kuan, P.-F. (2019). Quantitative decision-making in randomized Phase II studies with a time-to-event endpoint. *Journal of Biopharmaceutical Statistics*, 29(1), 189–202. doi:10.1080/10543406.2018.1489400

See Also

Other designs: `design_binary()`, `design_binary2()`, `design_continuous()`, `design_continuous2()`

Examples

```
# Single-arm study compared with a fixed control 3-month survival of 0.72
des <- design_tte(
  n_t = 30, n_c = 0, prior_t = prior_gamma(0.1, 1),
  criterion = criterion_posterior(theta_tv = 0.15, theta_mav = 0,
                                  gamma_go = 0.6, gamma_nogo = 0.8),
  milestone = 3,
  conduct = conduct_tte(accrual_time = c(0, 5, 10, 15),
                        accrual_rate = c(0.4, 2, 4.5),
                        dropout_hazard = 0.00226, cut_time = 15.44),
  fixed_c = 0.72
)
des
qdm_decide(des, data_t = data_tte(events = 6, exposure = 150))

# Concurrent control
design_tte(
  n_t = 30, n_c = 30, prior_t = prior_gamma(0.1, 1),
  prior_c = prior_gamma(0.1, 1),
  criterion = criterion_posterior(theta_tv = 0.15, theta_mav = 0,
                                  gamma_go = 0.6, gamma_nogo = 0.8),
  milestone = 3,
  conduct = conduct_tte(accrual_time = c(0, 12), cut_events = 25)
)

# Hazard ratio in the randomized study of the example of Huang et al.
# (2019) (2:1 randomization, analysis at 127 events): Go if HR < 0.67
# with probability at least 0.5, NoGo if HR >= 0.8 with probability at
# least 0.6
design_tte(
  n_t = 112, n_c = 56, prior_t = prior_gamma(0.01, 0.01),
  prior_c = prior_gamma(0.01, 0.01),
  criterion = criterion_posterior(theta_tv = 0.67, theta_mav = 0.8,
                                  gamma_go = 0.5, gamma_nogo = 0.6,
                                  direction = "less"),
  effect = "hazard_ratio",
  conduct = conduct_tte(accrual_time = c(0, 13), dropout_hazard = 0.003,
```

```
                        cut_events = 127)
)
```

`format.qdm_borrow` *Format an external data specification*

Description

Format an external data specification

Usage

```
## S3 method for class 'qdm_borrow'
format(x, digits = 4L, ...)
```

Arguments

`x`	An object of class `"qdm_borrow"`.
`digits`	Number of significant digits for numeric values.
`...`	Not used.

Value

A character string describing the external data and the power prior weight.

Examples

```
format(borrow(data_binary(n = 20, y = 8), a0 = 0.5))
```

`format.qdm_boundary` *Format a decision boundary*

Description

Formats the decision intervals of `qdm_boundary` as tables with one row per condition and one column per decision: the range of the observed quantity (numbers of responders or events, or observed means), and the corresponding observed response rates (binary endpoint) or milestone survival probabilities (time-to-event endpoint) and observed treatment effects.

Usage

```
## S3 method for class 'qdm_boundary'
format(x, digits = 4L, n = 20L, ...)
```

Arguments

x	An object of class "qdm_boundary".
digits	Number of significant digits for observed means, milestone survival probabilities, effects and conditions.
n	Maximum number of conditions shown in each table.
...	Not used.

Value

A character vector with one element per line of the tables (print.qdm_boundary prints them). For rows subset from the object, which no longer have its attributes, the formatted data frame.

Examples

```
des <- design_binary(
  n_t = 20, n_c = 0, prior_t = prior_beta(1, 1),
  criterion = criterion_posterior(theta_tv = 0.2, theta_mav = 0.05,
                                  gamma_go = 0.8, gamma_nogo = 0.8),
  fixed_c = 0.2
)
format(qdm_boundary(des, n_t = c(10, 20)))
```

---

format.qdm_calibration

Format a calibration

Description

Format a calibration

Usage

```
## S3 method for class 'qdm_calibration'
format(x, digits = 4L, ...)
```

Arguments

x	An object of class "qdm_calibration".
digits	Number of significant digits for numeric values.
...	Not used.

Value

A character vector with one element per line of the description (without the table of decision probabilities).

Examples

```
des <- design_binary(
  n_t = 12, n_c = 12,
  prior_t = prior_beta(0.5, 0.5), prior_c = prior_beta(0.5, 0.5),
  criterion = criterion_posterior(theta_tv = 0.2, theta_mav = 0.1,
                                  gamma_go = 0.5, gamma_nogo = 0.8)
)
format(qdm_calibrate(des, data.frame(pi_t = 0.4, pi_c = 0.4),
                     data.frame(pi_t = 0.7, pi_c = 0.4),
                     target_go = 0.05, target_nogo = 0.2))
```

---

format.qdm_conduct *Format the conduct of a study*

Description

Format the conduct of a study

Usage

```
## S3 method for class 'qdm_conduct'
format(x, digits = 4L, ...)
```

Arguments

x	An object of class "qdm_conduct".
digits	Number of significant digits for numeric values.
...	Not used.

Value

A character vector with one element per line of the description.

Examples

```
format(conduct_tte(accrual_time = c(0, 12), cut_events = 20))
```

format.qdm_criterion *Format a decision criterion*

Description

Format a decision criterion

Usage

```
## S3 method for class 'qdm_criterion'
format(x, digits = 4L, ...)
```

Arguments

x	An object of class "qdm_criterion".
digits	Number of significant digits for numeric values.
...	Not used.

Value

A character vector with one element per line of the description.

Examples

```
format(criterion_posterior(theta_tv = 0.2, theta_mav = 0.1,
                           gamma_go = 0.5, gamma_nogo = 0.8))
```

format.qdm_data *Format a data specification*

Description

Format a data specification

Usage

```
## S3 method for class 'qdm_data'
format(x, digits = 4L, ...)
```

Arguments

x	An object of class "qdm_data".
digits	Number of significant digits for numeric values.
...	Not used.

Value

A character string describing the data.

Examples

```
format(data_binary(n = 12, y = 7))
```

`format.qdm_decision` *Format a decision*

Description

Format a decision

Usage

```
## S3 method for class 'qdm_decision'
format(x, digits = 4L, ...)
```

Arguments

`x`	An object of class `"qdm_decision"`.
`digits`	Number of significant digits for numeric values.
`...`	Not used.

Value

A character vector with one element per line of the description.

Examples

```
des <- design_binary(
  n_t = 12, n_c = 12,
  prior_t = prior_beta(0.5, 0.5), prior_c = prior_beta(0.5, 0.5),
  criterion = criterion_posterior(theta_tv = 0.2, theta_mav = 0.1,
                                  gamma_go = 0.5, gamma_nogo = 0.8)
)
format(qdm_decide(des, data_binary(12, 9), data_binary(12, 4)))
```

format.qdm_design *Format a design*

Description

Format a design

Usage

```
## S3 method for class 'qdm_design'
format(x, digits = 4L, ...)
```

Arguments

x	An object of class "qdm_design".
digits	Number of significant digits for numeric values.
...	Not used.

Value

A character vector with one element per line of the description.

Examples

```
des <- design_binary(
  n_t = 12, n_c = 12,
  prior_t = prior_beta(0.5, 0.5), prior_c = prior_beta(0.5, 0.5),
  criterion = criterion_posterior(theta_tv = 0.2, theta_mav = 0.1,
                                  gamma_go = 0.5, gamma_nogo = 0.8)
)
format(des)
```

format.qdm_hypothetical *Format a hypothetical control specification*

Description

Format a hypothetical control specification

Usage

```
## S3 method for class 'qdm_hypothetical'
format(x, digits = 4L, ...)
```

Arguments

x	An object of class "qdm_hypothetical".
digits	Number of significant digits for numeric values.
...	Not used.

Value

A character string describing the hypothetical control.

Examples

```
format(hypothetical(mean = 1, r = 1))
```

---

format.qdm_prior *Format a prior specification*

Description

Format a prior specification

Usage

```
## S3 method for class 'qdm_prior'
format(x, digits = 4L, ...)
```

Arguments

x	An object of class "qdm_prior".
digits	Number of significant digits for numeric values.
...	Not used.

Value

A character string describing the prior distribution.

Examples

```
format(prior_beta(0.5, 0.5))
```

`hypothetical` *Hypothetical control for an uncontrolled study*

Description

Specifies the distribution of a hypothetical control group, used when the proof-of-concept study has no concurrent control group (Kang et al., 2026).

Usage

```
hypothetical(data = NULL, mean = NULL, r = NULL)
```

Arguments

`data`	Hypothetical control data created by `data_binary`, `data_binary2` (binary endpoints) or `data_tte` (time-to-event endpoint).
`mean`	Location of the hypothetical control mean, a number for one continuous endpoint or a numeric vector of length 2 for two continuous endpoints.
`r`	Positive ratio of the squared scale of the hypothetical control to that of the treatment group (continuous endpoints).

Details

For binary and time-to-event endpoints, give `data`: the hypothetical control data are combined with the prior of the control group, for example $\pi_c \sim \text{Beta}(a_c+y, b_c+n-y)$ for one binary endpoint, or a Gamma distribution of the hazard with shape a_c+d and rate b_c+E for d events in the total observed time E. For continuous endpoints, give `mean` and `r`: the control mean has the same t distribution as the posterior distribution of the treatment mean, with location `mean` and squared scale multiplied by `r`, that is $\mu_c \sim t_{\nu_{nt}}(\mu_{0c}, r\sigma^2_{nt}/\kappa_{nt})$.

Value

An object of class `"qdm_hypothetical"` with elements `type` (`"data"` or `"location"`), `data`, `mean` and `r`.

References

Kang, R., Yamaguchi, Y. and Han, C. (2026). Bayesian quantitative decision-making framework in cell and gene therapy development for rare diseases. *Journal of Biopharmaceutical Statistics*. doi:10.1080/10543406.2026.2655410

Examples

```
# Binary endpoint: 5 responders out of 15 hypothetical control subjects
hypothetical(data = data_binary(n = 15, y = 5))

# Continuous endpoint: control mean located at 1 with the same scale
hypothetical(mean = 1, r = 1)
```

```
# Time-to-event endpoint: 8 events in a total observed time of 60
hypothetical(data = data_tte(events = 8, exposure = 60))
```

`pbetabinomdiff` *Distribution function of the difference of two Beta-binomial proportions*

Description

Computes $\Pr(Y_t/m_t - Y_c/m_c \leq q)$ (or the upper tail $\Pr(Y_t/m_t - Y_c/m_c > q)$) for independent $Y_t \sim$ Beta-binomial(m_t, a_t, b_t) and $Y_c \sim$ Beta-binomial(m_c, a_c, b_c). This is the posterior predictive probability for the difference in response proportions of a future study with m_t and m_c subjects.

Usage

```
pbetabinomdiff(q, m_t, a_t, b_t, m_c, a_c, b_c, lower.tail = TRUE)
```

Arguments

`q`	Numeric vector of quantiles.
`m_t`, `m_c`	Numbers of future subjects in the treatment and control groups (whole numbers of at least 1).
`a_t`, `b_t`	Positive shape parameters of the treatment group.
`a_c`, `b_c`	Positive shape parameters of the control group.
`lower.tail`	Logical; if `TRUE` (default), probabilities are $\Pr(Y_t/m_t - Y_c/m_c \leq q)$, otherwise $\Pr(Y_t/m_t - Y_c/m_c > q)$.

Details

The probability is an exact finite sum. The probability mass functions are computed on the log scale, so large values of m_t and m_c do not cause overflow. Each difference is compared with `q` after a single division of the exact integer numerator $m_c Y_t - m_t Y_c$ by $m_t m_c$, so differences that are mathematically equal to a decimal `q` are classified correctly (they belong to the lower tail).

Value

A numeric vector of probabilities. The arguments `q`, `a_t`, `b_t`, `a_c` and `b_c` are recycled to a common length; each must have length 1 or the common length.

See Also

Other distributions: `cellprob_binary2()`, `pbetadiff()`, `phazardratio()`, `psurvdiff()`, `ptdiff()`, `rdirichlet()`

Examples

```
# Predictive probability that the future difference exceeds 0, with
# Jeffreys priors, 7/12 versus 4/12 responders and 20 future subjects per
# group
pbetabinomdiff(0, m_t = 20, a_t = 7.5, b_t = 5.5, m_c = 20, a_c = 4.5,
               b_c = 8.5, lower.tail = FALSE)
```

`pbetadiff` *Distribution function of the difference of two Beta variables*

Description

Computes $\Pr(\pi_t - \pi_c \le q)$ (or the upper tail $\Pr(\pi_t - \pi_c > q)$) for independent $\pi_t \sim \text{Beta}(a_t, b_t)$ and $\pi_c \sim \text{Beta}(a_c, b_c)$. This is the posterior probability used for a single binary endpoint.

Usage

```
pbetadiff(q, a_t, b_t, a_c, b_c, lower.tail = TRUE)
```

Arguments

`q`	Numeric vector of quantiles.
`a_t`, `b_t`	Positive shape parameters of the treatment group.
`a_c`, `b_c`	Positive shape parameters of the control group.
`lower.tail`	Logical; if `TRUE` (default), probabilities are $\Pr(\pi_t - \pi_c \le q)$, otherwise $\Pr(\pi_t - \pi_c > q)$.

Details

The upper tail is computed as

$$\Pr(\pi_t - \pi_c > q) = \Pr(\pi_c < -q) + \int f_c(x)\,\Pr(\pi_t > x + q)\,dx,$$

where the integral runs over the values of x with $0 \le x + q \le 1$, and the first term is zero for $q \ge 0$. The integral is evaluated by adaptive Gauss-Kronrod quadrature (the QUADPACK routine used by `integrate`), with the range split at quantiles of both Beta distributions so that concentrated distributions are integrated accurately. When the density of π_c is unbounded at 0 ($a_c < 1$) or at 1 ($b_c < 1$), the integral on that side of the median of π_c is evaluated in the variable $s = x^{a_c}$ or $s = (1-x)^{b_c}$, in which the integrand is bounded. The lower tail is computed directly from the same formula with the roles of the two groups exchanged, so both tails are accurate near 0.

Value

A numeric vector of probabilities. All arguments except `lower.tail` are recycled to a common length; each must have length 1 or the common length.

See Also

Other distributions: cellprob_binary2(), pbetabinomdiff(), phazardratio(), psurvdiff(), ptdiff(), rdirichlet()

Examples

```
# Posterior probability that the response rate difference exceeds 0.1,
# with Jeffreys priors and 7/12 versus 4/12 responders
pbetadiff(0.1, a_t = 7.5, b_t = 5.5, a_c = 4.5, b_c = 8.5,
          lower.tail = FALSE)
```

`phazardratio` *Distribution function of the ratio of two hazards*

Description

Computes $\Pr(\lambda_t/\lambda_c \le q)$ (or the upper tail $\Pr(\lambda_t/\lambda_c > q)$) for hazards with independent Gamma distributions, $\lambda_t \sim \text{Gamma}(a_t, b_t)$ and $\lambda_c \sim \text{Gamma}(a_c, b_c)$ (shape and rate). This is the posterior probability used for a time-to-event endpoint evaluated by the hazard ratio with a concurrent, external or hypothetical control.

Usage

```
phazardratio(q, shape_t, rate_t, shape_c, rate_c, lower.tail = TRUE)
```

Arguments

`q`	Numeric vector of quantiles (hazard ratios).
`shape_t`, `rate_t`	Positive shape and rate of the treatment hazard.
`shape_c`, `rate_c`	Positive shape and rate of the control hazard.
`lower.tail`	Logical; if `TRUE` (default), probabilities are $\Pr(\lambda_t/\lambda_c \le q)$, otherwise $\Pr(\lambda_t/\lambda_c > q)$.

Details

Because $2b_k\lambda_k$ has the chi-squared distribution with $2a_k$ degrees of freedom, $(b_t\lambda_t/a_t)/(b_c\lambda_c/a_c)$ has the F distribution with $2a_t$ and $2a_c$ degrees of freedom, and

$$\Pr(\lambda_t/\lambda_c \le q) = F_{2a_t,2a_c}\left(\frac{q\,a_c\,b_t}{a_t\,b_c}\right),$$

which is computed exactly by pf.

Value

A numeric vector of probabilities. All arguments except `lower.tail` are recycled to a common length; each must have length 1 or the common length.

See Also

Other distributions: cellprob_binary2(), pbetabinomdiff(), pbetadiff(), psurvdiff(), ptdiff(), rdirichlet()

Examples

```
# Posterior probability that the hazard ratio is below 0.7 with 20 events
# in 300 months of follow-up versus 30 events in 250 months (Gamma(0.1, 1)
# priors)
phazardratio(0.7, shape_t = 20.1, rate_t = 301, shape_c = 30.1,
             rate_c = 251)
```

---

plot.qdm_boundary *Plot a decision boundary*

Description

Draws the decision intervals computed by qdm_boundary as regions filled by the decision (Go, NoGo, Gray, or Miss if it occurs): the horizontal axis is the condition (for example, the number of control responders or the treatment group size) and the vertical axis is the observed quantity (numbers of responders or events, or the observed treatment mean) or, for continuous endpoints with a concurrent control group, the observed difference in means. The boundary values are written on the Go and NoGo regions next to the boundaries, for example the smallest number of responders that leads to a Go decision. With a concurrent control group and a binary endpoint, the observed response rate of the treatment group is shown on a secondary axis.

Usage

```
## S3 method for class 'qdm_boundary'
plot(
  x,
  col_go = "#658D1B",
  col_nogo = "#D91E49",
  col_gray = "#939597",
  col_miss = "#E69F00",
  labels = TRUE,
  label_size = NULL,
  base_size = 11,
  legend_position = "bottom",
  ...
)
```

Arguments

x	An object of class "qdm_boundary".

col_go, col_nogo, col_gray, col_miss
: Colors of the Go, NoGo, Gray and Miss decisions: color names or hexadecimal codes. The defaults are green, red, gray and orange. Transparency (an alpha value in the code) is not used in the heat maps.

labels
: TRUE (the default) to write the boundary values for every value of the horizontal axis (or, with more than 16 values, for about six of them, leaving out the first and the last), FALSE for none, or a numeric vector of the values of the horizontal axis at which they are written.

label_size
: Size of the boundary values, in millimeters, or NULL for the size of the axis tick labels, `0.8 * base_size / .pt`.

base_size
: Base font size of `ggplot2::theme_bw()`, in points, which sets the size of the axis labels, tick labels, legend and caption.

legend_position
: Position of the legend: `"bottom"` (the default), `"top"`, `"left"`, `"right"` or `"none"`. To place the legend inside the plot, add `ggplot2::theme()`.

...
: Not used; arguments given here are ignored with a warning.

Details

The plot is a ggplot object, so it can be changed further by adding **ggplot2** functions, for example `ggplot2::labs()`, `ggplot2::theme()` and `ggplot2::scale_fill_manual()`.

Value

A ggplot object.

Examples

```
des <- design_binary(
  n_t = 12, n_c = 12,
  prior_t = prior_beta(0.5, 0.5), prior_c = prior_beta(0.5, 0.5),
  criterion = criterion_posterior(theta_tv = 0.2, theta_mav = 0.1,
                                  gamma_go = 0.5, gamma_nogo = 0.8)
)
b <- qdm_boundary(des)
plot(b)
plot(b, labels = c(0, 4, 8), col_gray = "gray80", base_size = 14,
     legend_position = "right")
```

---

plot.qdm_calibration *Plot a calibration*

Description

Draws the probability that the Go criterion is met under scenario_go and the probability that the NoGo criterion is met under scenario_nogo against the threshold, with the targets (horizontal lines) and the chosen thresholds (vertical lines).

Usage

```
## S3 method for class 'qdm_calibration'
plot(
  x,
  col_go = "#658D1B",
  col_nogo = "#D91E49",
  linewidth = 1,
  base_size = 11,
  legend_position = "bottom",
  ...
)
```

Arguments

x	An object of class "qdm_calibration".
col_go, col_nogo	Colors of the curves, targets and thresholds of the Go and NoGo criteria: color names or hexadecimal codes. The defaults are green and red.
linewidth	Width of the calibration curves, in the units of the linewidth aesthetic of **ggplot2**.
base_size	Base font size of ggplot2::theme_bw(), in points, which sets the size of the axis labels, tick labels and legend.
legend_position	Position of the legend: "bottom" (the default), "top", "left", "right" or "none". To place the legend inside the plot, add ggplot2::theme().
...	Not used; arguments given here are ignored with a warning.

Details

The plot is a ggplot object, so it can be changed further by adding **ggplot2** functions, for example ggplot2::labs() and ggplot2::theme().

Value

A ggplot object.

Examples

```
des <- design_binary(
  n_t = 12, n_c = 12,
  prior_t = prior_beta(0.5, 0.5), prior_c = prior_beta(0.5, 0.5),
  criterion = criterion_posterior(theta_tv = 0.2, theta_mav = 0.1,
                                  gamma_go = 0.5, gamma_nogo = 0.8)
)
cal <- qdm_calibrate(des, data.frame(pi_t = 0.4, pi_c = 0.4),
                     data.frame(pi_t = 0.7, pi_c = 0.4),
                     target_go = 0.05, target_nogo = 0.2)
plot(cal)
plot(cal, linewidth = 0.6, base_size = 14, legend_position = "top")
```

`plot.qdm_oc` *Plot operating characteristics*

Description

Draws the probabilities of Go, NoGo and Gray decisions (and Miss decisions if they have positive probability) against the true treatment effect of the scenarios. For a time-to-event endpoint, the effect is the difference of the milestone survival probabilities $\exp(-\log(2)t_0/m)$ given by the median survival times m of the scenarios. For a design without a concurrent control group, the horizontal axis is the true response rate (binary endpoint), mean (continuous endpoint) or milestone survival probability (time-to-event endpoint) of the treatment group, and the thresholds of the criterion are shifted by the fixed control value, or otherwise by the mean (binary and time-to-event) or location (continuous) of the distribution of the control group. The scenarios must have distinct values on the horizontal axis, so scenarios with the same effect (for example, with several standard deviations or control parameters) are plotted separately.

Usage

```
## S3 method for class 'qdm_oc'
plot(
  x,
  col_go = "#658D1B",
  col_nogo = "#D91E49",
  col_gray = "#939597",
  col_miss = "#E69F00",
  linewidth = 1,
  point_size = 1.5,
  label_size = NULL,
  base_size = 11,
  legend_position = "bottom",
  overlay = FALSE,
  cell_values = TRUE,
  ...
)
```

Arguments

`x`	An object of class `"qdm_oc"`.
`col_go`, `col_nogo`, `col_gray`, `col_miss`	Colors of the Go, NoGo, Gray and Miss decisions: color names or hexadecimal codes. The defaults are green, red, gray and orange. Transparency (an alpha value in the code) is not used in the heat maps.
`linewidth`	Width of the curves, in the units of the `linewidth` aesthetic of **ggplot2**.
`point_size`	Size of the points, in millimeters.
`label_size`	Size of the text drawn in the plot, in millimeters: the labels of the thresholds, or the probabilities in the cells of the heat maps. `NULL` (the default) gives the size of the axis tick labels, `0.8 * base_size / ggplot2::.pt`.

`base_size` Base font size of `ggplot2::theme_bw()`, in points, which sets the size of the axis labels, tick labels, legend and caption.

`legend_position` Position of the legend: `"bottom"` (the default), `"top"`, `"left"`, `"right"` or `"none"`. To place the legend inside the plot, add `ggplot2::theme()`.

`overlay` Two endpoints only: `FALSE` (the default) for one heat map per decision, or `TRUE` for a single heat map of the decision with the highest probability in each cell.

`cell_values` Two endpoints only: `TRUE` (the default) to write the probabilities in the cells of the heat maps (with `overlay = TRUE`, the probability of the decision shown), or `FALSE` to leave them out. Scenarios that are not evaluable are marked "NE" in either case. The probabilities are rounded to two decimals, so a probability just below a step of the overlay (for example 0.499) is written as the value of the step (0.50) in a cell of the lower step.

`...` Not used; arguments given here are ignored with a warning.

Details

For two endpoints, the plot has one heat map per decision, with the true treatment effects of the two endpoints (or the true parameters of the treatment group without a concurrent control group) on the axes and the probability of the decision in each cell; scenarios that are not evaluable are marked "NE". The scenarios must have distinct values on the axes, so scenarios with several correlations (or several parameters of the control group) are plotted separately. Each panel is shaded from white (probability 0) to the color of its decision (probability 1); the heat maps have no legend. With `overlay = TRUE`, a single heat map is drawn instead: each cell has the color of the decision with the highest probability (Go, Gray or NoGo, or Miss if it occurs; ties go to the first in this order), shaded in five steps of that probability (below 0.5, 0.5 to 0.6, 0.6 to 0.7, 0.7 to 0.8, and 0.8 or more), with legends for the decisions and the steps (shown in gray whatever the colors of the decisions), as in the heat maps of Yamaguchi et al. (submitted, Figure 2).

The plot is a `ggplot` object, so it can be changed further by adding **ggplot2** functions, for example `ggplot2::labs()` for the title and the axis labels, `ggplot2::theme()` for the text and the legend, and `ggplot2::scale_colour_manual()` or `ggplot2::scale_linetype_manual()` for the curves of one endpoint. The colors of the heat maps are set by the color arguments.

Value

A `ggplot` object.

References

Yamaguchi, Y., Hu, H., Huang, Y., Han, C., Homma, G. and Wang, A. (submitted). Bivariate Bayesian quantitative decision-making for two endpoints in rare disease drug development.

Examples

```
des <- design_binary(
  n_t = 12, n_c = 12,
  prior_t = prior_beta(0.5, 0.5), prior_c = prior_beta(0.5, 0.5),
  criterion = criterion_posterior(theta_tv = 0.2, theta_mav = 0.1,
```

```
                                  gamma_go = 0.5, gamma_nogo = 0.8)
)
oc <- qdm_oc(des, data.frame(pi_t = seq(0.2, 0.9, by = 0.05), pi_c = 0.4))
plot(oc)

# Other colors, thinner lines, larger text and the legend on the right,
# with a title added by ggplot2
plot(oc, col_go = "darkgreen", col_nogo = "firebrick", linewidth = 0.6,
     base_size = 14, legend_position = "right") +
  ggplot2::labs(title = "Operating characteristics")

# Two binary endpoints: one heat map per decision, or a single heat map
# of the decision with the highest probability in each cell
prior <- prior_dirichlet(rep(0.25, 4))
des2 <- design_binary2(
  n_t = 6, n_c = 6, prior_t = prior, prior_c = prior,
  criterion = criterion_posterior(theta_tv = c(0.15, 0.15),
                                  theta_mav = c(0.1, 0.1),
                                  gamma_go = 0.6, gamma_nogo = 0.6),
  nMC = 500
)
eff <- expand.grid(theta1 = c(0, 0.2, 0.4), theta2 = c(0, 0.2, 0.4))
oc2 <- qdm_oc(des2, data.frame(pi_t1 = 0.2 + eff$theta1,
                               pi_t2 = 0.2 + eff$theta2, rho_t = 0.3,
                               pi_c1 = 0.2, pi_c2 = 0.2, rho_c = 0.3),
              seed = 1)
plot(oc2)
plot(oc2, overlay = TRUE)
plot(oc2, overlay = TRUE, cell_values = FALSE)
```

`print.qdm_borrow` *Print an external data specification*

Description

Print an external data specification

Usage

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

Arguments

`x`	An object of class `"qdm_borrow"`.
`...`	Passed to `format.qdm_borrow`.

Value

`x`, invisibly.

Examples

```
print(borrow(data_binary(n = 20, y = 8), a0 = 0.5))
```

`print.qdm_boundary` *Print a decision boundary*

Description

Prints the decision intervals of `qdm_boundary` as up to three tables, each with one row per condition (for example, the number of control responders or the treatment group size) and one column per decision. The first table gives the range of the observed quantity (numbers of responders or events, or observed means), the second the corresponding observed response rates or milestone survival probabilities, and the last the observed treatment effects. For example, the entry `"10-20"` in the column `Go` means that 10 to 20 responders lead to a Go decision. The object itself is a data frame with one row per interval (see `qdm_boundary`).

Usage

```
## S3 method for class 'qdm_boundary'
print(x, digits = 4L, n = 20L, ...)
```

Arguments

`x`	An object of class `"qdm_boundary"`.
`digits`	Number of significant digits for observed means, milestone survival probabilities, effects and conditions.
`n`	Maximum number of conditions shown in each table.
`...`	Not used.

Value

`x`, invisibly.

Examples

```
des <- design_binary(
  n_t = 12, n_c = 12,
  prior_t = prior_beta(0.5, 0.5), prior_c = prior_beta(0.5, 0.5),
  criterion = criterion_posterior(theta_tv = 0.2, theta_mav = 0.1,
                                  gamma_go = 0.5, gamma_nogo = 0.8)
)
b <- qdm_boundary(des)
print(b, n = 5)
# The intervals as a data frame
head(as.data.frame(b))
```

print.qdm_calibration *Print a calibration*

Description

Prints the chosen thresholds and the decision probabilities of the design with these thresholds under the two scenarios.

Usage

```
## S3 method for class 'qdm_calibration'
print(x, digits = 4L, ...)
```

Arguments

x	An object of class "qdm_calibration".
digits	Number of significant digits for numeric values.
...	Not used.

Value

x, invisibly.

Examples

```
des <- design_binary(
  n_t = 12, n_c = 12,
  prior_t = prior_beta(0.5, 0.5), prior_c = prior_beta(0.5, 0.5),
  criterion = criterion_posterior(theta_tv = 0.2, theta_mav = 0.1,
                                  gamma_go = 0.5, gamma_nogo = 0.8)
)
print(qdm_calibrate(des, data.frame(pi_t = 0.4, pi_c = 0.4),
                    data.frame(pi_t = 0.7, pi_c = 0.4),
                    target_go = 0.05, target_nogo = 0.2))
```

print.qdm_conduct *Print the conduct of a study*

Description

Print the conduct of a study

Usage

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

Arguments

x	An object of class "qdm_conduct".
...	Passed to format.qdm_conduct.

Value

x, invisibly.

Examples

```
conduct_tte(accrual_time = c(0, 12), cut_events = 20)
```

---

print.qdm_criterion *Print a decision criterion*

Description

Print a decision criterion

Usage

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

Arguments

x	An object of class "qdm_criterion".
...	Passed to format.qdm_criterion.

Value

x, invisibly.

Examples

```
print(criterion_predictive(theta_null = 0, m_t = 50, gamma_go = 0.9,
                           gamma_nogo = 0.7))
```

print.qdm_data *Print a data specification*

Description

Print a data specification

Usage

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

Arguments

x	An object of class "qdm_data".
...	Passed to format.qdm_data.

Value

x, invisibly.

Examples

```
print(data_continuous(n = 12, mean = 3.2, sd = 2))
```

print.qdm_decision *Print a decision*

Description

Print a decision

Usage

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

Arguments

x	An object of class "qdm_decision".
...	Passed to format.qdm_decision.

Value

x, invisibly.

Examples

```
des <- design_binary(
  n_t = 12, n_c = 12,
  prior_t = prior_beta(0.5, 0.5), prior_c = prior_beta(0.5, 0.5),
  criterion = criterion_predictive(theta_null = 0, m_t = 50,
                                   gamma_go = 0.9, gamma_nogo = 0.7)
)
print(qdm_decide(des, data_binary(12, 9), data_binary(12, 4)))
```

---

print.qdm_design *Print a design*

Description

Print a design

Usage

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

Arguments

x	An object of class "qdm_design".
...	Passed to format.qdm_design.

Value

x, invisibly.

Examples

```
des <- design_binary(
  n_t = 12, n_c = 0,
  prior_t = prior_beta(0.5, 0.5), prior_c = prior_beta(0.5, 0.5),
  criterion = criterion_posterior(theta_tv = 0.2, theta_mav = 0.1,
                                  gamma_go = 0.5, gamma_nogo = 0.8),
  hypothetical_c = hypothetical(data = data_binary(n = 12, y = 4))
)
print(des)
```

print.qdm_hypothetical *Print a hypothetical control specification*

Description

Print a hypothetical control specification

Usage

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

Arguments

x	An object of class "qdm_hypothetical".
...	Passed to format.qdm_hypothetical.

Value

x, invisibly.

Examples

```
print(hypothetical(data = data_binary(n = 15, y = 5)))
```

print.qdm_oc *Print operating characteristics*

Description

Print operating characteristics

Usage

```
## S3 method for class 'qdm_oc'
print(x, digits = 4L, ...)
```

Arguments

x	An object of class "qdm_oc".
digits	Number of significant digits for the probabilities.
...	Not used.

Value

x, invisibly.

Examples

```
des <- design_binary(
  n_t = 12, n_c = 12,
  prior_t = prior_beta(0.5, 0.5), prior_c = prior_beta(0.5, 0.5),
  criterion = criterion_posterior(theta_tv = 0.2, theta_mav = 0.1,
                                  gamma_go = 0.5, gamma_nogo = 0.8)
)
print(qdm_oc(des, data.frame(pi_t = c(0.4, 0.6), pi_c = 0.4)))
```

---

print.qdm_prior *Print a prior specification*

Description

Print a prior specification

Usage

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

Arguments

x	An object of class "qdm_prior".
...	Passed to format.qdm_prior.

Value

x, invisibly.

Examples

```
print(prior_nix(mu0 = 0, kappa0 = 1, nu0 = 3, sigma0 = 2))
```

prior_beta *Beta prior for a response rate*

Description

Creates a Beta prior distribution for the response rate of one group in an analysis of a single binary endpoint.

Usage

```
prior_beta(a, b)
```

Arguments

a, b	Positive shape parameters. `prior_beta(0.5, 0.5)` is the Jeffreys prior and `prior_beta(1, 1)` is the uniform prior.

Value

An object of class `c("qdm_prior_beta", "qdm_prior")` with elements `a` and `b`.

See Also

Other priors: `prior_dirichlet()`, `prior_gamma()`, `prior_niw()`, `prior_nix()`, `prior_vague()`

Examples

```
prior_beta(0.5, 0.5)
```

prior_dirichlet *Dirichlet prior for two binary endpoints*

Description

Creates a Dirichlet prior distribution for the four joint response probabilities of two binary endpoints in one group.

Usage

```
prior_dirichlet(alpha)
```

Arguments

alpha	A numeric vector of four positive concentration parameters in the cell order `"00"`, `"01"`, `"10"`, `"11"`.

Details

The four cells are ordered as `"00"`, `"01"`, `"10"` and `"11"`, where the first digit is the response on endpoint 1 and the second digit is the response on endpoint 2 (1 = response, 0 = no response). The marginal response rates are therefore $\pi_1 = p_{10} + p_{11}$ and $\pi_2 = p_{01} + p_{11}$. With `alpha = rep(0.25, 4)`, each marginal response rate has a Beta(0.5, 0.5) prior distribution.

Value

An object of class `c("qdm_prior_dirichlet", "qdm_prior")` with element `alpha`, a named numeric vector of length 4.

See Also

Other priors: `prior_beta()`, `prior_gamma()`, `prior_niw()`, `prior_nix()`, `prior_vague()`

Examples

```
prior_dirichlet(rep(0.25, 4))
```

---

prior_gamma *Gamma prior for a hazard rate*

Description

Creates a Gamma prior distribution for the hazard rate λ of one group in an analysis of a time-to-event endpoint with exponential event times. The milestone survival probability at time t_0 is $S(t_0) = \exp(-\lambda t_0)$.

Usage

```
prior_gamma(shape, rate)
```

Arguments

`shape`, `rate`	Positive shape and rate parameters (the prior mean of the hazard is `shape / rate`).

Details

The prior is equivalent to `shape` events observed during a total observed time of `rate`, so small values give a weakly informative prior: `prior_gamma(0.1, 1)` carries the information of a tenth of an event.

Value

An object of class `c("qdm_prior_gamma", "qdm_prior")` with elements `shape` and `rate`.

See Also

Other priors: prior_beta(), prior_dirichlet(), prior_niw(), prior_nix(), prior_vague()

Examples

```
prior_gamma(0.1, 1)
```

`prior_niw` *Normal-Inverse-Wishart prior for two continuous endpoints*

Description

Creates the conjugate Normal-Inverse-Wishart prior for the mean vector μ and covariance matrix Σ of a bivariate normal endpoint in one group:

$$\mu \mid \Sigma \sim N_2(\mu_0, \Sigma/\kappa_0), \quad \Sigma \sim \text{Inv-Wishart}(\nu_0, \Lambda_0).$$

Usage

```
prior_niw(mu0, kappa0, nu0, Lambda0)
```

Arguments

`mu0`	Numeric vector of length 2, the prior mean of μ.
`kappa0`	Positive prior sample size for μ.
`nu0`	Prior degrees of freedom of the inverse Wishart distribution; must be greater than 1.
`Lambda0`	A 2 x 2 symmetric positive definite scale matrix.

Value

An object of class `c("qdm_prior_niw", "qdm_prior")` with elements `mu0`, `kappa0`, `nu0` and `Lambda0`.

See Also

Other priors: prior_beta(), prior_dirichlet(), prior_gamma(), prior_nix(), prior_vague()

Examples

```
prior_niw(mu0 = c(0, 0), kappa0 = 1, nu0 = 4, Lambda0 = diag(2))
```

prior_nix *Normal-Inverse-chi-squared prior for a continuous endpoint*

Description

Creates the conjugate Normal-Inverse-chi-squared prior for the mean μ and variance σ^2 of a normally distributed endpoint in one group:

$$\mu \mid \sigma^2 \sim N(\mu_0, \sigma^2/\kappa_0), \quad \sigma^2 \sim \text{Inv-}\chi^2(\nu_0, \sigma_0^2).$$

Usage

```
prior_nix(mu0, kappa0, nu0, sigma0)
```

Arguments

`mu0`	Prior mean of μ.
`kappa0`	Positive prior sample size for μ.
`nu0`	Positive prior degrees of freedom for σ^2.
`sigma0`	Positive prior scale of σ, on the standard deviation scale (the scaled inverse chi-squared distribution has scale parameter `sigma0^2`).

Value

An object of class `c("qdm_prior_nix", "qdm_prior")` with elements `mu0`, `kappa0`, `nu0` and `sigma0`.

See Also

Other priors: `prior_beta()`, `prior_dirichlet()`, `prior_gamma()`, `prior_niw()`, `prior_vague()`

Examples

```
prior_nix(mu0 = 0, kappa0 = 1, nu0 = 3, sigma0 = 2)
```

prior_vague *Vague prior for continuous endpoints*

Description

Creates the vague (reference) prior for the parameters of a normally distributed endpoint. For one endpoint the prior density is $p(\mu, \sigma^2) \propto \sigma^{-2}$; for two endpoints it is $p(\mu, \Sigma) \propto |\Sigma|^{-3/2}$. With n observations the posterior distribution of the mean is a t distribution with $n - 1$ degrees of freedom for one endpoint and a bivariate t distribution with $n - 2$ degrees of freedom for two endpoints.

Usage

```
prior_vague()
```

Value

An object of class `c("qdm_prior_vague", "qdm_prior")`.

See Also

Other priors: `prior_beta()`, `prior_dirichlet()`, `prior_gamma()`, `prior_niw()`, `prior_nix()`

Examples

```
prior_vague()
```

`psurvdiff` *Distribution function of the difference of two milestone survival probabilities*

Description

Computes $\Pr(S_t - S_c \leq q)$ (or the upper tail $\Pr(S_t - S_c > q)$) for the milestone survival probabilities $S_k = \exp(-\lambda_k t_0)$, where the hazards have independent Gamma distributions, $\lambda_t \sim \text{Gamma}(a_t, b_t)$ and $\lambda_c \sim \text{Gamma}(a_c, b_c)$ (shape and rate). This is the posterior probability used for a time-to-event endpoint with a concurrent, external or hypothetical control.

Usage

```
psurvdiff(q, milestone, shape_t, rate_t, shape_c, rate_c, lower.tail = TRUE)
```

Arguments

`q`	Numeric vector of quantiles.
`milestone`	Positive milestone time t_0.
`shape_t`, `rate_t`	Positive shape and rate of the treatment hazard.
`shape_c`, `rate_c`	Positive shape and rate of the control hazard.
`lower.tail`	Logical; if `TRUE` (default), probabilities are $\Pr(S_t - S_c \leq q)$, otherwise $\Pr(S_t - S_c > q)$.

Details

The upper tail is computed as

$$\Pr(S_t - S_c > q) = \Pr(S_c \le -q) + \int g_c(\lambda) G_t\left(-\log(q + e^{-\lambda t_0})/t_0\right)\, d\lambda,$$

where g_c is the density of λ_c, G_t is the distribution function of λ_t, the integral runs over the values of λ with $0 < q + e^{-\lambda t_0} < 1$, and the first term is zero for $q \ge 0$. The integral is evaluated by adaptive Gauss-Kronrod quadrature (the QUADPACK routine used by `integrate`), with the range split at quantiles of the control hazard and at the values that correspond to quantiles of the treatment hazard. When the density of λ_c is unbounded at 0 ($a_c < 1$), the integral below its median is evaluated in the variable $u = \lambda^{a_c}$, in which the integrand is bounded. The lower tail is computed from the same formula with the roles of the two groups exchanged, so both tails are accurate near 0.

Value

A numeric vector of probabilities. All arguments except `lower.tail` are recycled to a common length; each must have length 1 or the common length.

See Also

Other distributions: `cellprob_binary2()`, `pbetabinomdiff()`, `pbetadiff()`, `phazardratio()`, `ptdiff()`, `rdirichlet()`

Examples

```
# Posterior probability that the 3-month survival is higher by more than
# 0.1 with 5 events in 40 months of follow-up versus 9 events in 35 months
# (Gamma(0.1, 1) priors)
psurvdiff(0.1, milestone = 3, shape_t = 5.1, rate_t = 41,
          shape_c = 9.1, rate_c = 36, lower.tail = FALSE)
```

---

ptdiff *Distribution function of the difference of two t variables*

Description

Computes $\Pr(T_t - T_c \le q)$ (or the upper tail $\Pr(T_t - T_c > q)$) for independent location-scale t variables $T_k = \mu_k + s_k X_k$, where X_k follows a t distribution with ν_k degrees of freedom. This is the posterior (or posterior predictive) probability used for a single continuous endpoint.

Usage

```
ptdiff(
  q,
  mu_t,
  scale_t,
  df_t,
```

```
  mu_c,
  scale_c,
  df_c,
  method = c("NI", "MM", "MC"),
  nMC = 10000L,
  lower.tail = TRUE
)
```

Arguments

`q`	Numeric vector of quantiles.
`mu_t`, `scale_t`, `df_t`	Location, positive scale and positive degrees of freedom of the treatment group.
`mu_c`, `scale_c`, `df_c`	Location, positive scale and positive degrees of freedom of the control group.
`method`	One of `"NI"` (default), `"MM"` or `"MC"`.
`nMC`	Number of Monte Carlo draws per element for `method = "MC"`.
`lower.tail`	Logical; if `TRUE` (default), probabilities are $\Pr(T_t - T_c \le q)$, otherwise $\Pr(T_t - T_c > q)$.

Details

Three methods are available.

`"NI"` Numerical integration over the whole real line by adaptive Gauss-Kronrod quadrature (the QUADPACK routine used by `integrate`). The integral is taken over the variable of the group with the smaller scale, for example $\int f_c(u) \Pr(T_t > u + q)\, du$ when $s_c \le s_t$. The range is split around the peak of the density and around the change of the other factor, and the parts far from the origin are integrated on a logarithmic scale, so that heavy tails and very different scales are handled. The absolute error is below 10^{-9} in the tests, which include scale ratios from 10^{-9} to 10^{3} and degrees of freedom from 0.5 to `Inf`.

`"MM"` The moment-matching approximation of Yamaguchi, Homma, Takeda and Maruo (submitted, Theorem 1), which replaces the difference by a single t distribution with matched second and fourth moments. It requires $\nu_t > 4$ and $\nu_c > 4$, is evaluated in closed form, and is the fastest method.

`"MC"` Monte Carlo simulation with `nMC` draws for each element, using the random number generator of R.

Degrees of freedom may be `Inf` (normal distributions). For the lower tail, the upper tail of $T_c - T_t$ at $-q$ is computed, so both tails are accurate near 0.

Value

A numeric vector of probabilities. All arguments from `q` to `df_c` are recycled to a common length; each must have length 1 or the common length.

References

Yamaguchi, Y., Homma, G., Takeda, K. and Maruo, K. (submitted). Moment-matching approximation for t-differences in clinical trials with continuous endpoints.

See Also

Other distributions: cellprob_binary2(), pbetabinomdiff(), pbetadiff(), phazardratio(), psurvdiff(), rdirichlet()

Examples

```
ptdiff(1, mu_t = 3, scale_t = 1, df_t = 15, mu_c = 1, scale_c = 1,
       df_c = 15, lower.tail = FALSE)
ptdiff(1, mu_t = 3, scale_t = 1, df_t = 15, mu_c = 1, scale_c = 1,
       df_c = 15, method = "MM", lower.tail = FALSE)
```

---

qdm_boundary *Decision boundary of a design*

Description

Finds the observed data that lead to each decision (Go, NoGo, Gray, or Miss) of a design: for example, how many responders out of n are needed for a Go decision, or how large the observed difference in means must be. Unlike the operating characteristics, the result does not depend on the true parameters. It is a table of decision intervals of an observed quantity (the numbers of responders or events, or an observed mean), one set of intervals for each value of a second quantity (the condition), together with the corresponding observed response rates, milestone survival probabilities and treatment effects. The intervals are printed as tables (print.qdm_boundary) and drawn as the regions of the decisions with the boundary values written on them (plot.qdm_boundary).

Usage

```
qdm_boundary(
  design,
  n_t = NULL,
  mean_t = NULL,
  sd_t = NULL,
  cov_t = NULL,
  events_t = NULL,
  exposure_t = NULL,
  data_c = NULL,
  seed = NULL
)
```

Arguments

design	A design created by design_binary, design_continuous, design_continuous2 or design_tte.
n_t	Numbers of subjects in the treatment group (whole numbers); by default that of the design. Several values are allowed for one binary endpoint without a concurrent control group and for one continuous endpoint. For a time-to-event endpoint, n_t only sets the default numbers of events.
mean_t	For one continuous endpoint, an optional numeric vector: the grid of observed treatment means on which the boundary values are searched. For two continuous endpoints, a list of two numeric vectors: the observed treatment means of endpoint 1 (the conditions) and the grid of observed treatment means of endpoint 2.
sd_t	Observed treatment standard deviations (one continuous endpoint; positive).
cov_t	Observed treatment covariance matrix (two continuous endpoints; a 2 x 2 positive definite matrix).
events_t	Numbers of events of the treatment group (time-to-event endpoint; whole numbers); by default 0:n_t.
exposure_t	Total observed times of the treatment group (time-to-event endpoint; positive).
data_c	Observed data of the concurrent control group, created by data_continuous, data_continuous2 or data_tte; required for those endpoints with a concurrent control group and NULL otherwise.
seed	Random seed for probabilities computed by Monte Carlo simulation, or NULL to use the current random number stream. If given, the previous state of the random number generator is restored on exit.

Details

The observed quantity and the conditions depend on the endpoint:

One binary endpoint The number of responders of the treatment group $y_t = 0, \ldots, n_t$. With a concurrent control group, one set of intervals for each number of control responders $y_c = 0, \ldots, n_c$ (n_t must be a single number; n_c is that of the design). Without one, one set for each sample size in n_t, which shows how the boundary changes with the sample size.

One continuous endpoint The observed treatment mean, for each combination of sd_t and n_t (at most one of them may have more than one value). With a concurrent control group, data_c gives the observed control data. The boundary values are located by bisection between the values of a grid, mean_t or by default a grid of 201 values that covers the thresholds of the criterion.

Two continuous endpoints The observed treatment mean of endpoint 2, located by bisection on the grid mean_t[[2]], for each observed treatment mean of endpoint 1 in mean_t[[1]], with the sample covariance matrix cov_t and n_t subjects (a single number). With a concurrent control group, data_c gives the observed control data.

One time-to-event endpoint The number of events of the treatment group (events_t), for each total observed time in exposure_t. With a concurrent control group, data_c gives the observed control data.

Two binary endpoints are not supported, because their decision depends on the four cell counts and cannot be summarized by one observed quantity.

The decision for each value is made as in `qdm_decide`. For an observed mean, the first and last intervals extend to minus and plus infinity; changes of the decision beyond the grid (whose ends are extended by two values far beyond them) or two changes of the same criterion between two neighboring grid values are not detected. With a conjugate (Normal-Inverse-chi-squared) prior, the posterior scale increases with the distance between the observed mean and the prior mean, so the probability of the Go region need not increase with the observed mean; the intervals then show where it decreases again. Probabilities computed by Monte Carlo simulation (`method = "MC"`, or posterior predictive probabilities of a time-to-event endpoint) make the boundary values approximate.

Value

An object of class `c("qdm_boundary", "data.frame")` with one row per decision interval and the columns

Conditions `n_t`, `n_c` and `y_c` (binary; only `n_t` without a concurrent control group), `n_t` and `sd_t` (continuous), `n_t`, `mean_t1` and `effect1` (two continuous endpoints; the observed treatment mean of endpoint 1 and its difference from the control) or `exposure_t` (time-to-event).

decision The decision (a factor with levels `"Go"`, `"NoGo"`, `"Gray"` and `"Miss"`).

from, to The ends of the interval of the observed quantity: the number of treatment responders or events (both ends included), or the observed treatment mean (of endpoint 2).

rate_from, rate_to Binary endpoint: the observed response rates y_t/n_t at the ends.

surv_from, surv_to Time-to-event endpoint evaluated by the milestone survival: the observed milestone survival probabilities $\exp(-t_0 d_t/E_t)$ at the ends.

effect_from, effect_to The observed treatment effect at the ends: the observed rate, mean or milestone survival minus that of the concurrent control group, or minus the fixed control value or the mean (location) of the distribution of the control group; for the hazard ratio, the observed hazard d_t/E_t divided by the observed control hazard (`NA` if the control group has no events), the fixed control hazard or the mean of the distribution of the control hazard. The values correspond to `from` and `to`, so `effect_from` is larger than `effect_to` when larger values of the observed quantity give smaller effects (for example, more events and the milestone survival).

The attributes `design`, `axes`, `titles`, `increasing`, `discrete` and `range` record the design and the information used by the print and plot methods.

See Also

Other operations: `qdm_calibrate()`, `qdm_decide()`, `qdm_oc()`

Examples

```
# Single-arm study with a fixed control response rate of 0.2: responders
# needed out of 10 to 30 subjects
des <- design_binary(
  n_t = 20, n_c = 0, prior_t = prior_beta(1, 1),
  criterion = criterion_posterior(theta_tv = 0.2, theta_mav = 0.05,
```

```
                                    gamma_go = 0.8, gamma_nogo = 0.8),
  fixed_c = 0.2
)
b <- qdm_boundary(des, n_t = c(10, 20, 30))
b
plot(b)

# Continuous endpoint with a concurrent control group: observed
# differences in means needed for each observed standard deviation
des2 <- design_continuous(
  n_t = 12, n_c = 12,
  prior_t = prior_vague(), prior_c = prior_vague(),
  criterion = criterion_posterior(theta_tv = 30, theta_mav = 15,
                                    gamma_go = 0.6, gamma_nogo = 0.7),
  method = "MM"
)
b2 <- qdm_boundary(des2, sd_t = c(20, 25, 30),
                   data_c = data_continuous(12, 0, 25))
b2
plot(b2)
```

`qdm_calibrate` *Calibration of the decision thresholds*

Description

Chooses the thresholds γ_{go} and γ_{nogo} of the criterion of a design so that the probabilities of a false Go decision and a false NoGo decision are controlled.

Usage

```
qdm_calibrate(
  design,
  scenario_go,
  scenario_nogo,
  target_go,
  target_nogo,
  grid = seq(0.01, 0.99, by = 0.01),
  nsim = 10000L,
  seed = NULL
)
```

Arguments

`design`	A design created by `design_binary`, `design_continuous`, `design_binary2`, `design_continuous2` or `design_tte`. Its thresholds are replaced by the calibrated ones. For two endpoints with a marginal rule (`rule = "and"` or `"or"`), the same threshold is used for both endpoints.

`scenario_go`, `scenario_nogo`
: Data frames with one row each, giving the scenarios used to calibrate γ_{go} and γ_{nogo} (see `qdm_oc` for the columns). For two binary endpoints, the correlations must be feasible.

`target_go`, `target_nogo`
: Upper bounds for the probabilities that the Go criterion is met under `scenario_go` and that the NoGo criterion is met under `scenario_nogo`, strictly between 0 and 1.

`grid`
: Candidate thresholds, strictly between 0 and 1.

`nsim`, `seed`
: As in `qdm_oc`. The probabilities for all values of `grid` are computed from the same outcomes (simulated studies or enumerated outcomes) and the same Monte Carlo draws.

Details

The threshold γ_{go} is the smallest value in `grid` for which the probability that the Go criterion is met under `scenario_go` (typically a scenario with no or a small treatment effect) is at most `target_go`. The threshold γ_{nogo} is the smallest value in `grid` for which the probability that the NoGo criterion is met under `scenario_nogo` (typically a scenario with the expected treatment effect) is at most `target_nogo`. Both probabilities decrease as the threshold increases, so the chosen thresholds are the least strict ones that meet the targets.

The decision probabilities of the design with the chosen thresholds are computed under both scenarios. If $\gamma_{go} + \gamma_{nogo} \leq 1$ (and, for two endpoints with the marginal rule `"or"`, for any thresholds), Miss decisions (both criteria met) can occur, and a warning is given when they have positive probability.

Value

An object of class `"qdm_calibration"`: a list with elements

gamma_go, gamma_nogo The chosen thresholds (`NA` if no value in `grid` meets the target).

prob_go, prob_nogo The probabilities that the Go criterion is met under `scenario_go` and that the NoGo criterion is met under `scenario_nogo`, at the chosen thresholds.

target_go, target_nogo The targets.

curve A data frame with the columns `gamma` (the grid), `prob_go` and `prob_nogo` (the two probabilities at each threshold).

oc A data frame of the decision probabilities with the chosen thresholds under the two scenarios (`NULL` if a threshold was not found).

design The design with the chosen thresholds (`NULL` if a threshold was not found).

scenario_go, scenario_nogo, method The scenarios and the method of computation.

See Also

Other operations: `qdm_boundary()`, `qdm_decide()`, `qdm_oc()`

Examples

```
des <- design_binary(
  n_t = 12, n_c = 12,
  prior_t = prior_beta(0.5, 0.5), prior_c = prior_beta(0.5, 0.5),
  criterion = criterion_posterior(theta_tv = 0.2, theta_mav = 0.1,
                                  gamma_go = 0.5, gamma_nogo = 0.8)
)
cal <- qdm_calibrate(des,
                     scenario_go = data.frame(pi_t = 0.4, pi_c = 0.4),
                     scenario_nogo = data.frame(pi_t = 0.7, pi_c = 0.4),
                     target_go = 0.05, target_nogo = 0.2)
cal
plot(cal)
cal$design
```

`qdm_decide` *Go/NoGo decision for observed data*

Description

Computes the probabilities of the Go and NoGo regions given the observed proof-of-concept data, and makes the decision of a design.

Usage

```
qdm_decide(design, data_t, data_c = NULL, seed = NULL)
```

Arguments

`design`	A design created by `design_binary`, `design_continuous`, `design_binary2`, `design_continuous2` or `design_tte`.
`data_t`	Observed data of the treatment group, created by `data_binary`, `data_continuous`, `data_binary2`, `data_continuous2` or `data_tte` (matching the endpoint of the design).
`data_c`	Observed data of the concurrent control group, created by the same function as `data_t`; `NULL` (the default) for a design without a concurrent control group.
`seed`	Random seed for probabilities computed by Monte Carlo simulation (a continuous design with `method = "MC"`, posterior probabilities for two binary endpoints, or posterior predictive probabilities for a time-to-event endpoint), or `NULL` to use the current random number stream. If given, the previous state of the random number generator is restored on exit. Not used otherwise.

Details

The decision is Go if only the Go criterion is met, NoGo if only the NoGo criterion is met, Gray if neither is met, and Miss if both are met (see `criterion_posterior` and `criterion_predictive`). A probability that is below a threshold by less than 10^{-10} is treated as equal to the threshold, so that ties that arise in symmetric settings are not decided by rounding errors.

The sample sizes of the observed data may differ from those of the design, which are used only by `qdm_oc` and `qdm_calibrate`.

Value

An object of class `"qdm_decision"`: a list with elements

decision `"Go"`, `"NoGo"`, `"Gray"` or `"Miss"`.

prob_go, prob_nogo Probabilities of the Go and NoGo regions (for two endpoints with a marginal rule, one probability per endpoint).

prob_regions For two endpoints, the probabilities of all regions (see `qdm_regions`); `NULL` otherwise.

met_go, met_nogo Whether the Go and NoGo criteria are met.

posterior_t, posterior_c Parameters of the posterior distributions of the treatment and control groups: for a binary endpoint, the Beta parameters `a` and `b`; for a continuous endpoint, the Normal-Inverse-chi-squared parameters `mu`, `kappa`, `nu` and `sigma` (on the standard deviation scale), so that the mean has the distribution $t_\nu(\mu, \sigma^2/\kappa)$. For a hypothetical control, `posterior_c` gives the distribution of the control mean in the same form (see `design_continuous`). For two binary endpoints, the Dirichlet parameters of the cells `"00"`, `"01"`, `"10"` and `"11"`. For two continuous endpoints, the Normal-Inverse-Wishart parameters `mu1`, `mu2`, `kappa`, `nu` and the elements `L11`, `L12` and `L22` of Λ, so that the mean has the distribution $t_{\nu-1}(\mu, \Lambda/(\kappa(\nu - 1)))$. For a time-to-event endpoint, the shape and rate of the Gamma distribution of the hazard. For a fixed control, `posterior_c` is the fixed value (named `value`, or `value1` and `value2` for two endpoints).

design, data_t, data_c The arguments.

See Also

Other operations: `qdm_boundary()`, `qdm_calibrate()`, `qdm_oc()`

Examples

```
des <- design_binary(
  n_t = 12, n_c = 12,
  prior_t = prior_beta(0.5, 0.5), prior_c = prior_beta(0.5, 0.5),
  criterion = criterion_posterior(theta_tv = 0.2, theta_mav = 0.1,
                                  gamma_go = 0.5, gamma_nogo = 0.8)
)
qdm_decide(des, data_t = data_binary(12, 9), data_c = data_binary(12, 4))

# Continuous endpoint
des2 <- design_continuous(
  n_t = 12, n_c = 12,
```

```
  prior_t = prior_vague(), prior_c = prior_vague(),
  criterion = criterion_posterior(theta_tv = 30, theta_mav = 15,
                                  gamma_go = 0.6, gamma_nogo = 0.7)
)
qdm_decide(des2, data_t = data_continuous(12, 18, 25),
           data_c = data_continuous(12, -12, 25))
```

---

qdm_oc *Operating characteristics of a design*

Description

Computes the probabilities of Go, NoGo, Gray and Miss decisions of a design under scenarios for the true parameters.

Usage

```
qdm_oc(design, scenarios, nsim = 10000L, seed = NULL)
```

Arguments

design	A design created by design_binary, design_continuous, design_binary2, design_continuous2 or design_tte.
scenarios	A data frame with one row per scenario. For a binary endpoint the columns pi_t and pi_c give the true response rates of the treatment and control groups. For a continuous endpoint the columns mu_t, sd_t, mu_c and sd_c give the true means and standard deviations. For two binary endpoints the columns pi_t1, pi_t2, rho_t, pi_c1, pi_c2 and rho_c give the true response rates of endpoints 1 and 2 (strictly between 0 and 1) and the correlation of the two endpoints in each group. For two continuous endpoints the columns mu_t1, mu_t2, sd_t1, sd_t2, rho_t, mu_c1, mu_c2, sd_c1, sd_c2 and rho_c give the true means, standard deviations and correlation of the two endpoints in each group (the correlation strictly between -1 and 1). For a time-to-event endpoint the columns median_t and median_c give the true median survival times (positive). The columns of the control group are not needed for a design without a concurrent control group. Other columns, such as labels, are kept in the result.
nsim	Number of simulated studies per scenario for designs whose operating characteristics are computed by simulation.
seed	Random seed for the simulation and for Monte Carlo estimates of probabilities, or NULL to use the current random number stream. If given, the previous state of the random number generator is restored on exit.

Details

For a binary endpoint, all possible outcomes of the proof-of-concept study are enumerated and the probabilities are exact (Kang et al., 2026, Section 4.3); `nsim` and `seed` are not used.

For a continuous endpoint, `nsim` studies are simulated under each scenario. Only the sufficient statistics are simulated: the sample mean $\bar{y} = \mu + \sigma Z/\sqrt{n}$ and the sample standard deviation $s = \sigma\sqrt{X/(n-1)}$, with $Z \sim N(0,1)$ and $X \sim \chi^2_{n-1}$. The same random numbers are used for all scenarios, so the results change smoothly across scenarios. The probabilities have a simulation standard error of at most $0.5/\sqrt{nsim}$ (0.005 for the default `nsim`). For each simulated study, the probabilities of the Go and NoGo regions are computed by the method of the design (see `design_continuous`), of which `method = "MM"` is the fastest. To explore many scenarios or designs, `method = "MM"` or a smaller `nsim` gives results quickly; the chosen design can then be checked with `method = "NI"` and the default `nsim`.

For two binary endpoints, all outcomes are enumerated (Yamaguchi et al., submitted, equation A.5) or `nsim` studies are simulated, as specified by `enumerate` in `design_binary2`. The response rates and correlation of each group are converted to cell probabilities by `cellprob_binary2`; a scenario whose correlation is outside the feasible range for its response rates is not evaluable, and its probabilities are `NA` (with a warning).

For two continuous endpoints, `nsim` studies are simulated under each scenario. Only the sufficient statistics are simulated: the sample means $\bar{y} = \mu + Lz/\sqrt{n}$ and the matrix of sums of squares and cross-products $S = LAA^TL^T$, where L is the Cholesky factor of the covariance matrix of the scenario, z has two independent standard normal elements and A is the Bartlett factor of a Wishart distribution with $n-1$ degrees of freedom. The same random numbers are used for all scenarios.

For a time-to-event endpoint, `nsim` studies are simulated under each scenario as specified by the conduct of the design (enrollment, dropout and analysis time; see `conduct_tte`), with exponential event times whose hazard is $\log(2)/M$ for the median survival time M of the scenario. The same random numbers are used for all scenarios. With a predictive criterion, every simulated study needs `nMC` simulated future studies (see `design_tte`), so the computing time grows with `nsim` times `nMC`.

The decision for each outcome is made as in `qdm_decide`. A warning is given when Miss decisions (both criteria met) have positive probability in some scenario.

Value

An object of class `c("qdm_oc", "data.frame")`: the scenarios with the columns `Go`, `NoGo`, `Gray` and `Miss` added. The attributes `design` and `method` record the design and the method of computation.

References

Kang, R., Yamaguchi, Y. and Han, C. (2026). Bayesian quantitative decision-making framework in cell and gene therapy development for rare diseases. *Journal of Biopharmaceutical Statistics*. doi:10.1080/10543406.2026.2655410

Yamaguchi, Y., Hu, H., Huang, Y., Han, C., Homma, G. and Wang, A. (submitted). Bivariate Bayesian quantitative decision-making for two endpoints in rare disease drug development.

See Also

Other operations: `qdm_boundary()`, `qdm_calibrate()`, `qdm_decide()`

Examples

```
des <- design_binary(
  n_t = 12, n_c = 12,
  prior_t = prior_beta(0.5, 0.5), prior_c = prior_beta(0.5, 0.5),
  criterion = criterion_posterior(theta_tv = 0.2, theta_mav = 0.1,
                                  gamma_go = 0.5, gamma_nogo = 0.8)
)
oc <- qdm_oc(des, data.frame(pi_t = seq(0.2, 0.8, by = 0.1), pi_c = 0.4))
oc
plot(oc)

# Continuous endpoint (moment-matching approximation for speed)
des2 <- design_continuous(
  n_t = 12, n_c = 12,
  prior_t = prior_vague(), prior_c = prior_vague(),
  criterion = criterion_posterior(theta_tv = 30, theta_mav = 15,
                                  gamma_go = 0.6, gamma_nogo = 0.7),
  method = "MM"
)
oc2 <- qdm_oc(des2, data.frame(mu_t = seq(0, 50, by = 10), sd_t = 25,
                               mu_c = 0, sd_c = 25),
              nsim = 2000, seed = 1)
oc2

# Two binary endpoints (small nMC for speed)
des3 <- design_binary2(
  n_t = 6, n_c = 6,
  prior_t = prior_dirichlet(rep(0.25, 4)),
  prior_c = prior_dirichlet(rep(0.25, 4)),
  criterion = criterion_posterior(theta_tv = c(0.15, 0.15),
                                  theta_mav = c(0.10, 0.10),
                                  gamma_go = 0.3, gamma_nogo = 0.4),
  nMC = 2000
)
effects <- expand.grid(theta1 = c(0, 0.15, 0.3), theta2 = c(0, 0.15, 0.3))
oc3 <- qdm_oc(des3, data.frame(pi_t1 = 0.2 + effects$theta1,
                               pi_t2 = 0.2 + effects$theta2, rho_t = 0.3,
                               pi_c1 = 0.2, pi_c2 = 0.2, rho_c = 0.3),
              seed = 1)
oc3
plot(oc3)
```

---

qdm_regions *Region numbers for two endpoints*

Description

Shows how the regions used by the decision criteria for two endpoints are numbered (Yamaguchi et al., submitted, Tables 1 and 2).

Usage

```
qdm_regions(prob = c("posterior", "predictive"), direction = "greater")
```

Arguments

prob	Either "posterior" (nine regions) or "predictive" (four regions).
direction	"greater" (the default) or "less", as in the criterion: a single value for both endpoints or one value per endpoint.

Details

For posterior probability, the target value (TV) and the minimum acceptable value (MAV) of each endpoint divide the plane of (θ_1, θ_2) into nine regions. Columns correspond to endpoint 1 and rows to endpoint 2. For posterior predictive probability, the null values divide the plane into four regions.

For an endpoint on which smaller effects are favorable (direction = "less" in the criterion), the inequalities of that endpoint are reversed, so that region 1 is always the most favorable one: for example, "theta2 < TV2", "TV2 <= theta2 < MAV2" and "theta2 >= MAV2".

Value

A character matrix of region labels (printed without quotes), with the conditions on endpoint 1 as column names and those on endpoint 2 as row names.

References

Yamaguchi, Y., Hu, H., Huang, Y., Han, C., Homma, G. and Wang, A. (submitted). Bivariate Bayesian quantitative decision-making for two endpoints in rare disease drug development.

Examples

```
qdm_regions("posterior")
qdm_regions("predictive")
# Endpoint 2 is a safety endpoint (smaller effects are favorable)
qdm_regions("posterior", direction = c("greater", "less"))
```

---

rdirichlet *Random generation from the Dirichlet distribution*

Description

Draws random vectors from the Dirichlet distribution by normalizing independent Gamma variables, using the random number generator of R.

Usage

```
rdirichlet(n, alpha)
```

Arguments

n	Number of draws (a whole number of at least 1).
alpha	Numeric vector of at least two positive concentration parameters.

Value

A numeric matrix with n rows and length(alpha) columns whose rows sum to 1 (a matrix is returned even when n = 1). Column names are taken from names(alpha).

See Also

Other distributions: cellprob_binary2(), pbetabinomdiff(), pbetadiff(), phazardratio(), psurvdiff(), ptdiff()

Examples

```
set.seed(1)
rdirichlet(3, c(1.5, 0.5, 2, 1))
```

Index

∗ **criteria**
  criterion_posterior, 6
  criterion_predictive, 9
∗ **data**
  data_binary, 11
  data_binary2, 11
  data_continuous, 12
  data_continuous2, 13
  data_tte, 13
∗ **designs**
  design_binary, 14
  design_binary2, 16
  design_continuous, 19
  design_continuous2, 22
  design_tte, 24
∗ **distributions**
  cellprob_binary2, 4
  pbetabinomdiff, 36
  pbetadiff, 37
  phazardratio, 38
  psurvdiff, 56
  ptdiff, 57
  rdirichlet, 69
∗ **operations**
  qdm_boundary, 59
  qdm_calibrate, 62
  qdm_decide, 64
  qdm_oc, 66
∗ **priors**
  prior_beta, 52
  prior_dirichlet, 52
  prior_gamma, 53
  prior_niw, 54
  prior_nix, 55
  prior_vague, 55

borrow, 3, *15*, *17*, *19*, *22*, *25*

cellprob_binary2, 4, *67*
cellprob_binary2(), *36*, *38*, *39*, *57*, *59*, *70*
conduct_tte, 5, *25*, *26*, *67*
criterion_posterior, 6, *15*, *17*, *19*, *22*, *25*, *65*
criterion_posterior(), *10*
criterion_predictive, 9, *15*, *17*, *19*, *20*, *22*, *25*, *26*, *65*
criterion_predictive(), *8*

data_binary, *4*, 11, *15*, *35*, *64*
data_binary(), *12–14*
data_binary2, *4*, 11, *17*, *18*, *35*, *64*
data_binary2(), *11–14*
data_continuous, *4*, 12, *19*, *60*, *64*
data_continuous(), *11–14*
data_continuous2, *4*, 13, *22*, *60*, *64*
data_continuous2(), *11*, *12*, *14*
data_tte, *4*, 13, *25*, *26*, *35*, *60*, *64*
data_tte(), *11–13*
design_binary, 14, *17*, *60*, *62*, *64*, *66*
design_binary(), *18*, *21*, *24*, *27*
design_binary2, 16, *62*, *64*, *66*, *67*
design_binary2(), *16*, *21*, *24*, *27*
design_continuous, 19, *60*, *62*, *64–67*
design_continuous(), *16*, *18*, *24*, *27*
design_continuous2, 22, *60*, *62*, *64*, *66*
design_continuous2(), *16*, *18*, *21*, *27*
design_tte, *5*, *6*, *9*, 24, *60*, *62*, *64*, *66*, *67*
design_tte(), *16*, *18*, *21*, *24*

format.qdm_borrow, 28, *44*
format.qdm_boundary, 28
format.qdm_calibration, 29
format.qdm_conduct, 30, *47*
format.qdm_criterion, 31, *47*
format.qdm_data, 31, *48*
format.qdm_decision, 32, *48*
format.qdm_design, 33, *49*
format.qdm_hypothetical, 33, *50*
format.qdm_prior, 34, *51*

hypothetical, *15*, *17*, *19*, *22*, *25*, 35

integrate, *37*, *57, 58*

pbetabinomdiff, 36
pbetabinomdiff(), *5*, *38, 39*, *57*, *59*, *70*
pbetadiff, 37
pbetadiff(), *5*, *36*, *39*, *57*, *59*, *70*
pf, *38*
phazardratio, *26*, 38
phazardratio(), *5*, *36*, *38*, *57*, *59*, *70*
plot.qdm_boundary, 39, *59*
plot.qdm_calibration, 40
plot.qdm_oc, 42
print.qdm_borrow, 44
print.qdm_boundary, *29*, 45, *59*
print.qdm_calibration, 46
print.qdm_conduct, 46
print.qdm_criterion, 47
print.qdm_data, 48
print.qdm_decision, 48
print.qdm_design, 49
print.qdm_hypothetical, 50
print.qdm_oc, 50
print.qdm_prior, 51
prior_beta, *15*, 52
prior_beta(), *53–56*
prior_dirichlet, *17*, 52
prior_dirichlet(), *52*, *54–56*
prior_gamma, *25*, 53
prior_gamma(), *52–56*
prior_niw, *22, 23*, 54
prior_niw(), *52–56*
prior_nix, *19, 20*, 55
prior_nix(), *52–54*, *56*
prior_vague, *19, 20*, *22, 23*, 55
prior_vague(), *52–55*
psurvdiff, *26*, 56
psurvdiff(), *5*, *36*, *38, 39*, *59*, *70*
ptdiff, *20*, 57
ptdiff(), *5*, *36*, *38, 39*, *57*, *70*

qdm_boundary, *24*, *28*, *39*, *45*, 59
qdm_boundary(), *63*, *65*, *67*
qdm_calibrate, *14*, *16*, *18, 19*, *22*, *24*, *26*, 62, *65*
qdm_calibrate(), *61*, *65*, *67*
qdm_decide, *14*, *16*, *19*, *22*, *24*, *61*, 64, *67*
qdm_decide(), *61*, *63*, *67*
qdm_oc, *14*, *16*, *18, 19*, *22–24*, *26*, *63*, *65*, 66
qdm_oc(), *61*, *63*, *65*
qdm_regions, *7, 8*, *10*, *65*, 68

rdirichlet, 69
rdirichlet(), *5*, *36*, *38, 39*, *57*, *59*