Package ‘SAMprior’

May 7, 2026

Type Package

Title Self-Adapting Mixture (SAM) Priors

Version 3.0.0

Maintainer Peng Yang `<pyang7@mdanderson.org>`

Description Implementation of the SAM prior and generation of its operating characteristics for dynamically borrowing information from historical data. For details, please refer to Yang et al. (2023) <doi:10.1111/biom.13927>.

Depends R (>= 3.5.0), RBesT, MatchIt

Imports Metrics, assertthat, checkmate, ggplot2

Suggests rmarkdown, knitr, testthat (>= 2.0.0), foreach, purrr, rstanarm (>= 2.17.2), scales, tools, broom, tidyr, parallel

VignetteBuilder knitr

License GPL (>= 3)

Encoding UTF-8

LazyData true

RoxygenNote 7.3.3

NeedsCompilation no

Author Peng Yang [aut, cre] (ORCID: <`https://orcid.org/0000-0002-0630-9506`>), Ying Yuan [aut] (ORCID: <`https://orcid.org/0000-0003-3163-480X`>)

Repository CRAN

Date/Publication 2026-04-27 22:10:02 UTC

Contents

`calibrate_cutoff_2arm` *Calibrate Posterior Probability Cutoff for a Two-Arm Comparative Trial*

Description

The `calibrate_cutoff_2arm` function is designed to calibrate the posterior probability cutoff for a two-arm comparative trial under the specified endpoint type and borrowing strategy: self-adapting mixture prior (SAM), robust MAP prior with fixed weight (rMAP), or non-informative prior (NP).

Usage

```
calibrate_cutoff_2arm(
  if.prior,
  nf.prior,
  prior.t = nf.prior,
  target = 0.05,
  n.t,
  n,
  theta.t = NULL,
  theta = NULL,
  delta,
  method = c("SAM", "rMAP", "NP"),
  alternative = c("greater", "less"),
  margin = 0,
  weight_rMAP = 0.5,
  method.w = "LRT",
  prior.odds = 1,
  interval = c(0.5, 0.999),
  rel.tol = 1e-05,
  oc_rel.tol = 1e-06,
  ...
)

## S3 method for class 'betaMix'
calibrate_cutoff_2arm(
```

```
  if.prior,
  nf.prior,
  prior.t = nf.prior,
  target = 0.05,
  n.t,
  n,
  theta.t = NULL,
  theta = NULL,
  delta,
  method = c("SAM", "rMAP", "NP"),
  alternative = c("greater", "less"),
  margin = 0,
  weight_rMAP = 0.5,
  method.w = "LRT",
  prior.odds = 1,
  interval = c(0.5, 0.999),
  rel.tol = 1e-05,
  oc_rel.tol = 1e-08,
  ...
)

## S3 method for class 'normMix'
calibrate_cutoff_2arm(
  if.prior,
  nf.prior,
  prior.t = nf.prior,
  target = 0.05,
  n.t,
  n,
  theta.t = NULL,
  theta = NULL,
  delta,
  method = c("SAM", "rMAP", "NP"),
  alternative = c("greater", "less"),
  margin = 0,
  weight_rMAP = 0.5,
  method.w = "LRT",
  prior.odds = 1,
  interval = c(0.5, 0.999),
  rel.tol = 1e-05,
  oc_rel.tol = 1e-06,
  sigma.t,
  sigma,
  n_sd_int = 8,
  ...
)
```

Arguments

`if.prior`	Informative prior constructed from historical data, represented (approximately) as a mixture of conjugate distributions.
`nf.prior`	Non-informative prior used for the mixture and as the robustifying component for the control arm prior.
`prior.t`	Prior used for the treatment arm. If missing, the default value is set to be `nf.prior`.
`target`	Target rejection probability under the calibration scenario, typically the desired type I error rate when the calibration scenario corresponds to the null hypothesis.
`n.t`	Sample size for the treatment arm.
`n`	Sample size for the control arm.
`theta.t`	The response rate (binary endpoints) or mean (continuous endpoints) for the treatment arm under the calibration scenario.
`theta`	The response rate (binary endpoints) or mean (continuous endpoints) for the control arm under the calibration scenario.
`delta`	Clinically significant difference used for the SAM prior. This argument is only used when `method = "SAM"`.
`method`	Borrowing strategy for the control arm. Must be one of `"SAM"`, `"rMAP"`, or `"NP"`.
`alternative`	Direction of the posterior decision. Must be one of `"greater"` (for superiority) or `"less"` (for inferiority).
`margin`	Clinical margin. Must be non-negative. Default is `0`.
`weight_rMAP`	Weight assigned to the informative prior component ($0 \leq$ `weight_rMAP` ≤ 1) for the robust MAP prior. This argument is only used when `method = "rMAP"`. The default value is 0.5.
`method.w`	Methods used to determine the mixture weight for SAM priors. The default method is "LRT" (Likelihood Ratio Test), the alternative option is "PPR" (Posterior Probability Ratio). See `SAM_weight` for more details.
`prior.odds`	The prior probability of H_0 being true compared to the prior probability of H_1 being true using PPR method. The default value is 1. See `SAM_weight` for more details.
`interval`	Search interval for the posterior probability cutoff. The default is from 0.5 to 0.999.
`rel.tol`	Tolerance passed to `uniroot`.
`oc_rel.tol`	Relative tolerance passed to scenario-level calculations. For continuous endpoints this controls numerical integration in the operating-characteristic engine; for binary endpoints this controls numerical integration in posterior probability calculations.
`...`	Additional arguments required by the endpoint-specific method. For continuous endpoints, this includes `sigma.t`, `sigma`, and optionally `n_sd_int`.
`sigma.t`	Known sampling standard deviation in the treatment arm.
`sigma`	Known sampling standard deviation in the control arm.
`n_sd_int`	Half-width of the numerical integration region for each arm, expressed as a multiple of the corresponding standard error.

Details

For binary endpoints, the function dispatches to `calibrate_cutoff_bin_2arm`. For continuous endpoints, it dispatches to `calibrate_cutoff_cont_2arm`.

The function calibrates the posterior probability cutoff C for a two-arm comparative trial so that the repeated-sampling rejection probability under the specified calibration scenario equals `target`. Inference is based on the posterior probability of the treatment effect $\theta_t - \theta$, where θ_t and θ denote the treatment-arm and control-arm parameters, respectively.

For `alternative = "greater"`, the posterior decision rule is based on

$$P(\theta_t - \theta > \Delta \mid D) > C,$$

where Δ denotes the clinical margin and D denotes the observed data. Under the default settings, calibration is performed under the null scenario $\theta_t = \theta = \theta_h$, where $\theta_h = E(\texttt{if.prior})$ is the posterior mean of the informative prior constructed from historical data. This corresponds to a superiority test when $\Delta = 0$.

For `alternative = "less"`, the posterior decision rule is based on

$$P(\theta_t - \theta < -\Delta \mid D) > C.$$

Calibration is again performed under the null scenario $\theta_t = \theta = \theta_h$, where $\theta_h = E(\texttt{if.prior})$. This corresponds to an inferiority test when $\Delta = 0$.

This function acts as a generic front-end and dispatches according to the class of `if.prior`. For binary endpoints, it calls `calibrate_cutoff_bin_2arm`. For continuous endpoints, it calls `calibrate_cutoff_cont_2arm`.

Value

A list with the following elements:

cutoff Calibrated posterior probability cutoff.

objective Value of the root-finding objective at the solution.

target Target rejection probability.

method Borrowing method used.

alternative Direction of the posterior decision.

margin Clinical margin used for inference.

theta True control-arm parameter under the calibration scenario.

theta.t True treatment-arm parameter under the calibration scenario.

interval Search interval used for calibration.

Methods (by class)

- `calibrate_cutoff_2arm(betaMix)`: Calibrate the posterior probability cutoff for a two-arm binary endpoint.
- `calibrate_cutoff_2arm(normMix)`: Calibrate the posterior probability cutoff for a two-arm continuous endpoint.

See Also

`calibrate_cutoff_bin_2arm`, `calibrate_cutoff_cont_2arm`

Examples

```
## Example: calibrate the posterior probability cutoff for a two-arm
## binary trial using a SAM prior for the control arm

## Informative prior constructed from historical data
if.prior <- mixbeta(c(1, 20, 40))

## Sample sizes
n.t <- 100
n <- 50

## Calibrate the posterior probability cutoff
PPC <- calibrate_cutoff_2arm(
  if.prior = if.prior,             ## Informative prior from historical data
  nf.prior = mixbeta(c(1, 1, 1)),  ## Non-informative prior for SAM prior
  prior.t = mixbeta(c(1, 1, 1)),   ## Prior for treatment arm
  target = 0.05,                   ## Targeted type I error rate
  n.t = n.t,                       ## Sample size for treatment arm
  n = n,                           ## Sample size for control arm
  theta.t = summary(if.prior)["mean"],  ## True treatment-arm response rate
  theta = summary(if.prior)["mean"],    ## True control-arm response rate
  method = "SAM",                  ## Borrowing method
  delta = 0.2,                     ## Clinically significant difference
  alternative = "greater",         ## Superiority test
  margin = 0                       ## Clinical margin
)

PPC
```

---

calibrate_cutoff_bin_2arm

Calibrate Posterior Probability Cutoff for a Two-Arm Comparative Trial with Binary Endpoint

---

Description

The `calibrate_cutoff_bin_2arm` function is designed to calibrate the posterior probability cutoff for a two-arm comparative trial with a binary endpoint under one borrowing strategy: self-adapting mixture prior (SAM), robust MAP prior with fixed weight (rMAP), or non-informative prior (NP).

Usage

```
calibrate_cutoff_bin_2arm(
  if.prior,
  nf.prior,
  prior.t = nf.prior,
  target = 0.05,
  n.t,
  n,
  theta.t = NULL,
  theta = NULL,
  delta,
  method = c("SAM", "rMAP", "NP"),
  alternative = c("greater", "less"),
  margin = 0,
  weight_rMAP = 0.5,
  method.w = "LRT",
  prior.odds = 1,
  interval = c(0.5, 0.999),
  rel.tol = 1e-05,
  oc_rel.tol = 1e-08
)
```

Arguments

`if.prior`	Informative prior constructed based on historical data for the control arm, represented (approximately) as a beta mixture prior.
`nf.prior`	Non-informative prior used for the mixture.
`prior.t`	Prior used for the treatment arm. If missing, the default value is set to be `nf.prior`.
`target`	Target rejection probability under the calibration scenario, typically the desired type I error rate when the calibration scenario corresponds to the null hypothesis.
`n.t`	Sample size for the treatment arm.
`n`	Sample size for the control arm.
`theta.t`	The response rate for the treatment arm under the calibration scenario.
`theta`	The response rate for the control arm under the calibration scenario.
`delta`	Clinically significant difference used for the SAM prior. This argument is only used when `method = "SAM"`.
`method`	Borrowing strategy for the control arm. Must be one of `"SAM"`, `"rMAP"`, or `"NP"`.
`alternative`	Direction of the posterior decision. Must be one of `"greater"` (for superiority) or `"less"` (for inferiority).
`margin`	Clinical margin. Must be non-negative. Default is `0`.
`weight_rMAP`	Weight assigned to the informative prior component ($0 \leq$ `weight_rMAP` ≤ 1) for the robust MAP prior. This argument is only used when `method = "rMAP"`. The default value is 0.5.

`method.w`	Methods used to determine the mixture weight for SAM priors. The default method is "LRT" (Likelihood Ratio Test), the alternative option is "PPR" (Posterior Probability Ratio). See `SAM_weight` for more details.
`prior.odds`	The prior probability of H_0 being true compared to the prior probability of H_1 being true using PPR method. The default value is 1. See `SAM_weight` for more details.
`interval`	Search interval for the posterior probability cutoff. The default is from 0.5 to 0.999.
`rel.tol`	Tolerance passed to `uniroot`.
`oc_rel.tol`	Relative tolerance passed to scenario-level calculations.

Details

The calibrated cutoff is chosen so that the repeated-sampling rejection probability under a specified scenario equals the target value.

The function solves for the posterior probability cutoff such that the scenario-level rejection probability returned by `eval_scenario_bin_2arm` matches `target`. Under null scenarios, this corresponds to calibration of the type I error.

Value

A list with the following elements:

cutoff Calibrated posterior probability cutoff.

objective Value of the root-finding objective at the solution.

target Target rejection probability.

method Borrowing method used.

alternative Direction of the posterior decision.

margin Clinical margin used for inference.

theta True control-arm response rate under the calibration scenario.

theta.t True treatment-arm response rate under the calibration scenario.

interval Search interval used for calibration.

`calibrate_cutoff_cont_2arm` *Calibrate Posterior Probability Cutoff for a Two-Arm Comparative Trial with Continuous Endpoint*

Description

The `calibrate_cutoff_cont_2arm` function is designed to calibrate the posterior probability cutoff for a two-arm comparative trial with a continuous endpoint under one borrowing strategy: self-adapting mixture prior (SAM), robust MAP prior with fixed weight (rMAP), or non-informative prior (NP).

Usage

```
calibrate_cutoff_cont_2arm(
  if.prior,
  nf.prior,
  prior.t = nf.prior,
  target = 0.05,
  n.t,
  n,
  sigma.t,
  sigma,
  theta.t = NULL,
  theta = NULL,
  delta,
  method = c("SAM", "rMAP", "NP"),
  alternative = c("greater", "less"),
  margin = 0,
  weight_rMAP = 0.5,
  method.w = "LRT",
  prior.odds = 1,
  interval = c(0.5, 0.999),
  rel.tol = 1e-05,
  oc_rel.tol = 1e-06,
  n_sd_int = 8
)
```

Arguments

`if.prior`	Informative prior constructed based on historical data for the control arm, represented (approximately) as a normal mixture prior.
`nf.prior`	Non-informative prior used for the mixture.
`prior.t`	Prior used for the treatment arm. If missing, the default value is set to be `nf.prior`.
`target`	Target rejection probability under the calibration scenario, typically the desired type I error rate when the calibration scenario corresponds to the null hypothesis.
`n.t`	Sample size for the treatment arm.
`n`	Sample size for the control arm.
`sigma.t`	Known sampling standard deviation for the treatment arm.
`sigma`	Known sampling standard deviation for the control arm.
`theta.t`	The mean for the treatment arm under the calibration scenario.
`theta`	The mean for the control arm under the calibration scenario.
`delta`	Clinically significant difference used for the SAM prior. This argument is only used when `method = "SAM"`.
`method`	Borrowing strategy for the control arm. Must be one of `"SAM"`, `"rMAP"`, or `"NP"`.
`alternative`	Direction of the posterior decision. Must be one of `"greater"` (for superiority) or `"less"` (for inferiority).

`margin`	Clinical margin. Must be non-negative. Default is `0`.
`weight_rMAP`	Weight assigned to the informative prior component ($0 \leq$ `weight_rMAP` ≤ 1) for the robust MAP prior. This argument is only used when `method = "rMAP"`. The default value is 0.5.
`method.w`	Methods used to determine the mixture weight for SAM priors. The default method is "LRT" (Likelihood Ratio Test), the alternative option is "PPR" (Posterior Probability Ratio). See `SAM_weight` for more details.
`prior.odds`	The prior probability of H_0 being true compared to the prior probability of H_1 being true using PPR method. The default value is 1. See `SAM_weight` for more details.
`interval`	Search interval for the posterior probability cutoff. The default is from 0.5 to 0.999.
`rel.tol`	Tolerance passed to `uniroot`.
`oc_rel.tol`	Relative tolerance passed to scenario-level numerical integration.
`n_sd_int`	Half-width of the numerical integration region for each arm, expressed as a multiple of the corresponding standard error.

Details

The calibrated cutoff is chosen so that the repeated-sampling rejection probability under a specified scenario equals the target value.

The function solves for the posterior probability cutoff such that the scenario-level rejection probability returned by `eval_scenario_cont_2arm` matches `target`. Under null scenarios, this corresponds to calibration of the type I error.

Value

A list with the following elements:

cutoff Calibrated posterior probability cutoff.

objective Value of the root-finding objective at the solution.

target Target rejection probability.

method Borrowing method used.

alternative Direction of the posterior decision.

margin Clinical margin used for inference.

theta True control-arm mean under the calibration scenario.

theta.t True treatment-arm mean under the calibration scenario.

interval Search interval used for calibration.

`eval_oc_bin_2arm` *Evaluate Multiple Scenarios for a Two-Arm Comparative Trial with Binary Endpoint*

Description

The `eval_oc_bin_2arm` function is designed to evaluate repeated-sampling operating characteristics for multiple scenarios in a two-arm comparative trial with a binary endpoint under one or more borrowing strategies: self-adapting mixture prior (SAM), robust MAP prior with fixed weight (rMAP), or non-informative prior (NP).

Usage

```
eval_oc_bin_2arm(
  if.prior,
  nf.prior,
  prior.t = nf.prior,
  theta,
  theta.t,
  n.t,
  n,
  delta,
  method = c("SAM", "rMAP", "NP"),
  cutoff,
  alternative = c("greater", "less"),
  margin = 0,
  weight_rMAP = 0.5,
  method.w = "LRT",
  prior.odds = 1,
  rel.tol = 1e-08
)
```

Arguments

`if.prior`	Informative prior constructed based on historical data for the control arm, represented (approximately) as a beta mixture prior.
`nf.prior`	Non-informative prior used as the robustifying component for the control arm prior.
`prior.t`	Prior used for the treatment arm. If missing, the default value is set to be `nf.prior`.
`theta`	A vector of true control arm response rates.
`theta.t`	A vector of true treatment arm response rates.
`n.t`	Sample size for the treatment arm.
`n`	Sample size for the control arm.

`delta`	Clinically significant difference used for the SAM prior. This argument is only used when `method = "SAM"`.
`method`	Borrowing methods to evaluate. Any subset of `c("SAM", "rMAP", "NP")`.
`cutoff`	Posterior probability cutoff specification. Either a single numeric value applied to all methods, or a named numeric vector/list with method-specific cutoffs, for example `c(SAM = 0.94, rMAP = 0.96, NP = 0.95)`.
`alternative`	Direction of the posterior decision. Must be one of `"greater"` (for superiority) or `"less"` (for inferiority).
`margin`	Clinical margin. Must be a non-negative scalar. The default value is `0`.
`weight_rMAP`	Weight assigned to the informative prior component ($0 \leq$ `weight_rMAP` ≤ 1) for the robust MAP prior. This argument is only used when `method = "rMAP"`. The default value is 0.5.
`method.w`	Methods used to determine the mixture weight for SAM priors. The default method is "LRT" (Likelihood Ratio Test), the alternative option is "PPR" (Posterior Probability Ratio). See `SAM_weight` for more details.
`prior.odds`	The prior probability of H_0 being true compared to the prior probability of H_1 being true using PPR method. The default value is 1. See `SAM_weight` for more details.
`rel.tol`	Relative tolerance passed to scenario-level numerical integration used in `post_summary_bin_2arm`.

Details

For each scenario, the function computes the repeated-sampling rejection probability, bias, RMSE, and mean borrowing weight using `eval_scenario_bin_2arm`.

The vectors `theta` and `theta.t` must have the same length. Each pair `(theta[i], theta.t[i])` defines one scenario.

The `cutoff` argument may be common across methods or method-specific. This is useful when each borrowing method is calibrated separately before operating characteristics are evaluated.

Value

A data frame with one row per scenario-method combination and columns:

scenario Scenario index.

theta True control arm response rate.

theta.t True treatment arm response rate.

delta_true True treatment effect, $\tau = \theta_t - \theta$.

method Borrowing method used.

alternative Direction of the posterior decision.

cutoff Posterior probability cutoff used.

margin Clinical margin used for inference.

reject_prob Repeated-sampling rejection probability.

bias Bias of the posterior mean estimator of θ.

rmse Root mean squared error of the posterior mean estimator of θ.

mean_weight Average borrowing weight under the specified method.

`eval_oc_cont_2arm` *Evaluate Multiple Scenarios for a Two-Arm Comparative Trial with Continuous Endpoint*

Description

The `eval_oc_cont_2arm` function is designed to evaluate repeated-sampling operating characteristics for multiple scenarios in a two-arm comparative trial with a continuous endpoint under one or more borrowing strategies: self-adapting mixture prior (SAM), robust MAP prior with fixed weight (rMAP), or non-informative prior (NP).

Usage

```
eval_oc_cont_2arm(
  if.prior,
  nf.prior,
  prior.t = nf.prior,
  theta,
  theta.t,
  n.t,
  n,
  sigma.t,
  sigma,
  delta,
  method = c("SAM", "rMAP", "NP"),
  cutoff,
  alternative = c("greater", "less"),
  margin = 0,
  weight_rMAP = 0.5,
  method.w = "LRT",
  prior.odds = 1,
  rel.tol = 1e-06,
  n_sd_int = 8
)
```

Arguments

`if.prior`	Informative prior constructed based on historical data for the control arm, represented (approximately) as a normal mixture prior.
`nf.prior`	Non-informative prior used as the robustifying component for the control arm prior.
`prior.t`	Prior used for the treatment arm. If missing, the default value is set to be `nf.prior`.
`theta`	A vector of true control arm means.
`theta.t`	A vector of true treatment arm means.

`n.t`	Sample size for the treatment arm.
`n`	Sample size for the control arm.
`sigma.t`	Known sampling standard deviation in the treatment arm.
`sigma`	Known sampling standard deviation in the control arm.
`delta`	Clinically significant difference used for the SAM prior. This argument is only used when `method = "SAM"`.
`method`	Borrowing methods to evaluate. Any subset of `c("SAM", "rMAP", "NP")`.
`cutoff`	Posterior probability cutoff specification. Either a single numeric value applied to all methods, or a named numeric vector/list with method-specific cutoffs, for example `c(SAM = 0.94, rMAP = 0.96, NP = 0.95)`.
`alternative`	Direction of the posterior decision. Must be one of `"greater"` (for superiority) or `"less"` (for inferiority).
`margin`	Clinical margin. Must be a non-negative scalar. The default value is `0`.
`weight_rMAP`	Weight assigned to the informative prior component ($0 \leq$ `weight_rMAP` ≤ 1) for the robust MAP prior. This argument is only used when `method = "rMAP"`. The default value is 0.5.
`method.w`	Methods used to determine the mixture weight for SAM priors. The default method is "LRT" (Likelihood Ratio Test), the alternative option is "PPR" (Posterior Probability Ratio). See `SAM_weight` for more details.
`prior.odds`	The prior probability of H_0 being true compared to the prior probability of H_1 being true using PPR method. The default value is 1. See `SAM_weight` for more details.
`rel.tol`	Relative tolerance passed to scenario-level numerical integration.
`n_sd_int`	Half-width of the numerical integration region for each arm, expressed as a multiple of the corresponding standard error.

Details

For each scenario, the function computes the repeated-sampling rejection probability, bias, RMSE, and mean borrowing weight using `eval_scenario_cont_2arm`.

The vectors `theta` and `theta.t` must have the same length. Each pair `(theta[i], theta.t[i])` defines one scenario.

The `cutoff` argument may be common across methods or method-specific. This is useful when each borrowing method is calibrated separately before operating characteristics are evaluated.

Value

A data frame with one row per scenario-method combination and columns:

scenario Scenario index.

theta True control arm mean.

theta.t True treatment arm mean.

delta_true True treatment effect, $\tau = \theta_t - \theta$.

method Borrowing method used.

alternative Direction of the posterior decision.

cutoff Posterior probability cutoff used.

margin Clinical margin used for inference.

reject_prob Repeated-sampling rejection probability.

bias Bias of the posterior mean estimator of θ.

rmse Root mean squared error of the posterior mean estimator of θ.

mean_weight Average borrowing weight under the specified method.

`eval_scenario_bin_2arm`

Evaluate One Scenario for a Two-Arm Comparative Trial with Binary Endpoint

---

Description

The `eval_scenario_bin_2arm` function is designed to evaluate repeated-sampling operating characteristics for a two-arm comparative trial with a binary endpoint under one borrowing strategy: self-adapting mixture prior (SAM), robust MAP prior with fixed weight (rMAP), or non-informative prior (NP).

Usage

```
eval_scenario_bin_2arm(
  if.prior,
  nf.prior,
  prior.t = nf.prior,
  n.t,
  n,
  theta.t,
  theta,
  cutoff,
  delta,
  method = c("SAM", "rMAP", "NP"),
  alternative = c("greater", "less"),
  margin = 0,
  weight_rMAP = 0.5,
  method.w = "LRT",
  prior.odds = 1,
  rel.tol = 1e-08
)
```

Arguments

`if.prior`	Informative prior constructed based on historical data for the control arm, represented (approximately) as a beta mixture prior.
`nf.prior`	Non-informative prior used as the robustifying component for the control arm prior.
`prior.t`	Prior used for the treatment arm. If missing, the default value is set to be `nf.prior`.
`n.t`	Number of subjects in the treatment arm.
`n`	Number of subjects in the control arm.
`theta.t`	True treatment arm response rate.
`theta`	True control arm response rate.
`cutoff`	Posterior probability cutoff used for decision making. Rejection occurs if the posterior probability exceeds `cutoff`.
`delta`	Clinically significant difference used for the SAM prior. This argument is only used when `method = "SAM"`.
`method`	Borrowing strategy for the control arm. Must be one of `"SAM"`, `"rMAP"`, or `"NP"`.
`alternative`	Direction of the posterior decision. Must be one of `"greater"` (for superiority) or `"less"` (for inferiority).
`margin`	Clinical margin. Must be a non-negative scalar. The default value is `0`.
`weight_rMAP`	Weight assigned to the informative prior component ($0 \leq$ `weight_rMAP` ≤ 1) for the robust MAP prior. This argument is only used when `method = "rMAP"`. The default value is 0.5.
`method.w`	Methods used to determine the mixture weight for SAM priors. The default method is "LRT" (Likelihood Ratio Test), the alternative option is "PPR" (Posterior Probability Ratio). See `SAM_weight` for more details.
`prior.odds`	The prior probability of H_0 being true compared to the prior probability of H_1 being true using PPR method. The default value is 1. See `SAM_weight` for more details.
`rel.tol`	Relative tolerance for numerical integration used in posterior probability calculations.

Details

The treatment effect is defined as $\tau = \theta_t - \theta$, where θ_t and θ denote the true response rates in the treatment and control arms, respectively.

For a given true scenario (θ_t, θ), this function computes the repeated-sampling rejection probability, bias, root mean squared error (RMSE), and mean borrowing weight. The rejection probability is accelerated by exploiting monotonicity of the posterior decision in the treatment-arm response count for each fixed control-arm response count.

Value

A one-row data frame with the following columns:

theta True control arm response rate.

theta.t True treatment arm response rate.

delta_true True treatment effect, $\tau = \theta_t - \theta$.

method Borrowing method used.

alternative Direction of the posterior decision.

cutoff Posterior probability cutoff used for decision making.

margin Clinical margin used for inference.

reject_prob Repeated-sampling rejection probability.

bias Bias of the posterior mean estimator of θ.

rmse Root mean squared error of the posterior mean estimator of θ.

mean_weight Average borrowing weight under the specified method.

`eval_scenario_cont_2arm`

Evaluate One Scenario for a Two-Arm Comparative Trial with Continuous Endpoint

Description

The `eval_scenario_cont_2arm` function is designed to evaluate repeated-sampling operating characteristics for a two-arm comparative trial with a continuous endpoint under one borrowing strategy: self-adapting mixture prior (SAM), robust MAP prior with fixed weight (rMAP), or non-informative prior (NP).

Usage

```
eval_scenario_cont_2arm(
  if.prior,
  nf.prior,
  prior.t = nf.prior,
  n.t,
  n,
  sigma.t,
  sigma,
  theta.t,
  theta,
  cutoff,
  delta,
  method = c("SAM", "rMAP", "NP"),
  alternative = c("greater", "less"),
  margin = 0,
```

```
    weight_rMAP = 0.5,
    method.w = "LRT",
    prior.odds = 1,
    rel.tol = 1e-06,
    n_sd_int = 8
  )
```

Arguments

`if.prior`	Informative prior constructed based on historical data for the control arm, represented (approximately) as a normal mixture prior.
`nf.prior`	Non-informative prior used as the robustifying component for the control arm prior.
`prior.t`	Prior used for the treatment arm. If missing, the default value is set to be `nf.prior`.
`n.t`	Number of subjects in the treatment arm.
`n`	Number of subjects in the control arm.
`sigma.t`	Known sampling standard deviation in the treatment arm.
`sigma`	Known sampling standard deviation in the control arm.
`theta.t`	True treatment arm mean.
`theta`	True control arm mean.
`cutoff`	Posterior probability cutoff used for decision making. Rejection occurs if the posterior probability exceeds `cutoff`.
`delta`	Clinically significant difference used for the SAM prior. This argument is only used when `method = "SAM"`.
`method`	Borrowing strategy for the control arm. Must be one of `"SAM"`, `"rMAP"`, or `"NP"`.
`alternative`	Direction of the posterior decision. Must be one of `"greater"` (for superiority) or `"less"` (for inferiority).
`margin`	Clinical margin. Must be a non-negative scalar. The default value is `0`.
`weight_rMAP`	Weight assigned to the informative prior component ($0 \leq$ `weight_rMAP` ≤ 1) for the robust MAP prior. This argument is only used when `method = "rMAP"`. The default value is 0.5.
`method.w`	Methods used to determine the mixture weight for SAM priors. The default method is "LRT" (Likelihood Ratio Test), the alternative option is "PPR" (Posterior Probability Ratio). See `SAM_weight` for more details.
`prior.odds`	The prior probability of H_0 being true compared to the prior probability of H_1 being true using PPR method. The default value is 1. See `SAM_weight` for more details.
`rel.tol`	Relative tolerance passed to numerical integration.
`n_sd_int`	Half-width of the numerical integration region for each arm, expressed as a multiple of the corresponding standard error.

Details

The treatment effect is defined as $\tau = \theta_t - \theta$, where θ_t and θ denote the true means in the treatment and control arms, respectively.

For a given true scenario (θ_t, θ), this function computes the repeated-sampling rejection probability, bias, root mean squared error (RMSE), and mean borrowing weight using one-dimensional numerical integration.

The rejection probability is computed by reducing the repeated-sampling decision rule to a one-dimensional integral over the control-arm sample mean.

Bias and RMSE are evaluated for the posterior mean estimator of the control arm mean θ. Both are computed from one-dimensional first and second moments of the control-arm posterior mean. The mean borrowing weight is computed by one-dimensional integration over the control-arm sample mean.

Under null scenarios, `reject_prob` corresponds to type I error. Under alternative scenarios, it corresponds to power.

Value

A one-row data frame with the following columns:

theta	True control arm mean.
theta.t	True treatment arm mean.
delta_true	True treatment effect, $\tau = \theta_t - \theta$.
method	Borrowing method used.
alternative	Direction of the posterior decision.
cutoff	Posterior probability cutoff used for decision making.
margin	Clinical margin used for inference.
reject_prob	Repeated-sampling rejection probability.
bias	Bias of the posterior mean estimator of θ.
rmse	Root mean squared error of the posterior mean estimator of θ.
mean_weight	Average borrowing weight under the specified method.

`get_OC` *Generating Operating Characteristics of SAM Priors Using Analytical Engines*

Description

The `get_OC` function generates operating characteristics of SAM priors using the analytical operating characteristic engines for two-arm trials with binary or continuous endpoints. As an option, the operating characteristics of robust MAP priors can also be generated for comparison.

Usage

```
get_OC(
  if.prior,
  theta.h,
  method.w,
  prior.odds,
  nf.prior,
  prior.t = nf.prior,
  delta,
  n,
  n.t,
  target = 0.05,
  if.rMAP = FALSE,
  weight.rMAP = 0.5,
  theta,
  theta.t,
  alternative = c("greater", "less"),
  margin = 0,
  rel.tol = 1e-05,
  oc_rel.tol = 1e-06,
  interval = c(0.5, 0.999),
  n_sd_int = 8,
  ...
)
```

Arguments

`if.prior`	Informative prior constructed from historical data, represented (approximately) as a mixture of conjugate distributions.
`theta.h`	Estimate of the treatment effect based on historical data. Included for interface compatibility. If missing, the default value is set to the posterior mean estimate from `if.prior`.
`method.w`	Methods used to determine the mixture weight for SAM priors. The default method is LRT (Likelihood Ratio Test), the alternative option can be PPR (Posterior Probability Ratio). See `SAM_weight` for more details.
`prior.odds`	The prior probability of H_0 being true compared to the prior probability of H_1 being true using PPR method. The default value is 1. See `SAM_weight` for more details.
`nf.prior`	Non-informative prior used as the robustifying component for the control arm prior.
`prior.t`	Prior used for the treatment arm. If missing, the default value is set to be `nf.prior`.
`delta`	Clinically significant difference used for the SAM prior.
`n`	Sample size for the control arm.
`n.t`	Sample size for the treatment arm.

`target`	Target type I error used to calibrate the posterior probability cutoff for each method. The default value is typically 0.05.
`if.rMAP`	Whether to evaluate the operating characteristics of the robust MAP prior for comparison. The default value is `FALSE`.
`weight.rMAP`	Weight assigned to the informative prior component ($0 \leq$ `weight.rMAP` ≤ 1) for the robust MAP prior.
`theta`	A vector of the response rate (binary endpoints) or mean (continuous endpoints) for the control arm.
`theta.t`	A vector of the response rate (binary endpoints) or mean (continuous endpoints) for the treatment arm.
`alternative`	Direction of the posterior decision. Must be one of `"greater"` (for superiority) or `"less"` (for inferiority).
`margin`	Clinical margin. Must be a non-negative scalar. The default value is `0`.
`rel.tol`	Tolerance passed to numerical root finding.
`oc_rel.tol`	Relative tolerance passed to operating characteristic evaluation.
`interval`	Search interval for the posterior probability cutoff.
`n_sd_int`	Half-width of the numerical integration region for each arm, expressed as a multiple of the corresponding standard error. Used for continuous endpoints only.
`...`	Additional parameters. For continuous endpoints, this includes `sigma`.

Details

Compared with the original simulation-based implementation, this function does not rely on trial simulation. Instead, it first calibrates the posterior probability cutoff for each borrowing method to achieve the target type I error, and then evaluates operating characteristics analytically across the requested scenarios.

For each borrowing method, the function first calibrates the posterior probability cutoff so that the repeated-sampling rejection probability under the boundary null scenario equals the target type I error `target`. Specifically, calibration is based on the first value of `theta`. Let $\theta =$ `theta`$[1]$ denote the control-arm parameter under the calibration scenario. Then the treatment-arm parameter is set to $\theta_t = \theta + margin$ when `alternative = "greater"`, and to $\theta_t = \theta - margin$ when `alternative = "less"`. Thus, when `margin = 0`, calibration is performed under the null scenario $\theta_t = \theta$, corresponding to no treatment effect difference between the treatment and control arms. After the cutoff is calibrated, the function evaluates the operating characteristics across all requested scenarios using the analytical two-arm engines.

Value

A data frame with one row per scenario-method combination. The columns are:

Scenarios Scenario index.

theta True control-arm parameter value.

theta.t True treatment-arm parameter value.

Methods Borrowing method, one of `"NP"`, `"rMAP"`, or `"SAM"`.

Cutoffs Calibrated posterior probability cutoff for the method.

Bias of theta Bias of the posterior mean estimator of θ.

RMSE of theta Root mean squared error of the posterior mean estimator of θ.

Weight Average borrowing weight under the method.

Probability of Rejection Repeated-sampling rejection probability.

References

Yang P, Zhao Y, Nie L, Vallejo J, Yuan Y. SAM: Self-adapting mixture prior to dynamically borrow information from historical data in clinical trials. *Biometrics* 2023;79(4):2857-2868.

Schmidli H, Gsteiger S, Roychoudhury S, O'Hagan A, Spiegelhalter D, Neuenschwander B. Robust meta-analytic-predictive priors in clinical trials with historical control information. *Biometrics* 2014;70(4):1023-1032.

See Also

`eval_oc_bin_2arm`, `eval_oc_cont_2arm`, `calibrate_cutoff_bin_2arm`, `calibrate_cutoff_cont_2arm`

Examples

```
## Example: operating characteristics for a two-arm binary trial
## using a SAM prior without rMAP comparison

## Informative prior constructed from historical data
if.prior <- mixbeta(c(1, 20, 40))

## Evaluate operating characteristics
OC <- get_OC(
  if.prior = if.prior,
  nf.prior = mixbeta(c(1, 1, 1)),   ## Non-informative prior for control mixture
  prior.t = mixbeta(c(1, 1, 1)),    ## Prior for treatment arm
  delta = 0.2,                      ## Clinically significant difference for SAM
  n = 50,                           ## Sample size for control arm
  n.t = 100,                        ## Sample size for treatment arm
  target = 0.05,                    ## Target type I error for cutoff calibration
  if.rMAP = FALSE,                  ## Do not include rMAP comparison
  theta = c(summary(if.prior)["mean"], summary(if.prior)["mean"]),
  theta.t = c(summary(if.prior)["mean"], 0.50),
  alternative = "greater",          ## Superiority test
  margin = 0                        ## Clinical margin
)

OC
```

`post_summary_bin_2arm` *Posterior Summary for Two-Arm Comparative Trial with Binary Endpoint*

Description

The `post_summary_bin_2arm` function is designed to compute the posterior summary for the treatment effect in a two-arm comparative trial with a binary endpoint under one of three borrowing strategies: self-adapting mixture prior (SAM), robust MAP prior with fixed weight (rMAP), or non-informative prior (NP).

Usage

```
post_summary_bin_2arm(
  x.t,
  x,
  if.prior,
  nf.prior,
  prior.t = nf.prior,
  n.t,
  n,
  delta,
  cutoff,
  method = c("SAM", "rMAP", "NP"),
  alternative = c("greater", "less"),
  margin = 0,
  weight_rMAP = 0.5,
  method.w = "LRT",
  prior.odds = 1,
  rel.tol = 1e-08
)
```

Arguments

`x.t`	Observed number of responses in the treatment arm.
`x`	Observed number of responses in the control arm.
`if.prior`	Informative prior constructed based on historical data for the control arm, represented (approximately) as a beta mixture prior.
`nf.prior`	Non-informative prior used as the robustifying component for the control arm prior.
`prior.t`	Prior used for the treatment arm. If missing, the default value is set to be `nf.prior`.
`n.t`	Sample size for the treatment arm.
`n`	Sample size for the control arm.

`delta`	Clinically significant difference used for the SAM prior. This argument is only used when `method = "SAM"`.
`cutoff`	Posterior probability cutoff used for decision making. The null hypothesis is rejected if the posterior tail probability exceeds `cutoff`.
`method`	Borrowing strategy for the control arm. Must be one of `"SAM"`, `"rMAP"`, or `"NP"`.
`alternative`	Direction of the posterior decision. Must be one of `"greater"` (for superiority) or `"less"` (for inferiority).
`margin`	Clinical margin. Must be a non-negative scalar. The default value is `0`.
`weight_rMAP`	Weight assigned to the informative prior component ($0 \leq$ `weight_rMAP` ≤ 1) for the robust MAP prior. This argument is only used when `method = "rMAP"`. The default value is 0.5.
`method.w`	Methods used to determine the mixture weight for SAM priors. The default method is "LRT" (Likelihood Ratio Test), the alternative option is "PPR" (Posterior Probability Ratio). See `SAM_weight` for more details.
`prior.odds`	The prior probability of H_0 being true compared to the prior probability of H_1 being true using PPR method. The default value is 1. See `SAM_weight` for more details.
`rel.tol`	Relative tolerance for numerical integration used to evaluate the posterior probability.

Details

The treatment effect is defined as $\tau = \theta_t - \theta$, where θ_t and θ denote the response rates in the treatment and control arms, respectively. Inference is based on the posterior distribution of τ given the observed response counts from the two arms.

The posterior for the treatment arm is obtained by updating `prior.t` using the observed response count `x.t`. The posterior for the control arm depends on the selected borrowing strategy:

- `SAM`: the prior for the control arm is a mixture of `if.prior` and `nf.prior`, with adaptive mixture weight determined by the SAM borrowing rule.
- `rMAP`: the prior for the control arm is a fixed mixture of `if.prior` and `nf.prior`, with fixed weight `weight_rMAP`.
- `NP`: the prior for control arm is `nf.prior`.

When `alternative = "greater"`, inference is based on $P(\theta_t - \theta > margin \mid X_t, X)$. When `alternative = "less"`, inference is based on $P(\theta_t - \theta < -margin \mid X_t, X)$.

The posterior mean and posterior variance of τ are defined as

$$E(\tau \mid X_t, X) = E(\theta_t \mid X_t) - E(\theta \mid X),$$

and

$$\mathrm{Var}(\tau \mid X_t, X) = \mathrm{Var}(\theta_t \mid X_t) + \mathrm{Var}(\theta \mid X),$$

where independence between the treatment and control arm posteriors is assumed conditional on the current and historical data.

Value

A list containing the posterior probability in the requested direction, posterior mean and variance of τ, decision indicator, borrowing weight used for the control arm prior, and the corresponding trial data and method information.

See Also

`SAM_weight`, `SAM_prior`

`post_summary_cont_2arm` *Posterior Summary for Two-Arm Comparative Trial with Continuous Endpoint*

Description

The `post_summary_cont_2arm` function is designed to compute the posterior summary for the treatment effect in a two-arm comparative trial with a continuous endpoint under one of three borrowing strategies: self-adapting mixture prior (SAM), robust MAP prior with fixed weight (rMAP), or non-informative prior (NP).

Usage

```
post_summary_cont_2arm(
  ybar_t,
  ybar,
  if.prior,
  nf.prior,
  prior.t = nf.prior,
  n.t,
  n,
  sigma.t,
  sigma,
  delta,
  cutoff,
  method = c("SAM", "rMAP", "NP"),
  alternative = c("greater", "less"),
  margin = 0,
  weight_rMAP = 0.5,
  method.w = "LRT",
  prior.odds = 1
)
```

Arguments

`ybar_t`	Observed sample mean in the treatment arm.
`ybar`	Observed sample mean in the control arm.
`if.prior`	Informative prior constructed based on historical data for the control arm, represented (approximately) as a normal mixture prior.
`nf.prior`	Non-informative prior used as the robustifying component for the control arm prior.
`prior.t`	Prior used for the treatment arm. If missing, the default value is set to be `nf.prior`.
`n.t`	Sample size for the treatment arm.
`n`	Sample size for the control arm.
`sigma.t`	Known standard deviation in the treatment arm.
`sigma`	Known standard deviation in the control arm.
`delta`	Clinically significant difference used for the SAM prior. This argument is only used when `method = "SAM"`.
`cutoff`	Posterior probability cutoff used for decision making. The null hypothesis is rejected if the posterior tail probability exceeds `cutoff`.
`method`	Borrowing strategy for the control arm. Must be one of `"SAM"`, `"rMAP"`, or `"NP"`.
`alternative`	Direction of the posterior decision. Must be one of `"greater"` (for superiority) or `"less"` (for inferiority).
`margin`	Clinical margin. Must be a non-negative scalar. The default value is `0`.
`weight_rMAP`	Weight assigned to the informative prior component ($0 \leq$ `weight_rMAP` ≤ 1) for the robust MAP prior. This argument is only used when `method = "rMAP"`. The default value is 0.5.
`method.w`	Methods used to determine the mixture weight for SAM priors. The default method is "LRT" (Likelihood Ratio Test), the alternative option is "PPR" (Posterior Probability Ratio). See `SAM_weight` for more details.
`prior.odds`	The prior probability of H_0 being true compared to the prior probability of H_1 being true using PPR method. The default value is 1. See `SAM_weight` for more details.

Details

The treatment effect is defined as $\tau = \theta_t - \theta$, where θ_t and θ denote the means in the treatment and control arms, respectively. Inference is based on the posterior distribution of τ given the observed sample means from the two arms.

The posterior for the treatment arm is obtained by updating `prior.t` using the observed sample mean `ybar_t`. The posterior for the control arm depends on the selected borrowing strategy:

- `SAM`: the control arm prior is a mixture of `if.prior` and `nf.prior`, with adaptive mixture weight determined by the SAM borrowing rule.
- `rMAP`: the control arm prior is a fixed mixture of `if.prior` and `nf.prior`, with fixed weight `weight_rMAP`.

- `NP`: the control arm prior is `nf.prior` only.

When `alternative = "greater"`, inference is based on $P(\theta_t - \theta > margin \mid \bar{Y}_t, \bar{Y})$. When `alternative = "less"`, inference is based on $P(\theta_t - \theta < -margin \mid \bar{Y}_t, \bar{Y})$.

The posterior mean and posterior variance of τ are defined as are defined as

$$E(\tau \mid \bar{Y}_t, \bar{Y}) = E(\theta_t \mid \bar{Y}_t) - E(\theta \mid \bar{Y}),$$

and

$$\mathrm{Var}(\tau \mid \bar{Y}_t, \bar{Y}) = \mathrm{Var}(\theta_t \mid \bar{Y}_t) + \mathrm{Var}(\theta \mid \bar{Y}),$$

where independence between the treatment and control arm posteriors is assumed conditional on the current and historical data.

Value

A list containing the posterior probability in the requested direction, posterior mean and variance of τ, decision indicator, borrowing weight used for the control arm prior, and the corresponding trial data and method information.

See Also

`SAM_weight`, `SAM_prior`

`PS_prior` *Calculating the Propensity Score-Integrated Informative Priors*

Description

The `PS_prior` function is designed to calculate the Propensity Score-Integrated (PS) informative prior constructed based on historical data.

Usage

```
PS_prior(
  formula,
  data,
  outcome,
  study,
  treat,
  method,
  distance,
  ratio,
  ps.method,
  trim
)

PS_prior.default(
```

```
  formula,
  data,
  outcome,
  study,
  treat,
  method,
  distance,
  ratio,
  ps.method,
  trim
)

PS_prior.beta(
  formula,
  data,
  outcome,
  study,
  treat,
  method,
  distance,
  ratio,
  ps.method,
  trim
)

PS_prior.norm(
  formula,
  data,
  outcome,
  study,
  treat,
  method,
  distance,
  ratio,
  ps.method,
  trim
)
```

Arguments

`formula`	A two-sided `formula` object containing the study indicator and covariates to be used in creating the distance measure used in the matching. This formula will be supplied to the functions that estimate the distance measure. For example, the formula should be specified as `G ~ X1 + X2 + ...` where `G` represents the name of study indicator and `X1` and `X2` are covariates.
`data`	A data frame containing the variables named in `formula` and possible other arguments.
`outcome`	The variable name of the outcome.

`study`	The variable name of the study indicator.
`treat`	The variable name of the treatment indicator.
`method`	The matching method to be used. The allowed methods are `"nearest"` for nearest neighbor matching (on the propensity score by default), `"optimal"` [method_optimal] for optimal pair matching, `"full"` [method_full] for optimal full matching, `"genetic"` [method_genetic] for genetic matching, `"cem"` [method_cem] for coarsened exact matching, `"exact"` [method_exact] for exact matching, `"cardinality"` [method_cardinality] for cardinality and template matching, and `"subclass"` [method_subclass] for subclassification. When set to `"NULL"`, no matching will occur, but propensity score estimation and common support restrictions will still occur if requested. See the linked pages for each method for more details on what these methods do, how the arguments below are used by each on, and what additional arguments are allowed.
`distance`	The distance measure to be used. Can be either the name of a method of estimating propensity scores (e.g., `"glm"`), the name of a method of computing a distance matrix from the covariates (e.g., `"mahalanobis"`), a vector of already-computed distance measures, or a matrix of pairwise distances. See [distance] for allowable options. The default is `"glm"` for propensity scores estimated with logistic regression using `glm()`. Ignored for some methods; see individual methods pages for information on whether and how the distance measure is used.
`ratio`	For methods that allow it, how many historical control units should be matched to each current control unit in $k:1$ matching. Should be a single integer value. See the individual methods pages for information on whether and how this argument is used. The default is 1 for 1:1 matching.
`ps.method`	PS method utilize to calculate an informative prior based on historical data. The allowed methods are `"Weighting"` or `"Matching"`. The default method is `"Weighting"`.
`trim`	Lower and upper bound of trimming used in `"Weighting"`. The default is [0.1,0.9].

Details

This function aims to calculate informative priors using historical data by incorporating covariate information to enhance borrowing strength and address prior-data conflicts.

Let G be the study indicator, where $G = 1$ indicate patient is from current control study, and $G = 0$ indicate patient is from historical control study. Given the covariates data X, the propensity score is defined as follows,

$$e(X) = \Pr(G = 1|X),$$

where distance allows different methods to estimate the propensity scores.

Calculate informative prior through PS matching is to identify a subset of historical data (D_h^*) that have similar PS as current control data (D). Various algorithms are available for PS matching, please refer to `method`. The informative prior can then be calculated based on the matched historical dataset.

Alternative, we can utilize the inverse probability of treatment weighting (IPTW) to adjust the distribution of X in historical data D_h, making it similar to that in D. Specifically, for the ith subject, we assign a weight α_i to the outcome y_i in D_h based on its PS $e(X_i)$ and a fixed weight $\alpha_i = 1$ to X_i in D, as follows:

$$\alpha_i = G_1 + (1 - G_i)\frac{e(X_i)}{1 - e(X_i)}.$$

To avoid extremely large weights that may compromise IPTW, symmetric trimming rule can be used to trim the tails of the PS distribution by input `trim` with default [0.1,0.9], that is to trim observations whose estimated PS is outside of this range.

To standardized α, we compute the effective sample size (ESS), which approximately reflects the level of precision or equivalently its sample size, retained in the sample after weight as $n_h^* = (\sum \alpha_i)^2 / \sum \alpha_i^2$. The standardized weight is given by

$$\alpha_i^* = G_i + (1 - G_i)\frac{G_i}{\sum \alpha_i / n_h^*}.$$

For binary endpoint $Y \sim Ber(\theta)$, the informative prior $\pi_1(\theta)$ can be constructed as follows,

$$\pi_1(\theta) \propto L(\theta | D_h, \alpha^*)\pi_0(\theta) = Beta(a + \sum \alpha_i^* y_i, b + n_h^* - \sum \alpha_i^* y_i)\},$$

where $\pi_0(\theta)$ is a non-informative prior, a natural choice is $Beta(a, b)$, with $a = b = 1$.

For continuous endpoint $Y \sim N(0, \sigma^2)$, suppose σ^2 is unknown, with non-informative prior $p(\theta, \sigma^2) \propto 1/\sigma^2$, $\pi_1(\theta)$ follows a student-t distribution with degree of freedom $n_h^* - 1$. Given that n_h^* is moderate and large, it can be approximated by a normal distribution $N(\bar{y}^*, {s^*}^2 / n_h^*)$ with

$$\bar{y}^* = \sum \alpha_i^* y_i / \alpha_i^*, \ \ {s^*}^2 = \sum \alpha_i^* (y_i - \bar{y}^*)^2 / (n_h^* - 1).$$

Value

Displays the informative prior calculated from historical data based on the selected PS method.

Functions

- `PS_prior.default()`: The function calculates the Propensity Score-Integrated informative prior based on historical data for binary and continuous endpoint.
- `PS_prior.beta()`: The function calculates the Propensity Score-Integrated informative prior based on historical data for binary endpoint.
- `PS_prior.norm()`: The function calculates the Propensity Score-Integrated informative prior based on historical data for continuous endpoint.

References

Zhao Y, Laird G, Chen J, Yuan Y. PS-SAM: doubly robust propensity-score-integrated self-adapting mixture prior to dynamically borrow information from historical data.

See Also

matchit

Examples

```
## Load example data
data('PS_SAM_data')
## Subset the data to contain historical data and current control
dat <- PS_SAM_data[PS_SAM_data$A == 0, ]
str(dat)

## Examples for binary endpoints
## Generate the informative prior based on historical data using PS Matching
summary(PS_prior(formula = 'G ~ X_1 + X_2 + X_3',
                 data = dat, ps.method = 'Matching', method = 'nearest',
                 outcome = 'Y_binary', study = 'G', treat = 'A'))

## Generate the informative prior based on historical data using PS Weighting
summary(PS_prior(formula = 'G ~ X_1 + X_2 + X_3',
                 data = dat, ps.method = 'Weighting',
                 outcome = 'Y_binary', study = 'G', treat = 'A'))

## Examples for continuous endpoints
## Generate the informative prior based on historical data using PS Matching
summary(PS_prior(formula = 'G ~ X_1 + X_2 + X_3',
                 data = dat, ps.method = 'Matching', method = 'nearest',
                 outcome = 'Y_continuous', study = 'G', treat = 'A'))

## Generate the informative prior based on historical data using PS Weighting
summary(PS_prior(formula = 'G ~ X_1 + X_2 + X_3',
                 data = dat, ps.method = 'Weighting',
                 outcome = 'Y_continuous', study = 'G', treat = 'A'))
```

`PS_SAM_data` *Simulated Data for the Construction of Propensity Score-Integrated Informative Priors*

Description

This dataset demonstrates the construction of a Propensity Score-Integrated (PS) SAM prior. It simulates a two-arm randomized clinical trial (RCT) with a 2:1 randomization ratio between treatment and control arms, considering both binary and continuous endpoints.

Usage

```
PS_SAM_data
```

Format

A data frame with 600 observations.

- "A" is the treatment assignment (1 = treated, 0 = control).

- "G" is the study indicator (1 = current, 0 = historical).
- "X_1" is a binary covariate.
- "X_2" is a continuous covariate.
- "X_3" is a continuous covariate.
- "Y_{binary}" is binary outcome.
- "$Y_{continuous}$" is continuous outcome.

Details

The dataset includes:

- Sample size for treatment arm: $n_t = 200$.
- Sample size for control arm: $n_c = 100$.
- Sample size for historical control study: $n_h = 300$.

Covariates for the control arm were generated from

$$X_1 \sim Ber(0.5),\ \ X_2 \sim N(0,1),\ \ X_3 \sim N(0.5,1),$$

where $Ber(\cdot)$ stands for Bernoulli distribution. Covariates for the historical controls were generated from a mixture distribution, with half were generated the same as for the control arm, while the other half were drawn from

$$X_1 \sim Ber(0.8),\ \ X_2 \sim N(-0.4,1),\ \ X_3 \sim N(-0.2,1).$$

For the binary endpoint, y_i were generated from the logit model:

$$logit(\Pr(y_i = 1|X_{1i}, X_{2i}, X_{3i}, A_i)) = -1.4 - 0.5X_{1i} + X_{2i} + 2X_{3i} + \lambda A_i,$$

where λ is the treatment effect size, and we let $\lambda = 0.9$ to generate a moderate treatment effect size so that they study has a reasonable power.

For the continuous endpoint, y_i were generated from the following normal model:

$$y_i = 1.8X_{1i} + 0.9X_{2i} - 2X_{3i} + \lambda A_i + \epsilon_i,$$

where we let $\lambda = 1$, and $\epsilon_i \sim N(0, 3.5^2)$.

This dataset enables evaluation of the PS-SAM prior's performance in addressing heterogeneity between the RCT control arm and historical controls.

Examples

```
# Load the dataset
data(PS_SAM_data)

# View the structure
str(PS_SAM_data)
```

`SAM_prior` *Calculating SAM priors*

Description

The `SAM_prior` function is designed to display the SAM prior, given the informative prior (constructed from historical data), non-informative prior, and the mixture weight calculated using `SAM_weight` function (*Yang, et al., 2023*).

Usage

```
SAM_prior(if.prior, nf.prior, weight, ...)

## S3 method for class 'betaMix'
SAM_prior(if.prior, nf.prior, weight, ...)

## S3 method for class 'gammaMix'
SAM_prior(if.prior, nf.prior, weight, ...)

## S3 method for class 'normMix'
SAM_prior(if.prior, nf.prior, weight, ..., sigma)
```

Arguments

`if.prior`	Informative prior constructed from historical data, represented (approximately) as a mixture of conjugate distributions.
`nf.prior`	Non-informative prior used for the mixture.
`weight`	Weight assigned to the informative prior component ($0 \leq$ `weight` ≤ 1), which should be determined by `SAM_weight` function.
`...`	Additional parameters required for different endpoints.
`sigma`	Variance used for constructing the non-informative prior for continuous endpoints.

Details

SAM prior is constructed by mixing an informative prior $\pi_1(\theta)$, constructed based on historical data, with a non-informative prior $\pi_0(\theta)$ using the mixture weight w determined by `SAM_weight` function to achieve the degree of prior-data conflict (*Schmidli et al., 2015, Yang et al., 2023*).

Let θ and θ_h denote the treatment effects associated with the current arm data D and historical data D_h, respectively. Let δ denote the clinically significant difference such that if $|\theta_h - \theta| \geq \delta$, then θ_h is regarded as clinically distinct from θ, and it is therefore inappropriate to borrow any information from D_h. Consider two hypotheses:

$$H_0 : \theta = \theta_h, \ H_1 : \theta = \theta_h + \delta \ or \ \theta = \theta_h - \delta.$$

H_0 represents that D_h and D are consistent (i.e., no prior-data conflict) and thus information borrowing is desirable, whereas H_1 represents that the treatment effect of D differs from D_h to such a degree that no information should be borrowed.

The SAM prior uses the likelihood ratio test (LRT) statistics R to quantify the degree of prior-data conflict and determine the extent of information borrowing.

$$R = P(D|H_0, \theta_h)/P(D|H_1, \theta_h) = P(D|\theta = \theta_h)/\max(P(D|\theta = \theta_h + \delta), P(D|\theta = \theta_h - \delta)),$$

where $P(D|\cdot)$ denotes the likelihood function. An alternative Bayesian choice is the posterior probability ratio (PPR):

$$R = P(D|H_0, \theta_h)/P(D|H_1, \theta_h) = P(H_0)/P(H_1) \times BF,$$

where $P(H_0)$ and $P(H_1)$ is the prior probabilities of H_0 and H_1 being true. BF is the Bayes Factor that in this case is the same as the LRT.

The SAM prior, denoted as $\pi_{sam}(\theta)$, is then defined as a mixture of an informative prior $\pi_1(\theta)$, constructed based on D_h and a non-informative prior $\pi_0(\theta)$:

$$\pi_{sam}(\theta) = w\pi_1(\theta) + (1 - w)\pi_0(\theta),$$

where the mixture weight w is calculated as:

$$w = R/(1 + R).$$

As the level of prior-data conflict increases, the likelihood ratio R decreases, resulting in a decrease in the weight w assigned to the informative prior and thus a decrease in information borrowing. As a result, $\pi_{sam}(\theta)$ is data-driven and has the ability to self-adapt the information borrowing based on the degree of prior-data conflict.

Value

Displays the SAM prior as a mixture of an informative prior (constructed based on the historical data) and a non-informative prior.

Methods (by class)

- `SAM_prior(betaMix)`: The function calculates the SAM prior for beta mixture distribution. The default `nf.prior` is set to be `mixbeta(c(1,1,1))` which represents a uniform prior `Beta(1,1)`.
- `SAM_prior(gammaMix)`: The function calculates the SAM prior for gamma mixture distribution. The default `nf.prior` is set to be `mixgamma(c(1,0.001,0.001))` which represents a vague gamma prior `Gamma(0.001,0.001)`.
- `SAM_prior(normMix)`: The function calculates the SAM prior for normal mixture distribution. The default `nf.prior` is set to be `mixnorm(c(1,summary(if.prior)['mean'], sigma))` which represents a unit-information prior.

References

Yang P, Zhao Y, Nie L, Vallejo J, Yuan Y. SAM: Self-adapting mixture prior to dynamically borrow information from historical data in clinical trials. *Biometrics* 2023; 79(4), 2857-2868.

Schmidli H, Gsteiger S, Roychoudhury S, O'Hagan A, Spiegelhalter D, Neuenschwander B. Robust meta-analytic-predictive priors in clinical trials with historical control information. *Biometrics* 2014; 70(4):1023-1032.

See Also

`SAM_weight`

Examples

```
set.seed(123)
## Examples for binary endpoints
## Suppose that the informative prior constructed based on historical data is
## beta(40, 60)
prior.historical <- mixbeta(c(1, 40, 60))
## Data of the control arm
data.control     <- rbinom(60, size = 1, prob = 0.42)
## Calculate the mixture weight of the SAM prior
wSAM <- SAM_weight(if.prior = prior.historical,
                   delta = 0.15,        ## Clinically significant difference
                   data = data.control  ## Control arm data
                   )
## Assume beta(1,1) as the non-informative prior used for mixture
nf.prior  <- mixbeta(nf.prior = c(1,1,1))
## Generate the SAM prior
SAM.prior <- SAM_prior(if.prior = prior.historical, ## Informative prior
                       nf.prior = nf.prior,         ## Non-informative prior
                       weight = wSAM                ## Mixture weight of the SAM prior
                       )
plot(SAM.prior)

## Examples for continuous endpoints
## Suppose that the informative prior constructed based on historical data is
## N(0, 3)
sigma      <- 3
prior.mean <- 0
prior.se   <- sigma/sqrt(100)
prior.historical <- mixnorm(c(1, prior.mean, prior.se), sigma = sigma)
## Data of the control arm
data.control <- rnorm(80, mean = 0, sd = sigma)
## Calculate the mixture weight of the SAM prior
wSAM <- SAM_weight(if.prior = prior.historical,
                   delta = 0.2 * sigma,    ## Clinically significant difference
                   data = data.control     ## Control arm data
                   )
## Assume unit-information prior N(0,3) as the non-informative prior used
## for the mixture
nf.prior         <- mixnorm(nf.prior = c(1,prior.mean, sigma),
```

```
                              sigma = sigma)
## Generate the SAM prior
SAM.prior <- SAM_prior(if.prior = prior.historical, ## Informative prior
                       nf.prior = nf.prior,         ## Non-informative prior
                       weight = wSAM                ## Mixture weight of the SAM prior
                       )
plot(SAM.prior)
```

SAM_weight *Calculating Mixture Weight of SAM Priors*

Description

The `SAM_weight` function is designed to calculate the mixture weight of the SAM priors according to the degree of prior-data conflicts (*Yang, et al., 2023*).

Usage

```
SAM_weight(if.prior, theta.h, method.w, prior.odds, data, delta, ...)

## S3 method for class 'betaMix'
SAM_weight(if.prior, theta.h, method.w, prior.odds, data, delta, n, r, ...)

## S3 method for class 'normMix'
SAM_weight(
  if.prior,
  theta.h,
  method.w,
  prior.odds,
  data,
  delta,
  m,
  n,
  sigma,
  ...
)

## S3 method for class 'gammaMix'
SAM_weight(if.prior, theta.h, method.w, prior.odds, data, delta, u, w, ...)
```

Arguments

`if.prior`	Informative prior constructed based on historical data, represented (approximately) as a mixture of conjugate distributions.
`theta.h`	Estimate of the treatment effect based on historical data. If missing, the default value is set to be the posterior mean estimate from `if.prior`.

`method.w`	Methods used to determine the mixture weight for SAM priors. The default method is "LRT" (Likelihood Ratio Test), the alternative option is "PPR" (Posterior Probability Ratio). See Details section for more information.
`prior.odds`	The prior probability of H_0 being true compared to the prior probability of H_1 being true using PPR method. The default value is 1. See Details section for more information.
`data`	Data of the control arm from the current trial, see Methods section for more details.
`delta`	Clinically significant difference used for the SAM prior.
`...`	Additional parameters required for different endpoints.
`n`	Number of subjects in the control arm for continuous endpoint.
`r`	Number of responses in the control arm for binary endpoint.
`m`	Mean estimate in the control arm for continuous endpoint.
`sigma`	Standard deviation in the control arm for continuous endpoint.
`u`	Number of events in the control arm for time-to-event endpoint.
`w`	Total observed time in the control arm for time-to-event endpoint.

Details

SAM prior is constructed by mixing an informative prior $\pi_1(\theta)$, constructed based on historical data, with a non-informative prior $\pi_0(\theta)$ using the mixture weight w determined by `SAM_weight` function to achieve the degree of prior-data conflict (*Schmidli et al., 2015, Yang et al., 2023*).

Let θ and θ_h denote the treatment effects associated with the current arm data D and historical data D_h, respectively. Let δ denote the clinically significant difference such that if $|\theta_h - \theta| \ge \delta$, then θ_h is regarded as clinically distinct from θ, and it is therefore inappropriate to borrow any information from D_h. Consider two hypotheses:

$$H_0 : \theta = \theta_h, \; H_1 : \theta = \theta_h + \delta \; or \; \theta = \theta_h - \delta.$$

H_0 represents that D_h and D are consistent (i.e., no prior-data conflict) and thus information borrowing is desirable, whereas H_1 represents that the treatment effect of D differs from D_h to such a degree that no information should be borrowed.

The SAM prior uses the likelihood ratio test (LRT) statistics R to quantify the degree of prior-data conflict and determine the extent of information borrowing.

$$R = P(D|H_0, \theta_h)/P(D|H_1, \theta_h) = P(D|\theta = \theta_h)/\max(P(D|\theta = \theta_h + \delta), P(D|\theta = \theta_h - \delta)),$$

where $P(D|\cdot)$ denotes the likelihood function. An alternative Bayesian choice is the posterior probability ratio (PPR):

$$R = P(D|H_0, \theta_h)/P(D|H_1, \theta_h) = P(H_0)/P(H_1) \times BF,$$

where $P(H_0)$ and $P(H_1)$ is the prior probabilities of H_0 and H_1 being true. BF is the Bayes Factor that in this case is the same as the LRT.

The SAM prior, denoted as $\pi_{sam}(\theta)$, is then defined as a mixture of an informative prior $\pi_1(\theta)$, constructed based on D_h and a non-informative prior $\pi_0(\theta)$:

$$\pi_{sam}(\theta) = w\pi_1(\theta) + (1 - w)\pi_0(\theta),$$

where the mixture weight w is calculated as:

$$w = R/(1 + R).$$

As the level of prior-data conflict increases, the likelihood ratio R decreases, resulting in a decrease in the weight w assigned to the informative prior and thus a decrease in information borrowing. As a result, $\pi_{sam}(\theta)$ is data-driven and has the ability to self-adapt the information borrowing based on the degree of prior-data conflict.

Value

The mixture weight of the SAM priors.

Methods (by class)

- `SAM_weight(betaMix)`: The function calculates the mixture weight of SAM priors for beta mixture distribution. The input `data` can be patient-level data (i.e., a vector of 0 and 1 representing the response status of each patient) or summary statistics (i.e., the number of patients and the number of responses).
- `SAM_weight(normMix)`: The function calculates the mixture weight of SAM priors for normal mixture distribution. The input `data` should be a vector of patient-level observations. The input `data` can be patient-level data (i.e., a vector of continuous response of each patient) or summary statistics (i.e., the mean estimate, number of subjects, and the standard deviation in the control arm).
- `SAM_weight(gammaMix)`: The function calculates the mixture weight of SAM priors for gamma mixture distribution. The input `data` can be patient-level data (i.e., a matrix with the first row as the censoring indicator and the second row recording the observed time) or summary statistics (i.e., the number of uncensored observations `u` and total observed time `w`).

References

Yang P, Zhao Y, Nie L, Vallejo J, Yuan Y. SAM: Self-adapting mixture prior to dynamically borrow information from historical data in clinical trials. *Biometrics* 2023; 79(4), 2857-2868.

Examples

```
set.seed(123)
## Examples for binary endpoints
## Example 1: no prior-data conflict
## Suppose that the informative prior constructed based on historical data is
## beta(40, 60)
prior.historical <- mixbeta(c(1, 40, 60))
## Data of control arm
data.control     <- rbinom(60, size = 1, prob = 0.42)
## Calculate the mixture weight of the SAM prior
```

```
wSAM <- SAM_weight(if.prior = prior.historical,
                   delta = 0.15,        ## Clinically significant difference
                   data = data.control  ## Control arm data
                   )
print(wSAM)

## Example 2: in the presence of prior-data conflict, where the current data
## has 12 responses in 60 patients
wSAM <- SAM_weight(if.prior = prior.historical,
                   delta = 0.15,    ## Clinically significant difference
                   ## Methods to determine mixture weight for the SAM priors
                   ## by Posterior Probability Ratio
                   method.w = 'PPR',
                   ## Prior odds of favoring no prior-data conflicts to
                   ## the presence of prior-data conflict
                   prior.odd = 1/9,
                   n = 60,          ## Number of patients in the control arm
                   r = 12           ## Number of responses in the control arm
                   )
print(wSAM)

## Example 3: in the presence of prior-data conflict, where the current data
## has 12 responses in 60 patients
wSAM <- SAM_weight(if.prior = prior.historical,
                   delta = 0.15, ## Clinically significant difference
                   n = 60,       ## Number of patients in the control arm
                   r = 12        ## Number of responses in the control arm
                   )
print(wSAM)

## Examples for continuous endpoints
## Example 1: no prior-data conflict
## Suppose that the informative prior constructed from historical data is
## N(0, 3)
sigma      <- 3
prior.mean <- 0
prior.se   <- sigma/sqrt(100)
prior.historical <- mixnorm(c(1, prior.mean, prior.se), sigma = sigma)
## Data of the control arm
data.control     <- rnorm(80, mean = 0, sd = sigma)
wSAM <- SAM_weight(if.prior = prior.historical,
                   delta = 0.3 * sigma,    ## Clinically significant difference
                   data = data.control     ## Control arm data
                   )
print(wSAM)

## Example 2: in the presence of prior-data conflict, where the current data
## has mean of 0.5
data.control     <- rnorm(80, mean = 1, sd = sigma)
wSAM  <- SAM_weight(if.prior = prior.historical,
                    delta = 0.3 * sigma,    ## Clinically significant difference
                    data = data.control     ## Control arm data
                    )
```

```
print(wSAM)

## Examples for survival endpoints
## Example 1: no prior-data conflict
## Suppose the survival times from historical data follows exp(1) distribution
## with random censoring time follows U(0.5, 5) distribution
T_hi <- rexp(100, rate = 1)
C_hi <- runif(100, min = 0.5, max = 5)
## Indicators of the uncensored events
delta_hi <- as.numeric(T_hi < C_hi)
## Observed survival times from historical data
U_hi     <- T_hi
U_hi[delta_hi == 0] <- C_hi[delta_hi == 0]
## Construct the informative prior based on simulated historical data
prior.historical <- mixgamma(c(1, sum(delta_hi), sum(U_hi)),
                             param = 'ab', likelihood = 'exp')
## Suppose the survival times from control data follows exp(0.95) distribution
## with random censoring time follows U(0.5, 5) distribution
T_ci <- rexp(100, rate = 0.95)
C_ci <- runif(100, min = 0.5, max = 5)
## Indicators of the uncensored events
delta_ci <- as.numeric(T_ci < C_ci)
## Observed survival times from control data
U_ci     <- T_ci
U_ci[delta_ci == 0] <- C_ci[delta_ci == 0]
## Data of the control arm
data.control     <- rbind(sum(delta_ci), sum(U_ci))
wSAM <- SAM_weight(if.prior = prior.historical,
                   delta = 0.2,            ## Clinically significant difference
                   data = data.control     ## Control arm data
                   )
print(wSAM)

## Example 2: in the presence of prior-data conflict, where the current survival
## times follows exp(2) distribution with random censoring time follows U(0.5, 5)
## distribution
T_ci <- rexp(100, rate = 2)
C_ci <- runif(100, min = 0.5, max = 5)
## Indicators of the uncensored events
delta_ci <- as.numeric(T_ci < C_ci)
## Observed survival times from control data
U_ci     <- T_ci
U_ci[delta_ci == 0] <- C_ci[delta_ci == 0]
## Data of the control arm
data.control     <- rbind(sum(delta_ci), sum(U_ci))
wSAM  <- SAM_weight(if.prior = prior.historical,
                    delta = 0.2,            ## Clinically significant difference
                    data = data.control     ## Control arm data
                    )
print(wSAM)
```

Index