

Package ‘abtest’

May 7, 2026

Type Package

Title Bayesian A/B Testing

Version 1.0.1

Description Provides functions for Bayesian A/B testing including prior elicitation options based on Kass and Vaidyanathan (1992) <[doi:10.1111/j.2517-6161.1992.tb01868.x](https://doi.org/10.1111/j.2517-6161.1992.tb01868.x)>. Gronau, Raj K. N., & Wagenmakers (2021) <[doi:10.18637/jss.v100.i17](https://doi.org/10.18637/jss.v100.i17)>.

License GPL (>= 2)

Encoding UTF-8

LazyData true

Depends R (>= 3.0.0)

Imports Rcpp (>= 0.12.14), mvtnorm, sn, qgam, truncnorm, plotrix, grDevices, RColorBrewer, Matrix, parallel

LinkingTo Rcpp

RoxygenNote 7.1.2

Suggests testthat

NeedsCompilation yes

Author Quentin F. Gronau [aut, cre],
Akash Raj [ctb],
Eric-Jan Wagenmakers [ths]

Maintainer Quentin F. Gronau <Quentin.F.Gronau@gmail.com>

Repository CRAN

Date/Publication 2021-11-22 09:00:11 UTC

Contents

ab-methods	2
ab_test	3
dprior	6
elicit_prior	8
extractors	10

plot_posterior	11
plot_prior	13
plot_robustness	14
plot_sequential	16
pprior	17
prob_wheel	19
qprior	20
seqdata	21
simulate_priors	22

Index	24
--------------	-----------

ab-methods	<i>Methods for ab objects</i>
------------	-------------------------------

Description

Methods defined for objects returned from the `ab_test` function.

Usage

```
## S3 method for class 'ab'
summary(object, digits = 3, raw = FALSE, ...)
```

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

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

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

Arguments

object, x	object of class ab as returned from <code>ab_test</code> .
digits	number of digits to print for the summary.
raw	if TRUE, the raw posterior samples are used to estimate the mean, sd, and quantiles for the summary of the posterior. If FALSE, parametric fits to the marginal posteriors are used to obtain the mean, sd, and quantiles. Specifically, a normal distribution is fitted for psi (logor) and beta; a log-normal distribution is fitted for or and rrisk; beta distributions are fitted for p1 and p2; a scaled beta distribution is fitted for arisk. These distributional fits are also used in <code>plot_posterior</code> .
...	further arguments, currently ignored.

Value

The `print` methods prints the Bayes factors, prior probabilities of the hypotheses, and posterior probabilities of the hypotheses (and returns nothing).

The `plot` method visualizes the prior probabilities of the hypotheses and posterior probabilities of the hypotheses (the next plots is obtained by hitting Return) using the `prob_wheel` function.

The `summary` methods returns the `ab` object that is guaranteed to contain posterior samples (i.e., it adds posterior samples if they were not included already). Additionally, it adds to the object a posterior summary matrix (i.e., `ab$post$post_summary`) for the posterior under H1 and the arguments `digits` (used for printing) and `raw` (added to `ab$input`).

 ab_test

Bayesian A/B Test

Description

Function for conducting a Bayesian A/B test (i.e., test between two proportions).

Usage

```
ab_test(
  data = NULL,
  prior_par = list(mu_psi = 0, sigma_psi = 1, mu_beta = 0, sigma_beta = 1),
  prior_prob = NULL,
  nsamples = 10000,
  is_df = 5,
  posterior = FALSE,
  y = NULL,
  n = NULL
)
```

Arguments

`data` list or data frame with the data. This list (data frame) needs to contain the following elements: `y1` (number of "successes" in the control condition), `n1` (number of trials in the control condition), `y2` (number of "successes" in the experimental condition), `n2` (number of trials in the experimental condition). Each of these elements needs to be an integer. Alternatively, the user can provide for each of the elements a vector with a cumulative sequence of "successes"/trials. This allows the user to produce a sequential plot of the posterior probabilities for each hypothesis by passing the result object of class "ab" to the `plot_sequential` function. Sequential data can also be provided in form of a data frame or matrix that has the columns "outcome" (containing only 0 and 1 to indicate the binary outcome) and "group" (containing only 1 and 2 to indicate the group membership). Note that the data can also be provided by specifying the arguments `y` and `n` instead (not possible for sequential data).

prior_par	list with prior parameters. This list needs to contain the following elements: mu_psi (prior mean for the normal prior on the test-relevant log odds ratio), sigma_psi (prior standard deviation for the normal prior on the test-relevant log odds ratio), mu_beta (prior mean for the normal prior on the grand mean of the log odds), sigma_beta (prior standard deviation for the normal prior on the grand mean of the log odds). Each of the elements needs to be a real number (the standard deviations need to be positive). The default are standard normal priors for both the log odds ratio parameter and the grand mean of the log odds parameter.
prior_prob	named vector with prior probabilities for the four hypotheses "H1", "H+", "H-", and "H0". "H1" states that the "success" probability differs between the control and the experimental condition but does not specify which one is higher. "H+" states that the "success" probability in the experimental condition is higher than in the control condition, "H-" states that the "success" probability in the experimental condition is lower than in the control condition. "H0" states that the "success" probability is identical (i.e., there is no effect). The one-sided hypotheses "H+" and "H-" are obtained by truncating the normal prior on the log odds ratio so that it assigns prior mass only to the allowed log odds ratio values (e.g., for "H+" a normal prior that is truncated from below at 0). If NULL (default) the prior probabilities are set to $c(0, 1/4, 1/4, 1/2)$. That is, the default assigns prior probability .5 to the hypothesis that there is no effect (i.e., "H0"). The remaining prior probability (i.e., also .5) is split evenly across the hypothesis that there is a positive effect (i.e., "H+") and the hypothesis that there is a negative effect (i.e., "H-").
nsamples	determines the number of importance samples for obtaining the log marginal likelihood for "H+" and "H-" and the number of posterior samples in case posterior = TRUE. The default is 10000.
is_df	degrees of freedom of the multivariate t importance sampling proposal density. The default is 5.
posterior	Boolean which indicates whether posterior samples should be returned. The default is FALSE.
y	integer vector of length 2 containing the number of "successes" in the control and experimental condition
n	integer vector of length 2 containing the number of trials in the control and experimental condition

Details

The implemented Bayesian A/B test is based on the following model by Kass and Vaidyanathan (1992, section 3):

$$\log(p1/(1 - p1)) = \beta - \psi/2$$

$$\log(p2/(1 - p2)) = \beta + \psi/2$$

$$y1 \text{ Binomial}(n1, p1)$$

$$y2 \text{ Binomial}(n2, p2).$$

"H0" states that $\psi = 0$, "H1" states that $\psi! = 0$, "H+" states that $\psi > 0$, and "H-" states that $\psi < 0$. Normal priors are assigned to the two parameters ψ (i.e., the test-relevant log odds ratio) and β

(i.e., the grand mean of the log odds which is a nuisance parameter). Log marginal likelihoods for "H0" and "H1" are obtained via Laplace approximations (see Kass & Vaidyanathan, 1992) which work well even for very small sample sizes. For the one-sided hypotheses "H+" and "H-" the log marginal likelihoods are obtained based on importance sampling which uses as a proposal a multivariate t distribution with location and scale matrix obtained via a Laplace approximation to the (log-transformed) posterior. If `posterior = TRUE`, posterior samples are obtained using importance sampling.

Value

returns an object of class "ab" with components:

- `input`: a list with the input arguments.
- `post`: a list with parameter posterior samples for the three hypotheses "H1", "H+" (in the output called "Hplus"), and "H-" (in the output called "Hminus"). Only contains samples if `posterior = TRUE`.
- `laplace`: a list with the approximate parameter posterior mode and variance/covariance matrix for each hypothesis obtained via a Laplace approximation.
- `method`: character that indicates the method that has been used to obtain the results. The default is "log-is" (importance sampling with multivariate t proposal based on a Laplace approximation to the log transformed posterior). If this method fails (for the one-sided hypotheses), method "is-sn" is used (i.e., importance sampling is used to obtain unconstrained samples, then a skew-normal distribution is fitted to the samples to obtain the results for the one-sided hypotheses). If `method = "is-sn"`, posterior samples can only be obtained for "H1".
- `logml`: a list with the estimated log marginal likelihoods for the hypotheses "H0" (i.e., "logml0"), "H1" (i.e., "logml1"), "H+" (i.e., "logmlplus"), and "H-" (i.e., "logmlminus").
- `post_prob`: a named vector with the posterior probabilities of the four hypotheses "H1", "H+", "H-", and "H0".
- `logbf`: a list with the log Bayes factor in favor of "H1" over "H0", the log Bayes factor in favor of "H+" over "H0", and the log Bayes factor in favor of "H-" over "H0".
- `bf`: a list with the Bayes factor in favor of "H1" over "H0" (i.e., "bf10"), the Bayes factor in favor of "H+" over "H0" (i.e., "bfplus0"), and the Bayes factor in favor of "H-" over "H0" (i.e., "bfminus0").

Author(s)

Quentin F. Gronau

References

- Kass, R. E., & Vaidyanathan, S. K. (1992). Approximate Bayes factors and orthogonal parameters, with application to testing equality of two binomial proportions. *Journal of the Royal Statistical Society, Series B*, 54, 129-144. doi: [10.1111/j.25176161.1992.tb01868.x](https://doi.org/10.1111/j.25176161.1992.tb01868.x)
- Gronau, Q. F., Raj K. N., A., & Wagenmakers, E.-J. (2021). Informed Bayesian Inference for the A/B Test. *Journal of Statistical Software*, 100. doi: [10.18637/jss.v100.i17](https://doi.org/10.18637/jss.v100.i17)

See Also

[elicit_prior](#) allows the user to elicit a prior based on providing quantiles for either the log odds ratio, the odds ratio, the relative risk, or the absolute risk. The resulting prior is always translated to the corresponding normal prior on the log odds ratio. The [plot_prior](#) function allows the user to visualize the prior distribution. The [simulate_priors](#) function produces samples from the prior distribution. The prior and posterior probabilities of the hypotheses can be visualized using the [prob_wheel](#) function. Parameter posteriors can be visualized using the [plot_posterior](#) function. The [plot_sequential](#) function allows the user to sequentially plot the posterior probabilities of the hypotheses (only possible if the data object contains vectors with the cumulative "successes"/trials).

Examples

```
# synthetic data
data <- list(y1 = 10, n1 = 28, y2 = 14, n2 = 26)

# Bayesian A/B test with default settings
ab <- ab_test(data = data)
print(ab)

# different prior parameter settings
prior_par <- list(mu_psi = 0.2, sigma_psi = 0.8,
                 mu_beta = 0, sigma_beta = 0.7)
ab2 <- ab_test(data = data, prior_par = prior_par)
print(ab2)

# different prior probabilities
prior_prob <- c(.1, .3, .2, .4)
names(prior_prob) <- c("H1", "H+", "H-", "H0")
ab3 <- ab_test(data = data, prior_prob = prior_prob)
print(ab3)

# also possible to obtain posterior samples
ab4 <- ab_test(data = data, posterior = TRUE)

# plot parameter posterior
plot_posterior(x = ab4, what = "logor")
```

 dprior

Prior Density

Description

Function for evaluating the prior density.

Usage

```
dprior(
  x1,
  x2 = NULL,
```

```

prior_par = list(mu_psi = 0, sigma_psi = 1, mu_beta = 0, sigma_beta = 1),
what = "logor",
hypothesis = "H1"
)

```

Arguments

x1	numeric vector with values at which the prior density should be evaluated.
x2	if what = "p1p2", value of p2 (i.e., the latent "success" probability in the experimental condition) at which the joint prior density should be evaluated. If what = "p2givenp1", the given value of p1 (i.e., the latent "success" probability in the control condition).
prior_par	list with prior parameters. This list needs to contain the following elements: mu_psi (prior mean for the normal prior on the test-relevant log odds ratio), sigma_psi (prior standard deviation for the normal prior on the test-relevant log odds ratio), mu_beta (prior mean for the normal prior on the grand mean of the log odds), sigma_beta (prior standard deviation for the normal prior on the grand mean of the log odds). Each of the elements needs to be a real number (the standard deviations need to be positive). The default are standard normal priors for both the log odds ratio parameter and the grand mean of the log odds parameter.
what	character specifying for which quantity the prior density should be evaluated. Either "logor" (i.e., log odds ratio), "or" (i.e., odds ratio), "p1p2" (i.e., the joint density of the latent "success" probability in the experimental and control condition), "p1" (i.e., latent "success" probability in the control condition), "p2" (i.e., latent "success" probability in the experimental condition), "p2givenp1" (i.e., conditional distribution of the latent "success" probability in the experimental condition given a "success" probability of p1 in the control condition), "rrisk" (i.e., relative risk, the ratio of the "success" probability in the experimental and the control condition), or "arisk" (i.e., absolute risk, the difference of the "success" probability in the experimental and control condition).
hypothesis	character specifying whether to evaluate the two-sided prior density (i.e., "H1"), the one-sided prior density with lower truncation point (i.e., "H+"), or the one-sided prior density with upper truncation point (i.e., "H-").

Value

numeric vector with the values of the prior density.

Note

Internally, the test-relevant prior is always a normal prior on the log odds ratio, consequently, if what is not "logor", the implied prior density for the quantity is returned.

Author(s)

Quentin F. Gronau

Examples

```
# prior parameters
prior_par <- list(mu_psi = 0, sigma_psi = 1,
                 mu_beta = 0, sigma_beta = 1)

# prior density
dprior(x1 = 0.1, prior_par = prior_par, what = "logor")
dprior(x1 = 1.1, prior_par = prior_par, what = "or")
dprior(x1 = 0.49, x2 = 0.51, prior_par = prior_par, what = "p1p2")
dprior(x1 = 0.45, prior_par = prior_par, what = "p1")
dprior(x1 = 0.45, prior_par = prior_par, what = "p2")
dprior(x1 = 0.49, x2 = 0.51, prior_par = prior_par, what = "p2givenp1")
dprior(x1 = 1.05, prior_par = prior_par, what = "rrisk")
dprior(x1 = 0.02, prior_par = prior_par, what = "arisk")

# also works for vectors
dprior(x1 = c(-0.1, 0, 0.1, 0.2), prior_par = prior_par, what = "logor")
```

elicit_prior

Elicit Prior

Description

Function for eliciting a prior distribution.

Usage

```
elicit_prior(
  q,
  prob,
  what = "logor",
  hypothesis = "H1",
  mu_beta = 0,
  sigma_beta = 1
)
```

Arguments

q	vector with quantiles for the quantity of interest.
prob	vector with probabilities corresponding to the quantiles (e.g., for the median the corresponding element of prob would need to be .5).
what	character specifying for which quantity a prior should be elicited. Either "logor" (i.e., log odds ratio), "or" (i.e., odds ratio), "rrisk" (i.e., relative risk, the ratio of the "success" probability in the experimental and the control condition), or "arisk" (i.e., absolute risk, the difference of the "success" probability in the experimental and control condition).

hypothesis	character specifying whether the provided quantiles correspond to a two-sided prior (i.e., "H1"), a one-sided prior with lower truncation point (i.e., "H+"), or a one-sided prior with upper truncation point (i.e., "H-").
mu_beta	prior mean of the nuisance parameter β (i.e., the grand mean of the log odds). The default is 0.
sigma_beta	prior standard deviation of the nuisance parameter β (i.e., the grand mean of the log odds). The default is 1.

Details

It is assumed that the prior on the grand mean of the log odds (i.e., β) is not the primary target of prior elicitation and is fixed (e.g., to a standard normal prior). The reason is that the grand mean nuisance parameter β is not the primary target of inference and changes in the prior on this nuisance parameter do not affect the results much in most cases (see Kass & Vaidyanathan, 1992). Nevertheless, it should be emphasized that the implemented approach allows users to set the prior parameters `mu_beta` and `sigma_beta` flexibly; the only constraint is that this takes place before the prior on the test-relevant log odds ratio parameter ψ is elicited. The `elicit_prior` function allows the user to elicit a prior not only in terms of the log odds ratio parameter ψ , but also in terms of the odds ratio, the relative risk (i.e., the ratio of the "success" probability in the experimental and the control condition), or the absolute risk (i.e., the difference of the "success" probability in the experimental and control condition). In case the prior is not elicited for the log odds ratio directly, the elicited prior is always translated to the closest corresponding normal prior on the log odds ratio. The prior parameters `mu_psi` and `sigma_psi` are obtained using least squares minimization.

Value

list with the elicited prior parameters. Specifically, this list consists of:

- `mu_psi` (prior mean for the normal prior on the test-relevant log odds ratio).
- `sigma_psi` (prior standard deviation for the normal prior on the test-relevant log odds ratio),
- `mu_beta` (prior mean for the normal prior on the grand mean of the log odds),
- `sigma_beta` (prior standard deviation for the normal prior on the grand mean of the log odds).

Note that the prior on the grand mean of the log odds is not part of the elicitation and is assumed to be fixed by the user (using the arguments `mu_beta` and `sigma_beta`). Consequently, the returned values for `mu_beta` and `sigma_beta` simply correspond to the input values.

Author(s)

Quentin F. Gronau

References

- Kass, R. E., & Vaidyanathan, S. K. (1992). Approximate Bayes factors and orthogonal parameters, with application to testing equality of two binomial proportions. *Journal of the Royal Statistical Society, Series B*, 54, 129-144. doi: [10.1111/j.25176161.1992.tb01868.x](https://doi.org/10.1111/j.25176161.1992.tb01868.x)
- Gronau, Q. F., Raj K. N., A., & Wagenmakers, E.-J. (2021). Informed Bayesian Inference for the A/B Test. *Journal of Statistical Software*, 100. doi: [10.18637/jss.v100.i17](https://doi.org/10.18637/jss.v100.i17)

See Also

The `plot_prior` function allows the user to visualize the elicited prior distribution.

Examples

```
# elicit prior
prior_par <- elicit_prior(q = c(0.1, 0.3, 0.5),
                        prob = c(.025, .5, .975),
                        what = "arisk")

print(prior_par)

# plot elicited prior (absolute risk)
plot_prior(prior_par = prior_par, what = "arisk")

# plot corresponding normal prior on log odds ratio
plot_prior(prior_par = prior_par, what = "logor")
```

 extractors

Extraction functions for ab objects

Description

Extraction functions for objects returned from the `ab_test` function.

Usage

```
get_bf(x, log = FALSE)

get_prior_prob(x)

get_post_prob(x)

get_post_samples(x, hypothesis = "H1")
```

Arguments

<code>x</code>	object of class "ab" as returned from <code>ab_test</code> .
<code>log</code>	determines whether the log Bayes factors are returned.
<code>hypothesis</code>	determines for which hypothesis posterior samples are returned. Needs to be either "H1", "H+", or "H-" (the default is "H1").

Value

`get_bf` returns the Bayes factors in favor of "H1", "H+", and "H-" (compared to H0). `get_prior_prob` returns the prior probabilities of the hypotheses. `get_post_prob` returns the posterior probabilities of the hypotheses. `get_post_samples` returns posterior samples for the specified hypothesis.

Examples

```
# synthetic data
data <- list(y1 = 10, n1 = 28, y2 = 14, n2 = 26)

# Bayesian A/B test with default settings
ab <- ab_test(data = data, posterior = TRUE)

# extract Bayes factors
get_bf(ab)

# extract prior probabilities
get_prior_prob(ab)

# extract posterior probabilities
get_post_prob(ab)

# extract posterior samples for H1
s <- get_post_samples(ab, hypothesis = "H1")
```

plot_posterior

Plot Posterior

Description

Function for plotting the posterior distribution.

Usage

```
plot_posterior(
  x,
  what = "logor",
  hypothesis = "H1",
  ci = 0.95,
  p1lab = "p1",
  p2lab = "p2",
  p1adj = 0.44,
  p2adj = 0.56,
  ...
)
```

Arguments

x	object of class "ab".
what	character specifying for which quantity the posterior should be plotted. Either "logor" (i.e., log odds ratio), "or" (i.e., odds ratio), "p1p2" (i.e., the marginal posteriors of the latent "success" probabilities in the experimental and control condition), "rrisk" (i.e., relative risk, the ratio of the "success" probability in the experimental and the control condition), or "arisk" (i.e., absolute risk, the

	difference of the "success" probability in the experimental and control condition).
hypothesis	character specifying whether to plot the two-sided posterior distribution (i.e., "H1"), the one-sided posterior distribution with lower truncation point (i.e., "H+"), or the one-sided posterior distribution with upper truncation point (i.e., "H-").
ci	numeric value specifying the <i>ci</i> % central credible interval. The default is 0.95 which yields a 95% central credible interval.
p1lab	determines p1 x-axis label. Only relevant for what = "p1p2".
p2lab	determines p2 x-axis label. Only relevant for what = "p1p2".
p1adj	determines p1 x-axis label adjustment. Only relevant for what = "p1p2".
p2adj	determines p2 x-axis label adjustment. Only relevant for what = "p1p2".
...	further arguments

Details

The resulting plot displays the posterior density for the quantity of interest and also displays the corresponding prior density. The values of the posterior median and a *ci*% central credible interval are displayed on top of the plot.

Author(s)

Quentin F. Gronau

Examples

```
# synthetic data
data <- list(y1 = 10, n1 = 28, y2 = 14, n2 = 26)

# Bayesian A/B test with default settings
ab <- ab_test(data = data, posterior = TRUE)

# plot parameter posterior
plot_posterior(x = ab, what = "logor")
plot_posterior(x = ab, what = "or")
plot_posterior(x = ab, what = "p1p2")
plot_posterior(x = ab, what = "rrisk")
plot_posterior(x = ab, what = "arisk")

# example of good width and height values for saving to file
cairo_pdf(file.path(tempdir(), "test_plot.pdf"),
          width = 530 / 72, height = 400 / 72)
plot_posterior(ab, what = "p1p2")
dev.off()
```

plot_prior

*Plot Prior***Description**

Function for plotting parameter prior distributions.

Usage

```
plot_prior(
  prior_par = list(mu_psi = 0, sigma_psi = 1, mu_beta = 0, sigma_beta = 1),
  what = "logor",
  hypothesis = "H1",
  p1 = 0.5,
  ...
)
```

Arguments

prior_par	list with prior parameters. This list needs to contain the following elements: mu_psi (prior mean for the normal prior on the test-relevant log odds ratio), sigma_psi (prior standard deviation for the normal prior on the test-relevant log odds ratio), mu_beta (prior mean for the normal prior on the grand mean of the log odds), sigma_beta (prior standard deviation for the normal prior on the grand mean of the log odds). Each of the elements needs to be a real number (the standard deviations need to be positive). The default are standard normal priors for both the log odds ratio parameter and the grand mean of the log odds parameter.
what	character specifying for which quantity the prior should be plotted. Either "logor" (i.e., log odds ratio), "or" (i.e., odds ratio), "p1p2" (i.e., plots the joint distribution of the latent "success" probability in the experimental and control condition), "p1" (i.e., latent "success" probability in the control condition), "p2" (i.e., latent "success" probability in the experimental condition), "p2givenp1" (i.e., plots the conditional distribution of the latent "success" probability in the experimental condition given a "success" probability of p1 in the control condition), "rrisk" (i.e., relative risk, the ratio of the "success" probability in the experimental and the control condition), or "arisk" (i.e., absolute risk, the difference of the "success" probability in the experimental and control condition).
hypothesis	character specifying whether to plot a two-sided prior (i.e., "H1"), a one-sided prior with lower truncation point (i.e., "H+"), or a one-sided prior with upper truncation point (i.e., "H-").
p1	value of the "success" probability in the control condition. Only used when what = "p2givenp1".
...	further arguments.

Note

Internally, the test-relevant prior is always a normal prior on the log odds ratio, however, the `plot_prior` function also allows one to plot the implied prior on different quantities.

Author(s)

Quentin F. Gronau

Examples

```
# prior parameters
prior_par <- list(mu_psi = 0, sigma_psi = 1,
                 mu_beta = 0, sigma_beta = 1)

# plot prior
plot_prior(prior_par = prior_par, what = "logor")
plot_prior(prior_par = prior_par, what = "or")
plot_prior(prior_par = prior_par, what = "p1p2")
plot_prior(prior_par = prior_par, what = "p1")
plot_prior(prior_par = prior_par, what = "p2")
plot_prior(prior_par = prior_par, what = "p2givenp1", p1 = 0.3)
plot_prior(prior_par = prior_par, what = "rrisk")
plot_prior(prior_par = prior_par, what = "arisk")
```

plot_robustness

Plot Bayes Factor Robustness Check

Description

Function for plotting Bayes factor robustness check results (i.e., prior sensitivity analysis).

Usage

```
plot_robustness(
  x,
  bftype = "BF10",
  log = FALSE,
  mu_range = c(0, 0.3),
  sigma_range = c(0.25, 1),
  mu_steps = 40,
  sigma_steps = 40,
  cores = 1,
  ...
)
```

Arguments

x	object of class "ab".
bftype	character that specifies which Bayes factor is plotted. Either "BF10", "BF01", "BF+0", "BF0+", "BF-0", or "BF0-".
log	Boolean that specifies whether the log Bayes factor is plotted.
mu_range	numeric vector of length two that specifies the range of mu_psi values to consider.
sigma_range	numeric vector of length two that specifies the range of sigma_psi values to consider.
mu_steps	numeric value that specifies in how many discrete steps the interval mu_range is partitioned.
sigma_steps	numeric value that specifies in how many discrete steps the interval sigma_range is partitioned.
cores	number of cores used for the computations.
...	further arguments passed to filled.contour.

Details

The plot shows how the Bayes factor changes as a function of the normal prior location parameter `mu_psi` and the normal prior scale parameter `sigma_psi` (i.e., a prior sensitivity analysis with respect to the normal prior on the test-relevant log odds ratio).

Value

Returns a `data.frame` with the `mu_psi` values, `sigma_psi` values, and corresponding (log) Bayes factors.

Author(s)

Quentin F. Gronau

Examples

```
## Not run:
# synthetic data
data <- list(y1 = 10, n1 = 28, y2 = 14, n2 = 26)

# Bayesian A/B test with default settings
ab <- ab_test(data = data)

# plot robustness check (i.e., prior sensitivity analysis)
p <- plot_robustness(ab)

# returned object contains the Bayes factors for the different prior settings
head(p)

## End(Not run)
```

plot_sequential *Plot Sequential Analysis*

Description

Function for plotting the posterior probabilities of the hypotheses sequentially.

Usage

```
plot_sequential(x, thin = 1, cores = 1, ...)
```

Arguments

x	object of class "ab". Note that the "ab" object needs to contain sequential data.
thin	allows the user to skip every k th data point for plotting, where the number k is specified via thin. For instance, in case thin = 2, only every second element of the data is displayed.
cores	number of cores used for the computations.
...	further arguments

Details

The plot shows the posterior probabilities of the hypotheses as a function of the total number of observations across the experimental and control group. On top of the plot, probability wheels (see also [prob_wheel](#)) visualize the prior probabilities of the hypotheses and the posterior probabilities of the hypotheses after taking into account all available data.

N.B.: This plot has been designed to look good in the following format: In inches, 530 / 72 (width) by 400 / 72 (height); in pixels, 530 (width) by 400 (height).

Author(s)

Quentin F. Gronau

Examples

```
### 1.

# synthetic sequential data (observations alternate between the groups)
# note that the cumulative number of successes and trials need to be provided
data <- list(y1 = c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 3, 3, 4, 4),
            n1 = c(1, 1, 2, 2, 3, 3, 4, 4, 5, 5, 6, 6, 7, 7, 8, 8, 9, 9, 10, 10),
            y2 = c(0, 1, 1, 2, 2, 3, 3, 4, 4, 5, 5, 6, 6, 7, 7, 8, 8, 9, 9, 9),
            n2 = c(0, 1, 1, 2, 2, 3, 3, 4, 4, 5, 5, 6, 6, 7, 7, 8, 8, 9, 9, 10))

# conduct Bayesian A/B test with default settings
ab <- ab_test(data = data)
print(ab)
```

```

# produce sequential plot of posterior probabilities of the hypotheses
# (using recommended width and height values for saving to file)
cairo_pdf(file.path(tempdir()), "test_plot.pdf"),
          width = 530 / 72, height = 400 / 72)
plot_sequential(ab)
dev.off()

### 2.

# synthetic sequential data (observations alternate between the groups)
# this time provided in the alternative format
data2 <- data.frame(outcome = c(1, 1, 0, 1, 0, 1, 0, 1, 0, 1,
                               0, 1, 0, 1, 1, 1, 1, 1, 1, 0),
                    group = rep(c(1, 2), 10))

# conduct Bayesian A/B test with default settings
ab2 <- ab_test(data = data2)
print(ab2)

# produce sequential plot of posterior probabilities of the hypotheses
# (using recommended width and height values for saving to file)
cairo_pdf(file.path(tempdir()), "test_plot2.pdf"),
          width = 530 / 72, height = 400 / 72)
plot_sequential(ab2)
dev.off()

## Not run:
### 3.
data(seqdata)

# conduct Bayesian A/B test with default settings
ab3 <- ab_test(data = seqdata)
print(ab3)

# produce sequential plot of posterior probabilities of the hypotheses
# (using recommended width and height values for saving to file)
cairo_pdf(file.path(tempdir()), "test_plot3.pdf"),
          width = 530 / 72, height = 400 / 72)
plot_sequential(ab3, thin = 4)
dev.off()

## End(Not run)

```

Description

Function for evaluating the prior cumulative distribution function (CDF).

Usage

```
pprior(
  q,
  prior_par = list(mu_psi = 0, sigma_psi = 1, mu_beta = 0, sigma_beta = 1),
  what = "logor",
  hypothesis = "H1"
)
```

Arguments

q	numeric vector with quantiles.
prior_par	list with prior parameters. This list needs to contain the following elements: mu_psi (prior mean for the normal prior on the test-relevant log odds ratio), sigma_psi (prior standard deviation for the normal prior on the test-relevant log odds ratio), mu_beta (prior mean for the normal prior on the grand mean of the log odds), sigma_beta (prior standard deviation for the normal prior on the grand mean of the log odds). Each of the elements needs to be a real number (the standard deviations need to be positive). The default are standard normal priors for both the log odds ratio parameter and the grand mean of the log odds parameter.
what	character specifying for which quantity the prior CDF should be evaluated. Either "logor" (i.e., log odds ratio), "or" (i.e., odds ratio), "rrisk" (i.e., relative risk, the ratio of the "success" probability in the experimental and the control condition), or "arisk" (i.e., absolute risk, the difference of the "success" probability in the experimental and control condition).
hypothesis	character specifying whether to evaluate the CDF for a two-sided prior (i.e., "H1"), a one-sided prior with lower truncation point (i.e., "H+"), or a one-sided prior with upper truncation point (i.e., "H-").

Value

numeric vector with the values of the prior CDF.

Note

Internally, the test-relevant prior is always a normal prior on the log odds ratio, consequently, if what is not "logor", the implied prior CDF for the quantity is returned.

Author(s)

Quentin F. Gronau

Examples

```
# prior parameters
prior_par <- list(mu_psi = 0, sigma_psi = 1,
                 mu_beta = 0, sigma_beta = 1)

# evaluate prior CDF
pprior(q = 0.1, prior_par = prior_par, what = "logor")
pprior(q = 1.1, prior_par = prior_par, what = "or")
pprior(q = 1.05, prior_par = prior_par, what = "rrisk")
pprior(q = 0.02, prior_par = prior_par, what = "arisk")

# also works for vectors
pprior(q = c(-0.1, 0, 0.1, 0.2), prior_par = prior_par, what = "logor")
```

prob_wheel	<i>Plot Probability Wheel</i>
------------	-------------------------------

Description

Function for visualizing prior and posterior probabilities of the hypotheses as a probability wheel.

Usage

```
prob_wheel(x, type = "posterior")
```

Arguments

x	object of class "ab".
type	character indicating whether to plot a probability wheel visualizing the prior probabilities of the hypotheses (i.e., type = "prior") or the posterior probabilities of the hypotheses (i.e., type = "posterior"). The default is "posterior".

Author(s)

Quentin F. Gronau

Examples

```
# synthetic data
data <- list(y1 = 10, n1 = 28, y2 = 14, n2 = 26)

# Bayesian A/B test with default settings
ab <- ab_test(data = data)
print(ab)

# visualize prior probabilities of the hypotheses
prob_wheel(ab, type = "prior")

# visualize posterior probabilities of the hypotheses
prob_wheel(ab, type = "posterior")
```

qprior

Prior Quantile Function

Description

Function for evaluating the prior quantile function.

Usage

```
qprior(  
  p,  
  prior_par = list(mu_psi = 0, sigma_psi = 1, mu_beta = 0, sigma_beta = 1),  
  what = "logor",  
  hypothesis = "H1"  
)
```

Arguments

p	numeric vector with probabilities.
prior_par	list with prior parameters. This list needs to contain the following elements: mu_psi (prior mean for the normal prior on the test-relevant log odds ratio), sigma_psi (prior standard deviation for the normal prior on the test-relevant log odds ratio), mu_beta (prior mean for the normal prior on the grand mean of the log odds), sigma_beta (prior standard deviation for the normal prior on the grand mean of the log odds). Each of the elements needs to be a real number (the standard deviations need to be positive). The default are standard normal priors for both the log odds ratio parameter and the grand mean of the log odds parameter.
what	character specifying for which quantity the prior quantile function should be evaluated. Either "logor" (i.e., log odds ratio), "or" (i.e., odds ratio), "rrisk" (i.e., relative risk, the ratio of the "success" probability in the experimental and the control condition), or "arisk" (i.e., absolute risk, the difference of the "success" probability in the experimental and control condition).
hypothesis	character specifying whether to evaluate the quantile function for a two-sided prior (i.e., "H1"), a one-sided prior with lower truncation point (i.e., "H+"), or a one-sided prior with upper truncation point (i.e., "H-").

Value

numeric vector with the values of the prior quantile function.

Author(s)

Quentin F. Gronau

Examples

```
# prior parameters
prior_par <- list(mu_psi = 0, sigma_psi = 1,
                 mu_beta = 0, sigma_beta = 1)

# evaluate prior quantile function
qprior(p = .1, prior_par = prior_par, what = "logor")
qprior(p = .7, prior_par = prior_par, what = "or")
qprior(p = .9, prior_par = prior_par, what = "rrisk")
qprior(p = .7, prior_par = prior_par, what = "arisk")

# also works for vectors
qprior(p = c(.1, .2, .5, .7, .9), prior_par = prior_par, what = "logor")
```

seqdata

Synthetic Sequential Data

Description

This data set contains synthetic sequential A/B data (500 observations in each of the two groups, where the observations are alternating between groups). `y1` denotes the number of successes for the first group, `n1` denotes the corresponding total number of observations for the first group. Similarly, `y2` denotes the number of successes for the second group and `n2` denotes the corresponding total number of observations for the second group.

Usage

```
seqdata
```

Format

A list with 4 elements.

Examples

```
data(seqdata)

# conduct Bayesian A/B test with default settings
ab <- ab_test(data = seqdata)
print(ab)

# produce sequential plot of posterior probabilities of the hypotheses
plot_sequential(ab, thin = 4)

# example of good width and height values for saving to file
cairo_pdf(file.path(tempdir(), "test_plot.pdf"),
          width = 530 / 72, height = 400 / 72)
plot_sequential(ab)
dev.off()
```

simulate_priors	<i>Simulate from Parameter Priors</i>
-----------------	---------------------------------------

Description

Function for simulating from the parameter prior distributions.

Usage

```
simulate_priors(
  nsamples,
  prior_par = list(mu_psi = 0, sigma_psi = 1, mu_beta = 0, sigma_beta = 1),
  hypothesis = "H1"
)
```

Arguments

nsamples	number of samples.
prior_par	list with prior parameters. This list needs to contain the following elements: mu_psi (prior mean for the normal prior on the test-relevant log odds ratio), sigma_psi (prior standard deviation for the normal prior on the test-relevant log odds ratio), mu_beta (prior mean for the normal prior on the grand mean of the log odds), sigma_beta (prior standard deviation for the normal prior on the grand mean of the log odds). Each of the elements needs to be a real number (the standard deviations need to be positive). The default are standard normal priors for both the log odds ratio parameter and the grand mean of the log odds parameter.
hypothesis	character specifying whether to sample from a two-sided prior (i.e., "H1"), a one-sided prior with lower truncation point (i.e., "H+"), or a one-sided prior with upper truncation point (i.e., "H-").

Value

a data frame with prior samples for the following quantities (see ?ab_test for a description of the underlying model):

- beta: prior samples for the grand mean of the log odds.
- psi: prior samples for the log odds ratio.
- p1: prior samples for the latent "success" probability in the control group.
- p2: prior samples for the latent "success" probability in the experimental group.
- logor: prior samples for the log odds ratio (identical to psi, only included for easier reference).
- or: prior samples for the odds ratio.
- rrisk: prior samples for the relative risk (i.e., the ratio of the "success" probability in the experimental and the control condition).

- *arisk*: prior samples for the absolute risk (i.e., the difference of the "success" probability in the experimental and control condition)

Author(s)

Quentin F. Gronau

Examples

```
# prior parameters
prior_par <- list(mu_psi = 0, sigma_psi = 1,
                 mu_beta = 0, sigma_beta = 1)

# obtain prior samples
samples <- simulate_priors(nsamples = 1000, prior_par = prior_par)

# plot, e.g., prior samples for absolute risk
hist(samples$arisk)
```

Index

* dataset

seqdata, [21](#)

ab-methods, [2](#)

ab_test, [2](#), [3](#), [10](#)

dprior, [6](#)

elicit_prior, [6](#), [8](#)

extractors, [10](#)

get_bf (extractors), [10](#)

get_post_prob (extractors), [10](#)

get_post_samples (extractors), [10](#)

get_prior_prob (extractors), [10](#)

plot.ab (ab-methods), [2](#)

plot_posterior, [2](#), [6](#), [11](#)

plot_prior, [6](#), [10](#), [13](#)

plot_robustness, [14](#)

plot_sequential, [3](#), [6](#), [16](#)

pprior, [17](#)

print.ab (ab-methods), [2](#)

print.summary.ab (ab-methods), [2](#)

prob_wheel, [3](#), [6](#), [16](#), [19](#)

qprior, [20](#)

seqdata, [21](#)

simulate_priors, [6](#), [22](#)

summary.ab (ab-methods), [2](#)